	GESTIÓN DE SERVICIOS ACADÉMICOS Y BIBLIOTECARIOS		CÓDIGO	FO-GS-15
			VERSIÓN	01
	ESQUEMA HOJA DE RESUMEN		FECHA	20/03/2023
			PÁGINA	1 de 2
ELABORÓ	REVISÓ	APROBÓ		
Jefe División de Biblioteca	Equipo Operativo de Calidad	Líder de Calidad		

RESUMEN TRABAJO DE GRADO

AUTOR(ES): NOMBRES Y APELLIDOS COMPLETOS

NOMBRE(S): ANGIE MADELEYNE APELLIDOS: GONZÁLEZ BARRERA

NOMBRE(S): CRISEL JAZMÍN APELLIDOS: AYALA LLANES

FACULTAD: INGENIERIA

PLAN DE ESTUDIOS: INGENIERÍA DE SISTEMAS

DIRECTOR:

NOMBRE(S): MILTON JESÚS APELLIDOS: VERA CONTRERAS

NOMBRE(S): DEYSI YURLEY APELLIDOS: ROLON FLOREZ

TÍTULO DEL TRABAJO (TESIS): TRAINING CENTER HIGH SCHOOL

Este proyecto realiza una extensión de la plataforma “Training Center UFPS” (<https://trainingcenter.cloud.ufps.edu.co>), con el fin de llevar la programación competitiva a nivel de colegios. Para así establecer una preparación temprana y lograr obtener mejores resultados en las competencias locales, nacionales y regionales, y también con el ánimo de despertar el interés por la programación en los estudiantes de bachillerato de la ciudad de Cúcuta. Para lograr tal fin se implementó la metodología usada los últimos años por el grupo de estudio de Programación Competitiva, perteneciente al semillero SILUX, del programa de Ingeniería de Sistemas de la Universidad Francisco de Paula Santander. Integrando a la plataforma un nuevo lenguaje de programación, como estrategia para generar motivación y despertar interés por la Programación Competitiva en los estudiantes de bachillerato de los colegios de la ciudad de Cúcuta.

PALABRAS CLAVES: Blockly, programación competitiva, colegios.

CARACTERÍSTICAS: PÁGINAS: 422 PLANOS: 0 ILUSTRACIONES: 133 CD ROOM: 0

TRAINING CENTER HIGH SCHOOL

Proyecto para optar por el título de Ingeniero de Sistemas

Autores

Angie Madeleyne González Barrera

Crisel Jazmín Ayala Llanes

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

CÚCUTA NORTE DE SANTANDER

FACULTAD DE INGENIERÍA

INGENIERÍA DE SISTEMAS

2022

TRAINING CENTER HIGH SCHOOL

Proyecto para optar por el título de Ingeniero de Sistemas

Autores

Angie Madeleyne González Barrera

Crisel Jazmín Ayala Llanes

Director

Profesor Milton Jesús Vera Contreras

UFPS

Co-Directora

Profesora Deisy Yurley Rolón

INSTITUCIÓN EDUCATIVA JULIO PEREZ FERRERO SEDE 4

UNIVERSIDAD FRANCISCO DE PAULA SANTANDER

CÚCUTA NORTE DE SANTANDER

FACULTAD DE INGENIERÍA

INGENIERÍA DE SISTEMAS

2022



ACTA DE SUSTENTACIÓN DE TRABAJO DE GRADO

FECHA: 21 DE DICIEMBRE DE 2022

HORA: 2:30 P. M.

LUGAR: AUDITORIO JJ MALDONADO 4 PISO AULA SUR - UFPS

PLAN DE ESTUDIOS: INGENIERÍA DE SISTEMAS

TÍTULO DEL TRABAJO DE GRADO: "TRAININGCENTER HIGH SCHOOL"

Jurados

ING.NELLY ROSANA DIAZ LEAL

ING.MATIAS HERRERA CACERES

ING.BORIS RAINIERO PÉREZ GUTIERREZ

DIRECTOR: MILTON JESÚS VERA CONTRERAS

CODIRECTOR: DEISY YURLEY ROLÓN

NOMBRE DEL ESTUDIANTE	CÓDIGO	CALIFICACIÓN	
		NÚMERO	LETRA
CRISEL JAZMIN AYALA LLANES	1151376	5,0	CINCO,CERO
ANGIE MADELEYNE GONZÁLEZ BARRERA	1151368	5,0	CINCO,CERO

LAUREADA

FIRMA DE LOS JURADOS

ING.NELLY ROSANA DIAZ LEAL

ING.MATIAS HERRERA CACERES

ING. BORIS RAINIERO PÉREZ GUTIÉRREZ

JUDITH DEL PÍLAR RODRIGUEZ TENJO

Coordinadora Comité Curricular

Dedicatoria

Este proyecto v dedicado a los estudiantes y profesores pertenecientes al grupo de estudio “Programación Competitiva” del semillero SILUX de la universidad Francisco de Paula Santander, los cuáles trabajan arduamente para que cada día tenga mejor representación y resultados en las diferentes competencias de programación.

Agradecimientos

Primeramente, agradecemos a Dios por la oportunidad de empezar una carrera profesional y llegar a culminarla. Gracias a Dios por la sabiduría y fortaleza en todo el tiempo de este camino recorrido.

A familiares, por el apoyo a lo largo de estos años, porque su apoyo fue gran aporte para llegar a la meta. Gracias por sembrar la semilla del estudio, regarla y abonarla con tanto sacrificio todos estos años.

De igual forma, agradecimientos a nuestro director de tesis Milton Jesús Vera Contreras por todo el apoyo dado a lo largo de estos años y también por el apoyo en el desarrollo de este proyecto, de inicio a fin.

A compañeros de carrera y grandes amigos, que siempre ofrecieron apoyo y acompañamiento. Gracias por compartir de la manera más extraordinaria esta experiencia de trabajo. Gracias por festejar cada logro obtenido y así mismo, entender cada frustración.

Muchas gracias al grupo de estudio “Programación Competitiva UFPS”, a sus integrantes y líderes, por esta gran oportunidad, por ofrecernos crecimiento tanto personal como profesional y, por todo el apoyo y colaboración en el desarrollo de este proyecto. Gracias porque fueron de gran motivación para poder finalizar y sacar este proyecto adelante.

Y, por último, pero no menos importante, muchas gracias a todos, los docentes, estudiantes, y personal de la UFPS que de una u otra manera aportaron su grano de arena en estos esfuerzos.

Angie González Barrera y Crisel Ayala Llanes.

TABLA DE CONTENIDO

1. Introducción	22
2. Presentación general del anteproyecto	23
2.1. Resumen	23
2.2. Título	23
2.3. Planteamiento del problema	24
2.4. Justificación	26
2.4.1. Análisis de justificación Pestel	27
2.4.2. Análisis de justificación geográfico	35
2.5. Objetivos	39
2.5.1. Objetivo general	39
2.5.2. Objetivos específicos	39
2.6. Alcance y delimitación	40
2.6.1. Alcance	40
2.7. Cronograma	41
2.8. Presupuesto	53
3. Programación competitiva	58
3.1. ICPC	59
3.2. Grupo de estudio del semillero SILUX UFPS	62
3.3. IOI	64

3.3.1. Contexto Mundial	65
3.3.2. Contexto Latinoamericano	65
3.3.3. Colombia	69
3.4. Propuesta Training Center High School	71
4. Redes, plataformas y lenguajes de programación para colegios y programación competitiva	79
4.1. Plataformas	81
4.2. Redes	88
4.3. Lenguajes	90
4.4. Programa Misión Tic 2022 Del Gobierno Nacional	96
4.5. Blockly	98
5. Desarrollo del proyecto	111
5.1. Metodología del proyecto	111
5.2. Levantamiento de información	116
5.3. Proceso de desarrollo de software	129
5.4. Prueba piloto	138
6. Nueva Versión Training Center High School	140
6.1. Requerimientos o funcionalidades	144
6.2. Proceso de desarrollo de software	148
6.3. Arquitectura de software	173
6.4. Experiencia de desarrollo y despliegue de integración continua	211

6.5. Pruebas de software	214
6.5.1. Alcance de las pruebas	214
6.5.2. Propósito de la evaluación	214
6.5.3. Motivadores de la prueba	215
6.5.4. Objetos a ser Evaluados	215
6.5.5. Ámbito de las pruebas	216
6.5.6. Lista de ideas de las pruebas	217
6.5.7. Enfoque de las pruebas	217
6.5.8. Herramientas para las pruebas	218
6.5.9. Métricas con SonarQube	218
6.5.10. Pruebas de rendimiento web	221
6.5.11. Pruebas de aceptación de software	222
6.6. Documentación de software	272
7. Prueba piloto en colegios	274
8. Divulgación científica y evaluación por pares	290
9. Productos	299
10. Conclusiones	302
11. Recomendaciones	304
12. Trabajo futuro	306
13. Referencias	308

LISTA DE TABLAS

Tabla 1. Análisis PESTEL ¿Por qué se necesita extender la plataforma training center a los colegios? Fuente: elaboración propia según literatura revisada y citada en este documento	27
Tabla 2 Justificación tomando como punto de vista el ámbito geográfico. Fuente: elaboración propia según literatura revisada y citada en este documento	35
Tabla 3 Actividades del proyecto. Fuente: Elaboración propia.	42
Tabla 4 Cronograma del proyecto. Fuente: Elaboración propia	47
Tabla 5 Presupuesto Total del Proyecto. Fuente: Elaboración Propia	54
Tabla 6 Presupuesto: Descripción de los gastos de personal. Fuente: Elaboración Propia.	55
Tabla 7 Presupuesto: Descripción de materiales y suministro. Fuente: Elaboración Propia.	55
Tabla 8 Presupuesto: Descripción de los equipos e infraestructura Fuente: Elaboración Propia.	56
Tabla 9 Ejes del marco de trabajo del grupo de estudio en Programación Competitiva UFPS. Fuente: Tesis Delgado & Lázaro 2018	72
Tabla 10 Redes, Plataformas y Lenguajes.	80
Tabla 11 Lineamientos para aplicar el diseño de la ciencia. Fuente: (Vera Contreras & Peña Reyes, 2013).	112
Tabla 12 Requerimientos funcionales	144
Tabla 13 Tecnologías actualizadas o agregadas para la nueva versión del software Training Center.	150
Tabla 14 Modificaciones en Front-end.	152
Tabla 15 Modificaciones en Back-end.	165
Tabla 16 Modelo de datos para users.	197

Tabla 17 Modelo de datos para contest_students. Elaboración propia	198
Tabla 18 Modelo de datos para contest.	199
Tabla 19 Modelo de datos para submission.	200
Tabla 20 Modelo de datos para contest_problems.	201
Tabla 21 Modelo de datos para syllabus_students.	202
Tabla 22 Modelo de datos para assignments_problems.	202
Tabla 23 Modelo de datos para problems.	203
Tabla 24 Modelo de datos para syllabus.	204
Tabla 25 Modelo de datos para assignments.	205
Tabla 26 Modelo de datos para syllabus_materials.	206
Tabla 27 Modelo de datos para categories.	207
Tabla 28 Modelo de datos para materials.	207
Tabla 29 Modelo de datos para blacklist_tokens.	208
Tabla 30 Modelo de datos para institutions.	209
Tabla 31 Módulos de la plataforma.	210
Tabla 32 Formato de las pruebas	223
Tabla 33 Formato de pruebas CP_01	224
Tabla 34 Formato de pruebas CP_02	226
Tabla 35 Formato de pruebas CP_03	228
Tabla 36 Formato de pruebas CP_04	231
Tabla 37 Formato de pruebas CP_05	234
Tabla 38 Formato de pruebas CP_06	237
Tabla 39 Formato de pruebas CP_07	238

Tabla 40 Formato de pruebas CP_08	241
Tabla 41 Formato de pruebas CP_09	243
Tabla 42 Formato de pruebas CP_10	245
Tabla 43 Formato de pruebas CP_11	248
Tabla 44 Formato de pruebas CP_12	251
Tabla 45 Formato de pruebas CP_13	254
Tabla 46 Formato de pruebas CP_14	257
Tabla 47 Formato de pruebas CP_15	259
Tabla 48 Formato de pruebas CP_16	262
Tabla 49 Formato de pruebas CP_17	264
Tabla 50 Formato de pruebas CP_18	266
Tabla 51 Formato de pruebas CP_19	269

LISTA DE FIGURAS

Ilustración 1 Calculadora cloud de Google. Fuente: Calculadora Google Cloud	57
Ilustración 2 Mapa regiones mundiales de ICPC. Fuente: ICPC (https://icpc.global/regionals/finder).	60
Ilustración 3 Región América latina. Fuente: ICPC (https://icpc.global/regionals/finder/sa)	61
Ilustración 4 Marco de Trabajo del Semillero SILUX (Delgado & Lázaro 2018)	71
Ilustración 5 Marco de Trabajo del semillero SILUX actualizado	75
Ilustración 6 Presentación Misión TIC 2022. Fuente: Ministerio de las TIC.	96
Ilustración 7 Tablero de edición de Blockly. Fuente: Blockly (https://developers.google.com/Blockly?HI=es-419)	99
Ilustración 8 Herramienta para desarrolladores de Blockly. Fuente: Blockly (https://developers.google.com/Blockly?HI=es-419)	102
Ilustración 9 Ejemplo implementación de generador de código.	103
Ilustración 10 Bloques de entrada.	104
Ilustración 11 Bloques de salida.	105
Ilustración 12 Bloques de lógica.	105
Ilustración 13 Bloques de ciclos.	106
Ilustración 14 Bloques de matemáticas.	106
Ilustración 15 Bloques de texto.	107
Ilustración 16 Bloques de listas.	107
Ilustración 17 Bloques de variables.	108
Ilustración 18 Bloques de funciones.	108

Ilustración 19 Implementación de bloques personalizados.	109
Ilustración 20 Implementación de Blockly en Training Center.	110
Ilustración 21 Marco de Investigación en ISD. Fuente: (R. Hevner et al., 2004)	112
Ilustración 22 Fases de la metodología y objetivos del proyecto.	114
Ilustración 23 Metodología del proyecto con los objetivos propuestos adheridos.	115
Ilustración 24 Pasos a seguir en el análisis y recopilación de información.	117
Ilustración 25 Encuesta para colegios propuesta Training Center High School.	118
Ilustración 26 Estadísticas pregunta “¿En su colegio se enseña programación de computadores?”.	122
Ilustración 27 Estadísticas pregunta “Si la respuesta anterior fue "si" seleccione el o los lenguajes que enseñan. Seleccione Ninguno si su respuesta fue "no”.	123
Ilustración 28 Estadísticas pregunta “¿En qué grados se enseña programación de computadores?”.	124
Ilustración 29 Estadísticas pregunta “¿Cuántas horas a la semana se enseña?”.	125
Ilustración 30 Estadísticas pregunta “¿Qué nivel de programación de computadores poseen los estudiantes al graduarse?”.	126
Ilustración 31 Estadísticas pregunta “¿Le interesaría recibir apoyo académico por parte de la UFPS para procesos relacionados con la enseñanza de programación de computadores?”.	127
Ilustración 32 Estadísticas pregunta “¿Le interesaría participar en competencias de programación (maratones de programación) llevadas a cabo por la UFPS?”.	128
Ilustración 33 Fases de la metodología Scrum. Fuente: (Pérez, 2019).	130
Ilustración 34 Reunión planificación actividades.	134
Ilustración 35 Tablero de Trello.	135

Ilustración 36 Tarea a alto nivel en Trello.	137
Ilustración 37 Pasos para la prueba piloto en colegios.	139
Ilustración 38 Ruta a seguir para hacer adaptaciones a Training Center.	140
Ilustración 39 Evolución 1 de la ruta a seguir para hacer adaptaciones en Training Center.	141
Ilustración 40 Evolución 2 de la ruta a seguir para hacer adaptaciones en Training Center.	142
Ilustración 41 Versión final de la ruta a seguir para hacer adaptaciones en Training Center.	143
Ilustración 42 Diagrama de arquitectura primera versión Training Center.	174
Ilustración 43 Diagrama de arquitectura a alto nivel.	175
Ilustración 44 Diagrama de arquitectura a bajo nivel.	176
Ilustración 45 Subcomponente routes.	178
Ilustración 46 Subcomponente services.	179
Ilustración 47 Subcomponente controllers.	180
Ilustración 48 Configuración del componente back-end.	180
Ilustración 49 Subcomponente models.	181
Ilustración 50 Subcomponente migrations.	182
Ilustración 51 Subcomponente Grader.	183
Ilustración 52 Subcomponente seeders.	183
Ilustración 53 Componente modules.	187
Ilustración 54 Componente layouts.	187
Ilustración 55 Componente de assets.	188
Ilustración 56 Componente models.	188
Ilustración 57 Componente services.	189
Ilustración 58 Componente resources.	189

Ilustración 59 Componente elements.	190
Ilustración 60 Configuración del componente front-end.	191
Ilustración 61 Diagrama de clases.	195
Ilustración 62 Modelo de Base de datos.	196
Ilustración 63 Control de versiones.	212
Ilustración 64 Código fuente primera versión de Training Center UFPS.	213
Ilustración 65 pruebas con SonarQube.	218
Ilustración 66 Análisis de líneas de código back-end	220
Ilustración 67 Análisis de líneas de código front-end	220
Ilustración 68 Pruebas de rendimiento web	221
Ilustración 69 Detalle del análisis de rendimiento web.	222
Ilustración 70 CP_01 Módulo ver problemas de colegios	226
Ilustración 71 CP_02 Listado de problemas	228
Ilustración 72 CP_03 Vista agregar problemas	231
Ilustración 73 CP_03 Confirmación agregar problemas	231
Ilustración 74 CP_04 Opción editar problema	234
Ilustración 75 CP_05 Opción eliminar problemas	236
Ilustración 76 CP_05 Mensaje de advertencia eliminar problema	236
Ilustración 77 CP_05 Confirmación eliminar problema	236
Ilustración 78 CP_06 Foro de discusión DISQUS por problema	238
Ilustración 79 CP_07 Opción buscar problemas	240
Ilustración 80 CP_07 Resultados de la búsqueda	240
Ilustración 81 CP_08 Categorías de problemas para colegios	242

Ilustración 82 CP_08 Categorías de problemas para universidades	242
Ilustración 83 CP_09 Opción ver ranking desde menú	244
Ilustración 84 CP_09 Listado de ranking con opción de filtrar	244
Ilustración 85 CP_10 Opción crear maratón	247
Ilustración 86 CP_10 Formulario crear maratón	247
Ilustración 87 CP_10 Notificación maratón creada	247
Ilustración 88 CP_11 Opción subir material por categoría	250
Ilustración 89 CP_11 Formulario anexar material a categoría	250
Ilustración 90 CP_11 Notificación material creado	251
Ilustración 91 CP_12 Listado de materiales por categoría	253
Ilustración 92 CP_12 Ver detalle de material	253
Ilustración 93 CP_13 Ejemplo tour página principal	256
Ilustración 94 CP_13 Ver tour desde página de detalle de problema	256
Ilustración 95 CP_14 Opción crear syllabus	259
Ilustración 96 CP_14 Notificación tarea creada:	259
Ilustración 97 CP_15 Opción ver tareas de syllabus	261
Ilustración 98 CP_15 Edición de tarea	261
Ilustración 99 CP_15 Estadísticas de envíos hechos por tarea	262
Ilustración 100 CP_16 Opción eliminar usuario desde usuario administrador	264
Ilustración 101 CP_17 Ver calificación de envío (Blockly)	266
Ilustración 102 CP_18 Separación de categorías (Por tipo colegio y universidad)	268
Ilustración 103 CP_18 Página principal Training Center High School	268

Ilustración 104 CP_19 Opción ver envíos de usuario desde ranking (opción disponible para administrador)	271
Ilustración 105 CP_19 Opción ver envíos en competencias (opción disponible para administrador)	271
Ilustración 106 CP_19 Opción ver envíos de estudiantes en tareas (disponible para usuario coach)	272
Ilustración 107 Documentación de software	273
Ilustración 108 Reunión Inicial planeación prueba piloto.	275
Ilustración 109 Reunión Inducción sobre Training Center High School a profesores de colegios.	277
Ilustración 110 Reunión Inducción sobre Training Center High School a profesores de colegios.	277
Ilustración 111 Reunión Inducción sobre Training Center High School a profesores de colegios.	278
Ilustración 112 Reunión Inducción sobre Training Center High School a profesores de colegios.	278
Ilustración 113 Reunión Inducción sobre Training Center High School a estudiantes de colegios.	281
Ilustración 114 Reunión Inducción sobre Training Center High School a estudiantes de colegios.	281
Ilustración 115 Reunión Inducción sobre Training Center High School a estudiantes de colegios.	282

Ilustración 116 Problemas maratón de entrenamiento, prueba piloto Training Center High School.	283
Ilustración 117 Maratón de entrenamiento, prueba piloto Training Center High School.	283
Ilustración 118 Solución de Ejercicio en lenguajes de bloques, en maratón de entrenamiento, prueba piloto Training Center High School.	284
Ilustración 119 Clasificación general maratón de entrenamiento, prueba piloto Training Center High School.	285
Ilustración 120 Estudiantes participantes en maratón de entrenamiento, prueba piloto Training Center High School.	286
Ilustración 121 Recopilación de datos de servidor de Google cloud 1	288
Ilustración 122 Recopilación de datos de servidor de Google cloud 2	289
Ilustración 123 Certificado Ponencia VII Encuentro Regional Semilleros de Investigación 1	290
Ilustración 124 Certificado Ponencia VII Encuentro Regional Semilleros de Investigación 2	291
Ilustración 125 Presentación Training Center High School como ponencia	292
Ilustración 126 Poster Training Center High School	293
Ilustración 127 Certificado ponencia XXIII Encuentro Nacional y XVII Encuentro Internacional de Semilleros de Investigación 1	294
Ilustración 128 Certificado ponencia XXIII Encuentro Nacional y XVII Encuentro Internacional de Semilleros de Investigación 2	295
Ilustración 129 Capacitaciones para mujeres emprendedoras en TIC. Fuente (Oriente noticias Canal TRO)	296
Ilustración 130 Certificado ponencia en Encuentro Internacional de Educación en Ingeniería ACOFI 2020 1	298

Ilustración 131 Certificado Ponencia en Encuentro Internacional de Educación en Ingeniería ACOFI 2020 2	298
Ilustración 132 Mapa de navegación Training Center	301
Ilustración 133 Vista niveles Training Center High School	306

LISTA DE ANEXOS

Anexo 1. Manual de Usuario - Estudiante	313
Anexo 2. Manual de Usuario – Coach-Docente	342
Anexo 3. Manual de Usuario – Administrador	365
Anexo 4. Documento Mockups	389
Anexo 5. Especificación de requerimientos	400
Anexo 6. Material de apoyo Blockly	413
Anexo 7. Ficha Técnica	420

1. Introducción

Desde hace varios años por medio del grupo de estudio de programación competitiva de la universidad Francisco de Paula Santander se vienen realizando maratones de programación anuales, en las cuales han participado varios estudiantes formando grupos de 3 personas para dar solución a una lista de problemas creados para cada maratón. Esto con el fin no sólo de incentivar a los estudiantes a seguir en las competencias de programación sino también para hacer una clasificación a la maratón nacional que regularmente se realiza en la ciudad de Bucaramanga. Esta maratón da lugar a una clasificación para la maratón regional latinoamericana. Este trabajo realizado por el grupo de estudio de programación competitiva ha logrado una consolidación mayor con el trabajo hecho por Gerson Lázaro y Melissa Delgado al realizar el marco de trabajo Training Center UFPS, con el fin de facilitar los entrenamientos de programación competitiva para los estudiantes de la universidad Francisco De Paula Santander. Con esto se evidenciaron grandiosos resultados tanto en la integración de más estudiantes en el grupo de estudio programación competitiva como en los resultados presentados en las maratones nacionales y latinoamericanas. Estas maratones nacionales y regionales han dado lugar para conocer a estudiantes de otras universidades y de otros lugares del país que se han logrado posicionar durante varios años en los primeros lugares de la maratón regional latinoamericana la cual da clasificación a la maratón mundial de programación. Estas relaciones sociales permiten conocer el proceso realizado por estos estudiantes para obtener esos logros, evidenciando y llevando a la conclusión la importancia de la preparación temprana para lograr mejores resultados en el ámbito de la programación competitiva.

2. Presentación general del anteproyecto

2.1. Resumen

Este proyecto realiza una extensión de la plataforma “Training Center UFPS” (<https://trainingcenter.cloud.ufps.edu.co>), con el fin de llevar la programación competitiva a nivel de colegios. Para así establecer una preparación temprana y lograr obtener mejores resultados en las competencias locales, nacionales y regionales, y también con el ánimo de despertar el interés por la programación en los estudiantes de bachillerato de la ciudad de Cúcuta. Para lograr tal fin se implementó la metodología usada los últimos años por el grupo de estudio de Programación Competitiva, perteneciente al semillero SILUX, del programa de Ingeniería de Sistemas de la Universidad Francisco de Paula Santander. Integrando a la plataforma un nuevo lenguaje de programación, como estrategia para generar motivación y despertar interés por la Programación Competitiva en los estudiantes de bachillerato de los colegios de la ciudad de Cúcuta.

2.2. Título

Training Center High School

2.3. Planteamiento del problema

Mundialmente el software ocupa un papel muy importante, es el motor de la economía en la Cuarta Revolución Industrial (Andreessen, 2011; Giorgio & Mon, 2018). Por esta razón se hace necesario que los estudiantes aprendan a programar desde el colegio, algo que ya está sucediendo en muchos países (Buitrago Flórez et al., 2017; Scasso et al., 2019). Por ejemplo, en Colombia, el año pasado el Ministerio de las Tecnologías de la Información y las Comunicaciones (MINTIC) anunció el inicio de un proyecto para enseñar a programar en los colegios (MINTIC, 2019; Decreto N.º 1008, capítulo 1, 2018). Sin embargo, Colombia aún no incluye la Programación de Computadores dentro de sus estándares curriculares de educación básica y media y los avances que hay en esta materia aún son incipientes (Buitrago Flórez et al., 2017). No obstante, existen colegios que enseñan a programar, como se puede evidenciar en los resultados de las competencias que organiza Fedesoft anualmente (Fedesoft, 2019) y los resultados en la Olimpiada Internacional de Informática (IOI, 2019.). Pero en Colombia y en Latinoamérica aún falta mucho camino por recorrer en esta materia: Según palabras del presidente de la República, Iván Duque Márquez, para el año 2020 se requieren 150.000 programadores que aún no se han formado (Goyeneche, 2019); en toda Latinoamérica sigue siendo insuficiente el talento humano en programación (Goyeneche, 2019) y posiblemente en el futuro será indispensable saber programar computadores para emplearse (Goyeneche, 2019).

Por otra parte, bajo las anteriores circunstancias, en el ámbito universitario los estudiantes logran muy buenos resultados en programación de computadores, a juzgar por el desempeño en

las competencias nacionales, latinoamericanas y mundiales, a las cuales Colombia ha asistido anualmente desde 2004 (ACIS, 2019) y en los últimos cinco años ha logrado lugares representativos en el ranking latinoamericano (ACIS, 2019; ICPC, 2018; UFPS, 2019). Parte del éxito en estas competencias surge de una dinámica de trabajo en equipo, autoaprendizaje, liderazgo y la motivación que genera la competencia (Bloomfield & Sotomayor, 2016; Halim & Halim, 2013; Lázaro et al., 2016; Carrillo et al., 2019). Además, también influye el trabajo colaborativo de internacionalización virtual, a través de las redes Red de Programación Competitiva (RPC, s. F.) Y Colombian Competitive Programming League (CCPL, 2019) (Contreras, 2019).

Tal es el caso del Grupo de estudio de Programación Competitiva del Semillero SILUX de la UFPS, el cual ha mostrado avances muy representativos en su desempeño desde su fundación en el año 2015 (Lázaro et al., 2016; Carrillo et al., 2019), con la mejor participación en el año 2019, al lograr ubicarse en el top 10 y top 20 de la competencia latinoamericana regional (ACIS, 2019). Este grupo de estudio sigue un Marco de Trabajo y usa una Plataforma Tecnológica “Training Center” (UFPS, 2018), ambos desarrollados, probados y mantenidos por estudiantes del mismo semillero. El marco de trabajo guía las actividades de aprendizaje, entrenamiento, motivación y divulgación, entre otras y la plataforma permite unificar y mantener integrada la información de los estudiantes, problemas, soluciones, competencias, resultados, ranking y demás información requerida. Así mismo, la plataforma soporta las competencias de programación de computadores en varios lenguajes de programación, específicamente C++, Java y Python (Lázaro et al., 2016; Carrillo et al., 2019). Adicionalmente, se ha identificado que hay

estudiantes que se vinculan desde primer semestre al semillero, pues han aprendido a programar en su colegio o de manera autónoma.

Considerando lo anterior, surge la oportunidad de ampliar el alcance del semillero a los colegios, con lo cual se aborda la necesidad de programadores que tiene el país, se apoya y reclutan talentos para seguir mejorando el desempeño en las competencias universitarias. Por lo tanto, este proyecto propone realizar una extensión a la Plataforma Training Center para su utilización en colegios, específicamente colegios de la ciudad de Cúcuta, siguiendo el Marco de Trabajo del Semillero SILUX. De esta manera, se procura inducir a los jóvenes desde edad temprana en la dinámica de la programación competitiva o maratones de programación. Para esto, es fundamental responder las siguientes preguntas de investigación ¿Qué plataformas y lenguajes de programación son los más apropiados para generar motivación en los estudiantes de colegios, acerca de las maratones de programación? ¿Qué modificaciones y nuevos requerimientos se deben realizar a la plataforma Training Center para su utilización en los colegios? ¿Qué metodologías y arquitecturas de software deben usarse para la modificación de esta plataforma? ¿Cómo realizar una adecuada transferencia de conocimiento a los diferentes colegios de la ciudad de Cúcuta, por medio de esta herramienta software?

2.4. Justificación

Para justificar el presente proyecto se usará la Análisis PESTEL (político, económico, social, tecnológico, ecológico y legal) (FODA-DAFO, 2020) y un análisis de ámbito geográfico (UFPS, Cúcuta, Colombia, Latinoamérica, mundo), lo cual se resume en las siguientes tablas.

2.4.1. Análisis de justificación Pestel

Tabla 1. Análisis PESTEL ¿Por qué se necesita extender la plataforma training center a los colegios? Fuente: elaboración propia según literatura revisada y citada en este documento

Político	Económico
<ul style="list-style-type: none"> ● La Programación de Computadores ya hace parte de la política pública a nivel nacional y mundial. ● El gobierno nacional, el Ministerio de las TIC, y el Ministerio de Educación y los órganos multilaterales como el Banco Interamericano de Desarrollo (BID), la CEPAL y la OCDE han establecido la necesidad de que la programación de computadores se 	<ul style="list-style-type: none"> ● La economía mundial actual gira en torno a las Tecnologías de la Información y la Comunicación; con el auge de la Cuarta revolución industrial, los modelos económicos tradicionales se están transformando a modelos más estratégicos, donde la creatividad, la innovación y el lanzamiento de productos y servicios de la mano del software, serán claves para mejorar la eficacia de las empresas; sumado a eso,

<p>convierta en un estándar curricular en las instituciones de educación superior (REPÚBLICA DE COLOMBIA, 2019).</p> <ul style="list-style-type: none"> ● La Olimpiada Internacional de Informática, es uno de los eventos más importantes en lo que se refiere a competencias de programación en el mundo, donde asisten cientos de estudiantes de Secundaria y Educación Superior, seleccionados entre la élite de los mejores programadores de cada país que participa (IOI, 2019). ● El gobierno nacional y los organismos multilaterales, apoyan muchos eventos y competencias que involucran programación de computadores y en los cuales la UFPS ha participado. Tal es el caso de la Maratón de Programación de Colegios que organiza Fedesoft, el programa Apps.co de MINTIC y las 	<p>la economía se está volviendo digital (Joyanes Aguilar, 2017).</p> <ul style="list-style-type: none"> ● Bajo este contexto, se puede detectar una necesidad urgente de programadores. De hecho, según el Banco Interamericano de Desarrollo (BID), para el año 2025 se espera que hagan falta más de 1,2 millones de programadores, sólo en América Latina (IDB, 2020). ● En las convocatorias de grandes empresas de TIC como Google, Facebook, Microsoft, o Huawei, haber participado en competencias de programación de computadores, aumenta la posibilidad de ser reclutado. Además, dentro de las pruebas de reclutamiento deben superarse pruebas similares a las que se enfrentan en Programación Competitiva.
---	---

<p>competencias como la Transport Hackathon.</p> <ul style="list-style-type: none"> ● A nivel nacional, se han lanzado algunas iniciativas por parte del Gobierno Colombiano, para que los estudiantes puedan aprender sobre programación básica (como es el caso del Proyecto “Programación para Niños y Niñas” publicado en el año 2019) (MINTIC, 2019). ● En algunos países ya se están llevando a cabo distintas estrategias y proyectos para que la programación sea parte de la enseñanza obligatoria de sus instituciones educativas, como es el caso de Program.ar en Argentina (Scasso et al., 2019), el Chile_Programa, y Progetiiger, de Estonia, como país pionero en este tipo de programas (HITSA, 2019.). 	<ul style="list-style-type: none"> ● Además de que el mundo está necesitando urgentemente programadores para las empresas, ellos deben tener muy bien desarrolladas una cantidad de competencias, como lo es la resolución de problemas y mejoramiento en las destrezas de programación, el trabajo en equipo y la comunicación; las Maratones de Programación pueden ser una útil herramienta para lograr dicho fin (Blandón Andrade, 2019). ● El involucramiento de los estudiantes de colegios en la programación competitiva pretende no sólo incentivarlos a ser creadores de soluciones tecnológicas sino también mostrarles los beneficios y oportunidades que esto les genera en su ámbito económico y laboral futuro.
--	--

<ul style="list-style-type: none"> ● Por todo esto en este proyecto se quiere aprovechar el apoyo del gobierno Nacional en el tema, para lograr incentivar en nuestra región por medio de la programación competitiva el involucramiento de los adolescentes de colegios al entorno de creación de soluciones a través de la tecnología. 	<p>Creando para ellos dentro del entorno difícil de frontera que se vive un futuro esperanzador.</p>
Social	Tecnológico
<ul style="list-style-type: none"> ● En lo que va de este año, han salido más de 5 noticias a nivel nacional referentes al tema de estudio. ● La revolución tecnológica que se ha dado en estos últimos años ha alcanzado y transformado profundamente a la sociedad. El saber programación será fundamental para aquellos que quieran salir adelante y obtener una carrera exitosa en la actual era digital. Tener 	<ul style="list-style-type: none"> ● La plataforma Training Center UFPS, fue desarrollada por el grupo de estudio de Programación Competitiva del Semillero SILUX, durante el año 2017, con el propósito de integrar los 5 aspectos importantes que se deben tener en cuenta para preparar al estudiante en las Competencias de Programación (Sesiones de entrenamiento, material de apoyo, repositorio de soluciones, competencias oficiales y metodología

<p>conocimientos acerca de tecnologías ayuda a un profesional para una mejor toma de decisiones y una ejecución más eficiente de su trabajo.</p> <ul style="list-style-type: none"> ● La tecnología ofrece una gran cantidad de beneficios a las personas que lo aprenden (empleo, viajes y formación) y considerando la situación de la ciudad de Cúcuta como frontera, se puede aprovechar de esta ventaja para formar programadores que puedan generar grandes ideas de emprendimiento. ● Al motivar a los estudiantes a competir en maratones de programación, se presenta una gran oportunidad para eliminar brechas sociales y reducir la desinformación alrededor de lo que es la programación de computadores, esto por medio de estrategias más divertidas y dinámicas. 	<p>de trabajo) (Lázaro et al., 2016). Entre las funcionalidades que diferencia esta plataforma de otros Jueces Online, se encuentran: (1) la posibilidad de subir ejercicios propios, de nivel fácil a difícil, con tiempos de ejecución personalizados, así como (2) poder subir el material de apoyo (videos, documentos y presentaciones) necesarios, y acorde se va explicando las temáticas en cada una de las sesiones; también permite (3) hacer seguimientos a las competencias que se ejecutan.</p> <ul style="list-style-type: none"> ● Como la tecnología avanza a pasos agigantados, es importante que Training Center reciba constante mantenimiento y sea modificada de manera permanente; las estadísticas observadas y en esta plataforma (mostradas más adelante en el marco teórico) indican el
--	--

<ul style="list-style-type: none">● La UFPS ya se ha involucrado en diferentes programas como Apps.co, el punto Vive Digital o las Hackatones y es importante que eso se haga extensivo a toda la región ya que hay una responsabilidad social universitaria.● Participar en maratones de programación ofrece la experiencia de compartir y competir con los mejores programadores de diferentes ciudades y países, permitiendo así no solo conocer nuevos puntos de vista, sino hacer nuevas conexiones con otras instituciones y posteriormente poder organizar eventos y realizar diferentes proyectos.	<p>alto uso que ella recibe diariamente, tanto de la UFPS como de otras universidades a nivel de la región nortesantandereana.</p> <ul style="list-style-type: none">● Ya existen herramientas que facilitan la enseñanza de la programación desde temprana edad, como los lenguajes de programación por bloques (la más popular, Scratch), el muy clásico lenguaje de programación Logo, y los movimientos apoyados por entidades gubernamentales como La Hora del Código. Se puede tomar alguna de estas alternativas para motivar y facilitar aún más la enseñanza de la programación competitiva en los colegios, de la mano de Training Center.● Si se motiva al estudiante a participar en competencias de programación, el podrá obtener conocimientos avanzados con
---	--

	<p>respecto a la resolución y abstracción de problemas y en un futuro puede especializarse y emplearse en otras áreas como la Inteligencia Artificial o el IOT.</p> <ul style="list-style-type: none">● También es importante resaltar que una de las metas futuras que se establecieron durante el desarrollo del marco de trabajo del grupo de estudio, fue mantener todo ese esfuerzo y trabajo realizado con el paso del tiempo, y para ello, es fundamental que las tecnologías creadas y que están al alcance de la mano (en este caso Training Center), sean de gran ayuda y se acoplen a mayor población, para que muchos puedan conocer el mundo de la programación y sus múltiples beneficios mencionados reiteradamente.
Ecológico	Legal

- Salvo por los esfuerzos de las multinacionales como Google, Amazon, IBM, etc, por la Computación Verde (Green Computing) no aplica para este proyecto. En ese sentido, Training Center opera en la nube (Cloud Computing) y así se planea mantenerlo en el presente proyecto.

- El Plan de Desarrollo del Gobierno Nacional 2018-2022 (PND, 2018) establece la necesidad de enseñar a programar en los colegios.
- El Plan Nacional Decenal de Educación de Colombia 2016-2026, mencionando uno de los desafíos estratégicos a cumplir, reitera que el uso continuo de las TIC de manera pedagógica, van ser fundamentales en la enseñanza, en la construcción de conocimiento, el aprendizaje, la investigación y la innovación (PNDE, 2016).
- El Documento CONPES 3975, motiva y presenta una política para impulsar la productividad y favorecer el bienestar de la comunidad, a través del uso estratégico de las tecnologías digitales tanto en el sector público como en el

	<p>privado, y así, poder enfrentar los desafíos relacionados con la Cuarta Revolución Industrial (Márquez et al., 2019).</p>
--	--

2.4.2. Análisis de justificación geográfico

Tabla 2 Justificación tomando como punto de vista el ámbito geográfico. Fuente: elaboración propia según literatura revisada y citada en este documento

Mundial	Latinoamericano
<ul style="list-style-type: none"> ● Según datos de la Olimpiada Internacional de Informática (IOI), edición 2019, participaron un total de 335 estudiantes de Instituciones educativas de más de 88 países (IOI, 2019). ● El movimiento “La hora del código” se realiza anualmente en más de 180 países alrededor del mundo (HOUR 	<ul style="list-style-type: none"> ● Varias organizaciones a nivel latinoamericano han apoyado el esfuerzo de enseñar programación competitiva hacia las instituciones educativas, como es caso de la Fundación Sadosky en Argentina (Fundación Sadosky, 2019) y el juez online en español “¡Acepta el reto!”.

<p>OF CODE, 2015) y es enormemente apoyado, tanto por entidades gubernamentales de todos estos países, como por distintivas celebridades.</p>	<ul style="list-style-type: none"> ● Participación de 7 países a nivel latinoamericano (Argentina, Bolivia, Colombia, Brasil, Chile, México y Venezuela) en la pasada Olimpiada Internacional de Informática, sumando un total de 23 estudiantes (IOI, 2019).
<p>Nacional</p>	<p>Local</p>
<ul style="list-style-type: none"> ● Según palabras del presidente de la República, Iván Duque Márquez, para el año 2020 se requieren 150.000 programadores (<i>Goyeneche, 2019</i>). ● En la pasada Olimpiada Internacional de Informática, Colombia logró participar con un total de 3 estudiantes (IOI, 2019). ● El año pasado, Colombia ocupó el segundo lugar en la reciente Maratón 	<ul style="list-style-type: none"> ● En estos últimos años, el desempeño de los estudiantes de las universidades de la región en las diferentes competencias de programación ha sido bastante bueno, colocando el nombre de la ciudad de Cúcuta en alto. ● La plataforma Training Center UFPS ha sido usada por diferentes universidades de la región nortesantandereana, tanto para entrenar, como para hacer uso de su

<p>Regional Latinoamericana de Programación (ICPC, 2019)</p> <ul style="list-style-type: none"> • Tal como se menciona en el Plan TIC 2018-2022, el gobierno colombiano desea impulsar la transformación digital sobre las distintas actividades de administración pública, mediante el uso y desarrollo de soluciones de software. 	<p>material educativo sobre programación y algoritmos.</p>
<p>La UFPS</p>	
<ul style="list-style-type: none"> • El grupo de estudio en Programación Competitiva de la UFPS (fundado en el año 2015) proporciona un espacio de aprendizaje y constante entrenamiento, con el objetivo de buscar e ir mejorando el desempeño de sus integrantes en las diferentes competencias de programación a nivel nacional, regional y mundial. <p>Para la preparación de los estudiantes universitarios en las competencias oficiales organizadas anualmente por ACM e ICPC, el grupo de estudio realiza sesiones de entrenamiento, y socialización de problemas y temáticas, además de una maratón oficial cada año, con el ánimo de preparar a los estudiantes y mejorar continuamente las habilidades para la resolución de problemas por medio de la programación.</p>	

Sin embargo, el grupo de estudio se enfrenta al constante desafío de mantenerse. Se ha evidenciado que pocos estudiantes de primeros semestres se vinculan al grupo de estudio, a diferencia de estudiantes de semestres superiores. Una de las metas que se quieren lograr en el grupo, es que haya mayor cantidad de estudiantes que se quieran vincular de manera temprana, ya que, además de lo mencionado anteriormente, esto también permite que los estudiantes logren mejores resultados en las competencias (algo que ha sido clave para otras instituciones como la Universidad Nacional o la Universidad de los Andes). En consecuencia, se hace evidente la necesidad de enseñar programación competitiva a nivel escolar, para que cuando los jóvenes inicien sus estudios universitarios, se vinculen al grupo de estudio con el conocimiento necesario.

- Otro beneficio que ofrece la promoción de la programación competitiva en las instituciones educativas de la ciudad de Cúcuta, es proveer a los estudiantes la oportunidad de prepararse de mejor forma para competencias a nivel de secundaria, como las Olimpiadas de Informática, las cuales no se han realizado en esta ciudad y cuya dinámica es bastante similar a las competencias oficiales de programación a nivel universitario.
- Por último, si la presente propuesta no se lidera desde la UFPS, muy seguramente, otros grupos o semilleros de investigación realizarán este trabajo, como anteriormente mencionado, hay una urgente necesidad de programadores; por eso es

muy importante aprovechar el momento actual, las tendencias y la oportunidad, para este tipo de iniciativas.

2.5. Objetivos

2.5.1. Objetivo general

- Desarrollar una extensión de la plataforma Training Center UFPS para promover la Programación Competitiva en colegios de la ciudad de Cúcuta.

2.5.2. Objetivos específicos

- Hacer una revisión sobre las plataformas y lenguajes de programación usadas en colegios y en programación competitiva para seleccionar el lenguaje más apropiado a usar en colegios de la ciudad de Cúcuta.
- Identificar los requerimientos y adaptaciones de la Plataforma Training Center UFPS para su utilización en colegios de la ciudad de Cúcuta.

- Realizar el análisis, diseño, implementación, pruebas y despliegue necesarios para que la plataforma Training Center UFPS se pueda usar en colegios de la ciudad de Cúcuta.
- Ejecutar una prueba piloto de uso de la plataforma Training Center UFPS en colegios de la ciudad de Cúcuta siguiendo el Marco de Trabajo del Semillero SILUX-

2.6. Alcance y delimitación

2.6.1. Alcance

- El presente proyecto se realizará como una extensión de la plataforma Training Center UFPS, del programa de Ingeniería de Sistemas de la Universidad Francisco de Paula Santander. Con el objetivo de seguir el trabajo hecho por el grupo de estudio de Programación competitiva, por medio de la plataforma Training Center, pero ahora dirigido a estudiantes de algunos colegios de la ciudad de Cúcuta.
- La interfaz gráfica de usuario (GUI) se adaptará para los estudiantes de Colegio, por lo que la plataforma tendrá dos vistas diferentes, una para Colegios y otra para Universidad.

Preliminarmente, la GUI de colegios debe permitir realizar entrenamientos de programación de una manera simple y divertida.

- Para este proyecto se reutilizará la arquitectura actual de la Plataforma Training Center, el código fuente y el repositorio y se mantendrá la misma licencia. Además, se hará una actualización de versiones de tecnologías con respecto a la versión actual.

- La prueba piloto se realizará en colegios de la ciudad de Cúcuta.

- Como entregables de este proyecto, se presentarán:
 - La plataforma Training center modificada, integrando un nuevo módulo para colegios, con el nuevo lenguaje de programación seleccionado.
 - Un manual para estudiantes y profesores-coach de colegio.
 - Una divulgación del proyecto (póster o ponencia).

- Este proyecto no pretende enseñar a programar a los estudiantes, sino motivarlos a que se involucren en el mundo de la programación competitiva. En consecuencia, los colegios de la prueba piloto serán seleccionados de manera que se tengan conocimientos previos de programación básica.

2.7. Cronograma

Teniendo en cuenta los objetivos del proyecto y las fases de la metodología a utilizar, se identifican las siguientes actividades y entregables:

Tabla 3 Actividades del proyecto. Fuente: Elaboración propia.

Objetivo	Fase	Actividad	Entregable
Hacer una revisión sobre las plataformas y lenguajes de programación usadas en colegios y en programación competitiva.	1	Hacer el análisis acerca de los lenguajes de programación útiles y tecnologías que se ajusten a los objetivos del proyecto, (lenguajes de bloques, tecnologías abiertas).	Documento acerca de las tecnologías seleccionadas para adaptar Training Center a los colegios.
		Realizar la elección y apropiación de las tecnologías a utilizar para la implementación de la plataforma “Training center School”.	

	4	<p>Documentar la implementación de las herramientas escogidas dentro de la plataforma.</p>	<p>Documentación de la integración de Training Center a las tecnologías escogidas en este proyecto.</p>
		<p>Registrar la información relevante encontrada para el proyecto.</p>	
<p>Identificar los requerimientos y adaptaciones de la Plataforma Training Center UFPS para su utilización en colegios de la ciudad de Cúcuta.</p>	1	<p>Análisis preliminar de la arquitectura de Training Center.</p>	<p>Documento con la descripción de las tecnologías ya utilizadas.</p>
		<p>Apropiación del funcionamiento interno y tecnologías ya utilizadas en Training center.</p>	
		<p>Recopilar requerimientos para la implementación de la extensión de Training</p>	<p>Documento de Requerimientos del sistema</p>

		Center de acuerdo a los objetivos establecidos.	
	4	Registrar los requerimientos funcionales y no funcionales del sistema.	Documento de Requerimientos del Sistema.
		Documentar las adaptaciones que va a necesitar Training Center.	Documento de Gestión de cambios en Training Center.
Realizar el análisis, diseño, implementación, pruebas y despliegue necesarios para que la plataforma Training Center UFPS se pueda	2	Realizar los mockups de las interfaces necesarias.	Documento de análisis y diseño del sistema
		Desarrollar las interfaces gráficas para la integración de las tecnologías seleccionadas junto a Training Center	
		Realizar los ajustes necesarios a	

usar en colegios de la ciudad de Cúcuta.	Training Center para la integración de las tecnologías a utilizar en programación para colegios.	
	Implementación de las tecnologías elegidas sobre la extensión de la plataforma Training center.	
	Realizar las pruebas de cada una de las funcionalidades implementadas.	Documento de pruebas de Software.
	Despliegue de la Plataforma Training Center con los ajustes Realizados.	URL de la plataforma Desplegada con los cambios y adaptaciones realizadas.

	4	Hacer el análisis y diseño de la extensión a implementar.	Documento de Análisis y Diseño del Sistema.
		Realizar registro documental de las pruebas realizadas al software.	Documento de Pruebas de Software.
Ejecutar una prueba piloto de uso de la plataforma Training Center UFPS en colegios de la ciudad de Cúcuta siguiendo el Marco de Trabajo del Semillero SILUX.	3	Seleccionar los colegios donde se realizará la prueba piloto.	Documento acerca del desarrollo de la prueba piloto.
		Realizar la prueba piloto por medio de una maratón de entrenamiento, utilizando la extensión desarrollada, y el marco de trabajo del Semillero SILUX, en cada uno de los colegios.	
	4	Documentar el desarrollo de la prueba	

		piloto en cada uno de los colegios.	
		Realizar un manual para estudiantes sobre la extensión de Training Center para colegios.	Manual de usuario de “Training Center High School”.

Tabla 4 Cronograma del proyecto. Fuente: Elaboración propia

Fase	Actividad	Mes 1				Mes 2				Mes 3				Mes 4				Mes 5				Mes 6			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
	Análisis preliminar de la arquitectura de Training Center.	■	■																						
	Apropiación del funcionamiento interno y tecnologías ya	■	■	■	■																				

	Training Center para la integración de las tecnologías a utilizar en programación para colegios.																									
	Implementación de las tecnologías elegidas sobre la extensión de la plataforma Training center.																									
	Realizar las pruebas de cada una de las funcionalidades implementadas.																									
	Despliegue de la Plataforma Training Center con																									

los ajustes Realizados.																									
Seleccionar los colegios donde se realizará la prueba piloto.																									
Realizar la prueba piloto por medio de una maratón de entrenamiento, utilizando la extensión desarrollada, y el marco de trabajo del Semillero SILUX, en cada uno de los colegios.																									
Documentar la implementación de las herramientas																									

<p>escogidas dentro de la plataforma.</p>																									
<p>Registrar la información relevante encontrada para el proyecto.</p>																									
<p>Registrar los requerimientos funcionales y no funcionales del sistema.</p>																									
<p>Documentar las adaptaciones que va a necesitar Training Center.</p>																									
<p>Hacer el análisis y diseño de la extensión a implementar.</p>																									

Tablas. Se incluyen los siguientes ítems:

Tabla 5 Presupuesto Total del Proyecto. Fuente: Elaboración Propia

PRESUPUESTO GLOBAL DEL TRABAJO DE GRADO							
RUBROS	FUENTE DE FINANCIACIÓN						TOTAL
	Estudiante		UFPS		Externa		
Gastos de Personal	\$	14.400.000	\$		\$		\$ 14.400.000
Materiales y Suministros	\$	1.200.000	\$		\$		\$ 1.200.000
Descripción de equipos e infraestructura.	\$		\$	3.594.000	\$		\$ 3.594.000
Divulgación del proyecto en eventos	\$		\$	2.000.000	\$		\$ 2.000.000
Total	\$	15.600.000	\$	5.594.000	\$	\$	21.194.000

Tabla 6 Presupuesto: Descripción de los gastos de personal. Fuente: Elaboración Propia.

DESCRIPCIÓN DE LOS GASTOS DE PERSONAL									
Nombre Estudiante	valor unitario / hora	Dedicación (hrs / semana)	FUENTE DE FINANCIACIÓN						SUBTOTAL (Semestral)
			Estudiante	UFPS	Externa				
Angie González	5.000/hora	40 (\$200.000)	\$	800.000 /mes	\$		\$		\$ 4.800.000
Crisel Ayala	5.000/hora	40 (\$200.000)	\$	800.000 /mes					4.800.000
Director	20.000/hora	10 (\$200.000)	\$	800.000 /mes					4.800.000
Total		84 horas	\$	2.400.000	\$		\$		14.400.000

Tabla 7 Presupuesto: Descripción de materiales y suministro. Fuente: Elaboración Propia.

DESCRIPCIÓN DE MATERIALES Y SUMINISTROS		
MATERIAL	FUENTE DE FINANCIACIÓN	SUBTOTAL (Semestral)



	Estudiante		UFPS		Externa		
Papelería	\$	30.000/mes	\$		\$		\$ 180.000
Servicio de Internet	\$	105.000/mes	\$		\$		\$ 630.000
Energía eléctrica	\$	40.000/mes	\$		\$		\$ 240.000
Material Adicional	\$	30.000/mes	\$		\$		\$ 150.000
Total	\$	205.000/mes	\$		\$		\$ 1.200.000

Tabla 8 Presupuesto: Descripción de los equipos e infraestructura Fuente: Elaboración Propia.

DESCRIPCIÓN DE EQUIPOS E INFRAESTRUCTURA							
DESCRIPCIÓN DE EQUIPOS E INFRAESTRUCTURA	FUENTE DE FINANCIACIÓN						SUBTOTAL (Semestral)
	Estudiante		UFPS		Externa		
Back-end como servicio en la Nube (Google Cloud o similar, todo el despliegue se realiza directamente en la nube, sin servidores físicos) x 6 meses. Se anexa en la figura 14 la estimación	\$		\$	349.000 COP (97.09 USD) / mes	\$		\$ 2.094.000

mensual en la calculadora de Google Cloud.							
Estación de trabajo (arriendo)	\$	300.000/mes	\$		\$		\$ 1.500.000
Total						\$	3.594.000

Compute Engine


1 x  

730 total hours per month

VM class: regular

Instance type: n1-standard-4

Region: iowa

[Sustained Use Discount](#): 30% 

[Effective Hourly Rate](#): USD 0.133

Estimated Component Cost: USD 97.09 per 1 month

Total Estimated Cost: USD 97.09 per 1 month

Estimate Currency


USD - US Dollar 

Ilustración 1 Calculadora cloud de Google. Fuente: Calculadora Google Cloud

3. Programación competitiva

La Programación Competitiva “es un deporte mental en el cual los participantes resuelven un conjunto de problemas bien especificados a través de programas de computador”. Esto permite que los estudiantes puedan mejorar sus destrezas en programación y en la resolución de problemas en general (Reyes, 2016). La programación competitiva permite a los estudiantes mejorar sus habilidades de resolución de problemas y programación.

Con el motivo de hacer efectiva la programación competitiva se implementan diferentes estrategias, como se detalla a continuación:

- Redes Sociales Académicas que permiten trabajo colaborativo y desarrollo de proyectos de manera colaborativa entre estudiantes y profesores de diversos lugares del mundo. Estas redes se consolidan con el tiempo y se unen a las ligas nacionales, regionales y mundiales, de manera similar a las ligas de los deportes. A través de estas redes se desarrollan actividades como:
 - Entrenamientos especializados o Training Camp, donde se reúnen varios equipos a compartir conocimientos y experiencias y a entrenar como en un deporte.
 - Competencias, de entrenamiento y oficiales, organizadas por las ligas según cada nivel geográfico.
- Divulgación de manera abierta en repositorios de código fuente y de documentos.
 - Divulgación de problemas y soluciones.

- Divulgación en libros y artículos académicos.
- Plataformas tecnológicas, que permiten articular todas las estrategias en línea, aprovechando las ventajas de las tecnologías.
- Reclutamiento de talentos, por parte del sector productivo y universidades que buscan investigadores a nivel de posgrado.

3.1. ICPC

De todas estas estrategias, la primera ha dado sus frutos y ya son cuarenta y cinco (45) años de la red mundial ICPC (International Collegiate Programming Contest) que reúne a todos los equipos del mundo, mediante relacionamiento de universidades y agremiaciones de Ciencias de la Computación, Ingeniería de Software, Sistemas de Información, Ingeniería de TI y, para el caso de Colombia, Ingeniería de Sistemas.

En Latinoamérica ICPC tiene una mesa directiva con representación de todos los países y en Colombia ICPC se lidera desde la Asociación Colombiana de Ingenieros de Sistemas, en articulación con la Red de Decanos de Ingeniería de Sistemas y afines (Redis).

ICPC inició como una competencia internacional de programación para estudiantes universitarios y evolucionó luego a una gran organización que funciona como la FIFA en fútbol y ATP en tenis. ICPC tiene sus raíces en 1970, cuando la primera competencia fue organizada por pioneros del Capítulo Alfa de la Sociedad de Honor de Ciencias de la Computación de la UPE. La iniciativa se extendió rápidamente en los Estados Unidos y Canadá como un programa

innovador para aumentar la ambición, la aptitud para resolver problemas y las oportunidades de los estudiantes más fuertes en el campo de la informática (ICPC, 2022).

Con el tiempo, este concurso evolucionó a ser una competencia de varios niveles y se realizó la primera ronda de campeonato en 1977. Desde ese momento el concurso se expandió hasta llegar a instancias mundiales, en las que universidades de diferentes partes del mundo organizan competencias regionales, las cuales permiten a los equipos avanzar a la competencia mundial que realiza cada año el ICPC World (Ilustración 2).



Ilustración 2 Mapa regiones mundiales de ICPC. Fuente: ICPC (<https://icpc.global/regionals/finder>).

Como se aprecia en la figura, en cada continente o subcontinente hay regiones ICPC consolidadas:

- Región Norteamérica: Compuesto por 12 zonas
- Europa
- Eurasia del norte
- África y arabia
- Asia Occidental
- Asia Oriental
- Asia pacifico
- América Latina

En el caso de Colombia, esta pertenece a la región de Latinoamérica Norte, junto a Venezuela y Ecuador (Ilustración 3).



Ilustración 3 Región América latina. Fuente: ICPC (<https://icpc.global/regionals/finder/sa>)

Las competencias ICPC estandarizaron mundialmente la programación competitiva, las cuales consisten en equipos de 3 estudiantes, quienes representan a su universidad. Estos estudiantes deben resolver entre 10 y 15 diferentes problemas que involucran ciencias básicas,

ciencias de la computación, dominio de tecnologías y lenguajes de programación y mucha creatividad e ingenio. En esta competencia se fomenta el trabajo en equipo, la creatividad, innovación y la capacidad de desenvolverse bajo presión. Además, son competencias que usan el inglés como idioma universal, por lo que en Latam y Colombia contribuyen a promover el bilingüismo.

En Colombia ICPC se ha estructurado en dos grandes ligas: la liga CCPL (Colombian Collegiate Programming League) concentrada en Bogotá y RPC (Red de Programación Competitiva) que se lidera por profesores de varias universidades y realiza competencias mensuales donde participan equipos desde México hasta Argentina. La UFPS es parte de RPC desde el año 2017 y administra en la nube la plataforma tecnológica, además de apoyar las diversas actividades que se realizan.

Considerando el modelo mundial, latinoamericano y nacional, la UFPS constituyó un Grupo de Estudio de Programación Competitiva, como parte del Semillero SILUX. Dicho grupo se ha logrado consolidar en los últimos seis años, llegando a ocupar el top 6 a nivel nacional, destacándose y convirtiéndose en un referente nacional.

3.2. Grupo de estudio del semillero SILUX UFPS

El grupo de estudio en Programación Competitiva de la UFPS (fundado en el año 2015) proporciona un espacio de aprendizaje y constante entrenamiento, con el objetivo de buscar e ir mejorando el desempeño de sus integrantes en las diferentes competencias de programación a nivel nacional, regional y mundial.

Para la preparación de los estudiantes universitarios en las competencias oficiales organizadas anualmente por ACM e ICPC, el grupo de estudio realiza sesiones de entrenamiento, y socialización de problemas y temáticas, además de una maratón oficial cada año, con el ánimo de preparar a los estudiantes y mejorar continuamente las habilidades para la resolución de problemas por medio de la programación.

Cada año el grupo de estudio se enfrenta al constante desafío de mantenerse, por cuanto las reglas de las competencias tienen límites de semestre y edad. En los inicios del semillero, identificó que pocos estudiantes de primeros semestres se vinculan al grupo de estudio, a diferencia de estudiantes de semestres superiores.

En consecuencia, una de las metas del grupo es que haya mayor cantidad de estudiantes que se quieran vincular de manera temprana, ya que, además de lo mencionado anteriormente, esto también permite que los estudiantes logren mejores resultados en las competencias (algo que ha sido clave para otras instituciones como la Universidad Nacional o la Universidad de los Andes).

En los últimos dos años se ha logrado aumentar la cantidad de estudiantes de primeros semestres y se ha avanzado en el reto de vincularse con los colegios, lo cual es parte de las propuestas de este trabajo de grado.

3.3. IOI

Siguiendo la idea anterior de vincularse a los colegios, se hace necesario revisar la Olimpiada Internacional de Informática (IOI). IOI es una prestigiosa competencia informática para estudiantes de secundaria y preparatoria de todo el mundo. Esta competencia es una de las cinco Olimpiadas científicas internacionales que se realizan anualmente y funciona de manera similar a ICPC.

La idea de iniciar las olimpiadas internacionales fue propuesta en la 24° Conferencia General de la UNESCO, en octubre de 1987, y su primera realización fue en Bulgaria (1989). La competencia está impulsada por Acer y respaldada por el Ministerio de Educación (MOE) de Singapur, la Oficina de Convenciones y Exposiciones de Singapur (SECB) y la Autoridad de Desarrollo de Medios de Infocomm (IMDA). (IOI,2021).

El objetivo principal del IOI es generar interés por la informática (ciencias de la computación) y también reunir a alumnos talentosos de diferentes países para que se integren y compartan experiencias científicas y culturales (IOI, 2022).

3.3.1. Contexto Mundial

El IOI es organizado anualmente en y por uno de los países participantes en la competencia. Cada país participante se encarga de enviar una delegación de cuatro concursantes y dos adultos que van como acompañantes. Los estudiantes durante dos días de competencia compiten individualmente resolviendo una serie de problemas informáticos para lograr acumular el máximo puntaje posible. Los demás días se organizan eventos y actividades culturales y recreativas.

Aunque los problemas de la competencia son algorítmicos, los participantes del concurso deben también demostrar habilidades informáticas como análisis de problemas, diseño de algoritmos, estructuras de datos, programación y pruebas.

Los ganadores del IOI pertenecen a los mejores jóvenes informáticos del mundo.

Con antelación a este concurso los participantes que llegan a él se preparan y cada país realiza diferentes pruebas clasificatorias para definir los estudiantes que participarán en la competencia mundial de la IOI.

3.3.2. Contexto Latinoamericano

La participación en el IOI no está abierta directamente a las personas, sino que cada país participante envía un equipo de cuatro estudiantes y la selección se coordina a través de un concurso nacional.

Cada país tiene su propio proceso de selección. A continuación, algunos de los concursos nacionales en cada país:

México:

La Olimpiada Mexicana de Informática (OMI) es un concurso a nivel nacional para jóvenes que tienen facilidad, habilidad y voluntad para resolver problemas prácticos mediante la lógica y el uso de computadoras, este concurso busca promover el desarrollo tecnológico en México y encontrar a los mejores programadores, los cuales pasan a hacer parte de la selección mexicana para participar en las Olimpiadas Internacionales de Informática (IOI) (*Olimpiada Mexicana De Informática, 2022*).

España:

La Olimpiada Informática Española es un concurso individual de programación algorítmica para estudiantes de educación secundaria, bachillerato y grado medio que se celebra anualmente. El premio principal para los ganadores es ser invitados a participar en la Olimpiada Informática Internacional (OIE, 2022).

La OIE es la final nacional de la olimpiada informática. Es posible clasificarse por dos vías:

- Olimpiada informática regional: Se clasifican entre uno y dos concursantes de cada olimpiada regional.

- Concurso clasificatorio abierto (online): Se clasifican los 10 mejores concursantes de toda España.

Como premio principal, los cuatro mejor clasificados en la final nacional forman la selección nacional que representa a España en la Olimpiada Internacional de Informática (IOI).

Los ganadores que proponga cada olimpiada regional se clasifican directamente a la final de la Olimpiada Informática Española.

Argentina:

OIA - Olimpíada Informática Argentina

Objetivos

El Programa O.I.A. Tiene como objetivos contribuir a la educación de los jóvenes fomentando su participación en actividades que demanden mayor preparación y superación en conocimientos de informática y, a través de esta disciplina en el rol del ser humano y en el desarrollo de las ciencias. También pretende detectar jóvenes que demuestran talento y aptitudes en este campo y apoyarlos en su futura formación. Es otro de sus objetivos, el contribuir a la capacitación y actualización docente fomentando el intercambio de conocimientos y experiencias con sus alumnos (OIA, 2022).

Chile:

Olimpiada Chilena de Informática

La OCI es una competencia de programación orientada a estudiantes de secundaria que pretende despertar el interés en la Computación, permitiendo que los participantes desarrollen habilidades de abstracción, creatividad, algoritmos, pensamiento sistemático y resolución de problemas, y comprendan la naturaleza científica de la disciplina. La OCI busca en los colegios a los potenciales talentos y los motiva para que participen en las actividades de capacitación y práctica a nivel regional en talleres organizados por los miembros de la organización de la OCI, con miras a las competencias regionales. Los mejores de cada región participan en la final nacional de la Olimpiada, y los participantes con el mejor desempeño a nivel nacional reciben una preparación intensiva. Cuatro de ellos ganarán el derecho a representar a Chile en la Olimpiada Internacional de Informática IOI (OCI, 2022).

República dominicana:

La Olimpiada Dominicana de Informática (ODI) es una competencia algorítmica y de programación dirigida a estudiantes de educación media que persigue principalmente estimular el estudio de las Ciencias de la Computación y la Informática.

La ODI es organizada por la SOCIEDAD DOMINICANA DE COMPUTACIÓN y a través de ella se establece el equipo dominicano que representa anualmente al país en la Olimpiada Internacional de Informática (IOI, por sus siglas en inglés International Olympiads in Informatics), una de las competencias de Ciencias de Computación más prestigiosas a nivel mundial (ODI, 2022).

Los objetivos de la ODI son:

1. Descubrir, estimular, incluir, retar y reconocer a jóvenes dominicanos que son excepcionalmente talentosos en el área de la Computación y Ciencias en general.
2. Dirigir la atención de los jóvenes dominicanos a la disciplina de la informática.
3. Promover el uso y aplicación de la informática como herramienta para la investigación y resolución de problemas.
4. Fomentar una actitud activa y crítica en los docentes frente a su rol de educadores.
5. Formar la selección dominicana para la competencia mundial International Olympiads in Informatics (IOI).

La competencia es individual y consiste en resolver cuatro problemas en cualquiera de los siguientes lenguajes: C, C++, Java, Python 2, Python 3 y Pascal. La ODI consiste en dos fases. La primera es una competición en línea abierta a todos los estudiantes de educación media hábiles. No más de veinte (20) estudiantes clasifican automáticamente a la segunda fase. En esta Final ODI se escoge el equipo dominicano que participará en la IOI.

3.3.3. Colombia

La Olimpiada Colombiana de Computación tiene tres rondas: Prueba Clasificatoria, Prueba Semifinal y Ronda Final. En la primera prueba se evalúan conocimientos y habilidades

básicas en un lenguaje de programación tal como C, Pascal o Basic. En las siguientes pruebas ya no solamente se pide el manejo de un lenguaje, sino también saber usarlo para resolver problemas retadores (OCC, 2022).

Niveles

Con el propósito de garantizar una participación equitativa para los estudiantes de secundaria, se toman las pruebas en dos niveles diferentes. En el Primer Nivel pueden participar los estudiantes de hasta grado noveno. En el Superior los de décimo y undécimo.

Participación en eventos internacionales

Los mejores participantes son invitados a entrenamientos en enero y en junio con el objetivo de participar en los siguientes eventos internacionales: Olimpiada Internacional de Informática, Competencia Iberoamericana de Informática por Correspondencia y las Rondas Abiertas de la Olimpiada Norteamericana de Computación. Los dos últimos se realizan en diferentes países y tanto las pruebas como las soluciones son enviadas a través de correo electrónico. El primero es presencial y se realiza cada año en un país diferente.

3.4. Propuesta Training Center High School

Tomando como punto de partida el reto de lograr consolidar modelos como la IOI e ICPC en la región y la experiencia de seis años y evolución exitosa del semillero SILUX, se tomó el Marco de Trabajo propuesto por Delgado & Lázaro (2018) y se propuso una extensión para vincularse con colegios y extenderse a otras instituciones.

La siguiente ilustración muestra los principales enfoques del marco de trabajo del semillero SILUX, propuesto por Delgado & Lázaro (2018) y usado con éxito, al punto de ubicarse top 6 a nivel nacional y clasificar todos los años a competencias latinoamericanas y aumentar la cantidad de equipos y estudiantes.

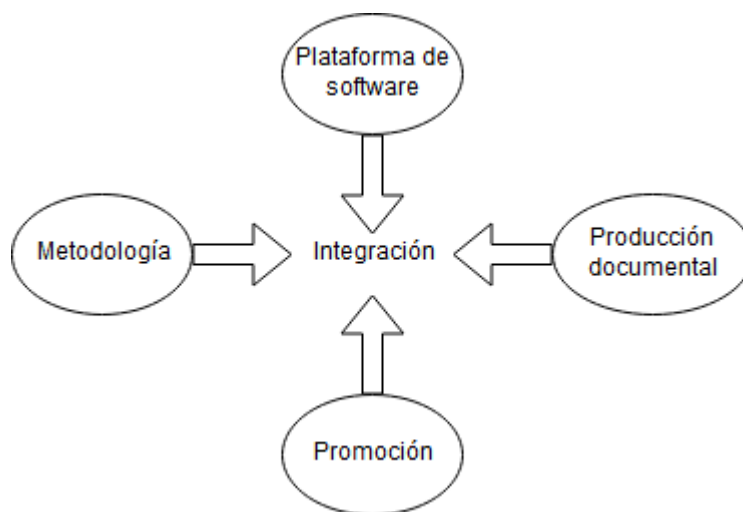


Ilustración 4 Marco de Trabajo del Semillero SILUX (Delgado & Lázaro 2018)

A continuación, se da un breve resumen de cada eje de este marco de trabajo propuesto, que se ha venido implementando en cada semestre académico:

*Tabla 9 Ejes del marco de trabajo del grupo de estudio en Programación Competitiva UFPS.
Fuente: Tesis Delgado & Lázaro 2018*

Eje	Descripción
Eje 1: Promoción	El objetivo del primer eje del marco de trabajo es fomentar la participación de los estudiantes en las competencias de programación para continuar representando a la universidad a nivel nacional e internacional.
Eje 2: Metodología	La metodología de trabajo del grupo define la forma de trabajar y la forma de llevar a cabo todas las actividades. Este eje estructura todo el trabajo a realizar y define cómo se orientará el grupo a lo largo del tiempo.
Eje 3: Producción documental	El tercer eje del marco de trabajo tiene como objetivo construir un banco documental que abarque todos los contenidos que se trabajan durante el semestre y que sirva para evidenciar las actividades realizadas.
Eje 4: UFPS Training Center	Todas las actividades desarrolladas en el grupo de estudio tendrán un punto de encuentro a través de una plataforma web

	llamada UFPS Training Center (Plataforma tecnológica). En este eje del proyecto se busca desarrollar dicha plataforma para realizar un seguimiento del trabajo realizado dentro del grupo de estudio.
Eje 5: Integración del Marco de Trabajo	El quinto eje del proyecto tiene como objetivo diseñar un plan de trabajo para integrar todos los componentes desarrollados de manera tal que sea replicable en el futuro.

En este marco de trabajo la Plataforma tecnológica Training Center ha sido un componente muy importante, por lo cual este proyecto se propuso extender la plataforma para colegios. Training Center UFPS fue utilizada por primera vez en abril del 2018, y desde esa fecha ha sido un componente que ha acompañado todas las actividades del Grupo de estudio de Programación Competitiva UFPS, ya que esta herramienta cuenta con el entorno de retroalimentación para que un estudiante pueda prepararse, estudiar, practicar, competir y mejorar sus habilidades tanto en programación competitiva como en otras áreas afines al programa de estudios de Ingeniería de Sistemas.

Otra ventaja que posee esta plataforma es la posibilidad de acceder al material multimedia y documental sobre los temas dictados en las charlas, ejercicios de práctica y un modo de aprendizaje guiado a las diferentes categorías de ejercicios, haciendo seguimiento del desempeño de los estudiantes durante este proceso de aprendizaje.

Este trabajo realizado por medio del marco de trabajo y su plataforma “Training Center UFPS” ha tenido grandes resultados y también se ha podido identificar estrategias y posibles medios de obtener aún mejores resultados. Es por eso que se propuso este proyecto, con el fin de empezar la preparación que se realiza hoy con el Semillero de Programación competitiva, pero desde una etapa más temprana de los estudiantes. Lo anterior con el propósito de obtener mejores resultados en las competencias de programación, ya que se observó por otros casos que una preparación temprana produce mejores resultados.

Por lo tanto, en este proyecto se propuso desarrollar una extensión de la ya existente plataforma “Training Center” enfocada para estudiantes de colegio, cambiando los complejos lenguajes de programación por un lenguaje de bloques como “Blockly”.

Si bien el propósito inicial del proyecto era meramente tecnológico, fruto de las iteraciones del proyecto, de las pruebas y retroalimentación y de la información obtenida, el director del Semillero SILUX, profesor Milton Jesús Vera Contreras, propuso el siguiente marco actualizado, el cual se encuentra en formulación, en fase inicial.



Ilustración 5 Marco de Trabajo del semillero SILUX actualizado

Descripción marco de trabajo actualizado

1. **Plataforma tecnológica:** Es el Core del proyecto y como tal la base de este marco de trabajo. La plataforma se actualizó por medio de una extensión a la misma, con el objetivo de llevar la programación competitiva a nivel de colegios. Por esta razón la plataforma ahora cuenta con la opción de resolver problemas en lenguaje de bloques el cuál es un lenguaje de programación más apropiado para estudiantes de colegio que deseen empezar a desarrollar o reforzar habilidades en programación.
2. **Repositorio**

- 2.1. Repositorio de material educativo: La plataforma contiene material educativo para las diferentes categorías de problemas con los que cuenta la plataforma. Este material puede ser agregado o actualizado por administradores o tutores, con el fin de resolver las necesidades que se presenten.
 - 2.2. Repositorio de problemas: La plataforma es alimentada por un banco de problemas a resolver, los cuales son agregados por administradores o tutores.
3. **Mecanismos de comunicación y divulgación (redes sociales):** La plataforma dentro de sí, actualmente iniciará manejando un foro que está disponible para cada problema (en lenguaje de bloques), con el fin de resolver dudas de dicho problema. Por otra parte, dentro del marco de trabajo se emplean herramientas de comunicación y divulgación como lo son Facebook y Discord para mantener informados a los integrantes, acerca de eventos realizados y a realizar.
4. **Mecanismo de colaboración y trabajo en red (roles, planes de trabajo, etc.):** Actualmente la metodología implementada es por medio de reuniones semanales, en las cuales se imparten diferentes temas. Para esto se designan diferentes estudiantes los cuales crean el material y preparan el tema para compartirlo en cada una de estas sesiones, realizando también pequeñas maratones como práctica de lo impartido.
5. **Mecanismos de motivación y promoción (ranking, incentivos, etc.):** Como modelo de motivación se ha venido trabajando con diferentes mecanismos para motivar a los

estudiantes y para promocionar la programación competitiva. Algunos de ellos ha sido el ranking que se maneja dentro de Training Center, el cual se basa en puntuaciones acumuladas por cada estudiante dependiendo de su trabajo realizado (cantidad de problemas resueltos), quienes más puntaje tengan se mantienen en los primeros lugares del ranking. El ranking se ha usado como herramienta para definir que estudiantes representan la universidad en las competencias nacionales de programación competitiva y también se ha usado para definir qué estudiantes ganan el privilegio de participar en diferentes campamentos de programación competitiva nacionales e internacionales. Para colegios se piensa en proponer a los profesores de colegios la realización de maratones de programación con ayuda de empresas patrocinadoras para poder proporcionar incentivos a los estudiantes, tales como camisetas y todo el tema logístico de la maratón. Con el objetivo de proporcionar estos espacios que motiven a los estudiantes a competir e involucrarse en la programación competitiva.

6. **Otros mecanismos específicos según el contexto (ejemplo: física y química):** Usar el marco de trabajo para resolver problemas de un área en específico. En un colegio el profesor podrá usar la plataforma como herramienta para que sus estudiantes resuelvan problemas de su área en específico como física o química.

7. Actividades

7.1. Sesiones explicativas y apoyo (teaching, mentoring, coaching): El marco de trabajo plantea y ejecuta actualmente sesiones semanales explicativas y de apoyo de

diferentes temas relevantes en programación competitiva. También se realiza material explicativo de las diferentes categorías que sirvan de apoyo permanente para cada estudiante.

7.2. Competencias de entrenamiento: Dentro de cada sesión realizada semanalmente se realizan o plantean competencias de entrenamiento, donde se proporcionan una serie de problemas a resolver en determinado tiempo.

7.3. Competencias oficiales: El trabajo anteriormente mencionado realizado bajo el marco de trabajo tiene como fin, la participación en las diferentes competencias oficiales. Iniciando con la maratón de programación interna la cual se instituyó como opción clasificatoria para la maratón nacional de programación. La maratón nacional busca la clasificación a la maratón regional latinoamericana. Teniendo equipos de representación en la maratón regional, el objetivo es poder obtener una clasificación a la maratón mundial que se realiza cada año en un lugar del mundo diferente. A nivel de colegios se desea que los estudiantes de colegios puedan llegar a participar de las competencias oficiales de la IOI, las cuales son dirigidas a estudiantes de colegio.

Este marco de trabajo busca que los estudiantes permanezcan en constante formación, crecimiento y evolución en lo referente a programación competitiva. Ahora con el objetivo de implementarlo a nivel de colegios para tener una preparación temprana y llegar a obtener mejores resultados en las diferentes competencias oficiales.

Tal como se indicó previamente, el foco de este proyecto es tecnológico, pero no es suficiente, se requiere método y voluntad de personas y organizaciones. En ese sentido, se adoptó la propuesta anterior como un Marco de Trabajo actualizado y se avanzó con el desarrollo del software, en cumplimiento de los objetivos del proyecto.

Corresponde a los profesores de Universidad y Colegio y a la Universidad e Instituciones de Educación básica y media, seguir mejorando la propuesta anterior.

4. Redes, plataformas y lenguajes de programación para colegios y programación competitiva

Aunque se puede desarrollar software desde cero, guiados por la imaginación e intuición, por las necesidades y requerimientos y el contexto, esto se queda en arte y técnica o hasta artesanía. Desde un enfoque de ingeniería es clave partir de un conocimiento científico o al menos más formal.

En consecuencia, antes de cualquier desarrollo de software es necesario revisar qué existe en el mundo, bien sea para reutilizar o para mejorar o superar.

Según la revisión realizada, existen diversas plataformas y lenguajes de programación con énfasis en colegios y programación competitiva. No es posible ser exhaustivos, por lo cual la revisión se limitó a las ligas oficiales ICPC e IOI, en cada región geográfica. A continuación, el detalle de hallazgos:

Tabla 10 Redes, Plataformas y Lenguajes.

REDES	PLATAFORMAS	LENGUAJES
RPC	Codeforces	Blockly
ICPC	Beecrowd	Diagramas de flujo
CODE.ORG	BOCA	PSEUDOCÓDIGO
CCPL	ACEPTA EL RETO	Python
	GUAPA	PHP
	VIRTUAL JUDGE	C
	GREENFOOT	JAVASCRIPT

	ALICE	UNITY
	SCRATCH	DFD
	MINECRAFT	VISUAL BASIC
	TYNKER	LENGUAJE DE ARDUINO
		PSEINT
		PC ²
	BYOB/SNAP! [26], MIT App Inventor [1], and Waterbear [6].	

4.1. Plataformas

A continuación, se presentan algunas plataformas que apoyan o basan su enfoque en la enseñanza de la programación para instituciones educativas:

Codeforces

Es una red social dedicada a concursos de programación competitiva. Codeforces es mantenida por un grupo de programadores de la Universidad ITMO liderados por Mike

Mirzayanov. En 2018 registró 600,000 usuarios registrados. Es una página utilizada por los mejores programadores competitivos como Gennady Korotkevich Petr Mitrichev, Benjamin Qi y Makoto Soejima además de ser considerado "demandado" para los desarrolladores de software (CODEFORCES, 2022).

La plataforma Codeforces es usada para concursos de programación competitiva. Cuenta con diferentes tipos de concurso los cuales son:

- "Codeforces Rounds" hechas una vez a la semana. Los cuales son concursos que duran alrededor de 2 horas
- Concursos educacionales con duración de 2 - 2.5 horas
- Resolución de problemas de rondas o concursos anteriores con propósitos de entrenamiento.

Por medio de un sistema de puntuación Elo los usuarios son evaluados y clasificados.

Los usuarios son clasificados en rangos en base a sus puntos usuarios con un puntaje mayor a 1900 son clasificados como "Primera División" y usuarios con un puntaje menor a 1900 son clasificados como "Segunda División", la "Tercera división" fue creada para los usuarios con un puntaje menor a 1600.

BEECROWD

Anteriormente llamado URI Online Judge. El 24 de octubre de 2021, se informó ampliamente que el juez en línea URI entró en una nueva fase y cambió su nombre a Beecrowd.

Es una comunidad global de desarrolladores comprometidos con seguir evolucionando como estudiantes y profesionales.

La plataforma de programación competitiva tiene más de 2,000 pruebas analíticas y de codificación disponibles en más de 20 lenguajes de programación diferentes. Todas las pruebas están disponibles en portugués e inglés.

Beecrowd Academic es un módulo exclusivo para Instituciones Educativas, Profesores y Entrenadores. Aquí usted puede crear cursos, listas de ejercicios y realizar un seguimiento del progreso de sus estudiantes dándoles comentarios en tiempo real (Beecrowd, 2022).

BOCA

Es un juez virtual usado para el juzgamiento automático de las soluciones propuestas por los equipos participantes en las maratones de programación organizadas por las Redes de programación RPC y CCPL.

ACEPTA EL RETO

Es un almacén ("repositorio") de problemas de programación en español, con un juez en línea incorporado. Cualquier usuario puede resolver los problemas propuestos y enviar su solución al juez para comprobar si es correcta. Este Juez en línea acepta soluciones en C, C++ y java (Acepta el reto, 2022).

El objetivo es que los usuarios pongan en práctica sus habilidades de programación básica, de algoritmos y estructuras de datos. Por tanto, ninguno de los problemas hace uso de interfaces gráficas de usuario, ni ficheros.

GUAPA

Es un Juez online nacido del Grupo de Algoritmia Avanzada y Programación Competitiva de la FES Acatlán de la UNAM, Hecho por/para estudiantes que desean profundizar en conocimientos matemáticos y de programación.

VIRTUAL JUDGE

Virtual Judge no es un juez en línea real. Puede tomar problemas de otros jueces en línea regulares y simular envíos a otros jueces en línea. Su objetivo es permitir la celebración de concursos cuando no tienes los datos de prueba (VIRTUAL JUDGE, 2022).

GREENFOOT

Es un software educativo visual e interactivo, diseñado para hacer del aprendizaje de programación algo fácil y divertido (Kölling et al., 2022). Las herramientas de visualización e interacción están integradas en el entorno.

Greenfoot enseña orientación a objetos con Java. Maneja actores los cuales están programados en código Java textual estándar, lo que brinda una combinación de experiencia de programación en un lenguaje tradicional basado en texto con ejecución visual.

ALICE

Alice es un innovador entorno de programación basado en bloques que facilita la creación de animaciones, la creación de narraciones interactivas o la programación de juegos sencillos en 3D. A diferencia de muchas de las aplicaciones de codificación basadas en rompecabezas, Alice motiva el aprendizaje a través de la exploración creativa. Alice está diseñado para enseñar habilidades de pensamiento lógico y computacional, principios fundamentales de programación y para ser una primera exposición a la programación orientada a objetos. El Proyecto Alice proporciona herramientas y materiales complementarios para la enseñanza utilizando Alice a través de un espectro de edades y temas con beneficios comprobados en la participación y retención de grupos diversos y desatendidos en la educación informática (Carnegie Mellon University, 2008).

SCRATCH

Scratch es una herramienta que “ayuda a los jóvenes a aprender, a pensar de forma creativa, a razonar sistemáticamente, y a trabajar de forma colaborativa — habilidades esenciales para la vida en el siglo 21” (SCRATCH, 2019).

El Lenguaje Scratch, fue diseñado para facilitar el aprendizaje de la programación, por medio de secuencias animadas que brinda un modelo de programación visual con diferentes elementos.

El manual de Scratch afirma que cuando se trabaja con Scratch se comprenden fácilmente conceptos matemáticos e informáticos que están muy bien integrados en el programa, como son: Los procesos interactivos (bucles), Los criterios condicionales (si, entonces, si-no), Las coordenadas en un plano, Las variables, etc. Y este conocimiento es adquirido de una manera que genera motivación. Ya que no es igual aprender en un contexto de aprendizaje tradicional, que usando herramientas visuales que facilitan el aprendizaje (Scratch, 2019).

MINECRAFT EDUCATIVO

Minecraft es un juego programable que podemos modificar mediante la programación. Se trata de un juego flexible y creativo, lo cual motiva mucho a los niños, y todos estos estímulos se pueden aprovechar para aprender a programar, entre otras muchas cosas. Para entender el amplio abanico de posibilidades educativas que ofrece: en Minecraft podemos trabajar tanto conceptos de electrónica básica con Redstone, como la creatividad de los alumnos al construir su propio mundo, pasando por el aprendizaje de la programación utilizando el lenguaje de programación JavaScript. (*Minecraft, Un Videojuego Educativo Para Aprender a Programar / Escuela De Programación, Robótica Y Pensamiento Computacional*, 2020)

Con este juego se puede trabajar por proyectos, fomentar el aprendizaje autónomo del alumno, trabajar la resolución de problemas creando retos que los alumnos tengan que superar e impulsar el trabajo en equipo.

Para aprovechar el aspecto educativo del juego y lograr objetivos pedagógicos, en 2016 se lanzó una versión especial. Esta versión es más restrictiva que el juego original, pero posee más funciones para que los profesores incluyan sus contenidos de asignaturas como matemáticas, ciencias, lenguas extranjeras o historia.

La versión educativa de Minecraft es una gran herramienta para trabajar en proyectos STEAM en el aula y fortalecer la resolución de problemas, creatividad, experimentación, pensamiento crítico, imaginación e ingenio, entre otras habilidades importantes para los niños.

TYNKER

Tynker impulsa la creatividad de más de 60 millones de estudiantes y sirve a miles de escuelas y educadores en todo el mundo (Tynker, 2013).

Maneja un aprendizaje interactivo basado en historias el cual permite que niños y adolescentes aprendan los conceptos básicos con sencillos desafíos de codificación basados en bloques antes de pasar a lenguajes basados en texto del mundo real como JavaScript y Python.

4.2. Redes

A continuación, se presentan algunas de las principales redes de competencias y aprendizaje de programación:

RPC

La Red de Programación Competitiva es la red de programadores más grande de Latinoamérica. La cual permite que las personas exploten sus habilidades algorítmicas mediante el trabajo en equipo. Su objetivo es llevar la Programación Competitiva a todas las instituciones educativas y universidades de Latinoamérica.

ICPC

El Concurso Internacional de Programación Universitaria es un concurso de programación algorítmica para estudiantes universitarios. Equipos de tres, en representación de su universidad, trabajan para resolver la mayoría de los problemas del mundo real, fomentando la colaboración, la creatividad, la innovación y la capacidad de desempeñarse bajo presión. A través del entrenamiento y la competencia, los equipos se desafían entre sí para elevar el nivel de lo posible. Sencillamente, es el concurso de programación más antiguo, grande y prestigioso del mundo (ICPC, 2022)

Code.org

Es una organización sin fines de lucro dedicada a ampliar el acceso a las ciencias de la computación en las escuelas y aumentar la participación de las mujeres jóvenes y estudiantes de otros grupos no suficientemente representados.

Su visión es que cada estudiante, en cada escuela, tenga la oportunidad de aprender ciencias de la computación de la misma manera que aprende biología, química o álgebra. Code.org es el encargado de organizar la campaña anual La Hora del Código, en la que participa el 10% de los estudiantes de todo el mundo (CODE.ORG, 2022).

La Hora del código

Es un movimiento global sin fines de lucro, liderado por la ONG norteamericana code.org, que está teniendo gran impacto en más de 180 países en 34 idiomas con un objetivo común: ayudar a los estudiantes de educación básica, secundaria y media a entender la programación computacional como un saber básico para desempeñarse como ciudadanos del siglo XXI. Code.org es una organización sin fines de lucro dedicada a expandir el acceso a la informática en las escuelas y a aumentar la participación de las mujeres y las minorías subrepresentadas. Su visión es que cada estudiante en cada escuela debe tener la oportunidad de aprender informática, al igual que la biología, la química o el álgebra (HOUR OF CODE, 2015).

CCPL

Colombian Collegiate Programming League. CCPL, a diferencia de RPC, es una red de programación que reúne equipos de distintas universidades colombianas para los entrenamientos que se realizan cada mes (CCPL, 2022).

4.3. Lenguajes

A continuación, se explican algunos de los principales lenguajes que se usan o tienen enfoque principal en aprendizaje básico de la programación:

Blockly

Blockly es una biblioteca de código abierto que facilita agregar programación visual basada en bloques a una aplicación. Está diseñado para ser flexible y admite un amplio conjunto de características para diferentes aplicaciones. Se ha utilizado para programar personajes animados en una pantalla; creando guiones de historias; robots de control; e incluso generar documentos legales. Pero Blockly no es en sí mismo un lenguaje; Los desarrolladores que usan Blockly crean sus propios lenguajes de bloque. Cuando los desarrolladores crean una aplicación usando Blockly, deben considerar cuidadosamente el estilo, qué bloques usar y qué API y características de lenguaje son adecuadas para su audiencia (Pasternak et al., 2017).

Diagramas de flujo

El diagrama de flujo o flujograma o diagrama de actividades es la representación gráfica de un algoritmo o proceso. Se utiliza en disciplinas como programación, economía, procesos industriales y psicología cognitiva.

Un diagrama de flujo es un diagrama que describe un proceso, sistema o algoritmo informático. Se usan ampliamente en numerosos campos para documentar, estudiar, planificar, mejorar y comunicar procesos que suelen ser complejos en diagramas claros y fáciles de comprender.

Pseudocódigo

Es una forma de expresar los distintos pasos que va a realizar un programa, de la forma más parecida a un lenguaje de programación. Su principal función es la de representar por pasos la solución a un problema o algoritmo, de la forma más detallada posible, utilizando un lenguaje cercano al de programación. El pseudocódigo no puede ejecutarse en un ordenador ya que entonces dejaría de ser pseudocódigo, como su propio nombre indica, se trata de un código falso (pseudo = falso), es un código escrito para que lo entienda el ser humano y no la máquina. (*Qué Es Pseudocódigo*, 2019)

Python

Python es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de *software*, la ciencia de datos y el *machine learning* (ML). Los desarrolladores

utilizan Python porque es eficiente y fácil de aprender, además de que se puede ejecutar en muchas plataformas diferentes (AWS, 2022).

PHP

Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

C

Es un lenguaje de programación (considerado como uno de los más importantes en la actualidad) con el cual se desarrollan tanto aplicaciones como sistemas operativos a la vez que forma la base de otros lenguajes más actuales como Java, C++ o C#.

JavaScript

Es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones de primera clase. Si bien es más conocido como un lenguaje de scripting (secuencias de comandos) para páginas web, y es usado en muchos entornos fuera del navegador, tal como Node.js, Apache couchdb y Adobe Acrobat javascript es un lenguaje de programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa (por ejemplo programación funcional) (MDN, 2022).

UNITY

Es una herramienta de desarrollo de videojuegos creada por la empresa Unity Technologies.

Unity es una herramienta que no engloba únicamente motores para el renderizado de imágenes, de físicas de 2D/3D, de audio, de animaciones y otros motores, sino que engloba además herramientas de networking para multijugador, herramientas de navegación navmesh para Inteligencia Artificial o soporte de Realidad Virtual (Martin, 2022).

DFD

Un diagrama de flujo de datos (DFD) traza el flujo de la información para cualquier proceso o sistema. Emplea símbolos definidos, como rectángulos, círculos y flechas, además de etiquetas de texto breves, para mostrar las entradas y salidas de datos, los puntos de almacenamiento y las rutas entre cada destino.

Visual Basic

Visual Basic es un lenguaje de programación orientado a objetos desarrollado por Microsoft. El uso de Visual Basic agiliza y simplifica la creación de aplicaciones .NET con seguridad de tipos (Microsoft, 2022).

Arduino

El lenguaje de programación de Arduino se puede dividir en tres grandes partes: estructura, valores (variables y constantes), y funciones.

- Funciones: Para controlar la tarjeta Arduino y realizar cálculos.
- Variables: Arduino tipos de datos y constantes.
- Estructura: Los elementos del código de Arduino (C++).

PC²

El sistema de control de concursos de programación, pronunciado "P-C-squared" o, a veces, simplemente "P-C-Two" para abreviar. Es un sistema de software diseñado para admitir operaciones de concursos de programación en una variedad de entornos informáticos. PC² permite a los concursantes enviar programas a través de una red a los jueces del concurso. Los jueces pueden volver a compilar el programa enviado, ejecutarlo, ver el código fuente y/o los resultados de la ejecución y enviar una respuesta al equipo. El sistema también es compatible con un modo de "evaluación automatizada" en el que la evaluación la realiza un software en lugar de jueces humanos (The PC² CCS, 2022).

Pseint

Pseint es un intérprete de pseudocódigo para estudiantes de programación de habla hispana. Su propósito principal es ser una herramienta para aprender y comprender los conceptos

básicos de programación y aplicarlos con un pseudocódigo en español de fácil comprensión (Pseint, 2022).

Al realizar el estudio de todas las herramientas disponibles para ser usadas en este proyecto, se decidió elegir Blockly como la herramienta más acorde para cumplir los objetivos planteados. Ya que es una herramienta de software libre, la cual cuenta con una amplia documentación y es de fácil entendimiento. Lo que permite que su implementación sea más sencilla al tener una clara explicación de su funcionamiento e integración. Además, que cuenta con un sitio web que permite crear bloques personalizados y la librería permite crear bloques propios para ser implementados según la lógica necesaria.

Otra de sus características principales es que cuenta con una serie de bloques que hacen operaciones básicas tales como: sumas, restas, creación de variables, listas etc.

También permite convertir ese código de bloques a lenguajes ejecutables como Python o PHP y al mismo tiempo proporciona la libertad de los bloques personalizados que se crean puedan ser convertidos al lenguaje que se desee, lo cual será explicado a detalle más adelante.

Todas estas redes, plataformas y lenguajes se relacionan entre sí. En algunos casos hay redes que crean su propio lenguaje y plataforma, por lo cual se confunden. A este respecto, este trabajo adopta el planteamiento teórico del Trabajo de Grado sobre Redes Sociales Académicas

de (Pájaro & Romero, 2022). En dicho trabajo se prioriza la red social académica y, aunque la plataforma es clave y muy importante, la tecnología es un artefacto y lo relevante es el trabajo en red.

Finalmente, y dado lo anterior, conviene revisar las iniciativas colaborativas desde sector gobierno y organizaciones no gubernamentales, tal como se presenta a continuación.

4.4. Programa Misión Tic 2022 Del Gobierno Nacional



Ilustración 6 Presentación Misión TIC 2022. Fuente: Ministerio de las TIC.

MISIÓN TIC 2022 es una estrategia de formación específica en programación orientada a jóvenes y ciudadanía en general, para ampliar las oportunidades en la construcción de trayectorias educativas y ocupacionales, garantizando el aprovechamiento de las oportunidades en el marco de las políticas de reactivación económica y economía naranja.

MISIÓN TIC en agosto del 2020 lanzó un pilotaje el cual tuvo gran respuesta por parte de todos los colombianos, alcanzando más de 128 mil inscritos. La cantidad de cupos inicial era de 2500, pero al ver el interés de los colombianos, se amplió a más de 5000.

Este programa de formación cuenta con dos rutas de aprendizaje:

La ruta de aprendizaje 1 va dirigida a estudiantes de grados 10° y 11° de establecimientos educativos oficiales del país. Los estudiantes podrán escoger entre fundamentos de programación y tecnologías digitales e industrias creativas digitales (MINTIC, 2022).

La ruta de aprendizaje 2 va dirigida a los colombianos mayores de 15 años que cuenten con su título de bachiller y que se encuentren interesados en desarrollar habilidades en programación.

El programa Misión Tic tiene 5 ciclos de formación:

- Programación básica
- Fundamentos de programación
- Desarrollo de software
- Desarrollo de aplicaciones web
- Desarrollo de aplicaciones móviles

Además de Misión TIC, el ministerio tiene otras iniciativas enfocadas en niños y niñas y adolescentes. Una de estas iniciativas es Greentic que usa lenguajes de bloques y tecnologías como Arduino e internet de las cosas. Estas iniciativas de bloques se inspiran en el trabajo de países como Argentina y Uruguay, que lleva grandes avances en lo referente a Pensamiento Computacional y STEM, que son otras líneas de pensamiento y trabajo.

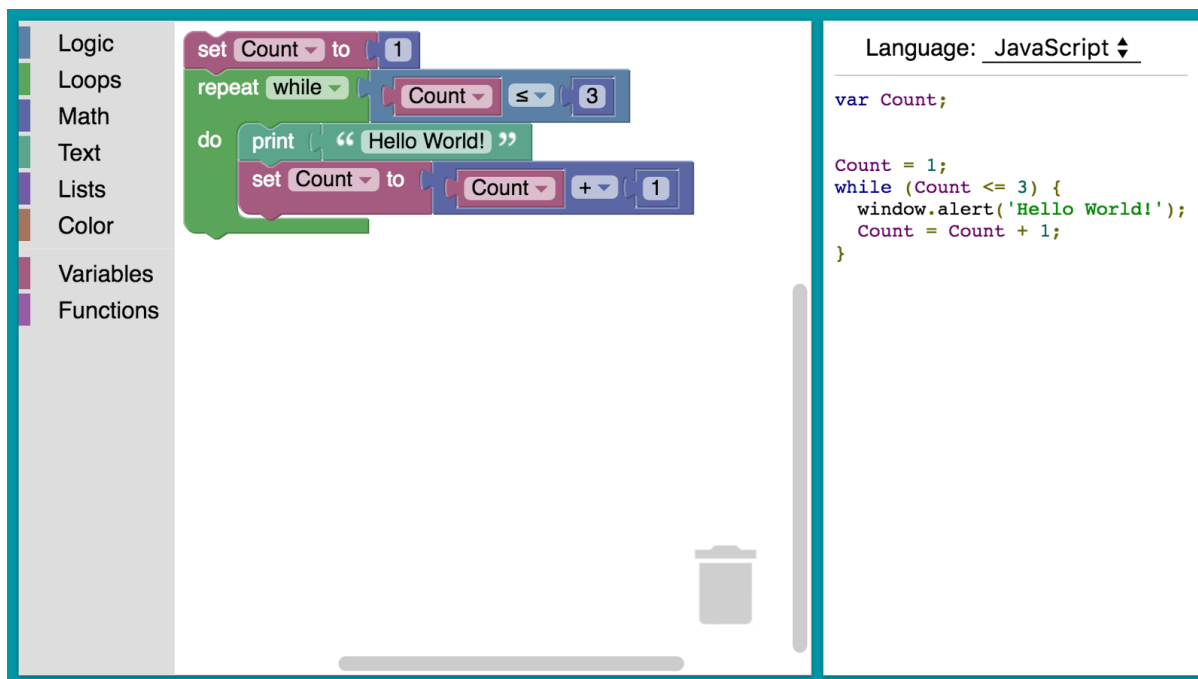
Manteniendo el foco tecnológico de este proyecto, conforme a los hallazgos previos, se avanza a la revisión del lenguaje de bloques Blockly.

4.5. Blockly

Como se dijo anteriormente, después de un análisis de diferentes herramientas y plataformas enfocadas en la enseñanza de la programación competitiva, y para estudiantes de instituciones educativas, se decidió escoger a Blockly como herramienta de enseñanza para colegios, esto hace parte de la implementación de esta nueva extensión para la plataforma UFPS Training Center de la mano del marco de trabajo del grupo de estudio.

Blockly es una biblioteca que agrega un editor de código visual a aplicaciones web y móviles. El editor de Blockly utiliza bloques gráficos entrelazados para representar conceptos de código como variables, expresiones lógicas, bucles, entre otros. Es un proyecto de Google y es software libre y de código abierto liberado bajo la Licencia de Apache 2.0. Blockly permite a los

usuarios aplicar principios de programación sin tener que preocuparse por la sintaxis o el uso de la línea de comandos (Blockly, 2022).



*Ilustración 7 Tablero de edición de Blockly. Fuente: Blockly
(<https://developers.google.com/Blockly?hl=es-419>)*

Como se puede observar en la ilustración 7, la biblioteca Blockly agrega un editor a la aplicación que representa los conceptos de codificación como bloques entrelazados. También genera código sintácticamente correcto en el lenguaje de programación de su elección. Esta librería también permite crear bloques personalizados para conectarse a su propia aplicación. Para el usuario final, Blockly es una forma intuitiva y visual de crear código.

Algunas de las principales características de Blockly son:

- **Código exportable:** Los usuarios pueden exportar sus programas basados en bloques a lenguajes de programación comunes (esta herramienta proporciona 5 lenguajes para exportar: Javascript, Python, PHP, Lua y Dart) y realizar una fácil transición a la programación basada en texto.
- **Código abierto:** Esta herramienta puede modificarse, ya que es de código abierto, y puede adaptarse, de acuerdo a las necesidades que tenga su implementación.
- **Extensible:** Blockly proporciona herramientas y tutoriales para que pueda ser modificado de acuerdo a las diferentes necesidades, agregando bloques personalizados, agregando código fuente exportable a esos nuevos bloques o eliminando bloques y funcionalidades innecesarias.
- **Altamente capaz:** Blockly Puede implementar tareas de programación complejas como calcular la desviación estándar en un solo bloque.

- Idiomas: Blockly ofrece traducción a más de 40 idiomas, para que sea ampliamente utilizado.

Proyectos que implementan Blockly

Actualmente Blockly está siendo usado por gran cantidad de proyectos, los cuales en su gran mayoría son educativos. Algunos de estos proyectos son App Inventor, CODE.ORG, Microsoft makecode, ozoblocly y Micro: bit.

Bloques de Blockly

Blockly incluye una gran cantidad de bloques predefinidos, desde funciones matemáticas hasta estructuras y lógica, pero también se puede personalizar para agregar más bloques nuevos. Para definir un nuevo bloque se requiere una definición de bloque y un generador; la *definición de bloque* describe la apariencia (interfaz de usuario) del bloque, mientras que el *generador* describe el proceso de convertir un bloque en código ejecutable (Blockly, 2022). Las definiciones y los generadores se pueden escribir en Javascript o se puede utilizar la *Herramienta para desarrolladores de bloques* (Ver ilustración 8).

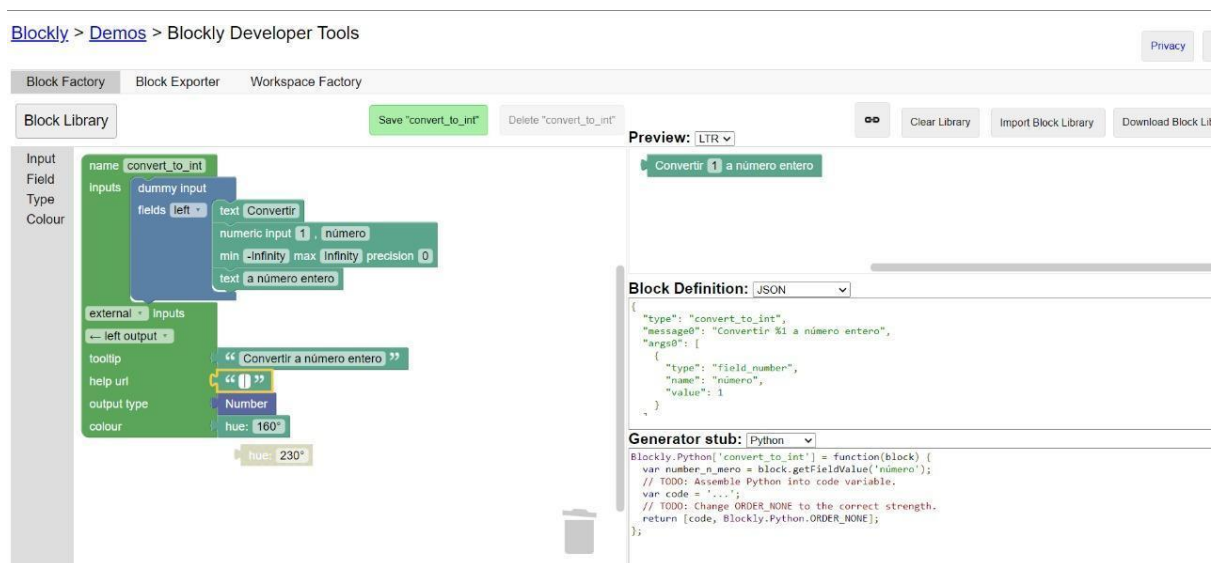


Ilustración 8 Herramienta para desarrolladores de Blockly. Fuente: Blockly (<https://developers.google.com/Blockly?hl=es-419>)

No sobra decir que la Herramienta para desarrolladores de Blockly también permite automatizar partes del proceso de configuración de Blockly, la compilación de la caja de herramientas y la configuración del espacio de trabajo o tablero de Blockly.

Generadores de código

Para el desarrollador, Blockly es una interfaz de usuario lista para crear un lenguaje visual que emite código generado por el usuario sintácticamente correcto. Blockly puede exportar bloques a muchos lenguajes de programación, como lo son:

- Javascript
- Python

- PHP
- Lua
- Dart

A su vez, Blockly permite integrar la opción de crear un generador a cualquier otro lenguaje de programación, por medio de objetos generadores de lenguaje. Esta es otra característica poderosa que ofrece esta herramienta. En el siguiente ejemplo se muestra la sintaxis de cómo se puede implementar un generador de código personalizado, tomado a partir de un bloque (sample_block) en un objeto generador de lenguaje (samplegenerator).

```
sampleGenerator['sample_block'] = function(block) {  
  return 'my code string';  
}
```

Ilustración 9 Ejemplo implementación de generador de código.

Implementación de Blockly en Training Center

Para implementar Blockly, en primer lugar fue necesario actualizar la versión de las herramientas y dependencias que componen la parte front-end (Aurelia.js), ya que el empaquetador no era compatible con esta librería; la actualización de este framework de desarrollo trajo la implementación de Webpack, un empaquetador de código fuente a una versión

más estándar soportada por la mayoría de los navegadores, esto permitió poder implementar todas las características que ofrece Blockly sin ningún inconveniente.

Para Training center, se personalizó la forma en que se listarán los bloques del tablero, por medio de categorías, separadas de la siguiente manera:

- Bloques de Entrada: Bloques para manejo de sentencias de entradas de texto o número, vía consola. Ver ilustración 10.



Ilustración 10 Bloques de entrada.

- Bloques de salida: Bloques para manejo de sentencias de salidas, vía consola. Ver ilustración 11.



Ilustración 11 Bloques de salida.

- Bloques de lógica: Bloques para manejo de sentencias de lógica booleana.



Ilustración 12 Bloques de lógica.

- Bloques de ciclos: Bloques para manejo de sentencias de bucles.



Ilustración 13 Bloques de ciclos.

- Bloques de Matemáticas: Bloques para manejo de sentencias de lógica matemática.



Ilustración 14 Bloques de matemáticas.

- Bloques de Texto: Bloques para manejo de cadenas de texto. Ver ilustración 15.

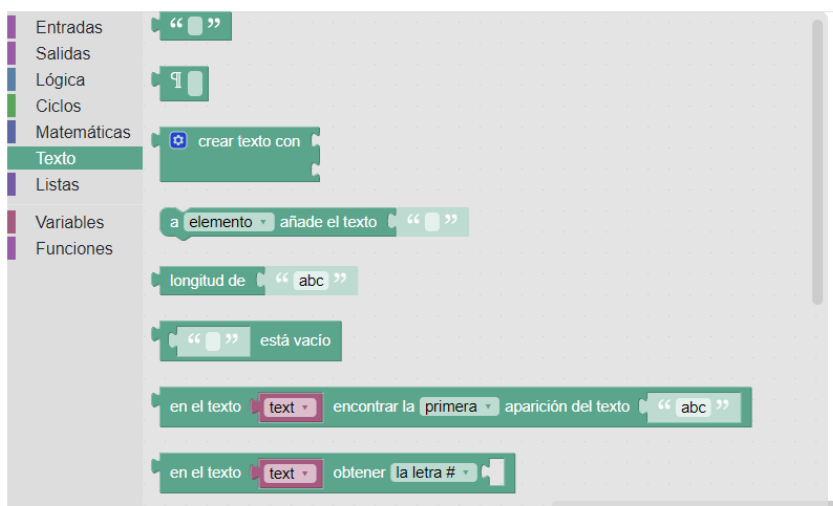


Ilustración 15 Bloques de texto.

- Bloques de Listas: Bloques para manejo de listas. Ver ilustración 16.



Ilustración 16 Bloques de listas.

- Bloques de variables: Bloques para creación y manejo de variables. Ver ilustración 17.



Ilustración 17 Bloques de variables.

- Bloques de funciones: Bloques para manejo e invocación de funciones.

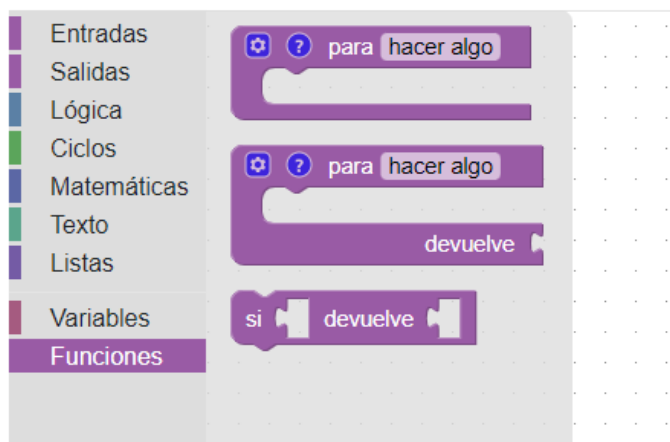


Ilustración 18 Bloques de funciones.

A su vez, también se implementaron bloques personalizados, que convierten a sintaxis de Python, y permitirán mitigar la ausencia de algunas funcionalidades para resolver ejercicios típicos de programación competitiva. En la siguiente ilustración se muestra la implementación de un bloque personalizado que convierte a números enteros desde números decimales.

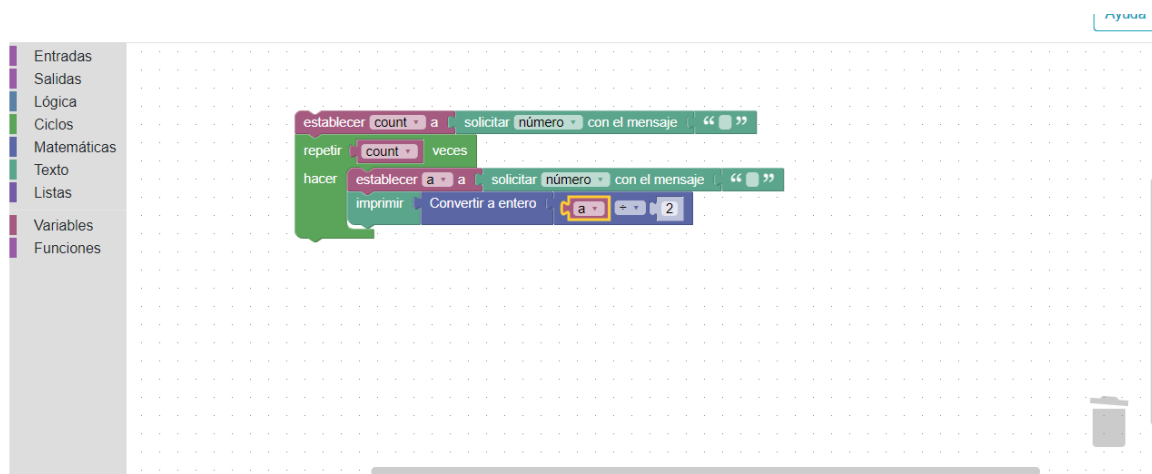


Ilustración 19 Implementación de bloques personalizados.

Para el presente proyecto, se utilizó el generador de código a Python, el cual, al momento de que el usuario cree una solución basada en sintaxis de bloques utilizando esta librería, el sistema deberá hacer la conversión a este lenguaje de programación y posteriormente, será enviado a su respectivo calificador que devolverá la respuesta, de acuerdo a los datos arrojados frente a la lista de los casos de prueba que el problema ya almacena. Esto hace que los diferentes componentes del sistema puedan ser reutilizados y que la arquitectura planteada durante la primera versión y posterior actualización de la plataforma sea escalable para ser utilizada en

otros ámbitos. Cabe mencionar que, en la sección de anexos, se adjunta un manual de uso más detallado del tablero de Blockly para Training Center, a disposición.

The screenshot displays the Blockly interface. At the top, there are two boxes: 'Entrada de ejemplo' (Example Input) containing the text '3', 'alfred', 'erick', and 'andres'; and 'Salida de ejemplo' (Example Output) containing 'Hola alfred', 'Hola erick', and 'Hola andres'. Below these is the text 'Ahora, intenta resolver este ejercicio' (Now, try to solve this exercise) and an 'Ayuda' (Help) button. The main workspace shows a script with the following blocks: 'establecer count a solicitar número con el mensaje " " ' (set count to request number with message " "), 'repetir count veces' (repeat count times), 'hacer' (do), 'establecer nombre a solicitar texto con el mensaje " " ' (set nombre to request text with message " "), 'imprimir' (print), 'crear texto con "Hola "' (create text with "Hola "), and 'nombre' (nombre).

Ilustración 20 Implementación de Blockly en Training Center.

Puntualizando, este proyecto persigue comenzar a vincularse con los colegios desde el semillero. Para tal efecto se propuso extender la Plataforma Training Center. En ese camino se identificó Blockly como la mejor alternativa, de tal forma que el resto del proyecto estará dedicado a cumplir el objetivo tecnológico y la prueba piloto con colegios.

5. Desarrollo del proyecto

5.1. Metodología del proyecto

Conforme a lo propuesto en el presente proyecto, se considerarán dos paradigmas de acuerdo a la Disciplina de los Sistemas de Información (ISD, por sus siglas en inglés Information Systems Discipline): el *paradigma de las ciencias del comportamiento*, que busca desarrollar teorías para explicar y predecir el comportamiento humano y organizacional en contextos mediados por las TIC (búsqueda de la verdad), y el *paradigma del diseño de la ciencia*, que busca ampliar las capacidades humanas y organizacionales a través de la creación o modificación de artefactos apoyados en las TIC (búsqueda de la utilidad) (R.Hevner et al., 2004) traducido y citado por (Vera Contreras & Peña Reyes, 2013).

De acuerdo a los paradigmas anteriormente mencionados, la investigación y la ingeniería consisten en un proceso iterativo de dos fases: “La primera fase sigue el paradigma de las ciencias del comportamiento, explicando y prediciendo los fenómenos relacionados con la necesidad organizacional. La segunda fase sigue el paradigma del diseño de la ciencia, creando y evaluando artefactos que contribuyan a subsanar la necesidad organizacional. Los resultados de la investigación pueden ser nuevos conocimientos o acciones en el mundo profesional, proporcionando un equilibrio entre el rigor y la relevancia.” (ver Figura 9) (Vera Contreras & Peña Reyes, 2013).

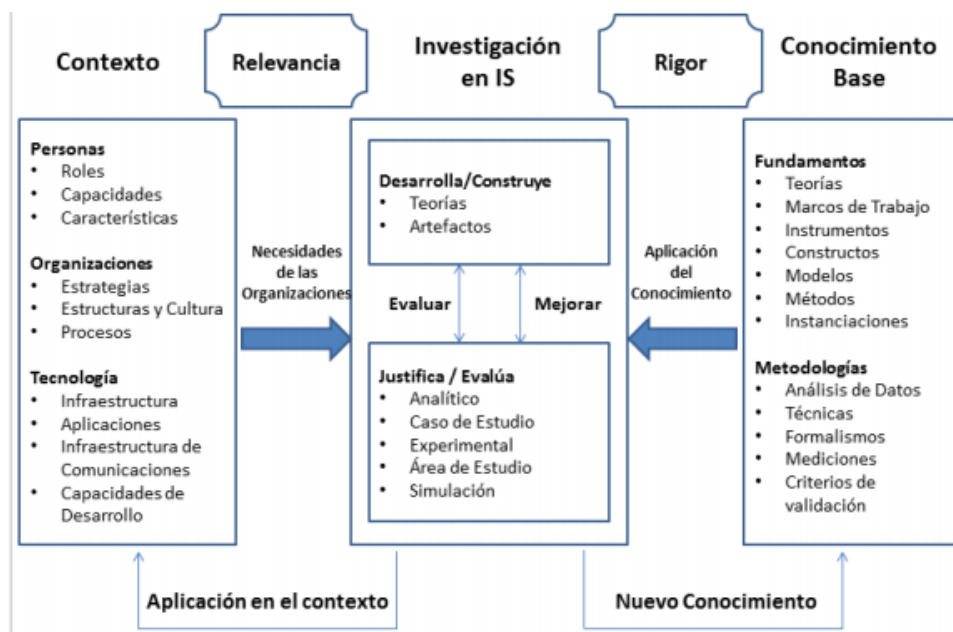


Ilustración 21 Marco de Investigación en ISD. Fuente: (R. Hevner et al., 2004)

Para que la aplicación del marco de Investigación en ISD sea exitoso y cumpla con los Paradigmas mencionados, es importante seguir los siete lineamientos que se resumen a Continuación:

Tabla 11 Lineamientos para aplicar el diseño de la ciencia. Fuente: (Vera Contreras & Peña Reyes, 2013).

Lineamiento	Descripción
1- Diseño como artefacto	El resultado de la investigación debe ser un artefacto o una instanciación de un artefacto.

2- Problema Relevante	El objetivo de la investigación debe ser desarrollar soluciones a Problemas relevantes en la práctica.
3- Evaluación	La utilidad, calidad y eficacia del artefacto debe demostrarse de Manera rigurosa.
4- Contribuciones a la Investigación	La investigación debe proporcionar contribuciones claras y Verificables.
5- Rigor de la Investigación	La investigación debe aplicar métodos rigurosos de construcción y evaluación del artefacto.
6- Diseño como un Proceso de investigación	La construcción y evaluación del artefacto es un proceso iterativo que satisface las normas y leyes de la disciplina al tiempo que logra el objetivo de desarrollar soluciones al problema.
7- Comunicación de la Investigación	Los resultados deben presentarse de manera efectiva tanto para los investigadores como para los profesionales.

Según lo dicho anteriormente, se hace evidente que no es trivial formular la metodología de un proyecto, ni para hacer investigación ni para hacer ingeniería. No obstante, sí resulta sencillo comprender que la metodología es el proceso que sigue el investigador o el ingeniero para conseguir sus objetivos. Dichos procesos deben ser rigurosos pero flexibles y debe ser explícito para garantizar la objetividad. Por lo tanto, aplicando los planteamientos del Marco de

Investigación en ISD, según la bibliografía citada previamente, para este proyecto se propone la siguiente metodología específica, representada gráficamente en la ilustración, y explicada a continuación:

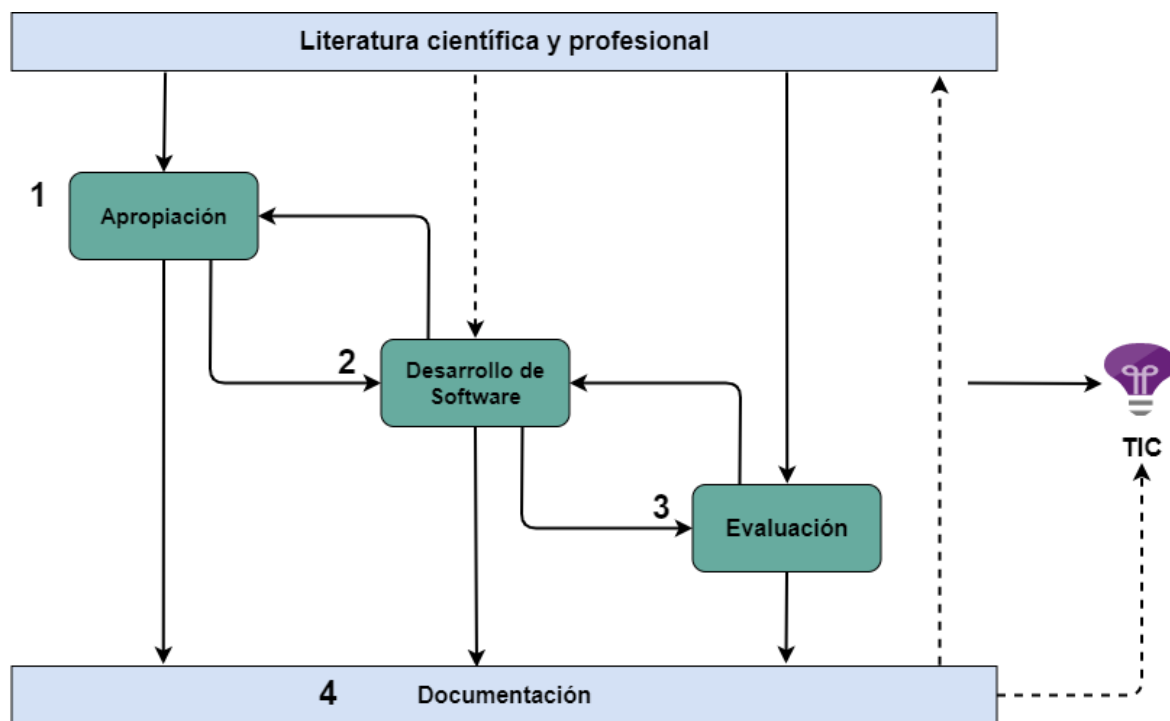


Ilustración 22 Fases de la metodología y objetivos del proyecto.

Esta metodología se divide en diferentes fases:

- La primera fase es la **apropiación** de conocimiento (revisión de literatura y bases teóricas). Aquí es muy importante tener un rigor de las fuentes de conocimiento, para poder comprender y responder: ¿qué está pasando? Y ¿por qué está pasando?

- La segunda fase es la de **experimentación**: Es ahí donde se aplica el conocimiento adquirido. Buscando responder a preguntas tales como: ¿cómo cambia la realidad? Y ¿por qué cambia la realidad?
- Después, en la tercera fase que es **evaluación** se verifica que el conocimiento obtenido es válido y repetible y que los cambios en la realidad ayudan a resolver el problema o satisfacer la necesidad.
- Finalmente, la cuarta fase es de **documentación**, en la cual se registran los resultados de las demás fases.

En la ilustración 23 se relaciona la metodología con los respectivos objetivos propuestos en este proyecto.

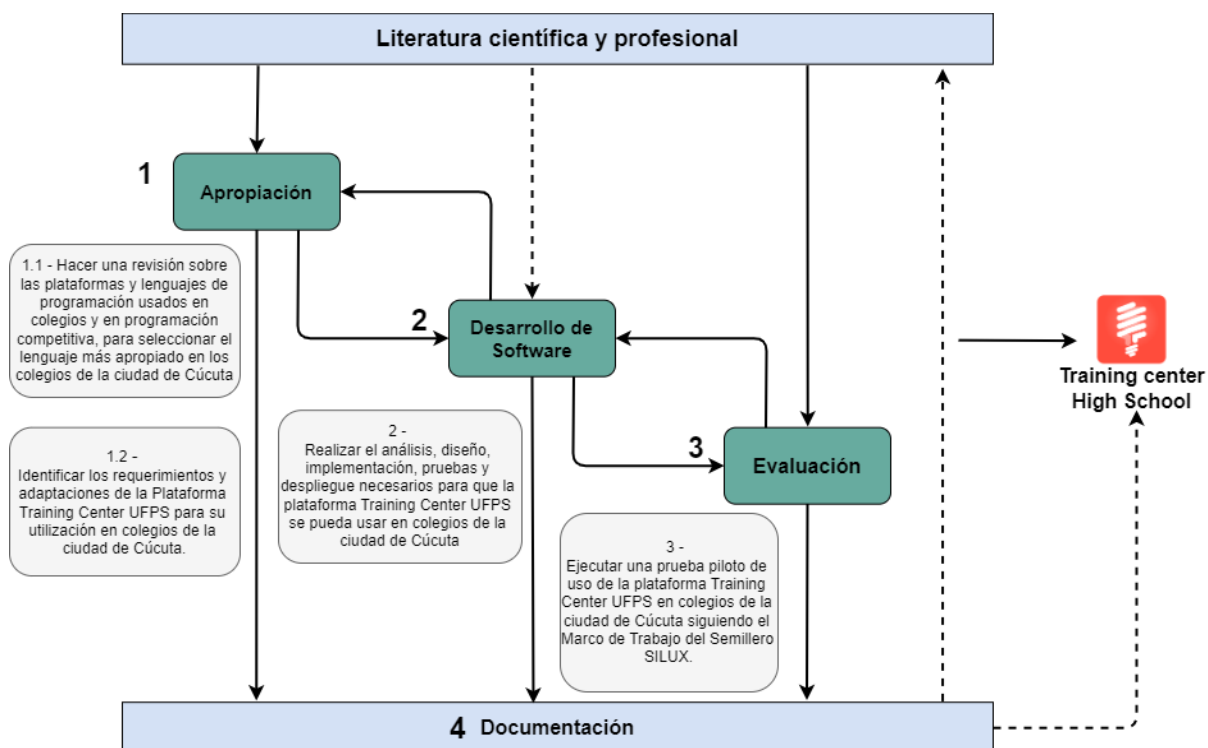


Ilustración 23 Metodología del proyecto con los objetivos propuestos adheridos.

El desarrollo del presente proyecto inició con el desarrollo del primer objetivo específico “Hacer una revisión sobre las plataformas y lenguajes de programación usadas en colegios y en programación competitiva para seleccionar el lenguaje más apropiado a usar en colegios de la ciudad de Cúcuta”.

5.2. Levantamiento de información

Como primera medida, para el desarrollo de esta nueva extensión para Training Center UFPS, se hizo un análisis de diferentes herramientas para enseñanza de programación y competencias para instituciones educativas. Como apoyo, se hizo una encuesta dirigida a colegios de la ciudad de Cúcuta sobre herramientas de enseñanza basadas en programación. En la siguiente imagen se muestra una gráfica de los pasos que se hicieron para recolectar y analizar esta información.

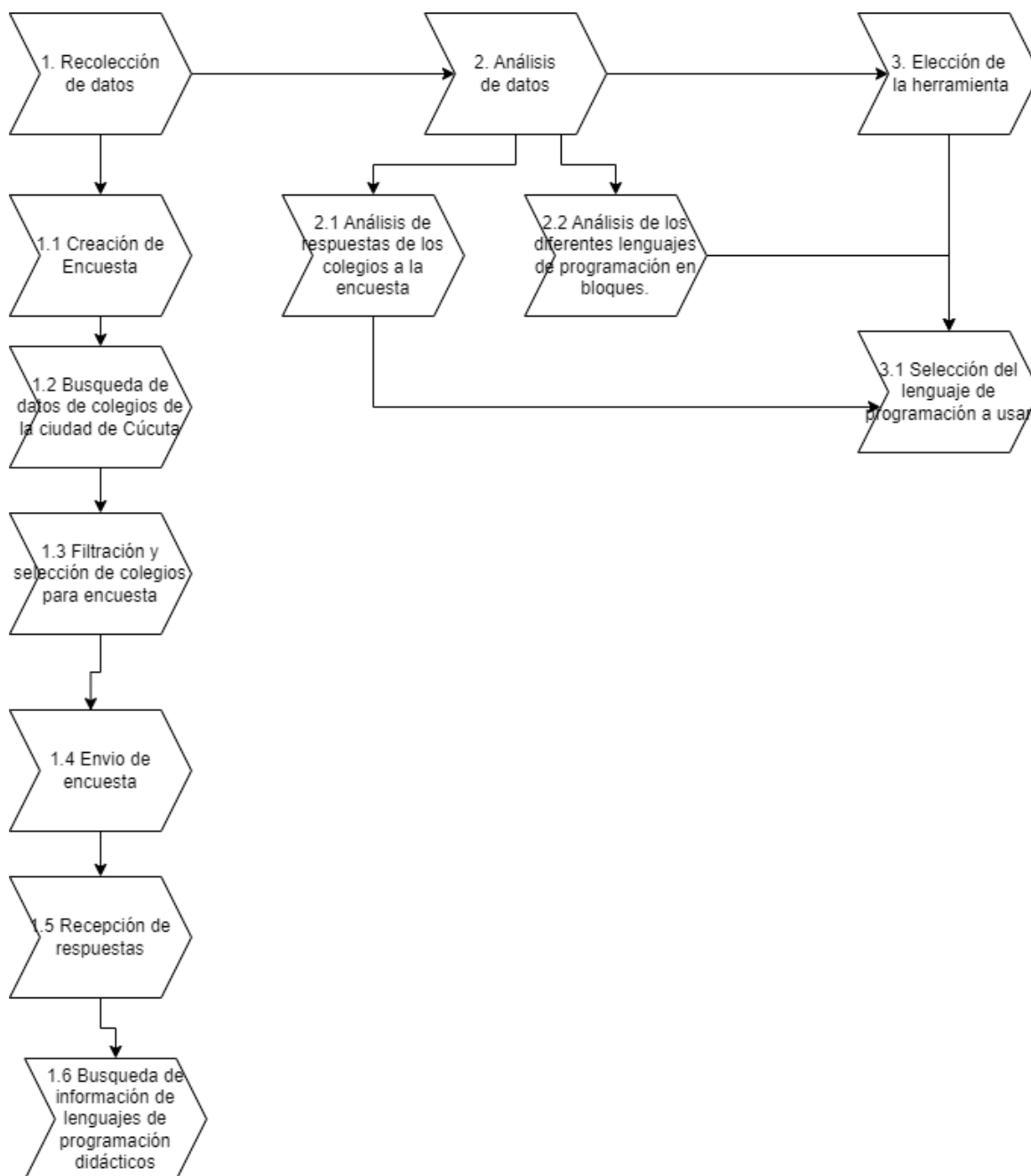



Ilustración 24 Pasos a seguir en el análisis y recopilación de información.


Se realizó la recolección de datos inicialmente creando una encuesta dirigida a varios colegios de la ciudad de Cúcuta, con el fin de conocer el estado de cada colegio en lo referente a

enseñanza de programación de computadores y la importancia que para ellos tiene esta área. Para esto se realizó un Formulario de Google con diferentes preguntas según los datos que se deseaban recolectar y que eran útiles para este proyecto (Ilustración 25).





Encuesta Proyecto Training Center High School

La siguiente encuesta tiene como fin recolectar información para el Proyecto Training Center High School, el cual busca integrar a los colegios con el semillero SILUX - Programación Competitiva de la UFPS. Si desea contactarnos puede hacerlo al correo electrónico miltonjesusvc@ufps.edu.co

Nombre del Colegio  Respuesta corta

Texto de respuesta corta

  Obligatorio

Dirección de Correo para contactar al colegio *

Texto de respuesta corta

Ilustración 25 Encuesta para colegios propuesta Training Center High School.

La encuesta contaba con la siguiente estructura:

- 1) Nombre del Colegio
- 2) Dirección de Correo para contactar al colegio
- 3) ¿En su colegio se enseña programación de computadores?
 - No
 - Si, programación con bloques (Scratch u otro)
 - Si, programación de sitios web
 - Otro
- 4) Si la respuesta anterior fue "si" seleccione el o los lenguajes que enseñan. Seleccione Ninguno si su respuesta fue "no"
 - Java
 - PHP
 - Javascript
 - Python
 - Bloques (Scratch u otro)
 - HTML
 - CSS
 - Ninguno
 - Otro
- 5) En qué grados se enseña programación de computadores

- 8
- 9
- 10
- 11
- Otro

6) ¿Cuántas horas a la semana se enseña programación?

- 0
- 1
- 2
- 3
- 4
- Otro

7) Según la percepción del maestro ¿Qué nivel de programación de computadores poseen los estudiantes al graduarse?

- Muy bajo
- Bajo
- Medio
- Alto
- Muy alto
- Otro

8) ¿Le interesaría recibir apoyo académico por parte de la Universidad Francisco de Paula Santander para procesos relacionados con la enseñanza de programación de computadores?

- Si
- No
- Otro

9) ¿Le interesaría participar en competencias de programación (maratones de programación) llevadas a cabo por la UFPS?

- Si
- No

10) Escriba las observaciones que considere sobre la Programación de Computadores en su colegio.

Posteriormente, luego de crear la encuesta, se realizó la petición de una base de datos con los correos de contacto de la mayoría de colegios de la ciudad de Cúcuta. Con el fin de poder enviar la encuesta a dichos colegios y realizar la recolección de datos.

De estos datos obtenidos se hicieron filtros manuales, para realizar el envío del correo con la encuesta a diligenciar. En este filtro se hizo un trabajo de búsqueda para cada colegio y con esto conocer si ellos contaban con una página web o alguna información en la que se pudiera corroborar que los correos electrónicos estaban actualizados y eran correctos.

La base de datos inicialmente tenía más de 150 colegios y después de la filtración quedaron aproximadamente 50 colegios a quienes se les envió la encuesta. De los colegios que respondieron a esta encuesta se recopilaron las siguientes respuestas agrupadas en gráficas:

Para la pregunta *¿En su colegio se enseña programación de computadores?*, más del 50% de los colegios encuestados respondieron “no”, una porción mayor al 12% respondieron que sí habían interactuado con algún lenguaje de programación ejecutable, otra porción del 12% conoce sobre mantenimiento de computadores, y, por último, una porción del 12.5% han interactuado con lenguajes de programación basados en bloques (Ilustración 26).

¿En su colegio se enseña programación de computadores?

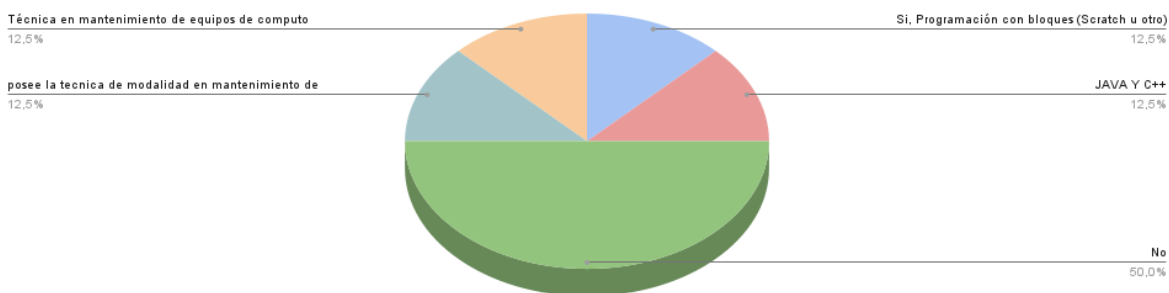


Ilustración 26 Estadísticas pregunta “¿En su colegio se enseña programación de computadores?”.

Para la pregunta *Si la respuesta anterior fue "si" seleccione el o los lenguajes que enseñan. Seleccione Ninguno si su respuesta fue "no"*, el 62% de los colegios respondieron que no enseñan ningún lenguaje, herramienta de programación u otra tecnología, el 37.5% de los colegios enseñan Java, PHP y C, el 12,5% enseñan sobre mantenimiento de equipos de cómputo, el 25% de los colegios enseña HTML a sus estudiantes, el 12.5% enseñan sobre bloques en Scratch, y ninguno de los colegios enseñan tecnologías como Javascript, Python y CSS (Ilustración 27).

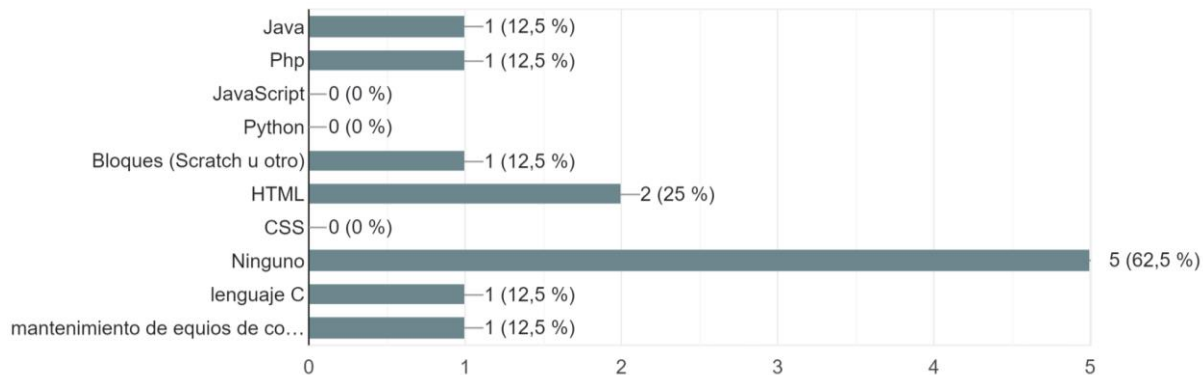


Ilustración 27 Estadísticas pregunta “Si la respuesta anterior fue "si" seleccione el o los lenguajes que enseñan. Seleccione Ninguno si su respuesta fue "no”.

En la pregunta *En que grados se enseña programación de computadores*, en ninguno de los colegios encuestados enseñan bases de programación desde 8° y 9° grado, el 12.5% de los colegios encuestados respondieron que enseñan programación desde 10° grado, el 37.5% respondieron que enseñan programación desde 11° grado, el 12.5% de los colegios encuestados respondieron que enseñan a nivel de técnica académica, el 12.5% enseñan bases de

programación en grados de primaria escolar, y el 37.5% de los colegios no enseñan programación en ninguno de sus cursos (Figura 28).

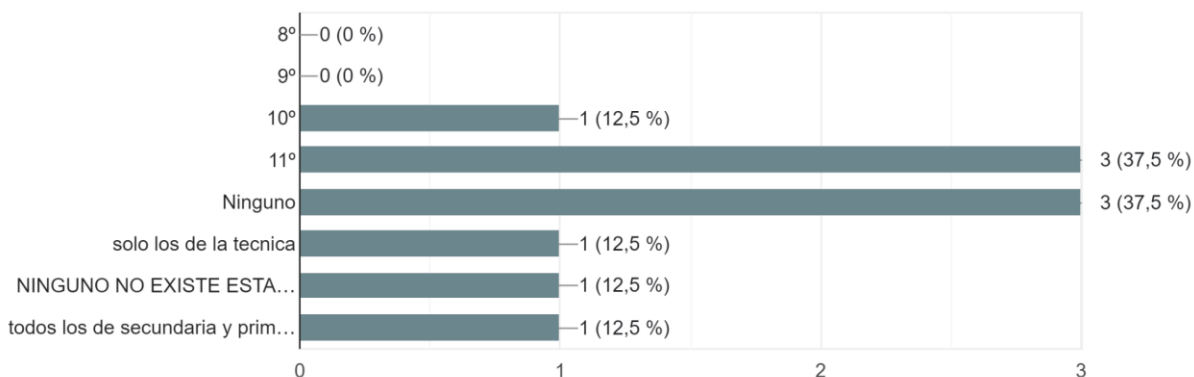


Ilustración 28 Estadísticas pregunta “¿En qué grados se enseña programación de computadores?”.

En la pregunta *¿Cuántas horas a la semana se enseña?*, el 50% de los colegios encuestados respondieron que dedican 0 horas a la semana en enseñanza de la programación, el 12.5% afirma que en sus cursos académicos no existe ninguna materia o sección en la que se enseñe programación, el 25% de los colegios enseña al menos 2 horas a la semana, y el 12.5% de los colegios afirma que enseñan programación al menos durante 5 horas a la semana (Ilustración 29).

¿Cuántas horas a la semana se enseña?

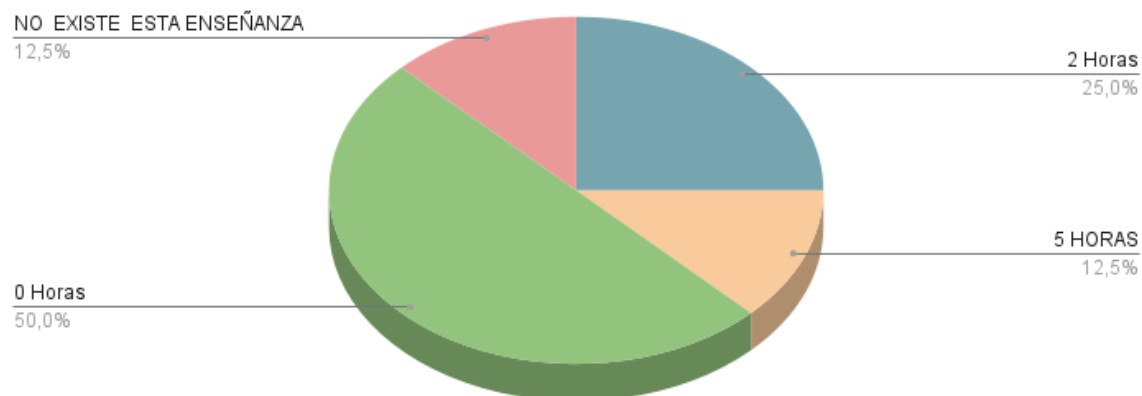


Ilustración 29 Estadísticas pregunta “¿Cuántas horas a la semana se enseña?”.

En la pregunta *Según la percepción del maestro ¿Qué nivel de programación de computadores poseen los estudiantes al graduarse?*, el 12.5% de los colegios afirma que el nivel de conocimiento en programación es alto, el 12.5% respondió que el nivel de enseñanza al momento de graduarse es medio, el 25% dice que el nivel de conocimiento es bajo, el 25% afirma que es muy bajo, el 12.5% afirma que no existe énfasis en enseñar programación, y el 12.5% de los colegios responde no enseñan programación (Ilustración 30).

Según la percepción del maestro ¿Que nivel de programación de computadores poseen los estudiantes al graduarse?

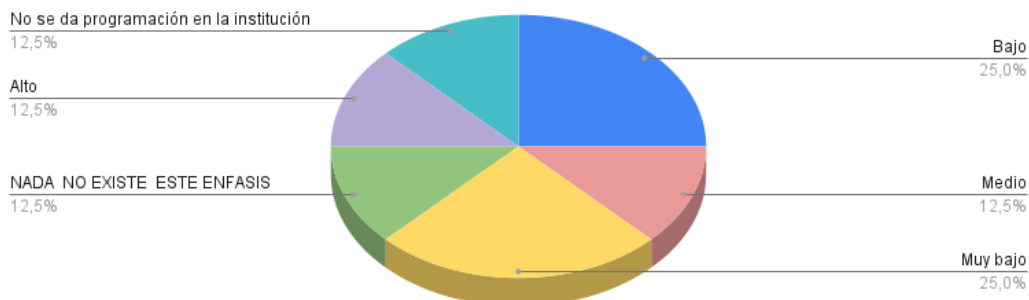


Ilustración 30 Estadísticas pregunta “¿Qué nivel de programación de computadores poseen los estudiantes al graduarse?”.

En la pregunta *¿Le interesaría recibir apoyo académico por parte de la Universidad Francisco de Paula Santander para procesos relacionados con la enseñanza de programación de computadores?*, la totalidad de los encuestados responde que están interesados en recibir apoyo académico por parte de la UFPS para apoyar procesos relacionados con la enseñanza de la programación (Ilustración 31).

¿Le interesaría recibir apoyo académico por parte de la Universidad Francisco de Paula Santander para procesos relacionados con la

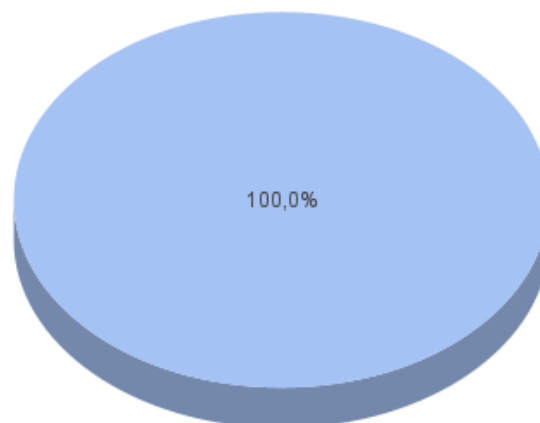


Ilustración 31 Estadísticas pregunta “¿Le interesaría recibir apoyo académico por parte de la UFPS para procesos relacionados con la enseñanza de programación de computadores?”.

En la pregunta *¿Le interesaría participar en competencias de programación (maratones de programación) llevadas a cabo por la UFPS?* El 75% de los colegios encuestados responde que, si están interesados, el 12.5% responde que están interesados en cuanto la estrategia que se proponga sea compacta, y el 12.5% no está interesado (Ilustración 32).

¿Le interesaría participar en competencias de programación (maratones de programación) llevadas a cabo por la UFPS?

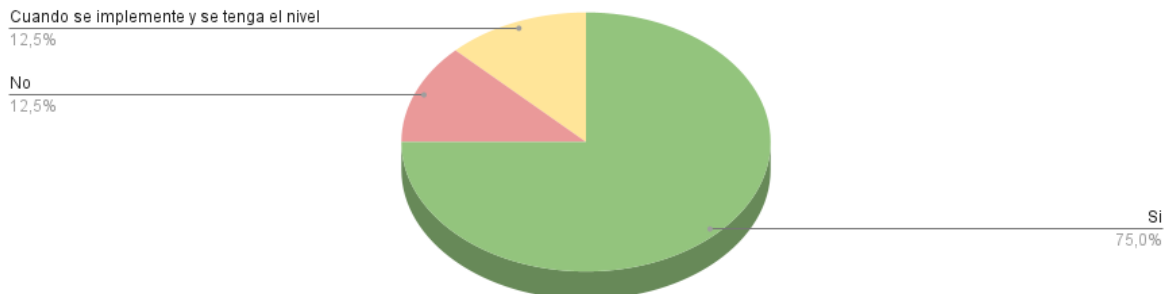


Ilustración 32 Estadísticas pregunta “¿Le interesaría participar en competencias de programación (maratones de programación) llevadas a cabo por la UFPS?”.

Después de analizar estos datos recolectados, se procedió a realizar una búsqueda de los diferentes lenguajes de bloques existentes para usar en programación e indagar cuál sería el más apropiado para este proyecto, teniendo en cuenta a la población que era dirigido y también el contexto de programación competitiva en el que se desarrollaría.

Luego de realizar la revisión y seleccionar el lenguaje “Blockly”, como el más apropiado para este proyecto, se procedió al desarrollo del segundo objetivo específico “Identificar los requerimientos y adaptaciones de la Plataforma Training Center UFPS para su utilización en colegios de la ciudad de Cúcuta”.

5.3. Proceso de desarrollo de software

Metodología de desarrollo

Scrum

Para este proyecto se ha seleccionado la metodología *Scrum*, como metodología para el desarrollo que será a nivel de aplicación Web. Scrum es una metodología ágil que proporciona una serie de iteraciones y fases a base de entregables funcionales, además de que brinda una estructura de trabajo para grupos de desarrollo pequeños y también es flexible en la adopción de cambios y nuevos requisitos (ARAQUE, 2017).

Esta metodología consta de una serie de fases las cuales se identificarán en la figura 33. Estas fases se irán repitiendo de forma cíclica (llamados Sprints), de tal manera que durante cada iteración se entregue una versión funcional del producto que vaya cumpliendo con los objetivos de desarrollo, y junto a la retroalimentación que esto ofrece, ofrecer mejoras y ajustes que satisfagan las necesidades de todos los involucrados. Se decidió que cada sprint para hacer entregas del software debería ser aproximadamente de una semana.

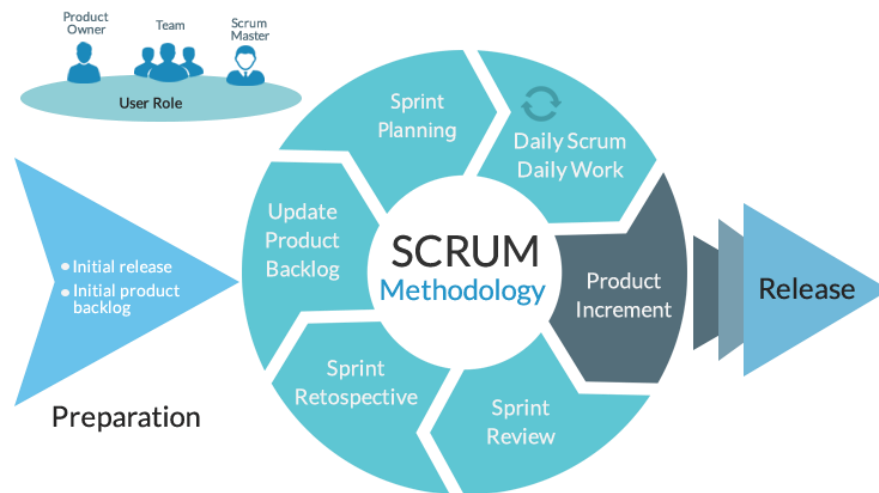


Ilustración 33 Fases de la metodología Scrum. Fuente: (Pérez, 2019).

SPRINTS

Los Sprints son el corazón de la metodología Scrum. Se podría definir como el contenedor para el resto de eventos de Scrum. Es un proceso continuo que no debe cambiar mientras esté en marcha el desarrollo del producto.

Gracias al sprint, es posible conocer la transparencia, el estado en el que está el desarrollo del proyecto, las carencias del Equipo Scrum para poder poner remedio y el feedback con el cliente.

MIEMBROS DEL EQUIPO SCRUM

- **Product Owner:** Es el representante que se comunica con el cliente y responsable de la gestión del Product backlog, es decir, es el responsable de la gestión de la lista de productos a entregar. Él se encargará de maximizar el valor del producto desarrollado por el Equipo Scrum y del retorno de la inversión.
- **Scrum Master:** Es el líder del equipo Scrum. Protegerá al equipo y les ayudará a eliminar impedimentos a la vez que colaborará con el Product Owner para priorizar el Product Backlog.
- **Equipo de Desarrollo:** Está compuesto por un equipo de desarrolladores, con experiencia y conocimiento, y su tarea es conseguir un incremento en el desarrollo del producto a partir de los elementos del Product Backlog seleccionados durante el Sprint Planning.

ARTEFACTOS DEL SCRUM

Los Artefactos Scrum son aquellos elementos que garantizan la transparencia y el registro de la información primordial del proceso de Scrum (Donetic, 2022).

- **Product Backlog:** Es una lista ordenada que contiene todo lo que podría ser necesario para el buen desarrollo del producto y es la única fuente de requisitos para hacer cambios en el producto.

- **Sprint Backlog:** Es el subconjunto de elementos del Product Backlog elegidos para abordarse durante el sprint. Estos elementos se componen de tareas técnicas más pequeñas que consiguen un incremento del software.
- **Incremento:** Es el resultado del sprint: es la suma de todas las tareas que el Equipo de Desarrollo ha ejecutado durante ese período de tiempo y que se ofrecerá al usuario final en forma de software, aportando un valor de negocio al producto que se está desarrollando.

CEREMONIAS DURANTE DEL DESARROLLO DE LOS SPRINTS

- **Sprint Planning:** Esto sucede al comienzo del sprint. Esta reunión revisa el Product Backlog con el Product Owner, discutiendo el trabajo a realizar en el Sprint. En esta ceremonia, cada miembro del equipo selecciona elementos de la lista de requerimientos y desarrollan una lista de objetivos para ser cumplidos durante el sprint. El único trabajo del Scrum Master es asegurarse de que la reunión exista como parte de Scrum y se mantenga a tiempo.
- **Daily:** Debe hacerse una vez por día, y no debe superar los quince minutos de duración. El equipo se reúne y se comunica el progreso, individual siempre con base en la meta del Sprint, ¿Qué hicimos ayer para contribuir al objetivo de este sprint?, ¿Qué vamos a hacer hoy para contribuir al objetivo de este sprint?, ¿hay algún impedimento que impida cumplir con este objetivo?

- **Sprint Review:** Es una reunión dedicada a mostrar el trabajo del equipo terminado para su inspección y adaptación por todos los miembros del equipo incluidos los stakeholders. De esta forma, las partes interesadas (en este caso, el director de proyecto, y las instituciones educativas) pueden ver cómo van las cosas y dedicarse a inspeccionar o adaptar el producto.
- **Sprint Retrospective:** Esta ceremonia consiste en obtener una retroalimentación rápida con el propósito de mejorar la cultura y desarrollo del producto. Se realiza al final de cada Sprint.
- **Refinamiento:** El refinamiento del Product Backlog es una práctica recomendada para asegurar de que el objetivo del sprint se cumple.

IMPLEMENTACIÓN DE SCRUM

Teniendo en cuenta las implicaciones que tiene Scrum, y por la cantidad de miembros del equipo de trabajo, se decide compartir roles dependiendo de las ceremonias que se desarrollan durante cada sprint. Con apoyo del director de proyecto se hicieron reuniones periódicas (presenciales y virtuales) hasta dos veces por semana, para revisar los objetivos, planificar y asignar tareas, hacer refinamiento de aquellas tareas que no se alineaban tan bien al cumplimiento del sprint, mostrar y explicar avances del producto de software (tanto de las mejoras y adaptaciones hechas a Training Center, como de la implementación del envío de soluciones en Blockly), y reflexionar sobre cómo se puede mejorar, y reevaluar aspectos que

potencialmente serían el objetivo para cumplir en el siguiente sprint. En la siguiente imagen se adjunta evidencia de una de las reuniones para la planificación de actividades con el apoyo del director del proyecto (el ingeniero Milton) y la co-directora (la ingeniera Deisy).

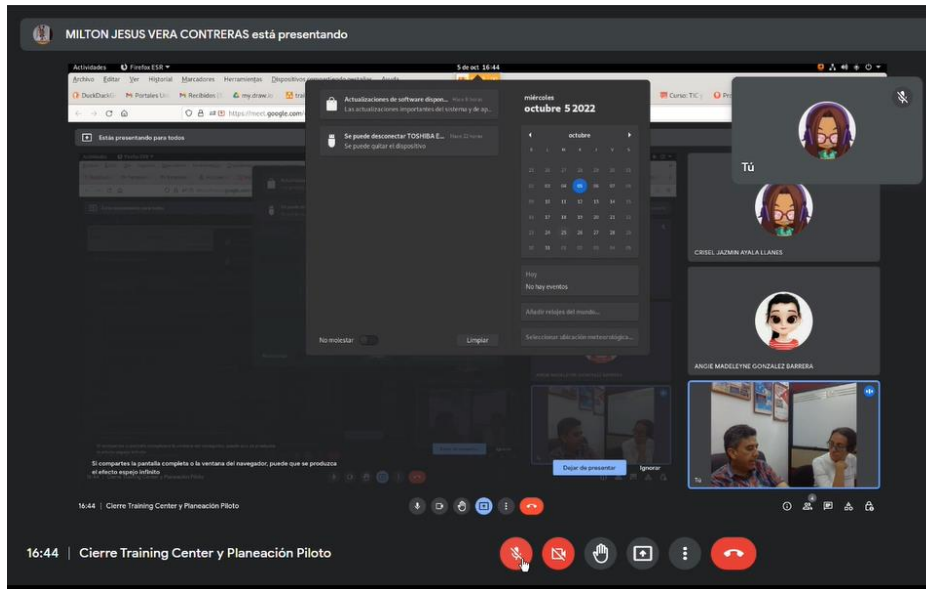


Ilustración 34 Reunión planificación actividades.

A su vez, se hizo necesario implementar herramientas que ayuden a gestionar de forma natural las tareas y objetivos para el desarrollo de software, entre otros, una herramienta de control de versiones como Git y Github, Calendario de Google para organizar reuniones periódicas para el desarrollo de los Sprints, Google Drive para almacenar evidencias y documentos de apoyo al avance de las tareas y objetivos, y, sobre todo, un software para poder administrar proyectos e iteraciones. Para el caso actual, se utilizó la herramienta Trello.

Trello es una aplicación para gestión de proyectos basada en el método Kanban y sirve para gestionar tareas, permitiendo organizar el trabajo en grupo de forma colaborativa mediante tableros virtuales compuestos de listas de tareas en forma de columnas. Es apropiada para la gestión de proyectos ya que se pueden representar distintos estados y compartirlas con diferentes personas que formen el proyecto. Con esta herramienta se permite mejorar las rutinas de trabajo del equipo generando prioridades, tiempos, avisos y otras opciones perfectas para organizar un proyecto en el que colaboran varias personas (Experto en Negocios Online, 2019).

Para este caso particular, se creó un tablero dividiendo las tareas en grupos de estados, de la siguiente manera: Product Backlog, To do, Doing, Testing, Done y Future tasks.

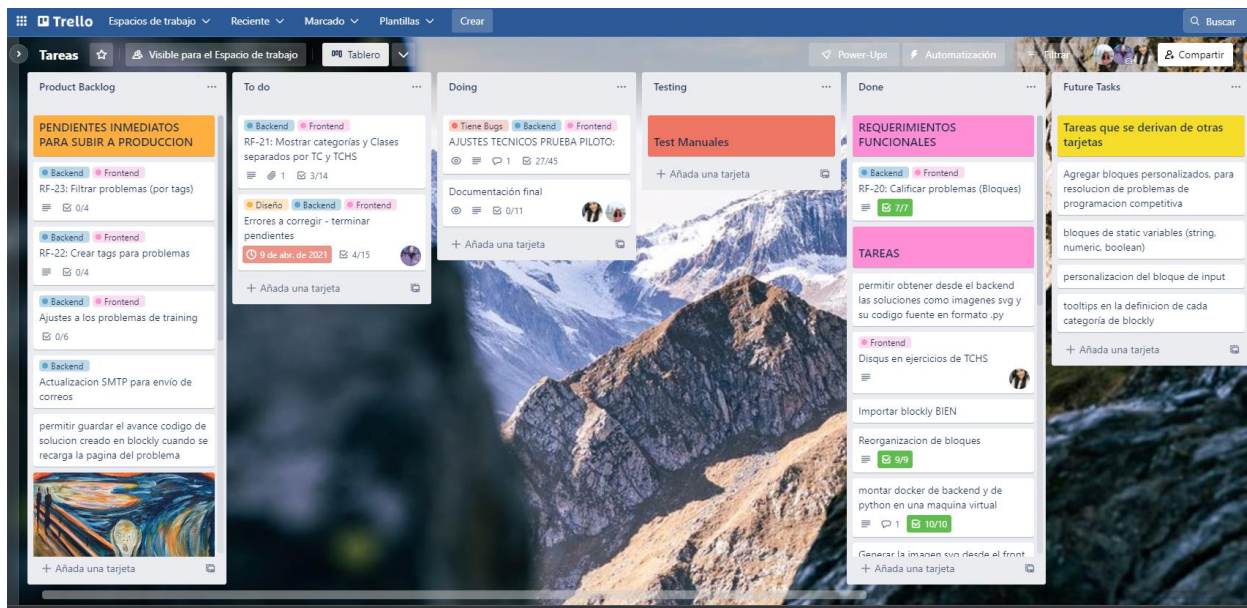


Ilustración 35 Tablero de Trello.

- **Product backlog:** Es una lista de trabajo ordenado por prioridades para el equipo de desarrollo que se obtiene de la hoja de ruta y sus requisitos. Los elementos más importantes se muestran al principio del backlog del producto para que el equipo sepa qué hay que entregar primero.
- **To do:** Lista de tareas que se van a realizar durante el transcurso del Sprint. Estas tareas están asignadas a uno de los miembros del equipo, y están estimadas de acuerdo al esfuerzo o puntos de estimación que él crea que vaya a colocar para que dicha tarea sea cumplida.
- **Doing:** Tareas que se están realizando durante la fase del sprint.
- **Testing:** Tareas que ya terminaron su proceso de desarrollo, y están en la fase de pruebas. Normalmente estos test suelen ser de tipo caja negra (tests manuales). Una tarea no puede pasar a estado Done, hasta que los resultados de las pruebas sean aceptables.
- **Done:** Tareas finalizadas; estas se discuten durante la review de cada sprint.
- **Future Tasks:** Lista de tareas o necesidades que surgen durante las reuniones de review y retrospectiva al final de cada sprint, pueden ser aceptadas para poder ser analizadas para pasar a estado de Product Backlog, o por lo contrario, pueden ser descartadas, por alejarse de los objetivos de alto nivel.

Las listas de tareas también se dividen de acuerdo a una jerarquía, basada por el nivel o la extensión que pueda tener en cuanto a tiempo y esfuerzos (las tareas de alto nivel se dividen en tareas de menor nivel que, terminadas en conjunto, hacen que la tarea de alto nivel sea cumplida). Los requerimientos funcionales del sistema, se consideran tareas de alto nivel; estas

tareas se dividen en pequeñas tareas que permitirán realizar de forma particular avance sobre estos requerimientos durante cada sprint (Ilustración 36).

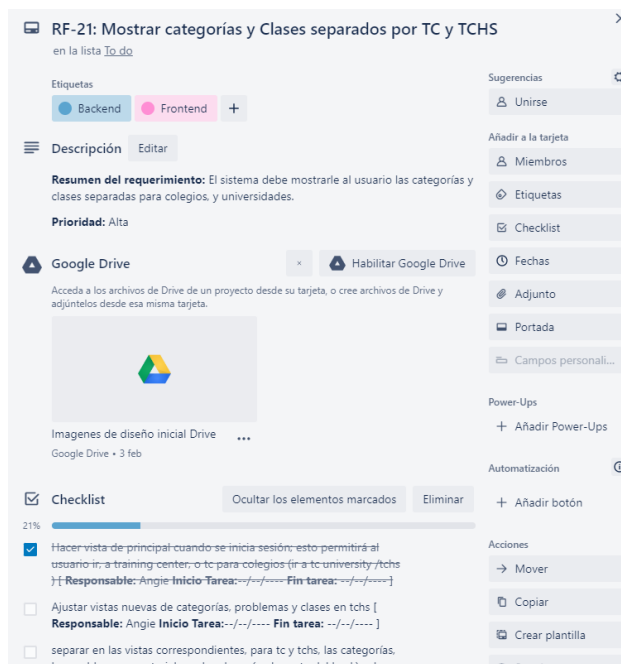


Ilustración 36 Tarea a alto nivel en Trello.

No sobra decir que el desarrollo de estas iteraciones va muy de la mano con el avance del cronograma del proyecto en la fase de desarrollo de software.

Es necesario aclarar que SCRUM es un marco metodológico que ofrece lineamientos, técnicas y buenas prácticas, pero no todas llegan a utilizarse. En ese sentido, en este proyecto SCRUM fue la línea base y la clave del éxito, de acuerdo a la documentación existente, es la cultura ágil, lo cual se logra con reuniones cortas y frecuentes donde se respondan las preguntas

esenciales: ¿qué se ha hecho? ¿qué están haciendo? ¿qué falta por hacer? ¿En qué necesitan ayuda?

Finalmente, siempre surge la duda de por qué ese marco metodológico de Scrum y no otra alternativa. En la práctica la justificación suele ser un sesgo de masificación, una tendencia, una moda. Para este caso particular, la mayoría de los estudiantes de Ingeniería de Sistemas de la UFPS que trabajan en empresas de desarrollo de software usan Scrum junto a otras opciones como Design Thinking. Entonces, el uso de Scrum se justifica en que es lo que se sabe y se domina en el contexto laboral de los autores del proyecto.

5.4. Prueba piloto

La prueba piloto es el tercer objetivo específico del presente proyecto. Con ella se plantea la socialización de la plataforma desarrollada, con profesores y alumnos de tres colegios de la ciudad de Cúcuta. Con el fin de dar a conocer el propósito y funcionamiento de la plataforma y así mismo llevar a cabo una maratón de entrenamiento con los estudiantes de colegio en la cual se implemente la metodología que ya se ha venido trabajando con el grupo de estudio. Esta prueba piloto proyecta obtener una retroalimentación acerca del funcionamiento de la plataforma y también sobre la experiencia de estudiantes y profesores con el acercamiento a la programación competitiva. Esto con el objetivo de seguir a futuro por medio del semillero SILUX y el grupo de estudio de programación competitiva, trabajando la programación competitiva con estudiantes de instituciones educativas.

Para el correcto desarrollo de esta prueba piloto se plantea la siguiente ruta a seguir.

Piloto Training Center High School

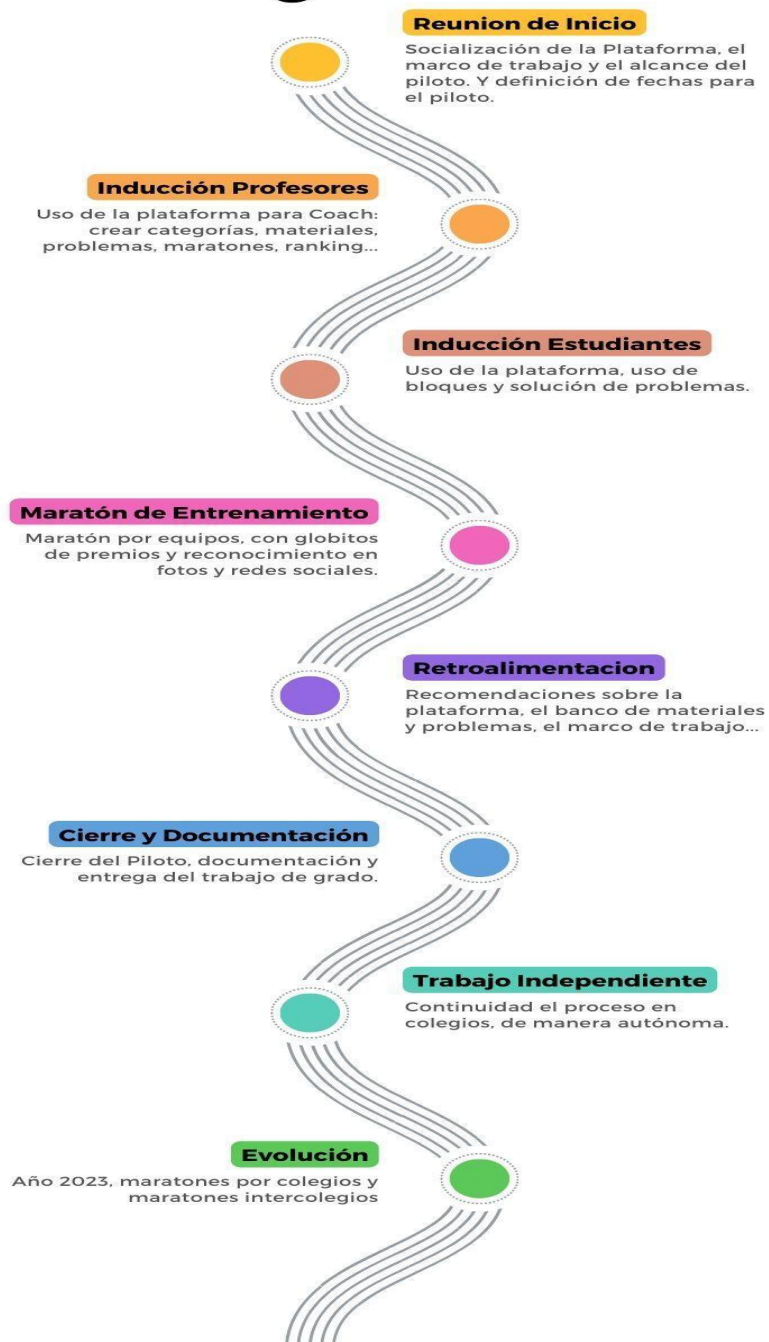


Ilustración 37 Pasos para la prueba piloto en colegios.

Esta hoja de ruta se retomará posteriormente al presentar el detalle del desarrollo de la prueba piloto.

6. Nueva Version Training Center High School

Según lo que dicta scrum se inició realizando un sprint planning donde se definió la ruta a seguir para llegar al cumplimiento de los objetivos. Esta ruta se definió de la siguiente manera:

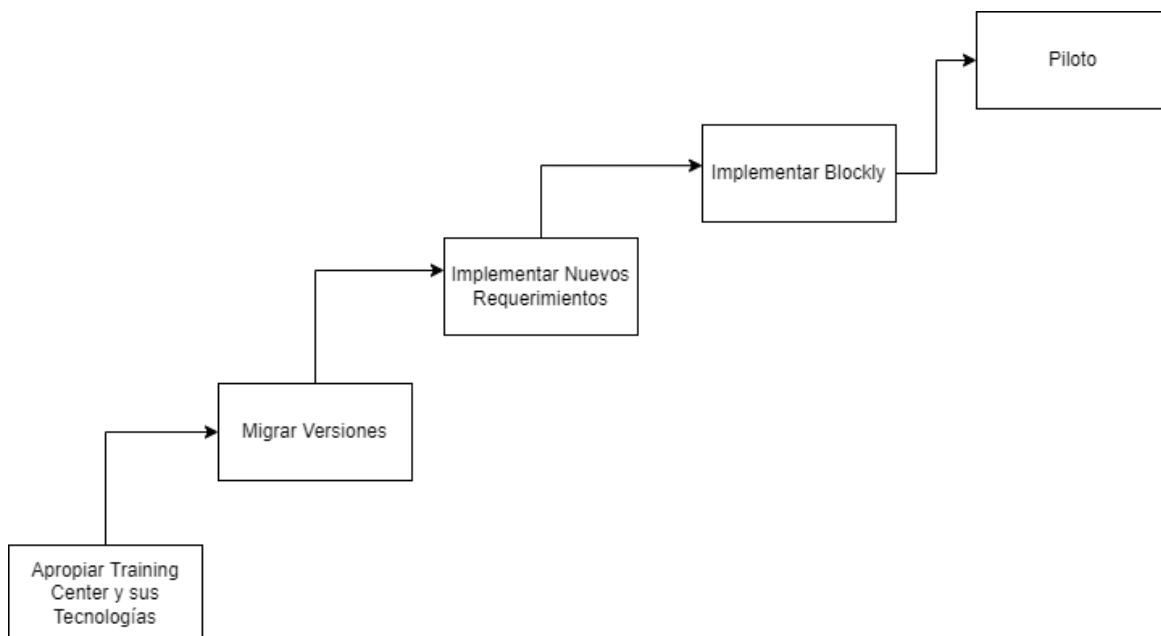


Ilustración 38 Ruta a seguir para hacer adaptaciones a Training Center.

Se dio inicio con la apropiación de conocimiento de Training center y sus tecnologías. Se identificó qué lenguaje de programación y framework usaba la plataforma y se procedió a realizar adquisición de conocimiento de las tecnologías no conocidas. De esta apropiación también se identificaron necesidades para llevar a cabo el cumplimiento de los objetivos de este proyecto. Dentro de las necesidades más importantes detectadas fue la migración de versiones de Training Center.

Durante la migración de versiones de la plataforma esta ruta fue cambiando según las necesidades presentadas. A continuación, se muestra la evolución y cambio de la ruta de desarrollo del proyecto (Ilustración 39).

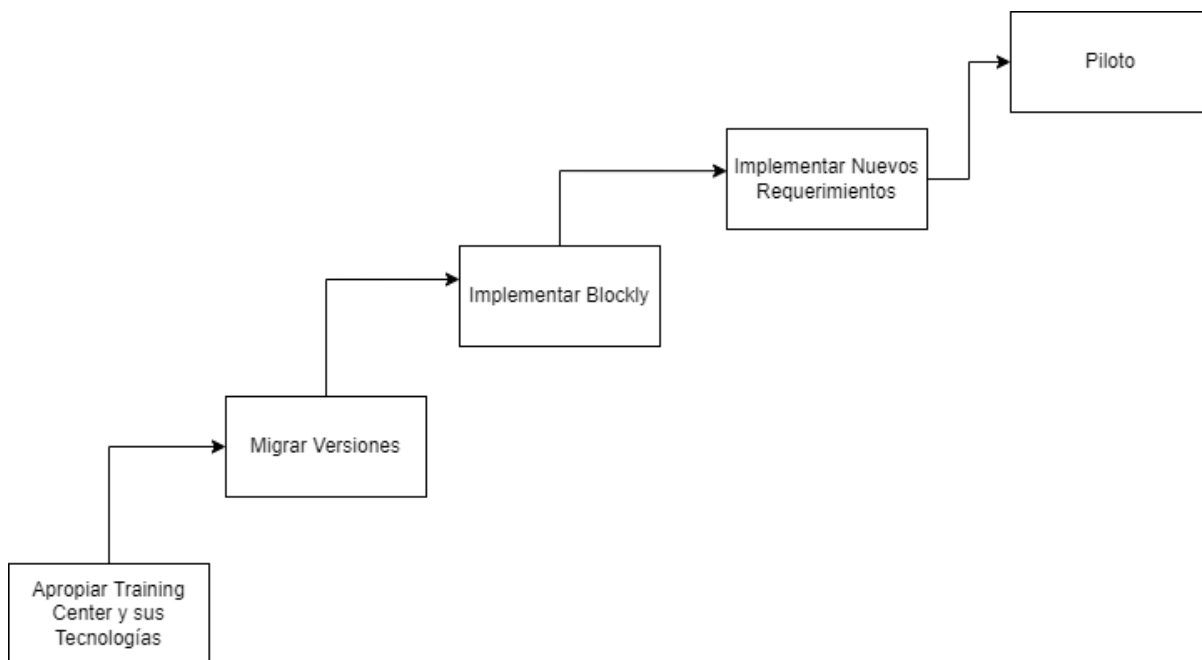


Ilustración 39 Evolución 1 de la ruta a seguir para hacer adaptaciones en Training Center.

Se vio la necesidad de implementar Blockly primero luego de migrar versiones para probar su funcionamiento con las nuevas versiones de la plataforma.

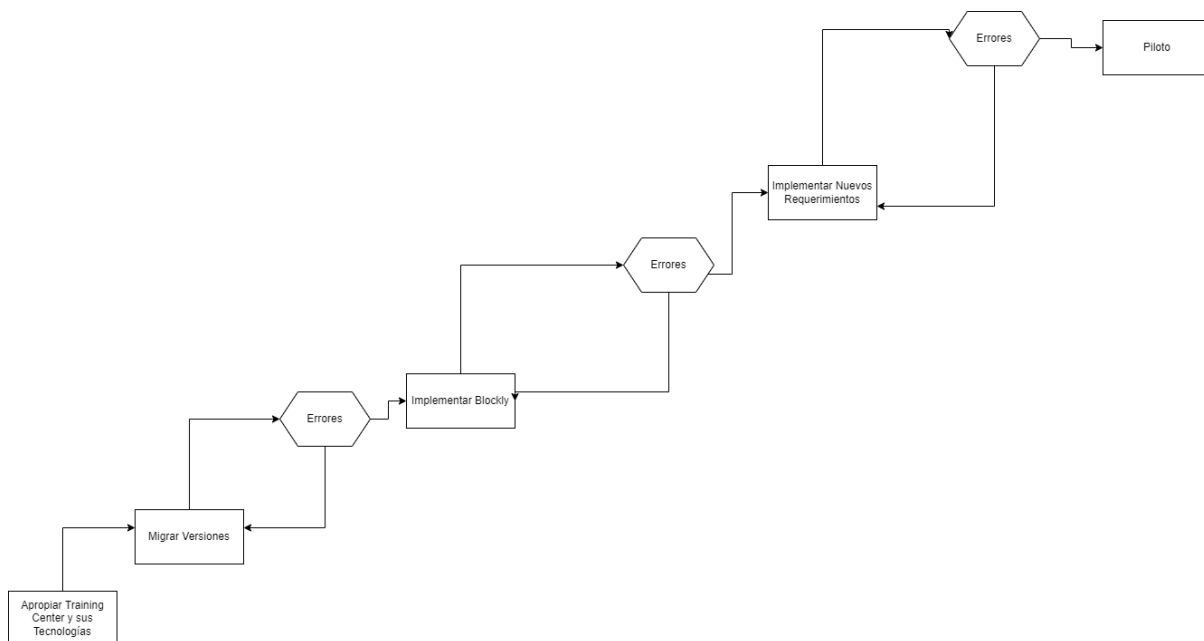


Ilustración 40 Evolución 2 de la ruta a seguir para hacer adaptaciones en Training Center.

Luego se evidenciaron algunos errores después la migración de versiones que debían ser atendidos para poder realizar las actividades siguientes en el proceso de desarrollo.

Convirtiéndose así en un proceso constante como lo muestra la imagen 40.

Seguidamente se evidenció que en algunos casos para la implementación de Blockly o el desarrollo de los nuevos requerimientos, se requería volver al proceso de actualización de

versiones para el correcto funcionamiento de cada proceso. Por lo tanto, finalmente la ruta llevada a cabo a lo largo del proceso de desarrollo fue la evidenciada en la siguiente imagen.

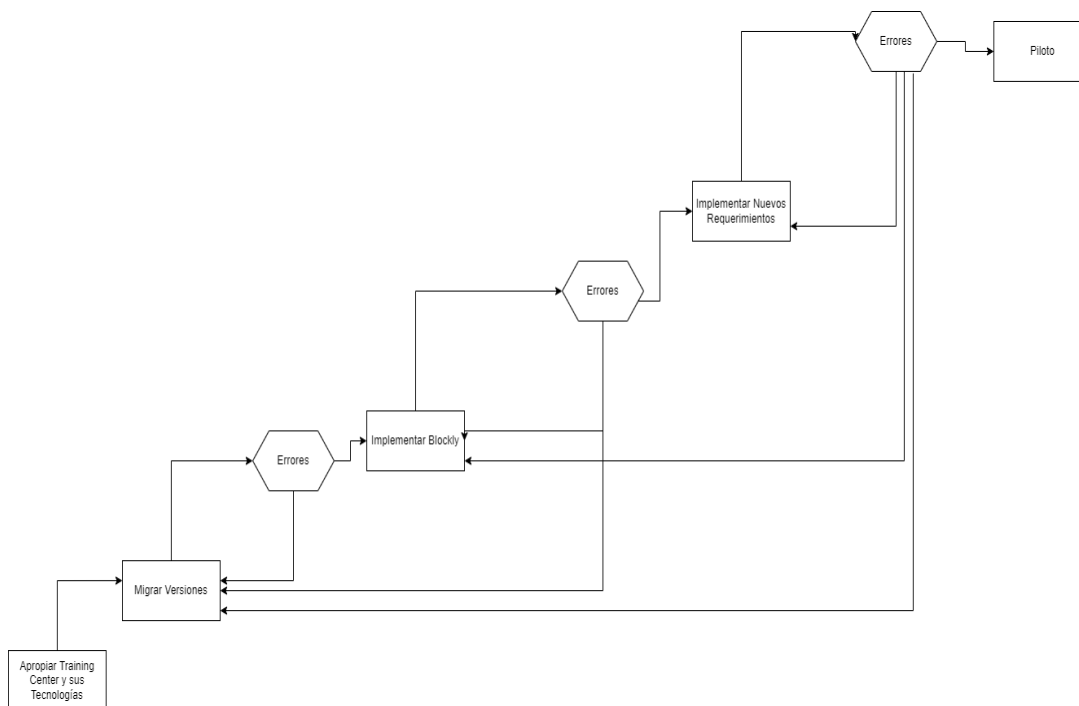


Ilustración 41 Versión final de la ruta a seguir para hacer adaptaciones en Training Center.

Esta ruta nos muestra el proceso realizado en el desarrollo de software del presente proyecto. El cuál representa también la mayoría de casos presentados en el mundo real del desarrollo de software, cuando se trabaja sobre un software ya existente.

6.1. Requerimientos o funcionalidades

La apropiación de la plataforma Training Center dio paso a la creación de requerimientos según las necesidades identificadas. Esto conllevó a listar las funcionalidades y requerimientos para el proyecto, lo que se debía modificar y adaptar y lo nuevo que se necesitaba implementar para llegar al cumplimiento de los objetivos. La lista de requerimientos que se definieron fue la siguiente:

Tabla 12 Requerimientos funcionales

Id.	Nombre	Descripción
RF-01	Implementar Blockly	Implementación de Blockly en Training Center, para la resolución de problemas en lenguaje de bloques.
RF-02	Listar Problemas	El sistema debe desplegar al usuario la lista de problemas almacenados en la plataforma con la posibilidad de filtrar bajo diferentes criterios.
RF-03	Agregar problemas	El sistema debe permitir a los docentes y administradores añadir nuevos problemas

		para que los estudiantes los solucionen en lenguajes de programación (Java, C++, Python, Blockly).
RF-04	Corregir problemas	Los administradores tienen la potestad de corregir problemas si tienen errores en su formato, estructura o entradas y salidas.
RF-05	Eliminar problemas	El administrador puede borrar problemas de la plataforma si no cumple con las características que debería contar un problema de programación.
RF-06	Acceder al foro de discusión (por problema)	El sistema debe permitir las discusiones de los estudiantes en cada problema.
RF-07	Filtrar problemas (por nombre), para colegios y universidades.	El sistema permite que los estudiantes filtren los problemas por nombre para módulo colegios y módulo universidades dependiendo del módulo en que se encuentre.

RF-08	Categorizar problemas	Los problemas en Training Center pueden agruparse bajo diferentes categorías globales dependiendo el tipo de categoría (colegio-universidad).
RF-09	Ver ranking	El sistema debe generar ranking de los usuarios según el número de problemas resueltos.
RF-10	Crear competencias	El sistema debe permitir crear competencias (maratones) de programación en tiempo real, en lenguaje de bloques.
RF-11	Añadir material de ayuda	El sistema debe permitir que los usuarios agreguen material de ayuda (PDF, videos).
RF-12	Ver material de ayuda	Los estudiantes podrán ver el material de ayuda para mejorar su entrenamiento.
RF-13	Crear Modo guiado (Tour)	Los usuarios de Training Center podrán acceder a un tour que permite el

		conocimiento de las nuevas funcionalidades de la plataforma.
RF-14	Resolver tareas de las clases	Los profesores pueden crear tareas (clases o syllabus) para que sus estudiantes resuelvan los ejercicios que propone con lenguaje de bloques y vean sus materiales.
RF-15	Recepcionar solución de tareas	Los estudiantes envían las tareas propuestas por el coach/docente a la plataforma. El sistema debe permitir la recepción de estas tareas.
RF-16	Eliminar usuarios	El sistema debe permitir al administrador eliminar usuarios no deseados.
RF-17	Calificar problemas	El sistema debe calificar automáticamente las soluciones enviadas por los estudiantes, a los diferentes problemas e indicarles su calificación.
RF-18	Mostrar categorías y clases separados por TC y TCHS	El sistema debe mostrarle al usuario las categorías y clases separadas para colegios, y universidades.

RF-19	Inspeccionar códigos en competencias (como usuario administrador)	El sistema debe permitir a un administrador inspeccionar códigos en competencias en vivo, para evitar posibles plagios.
-------	---	---

6.2. Proceso de desarrollo de software

Se inició el proceso de desarrollo de software con una revisión completa de la plataforma Training Center, para entender su arquitectura y funcionamiento. Se identificó en qué lenguajes y herramientas estaba construida, tanto para la parte del back-end como para la parte de front-end. Se realizó una búsqueda y lectura de documentación para el entendimiento a alto nivel de las herramientas utilizadas en la plataforma.

Dentro de este proceso de apropiación se detectaron necesidades de la plataforma Training Center, siempre alineado al objetivo principal, para posteriormente documentar implementación de cambios necesarios a realizar, siempre utilizando la metodología de desarrollo.

Se detectó que uno de los cambios necesarios para la implementación de la extensión de la plataforma era la actualización de dependencias de la plataforma; la primera versión de Training Center se lanzó en el año 2018, por lo tanto era evidente que varias de las versiones de las dependencias tanto del front-end como del back-end ya eran obsoletas. Como se mencionó

anteriormente, la implementación de Blockly también fué una razón fundamental para hacer la actualización de las librerías del proyecto de software. A continuación, se empezó a realizar la actualización de las dependencias de la plataforma.

Como se podrá observar más adelante en el diagrama de arquitectura, también se realizaron actualizaciones de versiones de los sandbox de C++, Java y Python y así mismo de la imagen base de sus respectivos Dockers de Ubuntu (para los calificadores) de la versión 16.04 a la versión 20.04. También se actualizó la versión de la imagen base del back-end de Node.js versión 8.9.4 a la versión 14 y del front-end a Nginx.

A nivel de frameworks y librerías, entre otras, se realizaron actualizaciones de versiones de Aurelia.js, Node.js, Sequelize.js y Bootstrap la cuál como consecuencia generaron nuevos bugs y se requirió actualizar la estructura de la mayoría de vistas a lo usado en la nueva versión para el correcto funcionamiento de las diferentes vistas de la plataforma.

A continuación, en la siguiente tabla se resume el listado de tecnologías que recibieron una actualización de versión o fueron agregadas en cada uno de los componentes que hacen parte del software:

Tabla 13 Tecnologías actualizadas o agregadas para la nueva versión del software Training Center.

Training Center	
Capa	Tecnologías
Grading	Docker Java C++ Python
Base de datos	Mariadb
Back-end	Docker Node.js Sequelize Express.js
Front-end	Docker Nginx Aurelia.js HTML 5 CSS Blockly Webpack

El proceso de actualización de dependencias de la plataforma Training Center requirió una gran cantidad de tiempo ya que cada actualización requería cambios en la forma de

implementar diferentes procedimientos y funcionalidades. Como resultado de estas actualizaciones de dependencias, se puede visualizar en la siguiente sección, los cambios realizados a la arquitectura de la aplicación.

Luego de finalizar esta fase, se inició el proceso de desarrollo de nuevos requerimientos y funcionalidades para la plataforma para dar con cumplimiento a los objetivos de este proyecto.

Las siguientes tablas (tabla 14 y tabla 15) muestran las funcionalidades nuevas y modificadas en front-end y back-end del presente proyecto, frente a las funcionalidades de la primera versión de Training Center.

Tabla funcionalidades a nivel de Front-end

Tabla 14 Modificaciones en Front-end.

Funcionalidad	Funcionalidad nueva, modificada, sin modificar	Modificado por (otros proyectos externos a este)	Detalle	Usuario que puede realizar esta acción
Listar Problemas	Modificada	Proyecto Training Center High School	<p>Se ajusta el redireccionamiento de los links de los problemas de la plataforma, teniendo en cuenta si ese problema es de colegio o universidad.</p> <p>Se actualizaron versiones de Aurelia js y Bootstrap.</p>	Administrador, coach, estudiante

			Se implementó Webpack para empaquetar ficheros o módulos	
Agregar problemas	Modificada	Proyecto Training Center High School	Se agregó la opción para elegir si el problema a crear es de tipo colegio o universidad. Se actualizaron versiones de Aurelia js y Bootstrap. Se implementó Webpack para empaquetar ficheros o módulos	Administrador, coach
Corregir problemas	Modificada	Proyecto Training Center High School	Se actualizaron versiones de Aurelia js y Bootstrap.	Administrador

			Se implementó Webpack para empaquetar ficheros o módulos	
Eliminar problemas	Modificada	Proyecto Training Center High School	Se actualizaron versiones de Aurelia js y Bootstrap. Se implementó Webpack para empaquetar ficheros o módulos	Administrador
Acceder al foro de discusión (por problema)	Nuevo	Proyecto Training Center High School	Se implementó Disqus como herramienta para la interacción de los usuarios con respecto a cada problema. Con el fin de poder aclarar y resolver dudas acerca del problema.	Administrador, coach, estudiante

<p>Filtrar problemas (por nombre)</p>	<p>Modificada</p>	<p>Proyecto Training Center High School</p>	<p>El buscador permite buscar problema por nombre y al listar muestra una columna en la que se puede identificar a qué tipo de institución pertenece el problema</p> <p>Se ajusta el link que redirecciona al problema para si es de tipo colegio redireccione a la vista de solución de problemas en Blockly.</p> <p>Se actualizaron versiones de dependencias (Bootstrap). Se acomodaron temas visuales que se descuadraron luego de la actualización de dependencias.</p>	<p>Administrador, coach, estudiante</p>
---------------------------------------	-------------------	---	--	---

Categorizar problemas	Nuevo	Proyecto Training Center High School	<p>Se ajusta el link que redirecciona al problema, para que, si es de tipo colegio redireccione a la vista de solución de problemas en Blockly.</p> <p>Se actualizaron versiones de dependencias (Bootstrap). Se acomodaron temas visuales que se descuadraron luego de la actualización de dependencias.</p> <p>Se implementó Webpack para empaquetar ficheros o módulos.</p>	Administrador, coach, estudiante
Ver ranking	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias (Bootstrap). Se	Administrador, coach, estudiante

			acomodaron temas visuales que se descuadraron luego de la actualización de dependencias.	
Crear competencias	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias (Bootstrap). Se acomodaron temas visuales que se descuadraron luego de la actualización de dependencias.	Administrador, coach, estudiante
Añadir material de ayuda	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias (Bootstrap). Se acomodaron temas visuales que se descuadraron luego de la actualización de dependencias.	Coach, estudiante

Ver material de ayuda	Modificada	Proyecto Training Center High School	Se creó una sección en la que los estudiantes pueden acceder a un material de ayuda para la solución de problemas con Blockly.	Administrador, coach, estudiante
Crear Modo guiado	Nuevo	Proyecto Training Center High School	Se implementó herramienta que realiza un tour al usuario mostrando los cambios realizados, para ayudar al usuario a familiarizarse con la plataforma.	Administrador, coach, estudiante
Eliminar usuarios	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias (Bootstrap). Se acomodaron temas visuales que se descuadraron luego de la	Administrador

			<p>actualización de dependencias. Se cambió la gama de colores y fuentes.</p> <p>Se implementó Webpack para empaquetar ficheros o módulos</p>	
<p>Calificar problemas (Blockly)</p>	Nuevo	<p>Proyecto Training Center High School</p>	<p>Se muestra al usuario un mensaje indicando que su solución fue enviada y que pronto recibirá respuesta del Juez.</p>	Estudiante
<p>Mostrar categorías y clases separados por Training center y Training center high school</p>	Nuevo	<p>Proyecto Training Center High School</p>	<p>Se actualizaron versiones de dependencias (Bootstrap). Se acomodaron temas visuales que se descuadraron luego de la</p>	Administrador, coach, estudiante

			<p>actualización de dependencias. Se cambió el tono de colores.</p> <p>Se implementó Webpack para empaquetar ficheros o módulos</p>	
Home de Training Center	Nuevo	Proyecto Training Center High School	Se creó una página de inicio para Training Center. La cual incluye dentro de sí acceso a login, registro, información sobre los desarrolladores, material público y otras secciones para más información la cual está sujeta a futuras actualizaciones.	Administrador, coach, estudiante
Registro Usuarios.	Nuevo	Proyecto Training Center High School	Al formulario de registro usuario se agregaron los campos tipo de	Estudiante

			<p>institución e institución. Para poder separar los usuarios dependiendo si son de Colegio o Universidad.</p> <p>Se hacen ajustes para agregar esta sección a la vista “Home” y se muestre a través de un modal.</p>	
Inspeccionar códigos en competencias	Nuevo	Proyecto Seminario integrador I semestre 2022	Permite ver el código de cada envío realizado en competencia, con permiso de administrador. Esto con el fin de que los administradores puedan validar la autenticidad de cada código.	Administrador, coach, estudiante

Inspeccionar códigos en competencias	Modificado	Proyecto Training Center High School	Se ajustan Bugs al ver detalle del envío.	Administrador, coach, estudiante
Inspeccionar códigos en ranking	Nuevo	Proyecto Seminario integrador I semestre 2022	Permite ver el código de cada envío realizado, mostrando en la tabla de ranking una opción “ver envíos”. Solo para usuario administrador.	Administrador, coach, estudiante
Inspeccionar códigos en ranking	Nuevo	Proyecto Training Center High School	Se ajustan Bugs al ver detalle del envío.	Administrador, coach, estudiante
Ver ranking	Modificado	Proyecto Seminario integrador I semestre 2022	Se agregan filtros a ranking que permite filtrar los usuarios por tipo de institución (Colegio - Universidad), institución, año y categoría.	Administrador, coach, estudiante

			Se agrega columna para especificar a qué institución pertenece.	
Ver maratones	Modificado	Proyecto Training Center High School	Se agrega filtro para poder realizar la búsqueda por tipo de institución (Colegios - Universidad) dentro de las maratones.	Administrador, coach, estudiante
Ver envíos en bloques	Nuevo	Proyecto Training Center High School	Se crea la opción para poder visualizar el envío del problema realizado en lenguaje de bloques por medio de un modal que se despliega desde la pantalla “mis envíos”. Mostrando así la solución enviada en	Coach, estudiante

			<p>formato .svg. Estas soluciones pueden ser descargadas en lenguaje Python.</p> <p>Se actualizaron versiones de dependencias (Bootstrap).Se modificaron temas de look and feel.</p>	
Login	Modificada	Proyecto Training Center High School	Se hacen ajustes para agregar esta sección en la vista “Home”, el cual se puede acceder desde el menú y se muestra un modal correspondiente al ingreso del usuario.	Administrador, coach, estudiante

Tabla funcionalidades Back-end

Tabla 15 Modificaciones en Back-end.

Funcionalidad	Nueva, modificada, sin modificar	Modificado por	Detalle	Usuario que puede realizar esta acción
Listar Problemas	Modificada	Proyecto Training Center High School	<p>Se ajusta el redireccionamiento de los problemas para que tenga en cuenta si es de colegio o universidad.</p> <p>Se actualizaron versiones de dependencias docker, mysql y sequelize.js.</p>	Administrador, coach, estudiante

Agregar problemas	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias docker, mysql y sequelize.js.	Administrador, coach
Calificar problemas (Java, C++, Python)	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias docker, mysql y sequelize.js.	Coach, estudiante
Corregir problemas	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias docker, mysql y sequelize.js.	Administrador

Eliminar problemas	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias docker, mysql y sequelize.js.	Administrador
Ver ranking	Modificada	Proyecto Seminario integrador I semestre 2022	Se agregan filtros a ranking que permite filtrar los usuarios por tipo de institución (Colegio - Universidad), institución, año y categoría. Se agrega columna para especificar a qué institución pertenece.	Administrador, coach, estudiante
Ver maratones	Nueva	Proyecto Training Center High School	Se agrega filtro por institución (Colegio - Universidad).	Administrador, coach, estudiante

Categorizar problemas	Modificada	Proyecto Training Center High School	Se agrego el tipo de categoría para que al listar categorías, muestre los problemas dependiendo el tipo de categoría (Universidad - Colegio) Se actualizaron versiones de dependencias docker, mysql y sequelize.js.	Administrador, coach, estudiante
Crear competencias en Blockly	Nueva	Proyecto Training Center High School	Se agrega la creación de maratones que permite agregar problemas en lenguajes de bloques (Maratones para colegios).	Administrador, coach, estudiante

Eliminar usuarios	Modificada	Proyecto Training Center High School	Se actualizaron versiones de dependencias docker, mysql y sequelize.js.	Administrador
Calificar problemas (Blockly)	Nueva	Proyecto Training Center High School	Se pueden calificar problemas de Blockly, mediante código fuente en Python. Se guardan envíos en Blockly en formato .svg.	Coach, estudiante
Mostrar categorías y clases separados por Training center y Training center high School	Nueva	Proyecto Training Center High School	Permite crear categorías y clases ya sea para colegios o universidad. Al mostrarlo los separa dependiendo de a qué módulo se ingresa	Administrador, coach, estudiante

			Se actualizaron versiones de dependencias docker, mysql y sequelize.js.	
Hacer competencias (maratones)	Modificada	Proyecto Training Center High School	Se modificó para que se permita hacer maratones en vivo con problemas que puedan ser resueltos en lenguaje de bloques.	Administrador, coach, estudiante
Registro Usuario	Modificada		Se ajusta para registrar a la información del usuario tipo de institución (Colegio - Universidad) e institución a la que pertenece.	Estudiante

Inspeccionar códigos en competencias	Nueva	Proyecto Seminario integrador I semestre 2022	Permite ver los códigos enviados mientras está en ejecución una maratón.	Estudiante
Inspeccionar códigos en ranking	Nueva	Proyecto Seminario integrador I semestre 2022	Permite ver el código de cada envío realizado, mostrando en la tabla de ranking una opción “ver envíos”	Coach, administrador
Filtro de institución para ranking	Nueva	Proyecto Seminario integrador I semestre 2022	Se agrega filtro para poder realizar la búsqueda por institución (Colegio - Universidad).	Administrador, coach, estudiante
Filtro de institución para maratones	Nueva	Proyecto Training Center High School	Se agrega filtro para poder realizar la búsqueda por institución (Colegio - Universidad).	Administrador, coach, estudiante

Ver envíos	Nueva	Proyecto Training Center High School	Se crea la opción para poder visualizar el envío del problema realizado en lenguaje de bloques por medio de un modal que se despliega desde la pantalla “mis envíos”. Mostrando así la solución en formato .svg.	Administrador, coach, estudiante
Vista Login	Modificada	Proyecto Training Center High School	Se modifica para agregar el login a la pantalla nueva “Home”, el cual se puede acceder desde el menú y muestra un modal correspondiente al ingreso del usuario.	Administrador, coach, estudiante

A nivel de front-end se inició con la creación de mockups para las nuevas vistas planeadas para la extensión de la plataforma Training Center. Los mockups creados se pueden encontrar como documentos adjuntos en este proyecto.

6.3. Arquitectura de software

Ajustes realizados a la arquitectura del sistema

Para entender los cambios realizados en la arquitectura de Training Center se realiza señalización en los diferentes diagramas por medio de colores.

- Azul: Indica que se hizo modificación en ese objeto, archivo o carpeta.
- Verde: Indica que se agregó ese nuevo objeto, archivo o carpeta.
- Amarillo: Indica actualización de dependencias en ese objeto, archivo o carpeta.

Diagrama inicial de arquitectura: A continuación, se muestra el diseño de arquitectura de la primera versión de Training Center (Ilustración 42).

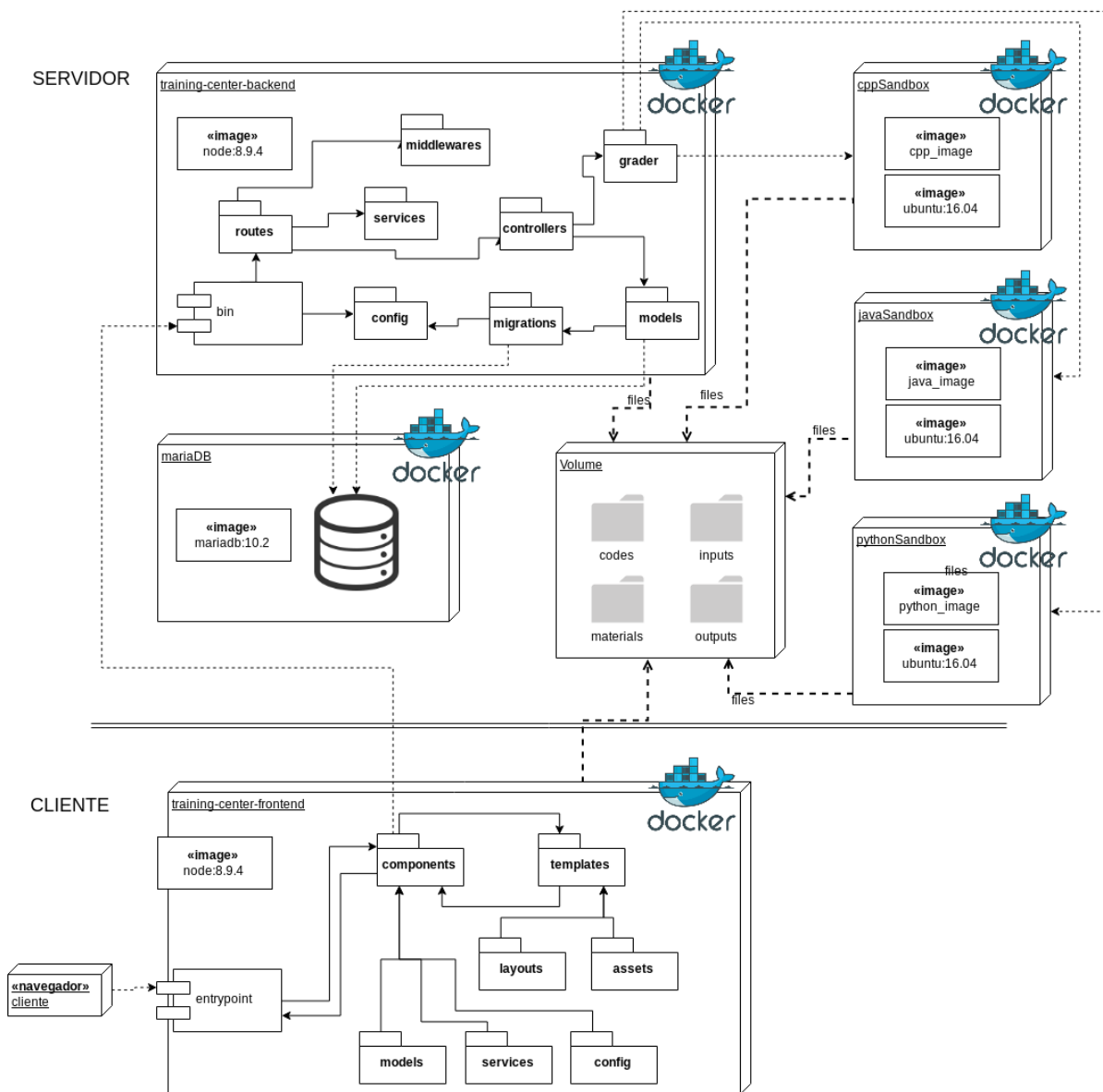


Ilustración 42 Diagrama de arquitectura primera versión Training Center.

Nuevo diagrama de arquitectura: A continuación, se muestra el diseño de arquitectura de la nueva versión de Training Center, tanto a alto como a bajo nivel (Ilustración 43 y 44).

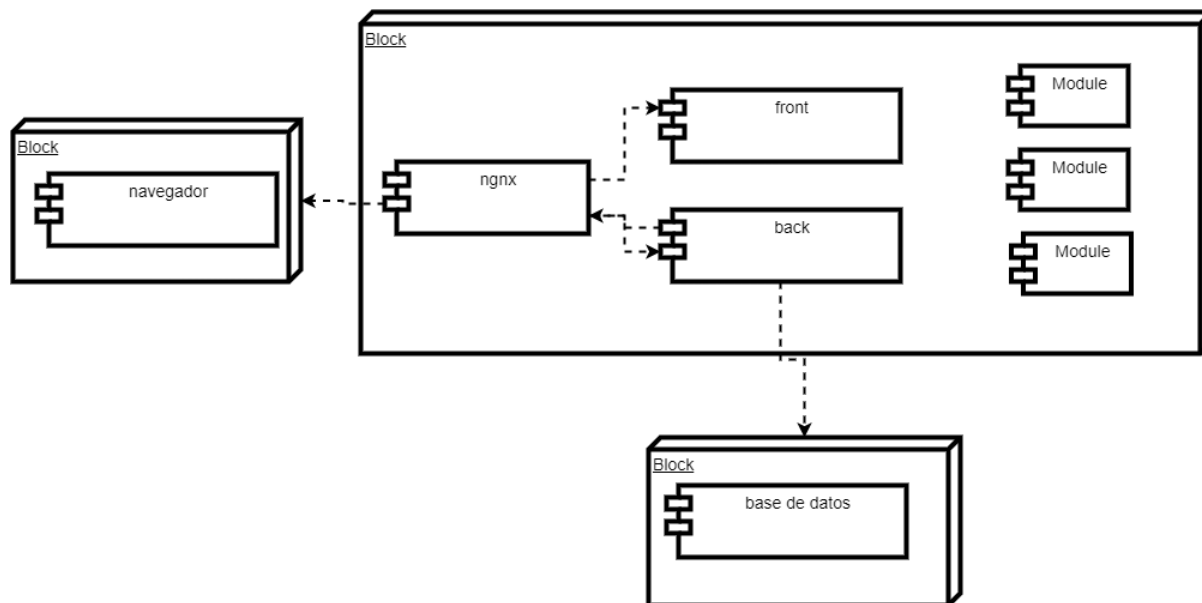


Ilustración 43 Diagrama de arquitectura a alto nivel.

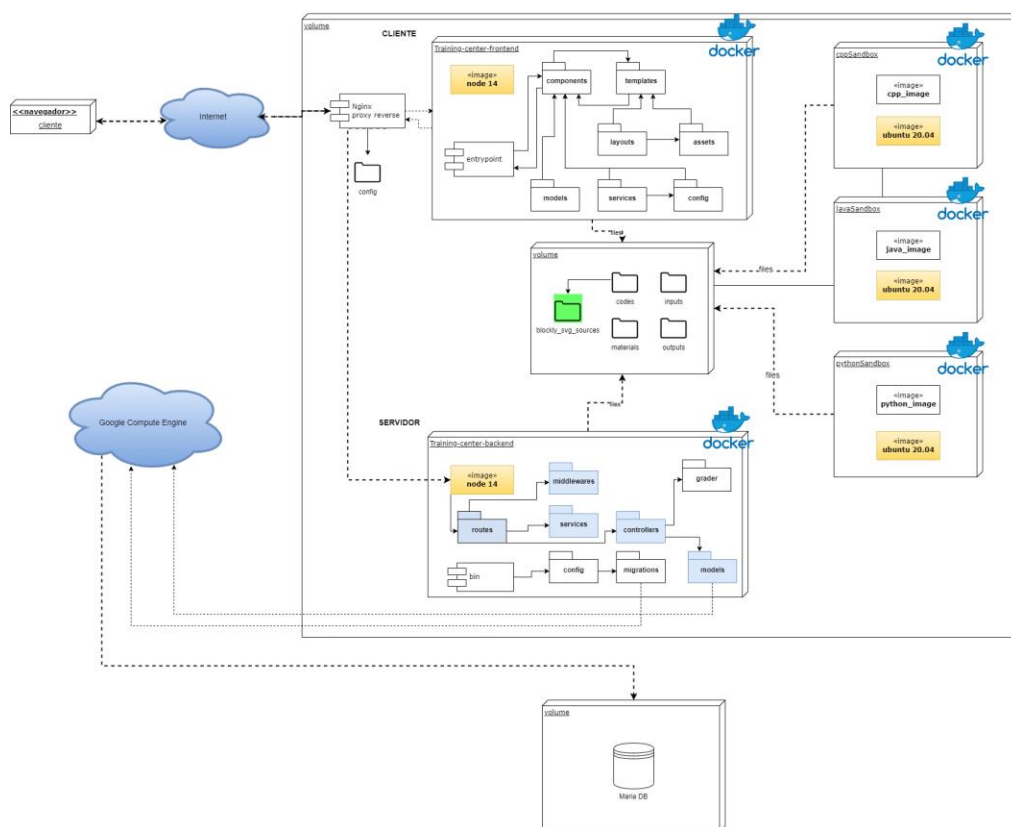


Ilustración 44 Diagrama de arquitectura a bajo nivel.

Especificación general de la arquitectura

La plataforma está diseñada bajo una arquitectura de cliente - servidor (dos capas). Cada una de las capas es diseñada como una aplicación completa y funcional, con interdependencia una de la otra. A continuación, se describe el funcionamiento de cada capa.

Capa servidor

El servidor es el encargado de procesar la información enviada desde el cliente y devolver las respuestas que sean necesarias. Esta capa contiene en su interior varios componentes interrelacionados entre sí:

- training-center-back-end: El núcleo central de este servidor es el “training-center-back-end”, aplicación que funciona como eje central de procesamiento obteniendo la información necesaria de los demás componentes y que mantiene una constante comunicación con la aplicación cliente. A continuación, se definen los subcomponentes de esta aplicación, para posteriormente definir su funcionamiento en conjunto:

- Bin: Se encarga de recibir las peticiones desde el cliente y devolver las respuestas.
- Routes: Conjunto de utilidades para analizar las peticiones recibidas y “enrutarlas” hacia el componente encargado de su procesamiento.

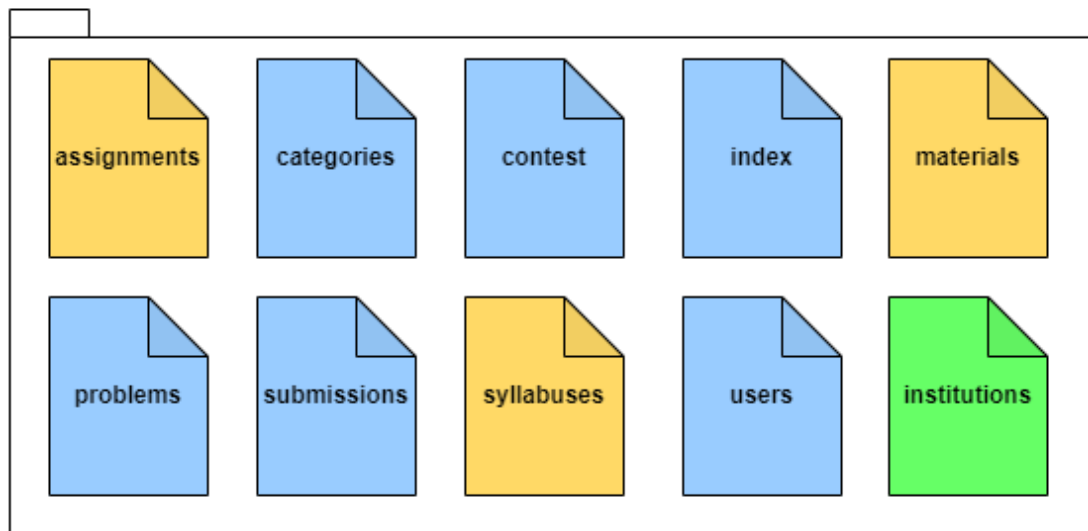
routes

Ilustración 45 Subcomponente routes.

Ajustes nuevos: Se agrega nuevos enrutadores, para la gestión de materiales, problemas y maratones para colegios y universidades, filtros de problemas y maratones, se agrega el enrutador “institutions” con sus diferentes endpoints, para la gestión de instituciones, tanto colegios como universidades. Se modifican categorías para modificar el enrutamiento del módulo materiales.

- **Middleware:** Son pequeños bloques de código que se ejecutan entre la petición que hace el usuario hasta que la petición llega al servidor, definiendo el flujo y restricciones de la aplicación.

- Services: Utilidades genéricas que pueden ser de ayuda en diferentes operaciones y se encapsulan para evitar redundancia de código.

services

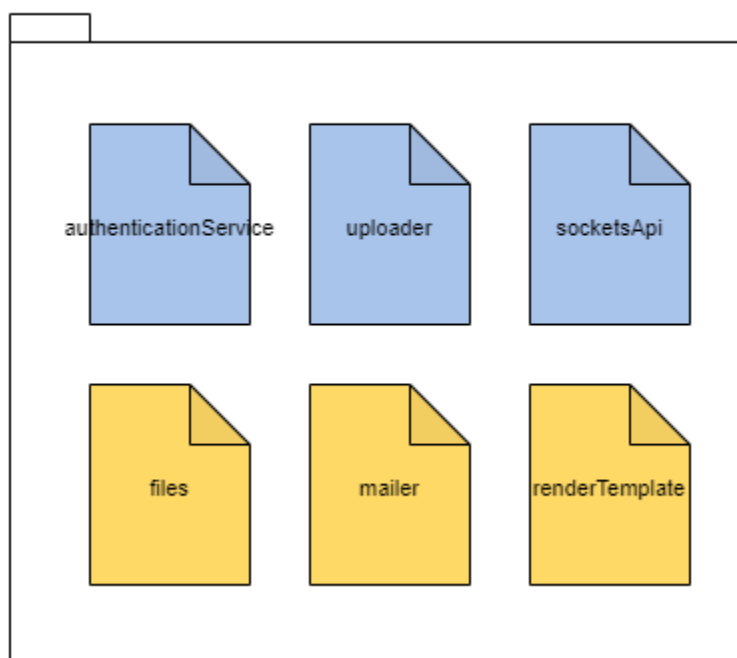


Ilustración 46 Subcomponente services.

Ajustes nuevos: Se ajusta el servicio de manejo de archivos (files), para poder gestionar envíos en Blockly, validar que la extensión del archivo y la información recibida sea válida.

- Controlllers: Lógica de la aplicación.

Ajustes nuevos: Se hacen ajustes en todos los controladores, para gestionar problemas, materiales, categorías, clases, competencias y usuarios para colegios y universidades. Se agrega el controlador “institutions”, para manejo de instituciones tanto

colegios como universidades. Se ajusta sintaxis de operaciones a la base de datos con sequelize.js.

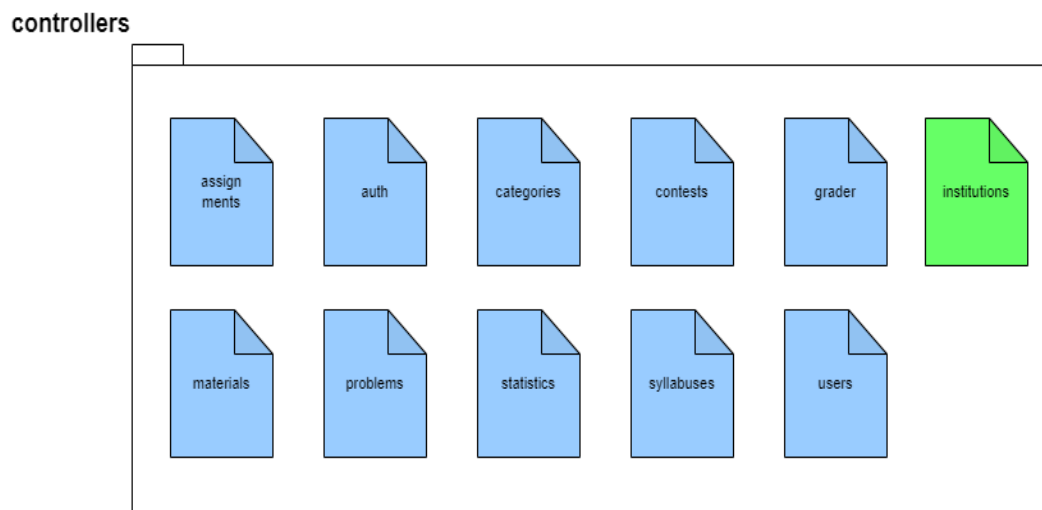


Ilustración 47 Subcomponente controllers.

- Config: Archivos de configuración de la aplicación.

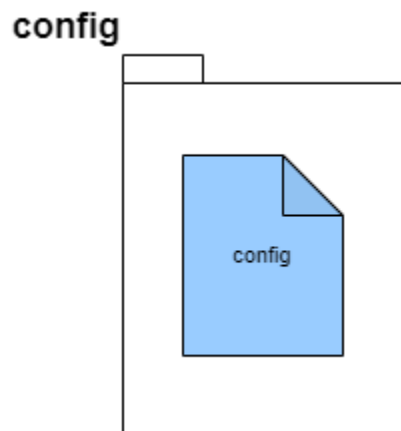


Ilustración 48 Configuración del componente back-end.

- **Models:** Modelos de datos (en este punto se gestiona la conexión con la base de datos).

Ajustes nuevos: Se ajusta la sintaxis de los modelos de base de datos a Sequelize en su versión más reciente. Se agregan nuevos índices, relaciones y modelos para gestionar problemas, categorías, competencias, usuarios e instituciones de colegios y universidades.

Models

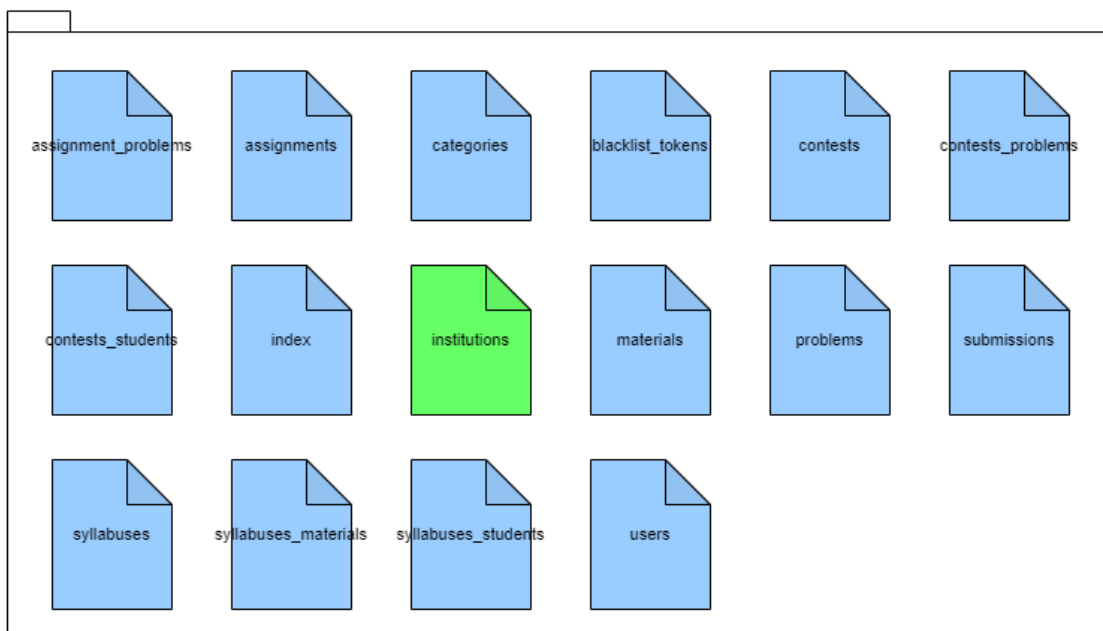


Ilustración 49 Subcomponente models.

- **Migrations:** Se usan para hacer un seguimiento de los cambios en la base de datos. Con las migraciones se puede transferir la base de datos existente a otro estado y viceversa, esas transiciones de estado se guardan en archivos de migración, que describen cómo llegar al nuevo estado y cómo revertir los cambios para volver al estado anterior.

Ajustes nuevos: Con la actualización de los modelos de base de datos, se generaron los nuevos archivos de migración.



Ilustración 50 Subcomponente migrations.

- Grader: componente encargado de comunicarse con los contenedores “sandboxing” para la ejecución de las soluciones de los estudiantes.

Ajustes nuevos: Se hacen ajustes en los scripts para ejecutar envíos en Python 3.x.

Se hacen ajustes para gestionar archivos y calificar problemas con la herramienta Blockly.

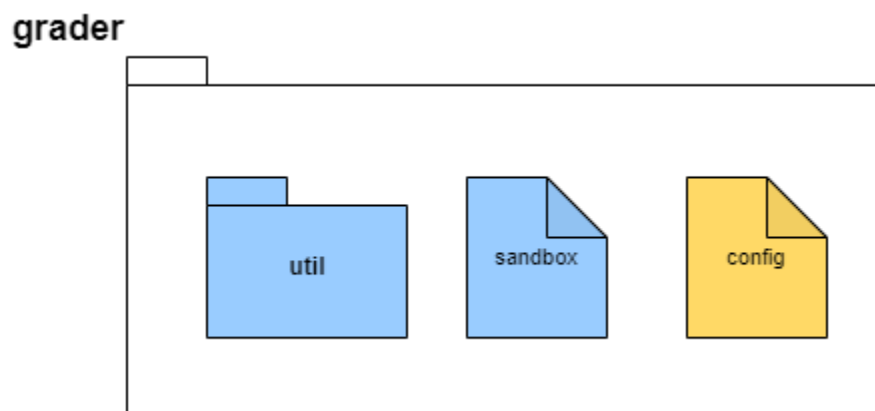


Ilustración 51 Subcomponente Grader.

- Seeders: Se usa para crear datos por defecto en la base de datos.

Ajustes nuevos: Se crean nuevas configuraciones de registros por defecto para usuarios e instituciones tanto colegios como universidades.

seeders

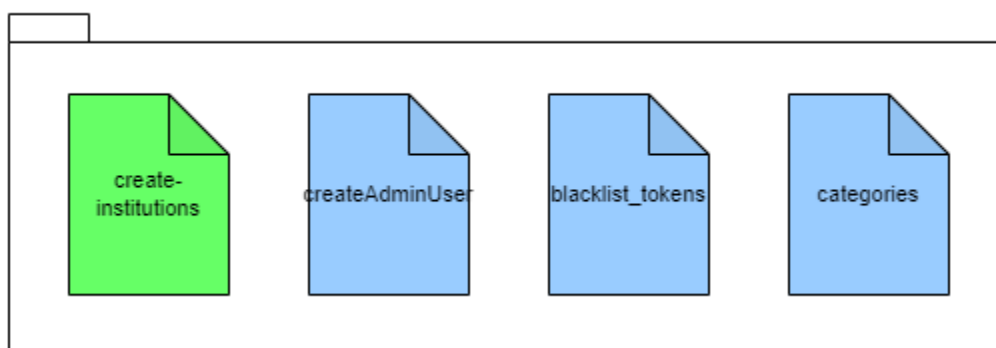


Ilustración 52 Subcomponente seeders.

En conjunto, el funcionamiento del back-end es el siguiente: “bin” se encuentra permanentemente escuchando peticiones desde el cliente. Al llegar una petición, esta es capturada por un router específico que la redirige al controlador en el que debe procesarse, utilizando los servicios que sean necesarios, y en el proceso de redirección la solicitud cruza los middlewares para definir entre otros aspectos, que el usuario esté autorizado. En caso de ser necesario, el controlador se comunica con los modelos (models), los cuales a través de migrations obtienen y actualizan la información almacenada en la base de datos. Durante todo el proceso, se toman en cuenta los parámetros establecidos globalmente en config. Un caso especial ocurre cuando la solicitud recibida es un ejercicio para ser calificado. En este caso, el controlador llama al Grader, el cual se comunicará con el entorno seguro específico para el lenguaje de la solución que se encargará de su ejecución y calificación.

- Cppsandbox: Contenedor Docker que funciona a modo de sandbox para ejecutar en entorno seguro las soluciones de los estudiantes recibidas en lenguaje C++. Este container cuenta con el compilador GNU G++ en su versión 5.4.0 (Este compilador se encuentra bajo licencia libre gplv3). Recibe las peticiones desde el Grader y toma la solución y las entradas desde las carpetas compartidas en “volume”.

Ajustes nuevos: Se realiza la actualización de la versión de ubuntu a la 20.04.

- Javasandbox: Contenedor Docker que funciona a modo de sandbox para ejecutar en entorno seguro las soluciones de los estudiantes recibidas en lenguaje java. Este container cuenta con el compilador incluido en el JDK (Java Development Kit) en su versión 8

(Compartido bajo licencia libre Oracle Binary Code License). Recibe las peticiones desde el Grader y toma la solución y las entradas desde las carpetas compartidas en “volume”.

Ajustes nuevos: Se realiza la actualización de la versión de ubuntu a la 20.04.

- **Pythonsandbox:** Contenedor Docker que funciona a modo de sandbox para ejecutar en entorno seguro las soluciones de los estudiantes recibidas en lenguaje python. Este container cuenta con el compilador python en su versión 3 (Compartido bajo licencia libre GPL). Recibe las peticiones desde el Grader y toma la solución y las entradas desde las carpetas compartidas en “volume”.

Ajustes nuevos: Se realiza la actualización de la versión de ubuntu a la 20.04.

- **Mariadb:** Gestor de base de datos. La plataforma cuenta con una base de datos que almacena toda la información no estática de la aplicación usando este gestor.

Ajustes nuevos: En la anterior versión, el gestor de mariadb estaba almacenado dentro de una instancia de Docker, pero ahora, esta base de datos está administrada por medio de una instancia de máquina virtual monitoreada por servicios de Google Cloud. Esto se hizo con el fin de mitigar posibles vulnerabilidades de seguridad que han generado problemas de borrado de datos.

- **Volume:** Es un espacio de almacenamiento compartido. Aquí se almacenan los archivos estáticos de la plataforma, y de allí son tomados los códigos, entradas y salidas por los contenedores para realizar las calificaciones.

Ajustes nuevos: Se agrega la ruta “Blockly_svg_sources” para el manejo de los envíos realizados en lenguaje de bloques (Blockly); dentro de esa carpeta, se almacena los .svg con el formato de submissions hechos en esa herramienta.

Capa cliente

La capa del cliente es una aplicación web completa que se ejecuta en cualquier navegador web moderno. Cuenta con los siguientes componentes:

- **Entrypoint (index.html):** Punto de comunicación entre el navegador y la aplicación. Todas las solicitudes e interacciones de la aplicación son hechas a través de este componente.
- **Modules:** Los componentes (también llamados módulos) definen la lógica de la aplicación en el cliente. Se comunican en tiempo real con los templates.

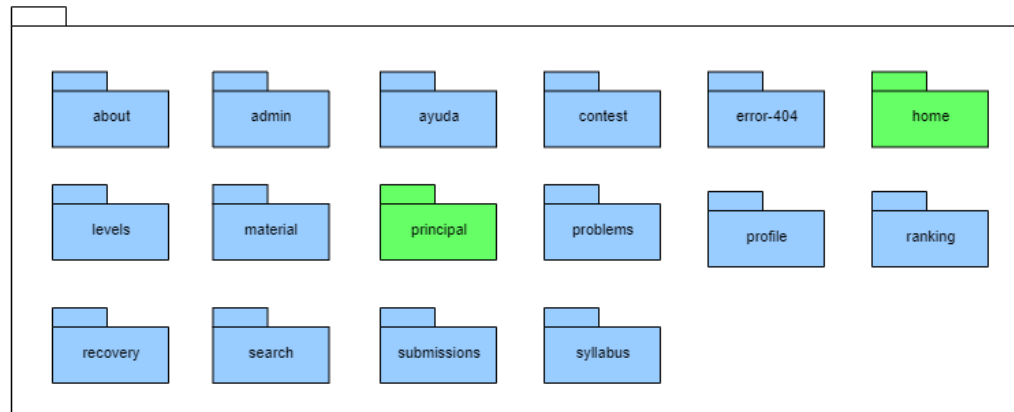
modules

Ilustración 53 Componente modules.

- Templates: Son unidades de diseño de interfaz de usuario conectados a los componentes para tener la información actualizada en tiempo real.
- Layouts: Los layouts definen plantillas de diseño que pueden ser reutilizadas por diferentes templates.

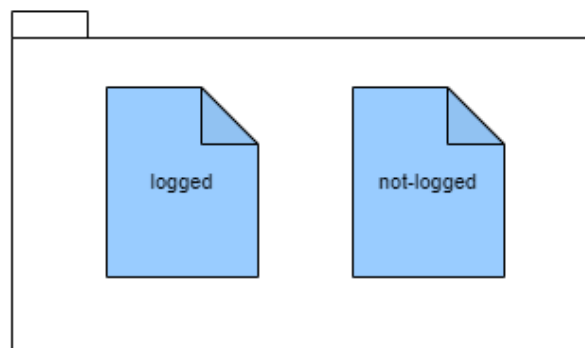
layouts

Ilustración 54 Componente layouts.

- Assets: Imágenes, estilos, y demás componentes gráficos.

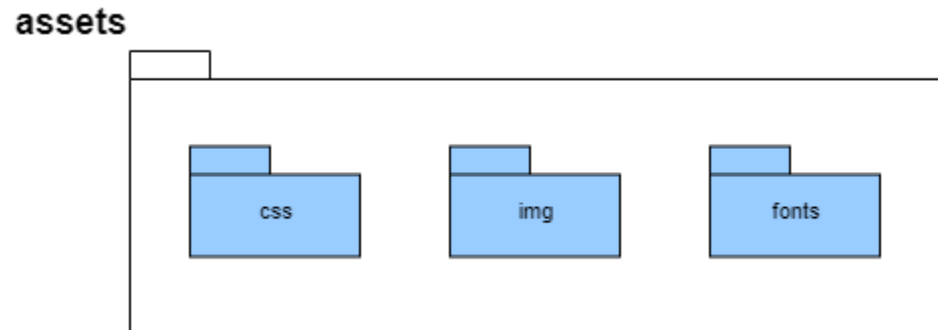


Ilustración 55 Componente de assets.

- Models: Los modelos permiten mantener una estructura de objetos en la aplicación cliente.

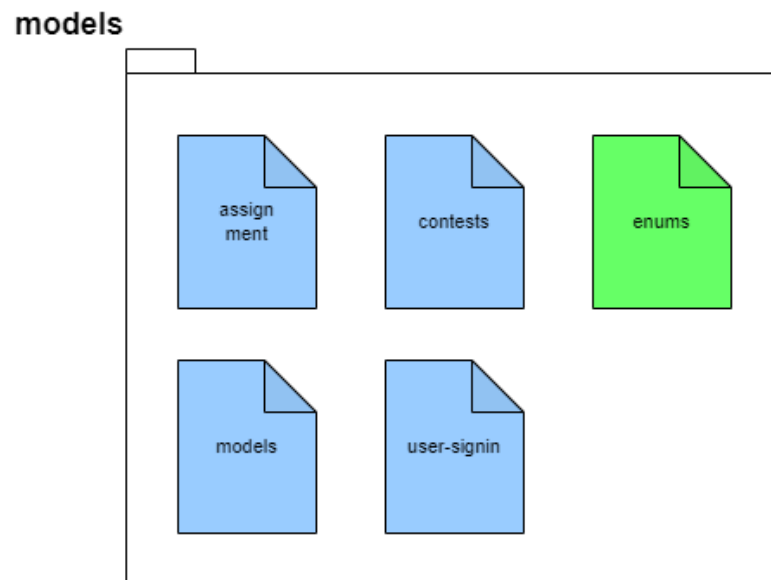


Ilustración 56 Componente models.

- Services: Servicios que permiten encapsular tareas repetitivas para evitar duplicaciones de código.

services

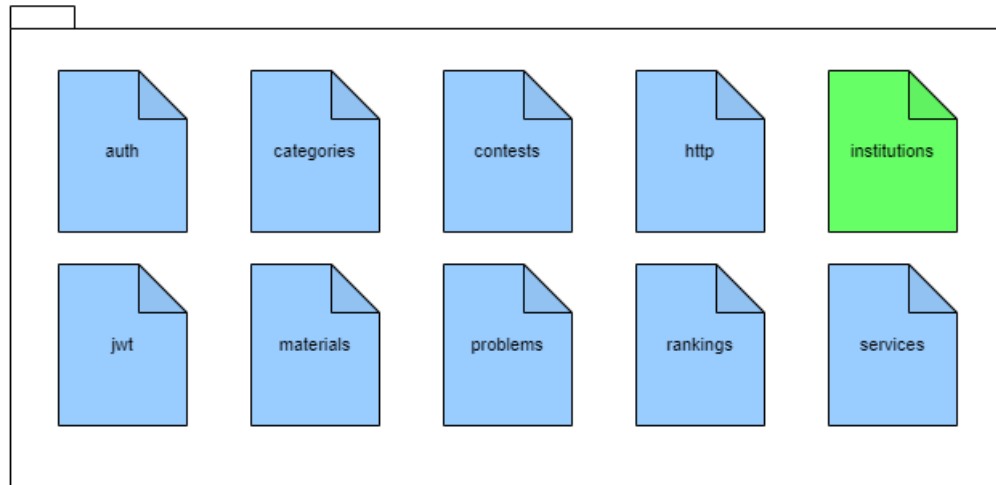


Ilustración 57 Componente services.

- Resources:

resources

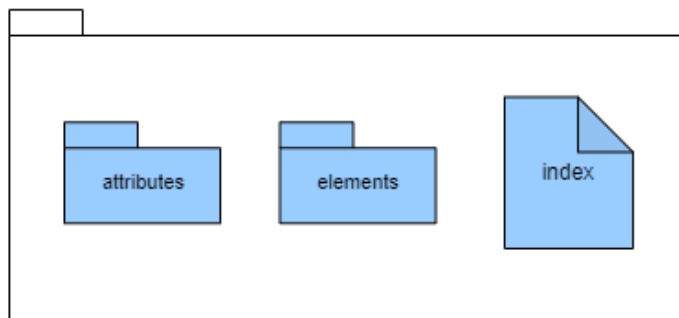


Ilustración 58 Componente resources.

- Elements : Los elementos personalizados son una herramienta que permite ser configurable, adaptable, y podrá ser utilizado en toda la aplicación.

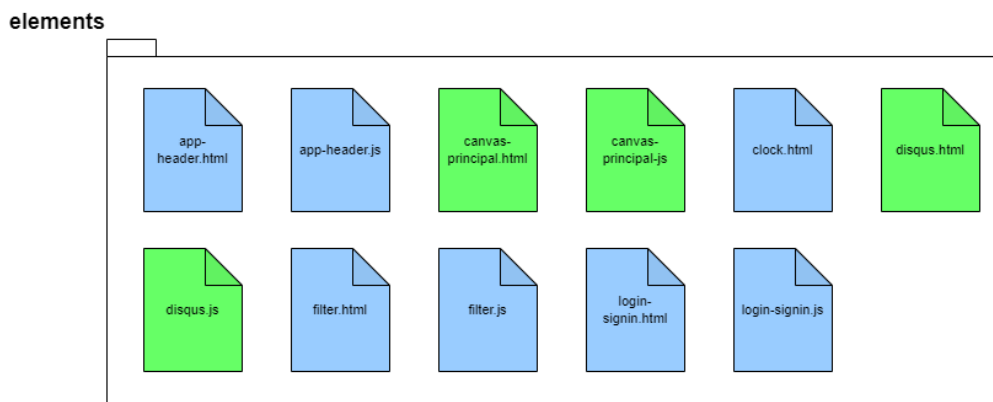


Ilustración 59 Componente elements.

- **Config:** Configuraciones generales de la aplicación. Estos componentes se relacionan de la siguiente manera: El usuario ingresa a la aplicación a través del endpoint (pasando primeramente por el proxy reverse), que inmediatamente lo dirige al componente solicitado (mediante un enrutamiento transparente realizado por las mismas configuraciones de los componentes). Cada componente define una lógica del negocio y tiene adjunto un template con el cual realiza un “data binding” que permite comunicación en tiempo real: los cambios realizados en el componente luego de un procesamiento o de una conexión al servidor (pasando por el proxy reverse) se despliegan inmediatamente en la interfaz, mientras los cambios ingresados por el usuario al template se actualizan inmediatamente al componente. Para esta labor el componente se basa en los models que definen el esqueleto de la aplicación, y los services que son utilidades transversales a la aplicación. Por su parte, los templates se ayudan de los layouts y los assets para desplegar la información correctamente y de forma agradable al usuario. Toda la ejecución sigue los parámetros establecidos en config.

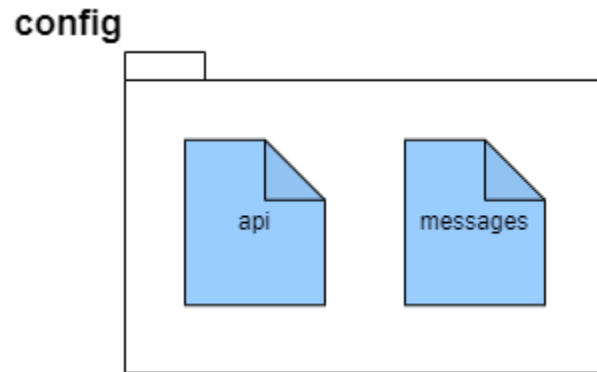


Ilustración 60 Configuración del componente front-end.

Cabe destacar que la comunicación con el servidor ocurre en los componentes, aunque para una mayor modularidad y simplicidad, generalmente estos componentes se apoyan en servicios para realizar estas tareas.

Comunicación entre cliente y servidor

La comunicación entre cliente y servidor se realiza utilizando REST (transferencia de estado

Representacional) utilizando JSON como el formato de medios transmitidos.

Esto nos permite una separación e interdependencia sencilla entre el cliente y el servidor, facilidad para posibles migraciones y en caso de posteriores aplicaciones (por ejemplo, aplicaciones móviles para complementar la plataforma) muy fácilmente podrían conectarse al mismo servidor, siendo necesario solo el desarrollo del cliente.

Por su parte, desde el navegador todas las conexiones se realizan utilizando la interfaz Fetch de javascript, en lugar del clásico xmlhttprequest. Si bien Fetch no tiene compatibilidad con navegadores antiguos (especialmente internet explorer), utilizar fetch trae como ventaja fundamental su apoyo directo en “promesas” que nos permite simplificar la aplicación en momentos de espera de las conexiones. Sin embargo, para no perder del todo la compatibilidad, se utiliza un polyfill desarrollado por github y compartido bajo código libre que permite la utilización de fetch en navegadores que no lo soportan nativamente.

Para la comunicación entre cliente y servidor, se agregó una capa intermedia, con una configuración del proxy reverso de Nginx. El proxy reverso de Nginx es un tipo de servidor proxy que normalmente se sitúa detrás del firewall en una red privada y dirige las peticiones de la capa cliente a la capa servidor. Un proxy reverso proporciona un nivel adicional de abstracción y control para garantizar el flujo fluido del tráfico de red entre los clientes y los servidores. Desde la perspectiva del cliente, éste carga el contenido del host al que accede. Sin embargo, NGINX carga el contenido real del servidor remoto y lo reenvía al cliente. Este es un nuevo componente incluido en la arquitectura de la aplicación.

Arquitectura encapsulada

En el diagrama de arquitectura general se puede observar cómo los componentes (excepto volumes que es un folder de intercambio) se encuentran definidos como contenedores docker. Esto permite encapsular fácilmente los diferentes componentes, permitiendo un despliegue más simple y controlado bajo cualquier hardware sin necesidad de excesivas configuraciones. De esta

manera, cada componente resulta en un contenedor independiente, que puede comunicarse con los demás con facilidad, pero abstrayendo sus operaciones a su espacio contenido. Del mismo modo, esto permite una mayor facilidad para la migración.

Escalabilidad

Hay dos cuestiones de escalabilidad que se han priorizado desde la creación de la plataforma. En primer lugar, si bien el flujo de usuarios se encuentra acotado por los estudiantes del grupo de estudio, de la carrera y en este momento de algunos colegios y universidades de la ciudad de Cúcuta y estos no son números muy altos, si debe considerarse que los usuarios utilizarán la plataforma generando operaciones pesadas (calificación de sus soluciones) que pueden tardar varios segundos de procesamiento. Estos procesamientos de larga duración no deben afectar el rendimiento de la plataforma. En segundo lugar, aunque originalmente se diseñó la plataforma para funcionar con java, C++, Python por ser los lenguajes actualmente aceptados en competencias, y en su nueva actualización el lenguaje visual Blockly para competencias y/o enseñanza de programación para colegios, sabemos que a futuro podría ser necesario incorporar nuevos lenguajes, bien sea porque las competencias de programación hacen oficial un nuevo lenguaje aceptado, o porque en la Universidad se desea trabajar un lenguaje diferente. Por tanto, siempre surge la necesidad de integrar fácilmente nuevos lenguajes a la plataforma. La primera cuestión es inmediatamente resuelta por la arquitectura. Al separar el front-end del back-end, cada usuario puede acceder a la aplicación en su computador sin importar la saturación que esté ocurriendo en el servidor. Pero no solo esto, sino que la aplicación del servidor también está separada (por un lado el back-end procesando las comunicaciones, y por otro los contenedores

sandbox ejecutando las soluciones de los usuarios). Esta separación permite que el back-end siga respondiendo las peticiones de los usuarios mientras la saturación queda únicamente en las calificaciones. Así los demás usuarios podrán seguir leyendo problemas, solucionándolos y utilizando la plataforma sin problema. Inevitablemente esto implica que las calificaciones pueden tardar unos segundos debido a la saturación, pero gracias al enfoque el usuario puede seguir trabajando normalmente mientras espera su calificación, y una vez esta es obtenida se despliega en tiempo real en su interfaz. La segunda cuestión es aún más arquitectural: Un primer enfoque permitiría tener un único contenedor Sandbox de calificación con las herramientas necesarias para la calificación en todos los lenguajes disponibles. Sin embargo, en el momento de intentar añadir nuevos lenguajes, inevitablemente sería necesario una reestructuración de todo el código para añadir las nuevas herramientas e instrucciones que el lenguaje requiera. Esto implica un gasto considerable de tiempo y recursos, así como mayores riesgos de errores inducidos en

El cambio. Justo por eso se planteó en primera versión de la plataforma, un contenedor independiente por cada lenguaje. De esta forma, en el momento de añadir un nuevo lenguaje, como es el caso presentado en esta extensión de la plataforma, basta con crear un nuevo contenedor con sus configuraciones y archivos necesarios, o adaptar un contenedor existente para calificar problemas en esa sintaxis base (como es el caso de Blockly convertido a lenguaje Python). Posteriormente se configura el componente Grader para indicarle que hay un nuevo lenguaje disponible, e inmediatamente después puede someterse a pruebas, pues ya estará listo para usar.

Modelo de clases

A continuación, se muestra el diagrama de clases de la aplicación (Ilustración 61).

A continuación, se muestra el diagrama de modelo de datos de la aplicación (ilustración 62).

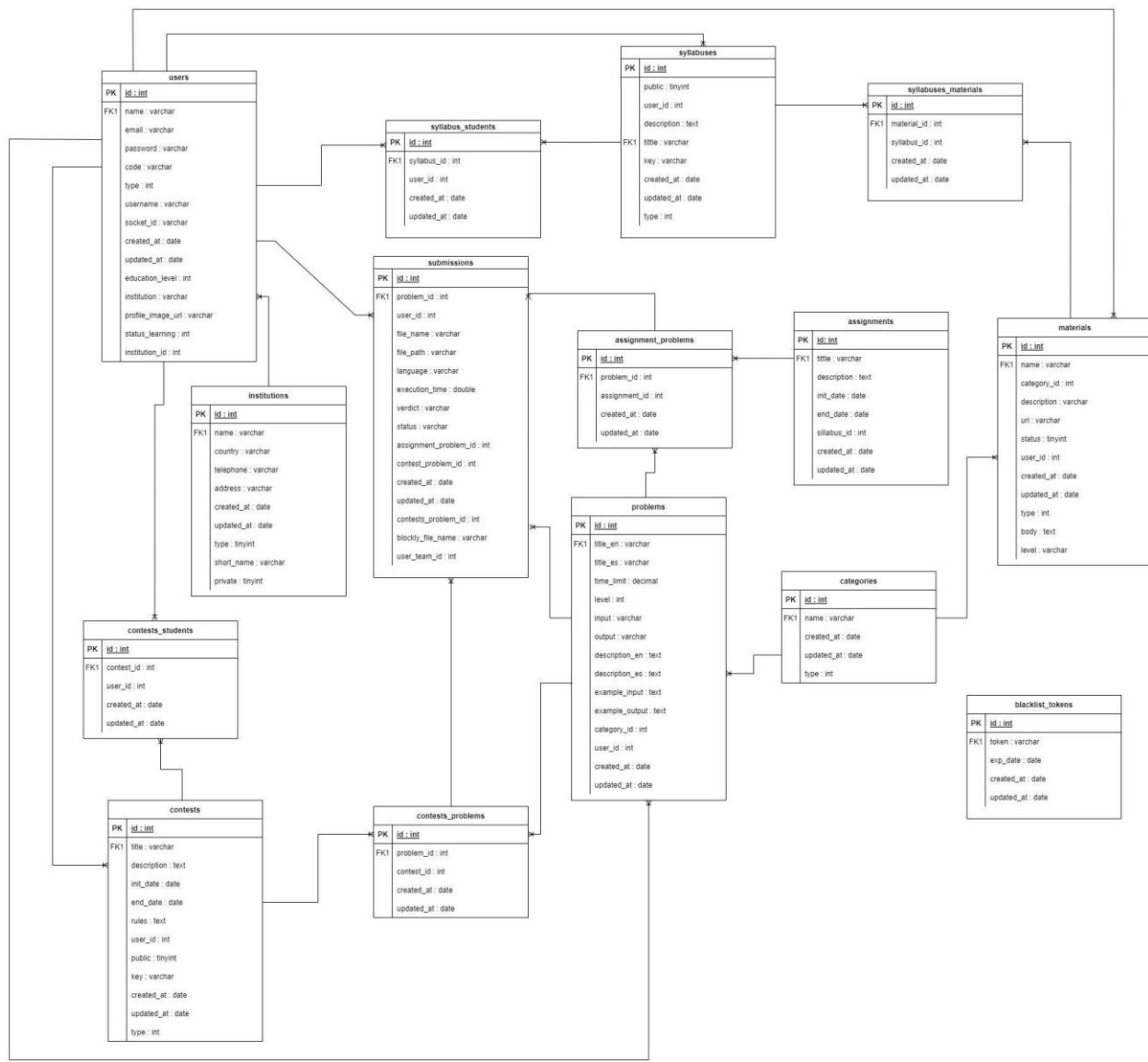


Ilustración 62 Modelo de Base de datos.

Para ver esta imagen más a detalle, ver los documentos y anexos digitales.

Diccionario de la base de datos

A continuación, se explican los modelos de base de datos de la plataforma Training Center UFPS, con las nuevas actualizaciones aplicadas.

Tabla 16 Modelo de datos para users.

Tabla	Users		
Descripción	Usuarios de la plataforma		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador del usuario.
Name	Varchar		Nombre del usuario.
Email	Varchar		Email del usuario.
Password	Varchar		Contraseña del usuario (guardada encriptada con SALT HASH)
Code	Varchar		Código del usuario en la Universidad.
Type	Int		0: estudiante 1: docente/coach 2: administrador

Username	Varchar		Nombre del usuario para mostrar.
Total_submissions	Int		Total de soluciones enviadas.
Accepted_submissions	Int		Total de soluciones correctas.
Created_at	Datetime		Fecha de creación del usuario.
Updated_at	Datetime		Fecha de actualización del usuario.

Tabla 17 Modelo de datos para contest_students. Elaboración propia

Tabla	Contest_students		
Descripción	Tabla de transición que relaciona los estudiantes inscritos en una maratón específica.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la relación.
User_id	Int	FK	Identificador del usuario.
Contest_id	Int	FK	Identificador de la maratón.
Created_at	Datetime		Fecha de creación del registro.

Updated_at	Datetime		Fecha de última actualización del registro.
------------	----------	--	---

Tabla 18 Modelo de datos para contest.

Tabla	Contest		
Descripción	Maratón de programación		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la maratón
Title	Varchar		Título de la maratón.
Description	Varchar		Descripción de la maratón.
Init_date	Datetime		Fecha de inicio de la maratón.
End_date	Datetime		Fecha de finalización de la maratón.
Rules	Text		Reglas de la maratón
User_id	Int	FK	Usuario creador de la maratón.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 19 Modelo de datos para submission.

Tabla	Submissions		
Descripción	Soluciones enviadas a la plataforma.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la solución.
Problem_id	Int	FK	Identificador del problema solucionado.
User_id	Int		Identificador del estudiante que envía la solución.
File_name	Varchar		Nombre del archivo enviado.
File_path	Varchar		Ruta al archivo enviado.
Language	Varchar		Nombre del lenguaje de programación.
Execution_time	Double		Tiempo de ejecución de la solución.
Veredict	Varchar		Calificación dada por la plataforma.
Status	Varchar		Status del problema.

Assignment_proble m_id	Int	FK	Identificador del problema en la tarea (si aplica).
Contest_proble m_id	Int	FK	Identificador del problema en el contest (si aplica).
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 20 Modelo de datos para contest_problems.

Tabla	Contest_problems		
Descripción	Tabla de transición que asocia una maratón de programación con sus problemas.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la relación.
Problem_id	Int	FK	Identificador del problema.
Contest_id	Int	FK	Identificador de la maratón.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 21 Modelo de datos para syllabus_students.

Tabla	Syllabus_students		
Descripción	Tabla de transición que asocia clases con los estudiantes matriculados en ella.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la relación.
Syllabus_id	Int	FK	Identificador de la clase.
User_id	Int	FK	Identificador del estudiante.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 22 Modelo de datos para assignments_problems.

Tabla	Assignments_problems		
Descripción	Tabla de transición para asociar tareas con sus problemas.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la relación.
Problem_id	Int	FK	Identificador del problema.

Assignment_id	Int	FK	Identificador de la tarea.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 23 Modelo de datos para problems.

Tabla	Problems		
Descripción	Problemas almacenados en la plataforma.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador del problema.
Title_en	Varchar		Título en inglés.
Title_es	Varchar		Título en español.
Time_limit	Decimal		Tiempo límite del problema.
Level	Int		Nivel entre 1 (muy facil) y 10 (muy dificil).
Input	Varchar		Ruta a la entrada del problema.
Output	Varchar		Ruta a la salida del problema.
Description_en	Varchar		Descripción del problema en inglés.

Description_es	Varchar		Descripción del problema en español.
Example_input	Text		Entrada de ejemplo.
Example_output	Text		Salida de ejemplo.
Category_id	Int	FK	Identificador de la categoría del problema.
User_id	Int	FK	Identificador del autor del problema
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 24 Modelo de datos para syllabus.

Tabla	Syllabus		
Descripción	Clases almacenadas en la plataforma		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la clase.
Title	Varchar		Título de la clase.
Description	Varchar		Descripción de la clase.
User_id	Int	FK	Identificador del docente/coach creador de la

			clase.
Public	Tinyint		0: clase privada 1: clase pública
Key	Varchar		En caso de ser privada, el docente/coach asigna una clave que los estudiantes deben conocer para acceder. Esta se guarda en este campo.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 25 Modelo de datos para assignments.

Tabla	Assignments		
Descripción	Tareas almacenadas en la plataforma		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la tarea.
Title	Varchar		Título de la tarea.
Description	Varchar		Descripción de la tarea.
Init_date	Datetime		Fecha de inicio de la tarea.

End_date	Datetime		Fecha de finalización de la tarea.
Syllabus_id	Int	FK	Syllabus al cual pertenece la tarea.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 26 Modelo de datos para syllabus_materials.

Tabla	Syllabus_materials		
Descripción	Tabla de transición que asocia materiales con el syllabus al que pertenecen		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la relación.
Material_id	Int	FK	Identificador del material.
Syllabus_id	Int	FK	Identificador del syllabus.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 27 Modelo de datos para categories.

Tabla	Categories		
Descripción	Categorías de ejercicios y materiales		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador de la categoría.
Name	Varchar		Nombre de la categoría.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 28 Modelo de datos para materials.

Tabla	Materials		
Descripción	Materiales disponibles en la plataforma.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Identificador del material.
Name	Varchar		Nombre del material.
Category_id	Int	FK	Id de la categoría a la cual pertenece el

			material.
User_id	Int	FK	Id del usuario que subió el material a la plataforma.
Description	Varchar		Descripción del material.
URL	Varchar		URL del material.
Status	Tinyint		1: Interno 2: Externo
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de última actualización del registro.

Tabla 29 Modelo de datos para blacklist_tokens.

Tabla	Blacklist_tokens		
Descripción	Lista de tokens de usuario que han sido revocados.		
Campo	Tipo	Adicional	Descripción
Id	Int	PK	Id del token
Token	Varchar		Token JWT.

Exp_date	Datetime		Fecha de vencimiento del token.
Created_at	Datetime		Fecha de creación del registro.
Updated_at	Datetime		Fecha de actualización del registro.

Tabla 30 Modelo de datos para institutions.

Tabla	Institutions		
Descripción	Instituciones educativas almacenadas		
Campo	Tipo	Adicional	Descripción
Id	Varchar	PK	Id de la institución
Name	Varchar		Nombre de la institución
Country	Varchar		País de la institución
Telephone	Varchar		Teléfono
Address	Varchar		Dirección
Created_at	Date		Fecha de creación del registro.
Updated_at	Date		Fecha de actualización del registro.

Type	Tinyint		Tipo de institución
Short_name	Varchar		
Private	Tinyint		

Módulos de la plataforma

La plataforma Training Center UFPS, en su nueva versión está definida en los siguientes módulos:

Tabla 31 Módulos de la plataforma.

Módulo	Descripción
Inicialización y autenticación	Inicialización del esquema, diseño y estructura de la aplicación, y autenticación de los usuarios en la plataforma. Permite que los usuarios de diferentes instituciones educativas se registren y autenticuen.
Materiales y guías de estudio	Creación, actualización y revisión de materiales y guías de estudio, y despliegue a los usuarios para su visualización. Estos materiales se pueden dividir dependiendo si está orientado para universidades o colegios.

Administración de la plataforma	Herramienta para la administración de usuarios, problemas y guías en la plataforma (solo usuarios administradores). También permite inspeccionar códigos de soluciones realizadas por los usuarios.
Estadísticas y perfiles de usuario	Módulo para desplegar estadísticas, rankings, perfiles de usuario, y resultados a los docentes/coach.
Problemas	Creación, edición y eliminación de problemas, visualización de los mismos y motor de calificación. Están categorizados en problemas de tipo universidad o de tipo colegio.
Modo guiado/syllabus	Módulo para la supervisión de los docentes/coach al trabajo de los estudiantes.
Competencias (maratones)	Módulo para la creación, administración y desarrollo de maratones de programación, para universidades o colegios, así como para la ejecución de las mismas.

6.4. Experiencia de desarrollo y despliegue de integración continua

Durante del desarrollo de este proyecto se realizaron mejoras para el despliegue e integración de nuevos ajustes, ya que se hacía fundamental hacer que los pasos de despliegue a servidor fueran menos complicados de hacer; como primera medida, se usó Git como control de

versiones (que ya se usaba desde los inicios de la plataforma), esto permite poder acceder a diferentes escenarios y momentos de la historia de cambios de la aplicación, y revisar su funcionamiento; usando la línea de comandos, o una interfaz gráfica, se pueden documentar y crear commits de los nuevos cambios para evidenciar que las tareas de un sprint están finalizadas (Ilustración 63).

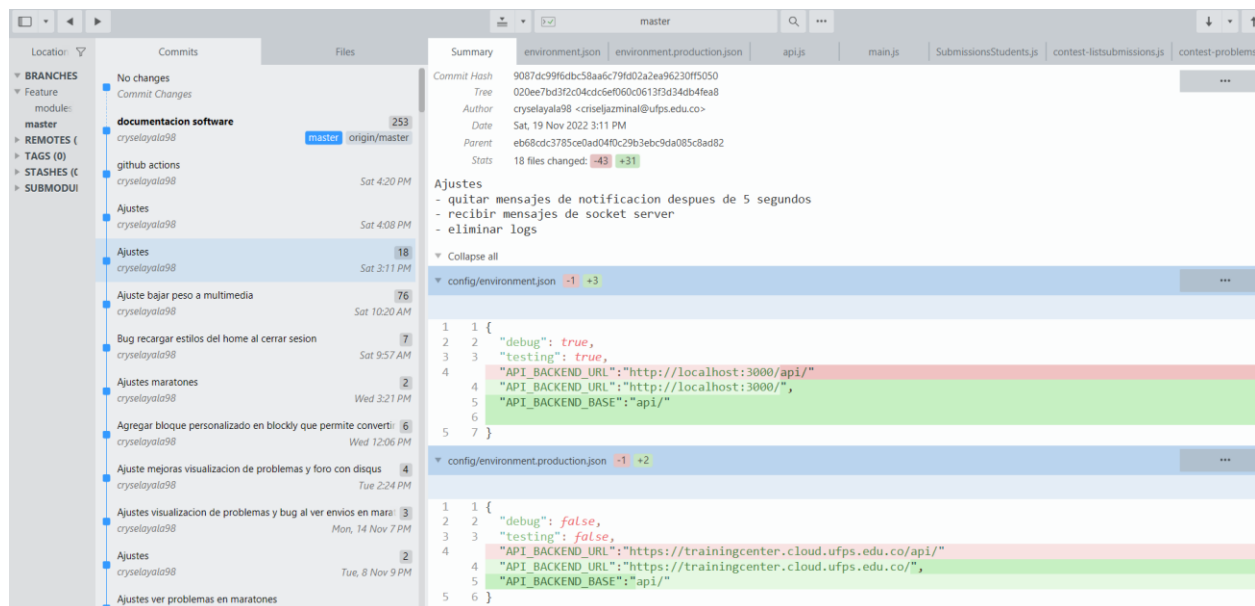


Ilustración 63 Control de versiones.

A su vez, se creó un release de Github, que permite acceder al código fuente de la primera versión de la plataforma Training Center, esto con el objetivo de que cualquier usuario pueda instalarla y probar características propias de esa versión (ilustración 64).

Releases / Training_Center_Version_1

The screenshot shows a GitHub release page for 'Training Center'. At the top, it says 'Training Center' with a 'Latest' badge. Below that, it indicates that 'cryselayala98' released this on May 23, with a commit hash of 'bd47220'. The release name is 'Training_Center_Version_1'. Under the 'Assets' section, there are two items: 'Source code (zip)' and 'Source code (tar.gz)', both dated May 23.

Ilustración 64 Código fuente primera versión de Training Center UFPS.

Por medio de la herramienta Github Actions, se pudieron automatizar algunas tareas repetitivas del flujo de trabajo, como, por ejemplo, la creación de pruebas de calidad de código al momento de realizar commits de nuevos cambios en la aplicación. Se crearon una serie de scripts que tienen como objetivo, subir cambios en el ambiente respectivo (desarrollo, producción) y desplegarlos en el servidor donde está alojada la plataforma; estos scripts hacen parte del código fuente de la misma.

Integración con otros proyectos en paralelo

Durante el primer semestre del año 2022, 3 estudiantes de la asignatura Seminario Integrador, realizaron un proyecto, basado en ajustes a la plataforma Training Center, para permitir la gestión de instituciones educativas, filtros más personalizados, y visualización de códigos desde usuario admin; usando control de versiones (git) se logró unificar los cambios

hechos para el desarrollo del proyecto a la nueva versión de Training Center, sin ningún inconveniente.

6.5. Pruebas de software

Para verificar la calidad del software UFPS Training Center V.2.0, se somete a varias pruebas, a través de diferentes estrategias para garantizar que el sistema cumple con éxito los requisitos recomendados y que el código fuente entregado es óptimo para futuras modificaciones.

6.5.1. Alcance de las pruebas

El alcance de desarrollo de pruebas incluye la ejecución y documentación de todas las pruebas existentes de los requerimientos funcionales del sistema, así como la identificación y análisis del rendimiento y el nivel de seguridad que proporciona, así como la usabilidad percibida por los usuarios del mismo.

6.5.2. Propósito de la evaluación

La presente evaluación a la plataforma se realiza para detectar errores y corregirlos antes de su puesta en marcha con usuarios reales, para verificar su correcto funcionamiento, analizar el rendimiento y resolver las alertas de seguridad que puedan hacer vulnerable al sistema ante potenciales ataques.

6.5.3. Motivadores de la prueba

Las pruebas presentadas a continuación buscan reducir el índice de fallas de la plataforma una vez puesta en marcha. Resulta casi inevitable cometer errores durante el desarrollo, pero contar con un conjunto de pruebas que detectan estos errores a tiempo permite corregir estos errores a tiempo, antes de salir a producción.

6.5.4. Objetos a ser Evaluados

La plataforma está compuesta por 3 partes que serán evaluadas de forma independiente, y como conjunto:

- **Aplicación front-end:** Aplicación que se comunica con el usuario a través del navegador. Las pruebas realizadas sobre este componente buscan verificar que las interacciones que los usuarios deben realizar tengan los comportamientos deseados.
- **Aplicación back-end:** Aplicación de servidor que se encarga de procesar las peticiones de los usuarios y darles respuesta. Las pruebas realizadas sobre este componente buscan verificar que los procesamientos realizados arrojan las respuestas correctas y esperadas por el usuario.
- **Aplicación Grader:** Aplicación que se conecta con el back-end, y que se encarga de la ejecución de los códigos en un entorno seguro. Las pruebas a este componente se centran en comprobar la correcta calificación de los envíos de los estudiantes (tanto cuando estos

envíos son correctos, como cuando contienen fallan), así como verificar la seguridad del entorno en el cual se ejecutan estas soluciones.

6.5.5. Ámbito de las pruebas

El conjunto de pruebas a realizar busca cubrir el mayor porcentaje posible de código (idealmente el 100% del mismo) para poder garantizar un óptimo funcionamiento de las diferentes partes que componen la plataforma.

Adicionalmente, pero no menos importante, se analizan diferentes métricas que permitan detectar errores de forma general, y evaluar las diferentes características de la plataforma.

Las pruebas a las que es sometido el software entregado son:

- Pruebas de aceptación de software
- Análisis de código estático con SonarQube
- Análisis de rendimiento de plataforma

Para las pruebas manuales se toma cada flujo de la aplicación y se describen los criterios de aceptación para dar cómo aceptada dicha prueba, finalmente se agregan las evidencias. En el análisis de código estático con SonarQube se definen unas métricas de calidad (Quality Gates) que debe superar cómo mínimo los resultados del análisis para decir que el código fuente es de

calidad. Las pruebas de rendimiento web permiten monitorear la velocidad y peso de las peticiones de recursos front-end, para brindar la mejor experiencia de usuario posible. Por otro lado, también correlacionan pruebas que permitan identificar el nivel de seguridad que ofrece la aplicación y la facilidad de uso que ofrece.

6.5.6. Lista de ideas de las pruebas

- Evaluar en los casos de uso especificados el flujo correcto y los flujos alternos de la aplicación
- Evaluar el correcto cumplimiento de los requerimientos del software.
- Cumplir con los estándares de los lenguajes y tecnologías utilizadas.
- Evaluar aspectos de calidad del software.
- Evaluar el rendimiento de la plataforma.
- Analizar aspectos de seguridad para detectar posibles vulnerabilidades que puedan poner en riesgo información delicada dentro de la aplicación.

6.5.7. Enfoque de las pruebas

Se realizan sobre la plataforma:

- Evaluación de métricas del software con herramientas externas.
- Pruebas de aceptación
- Pruebas de rendimiento web

6.5.8. Herramientas para las pruebas

- **SonarQube:** Plataforma para la evaluación de código fuente que permite analizar código duplicado, estándares de codificación, pruebas unitarias, cobertura de código, complejidad ciclomática, potenciales errores, comentarios y diseño del software.
- **Yellow Lab Tools:** Es un proyecto de código abierto que permite probar una página web (a través de una URL) y detecta problemas de rendimiento y de calidad del código front-end.

6.5.9. Métricas con SonarQube

Las métricas de SonarQube son aplicadas al front-end y al back-end (No al Grader pues contiene una combinación de tecnologías no soportada por la herramienta).

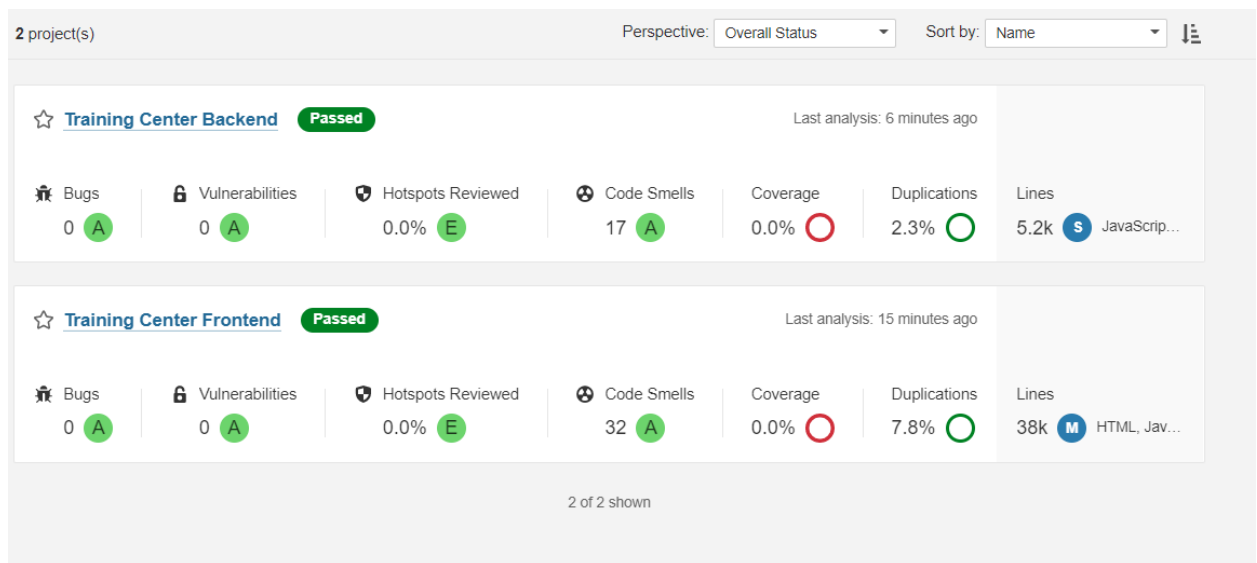


Ilustración 65 pruebas con SonarQube.

Análisis de SonarQube:

Bugs: 0, calificación A en ambos componentes. La primera evaluación realizada detectó 7 bugs en el Back-end y 23 en el Front-end colocándolos en calificación. Gracias a las propias indicaciones de SonarQube estos bugs fueron eliminados de la plataforma.

Vulnerabilidades: SonarQube no detectó vulnerabilidades en los componentes. Por tanto la calificación es A.

Errores de mantenibilidad: Aunque se detectaron algunos errores de mantenibilidad (17 en el back-end y 32 en el front-end), la calificación sigue siendo A, dado que son errores menores. De cualquier forma, se plantea la posibilidad de la eliminación de dichos errores para futuras actualizaciones.

Cobertura de las pruebas: La configuración utilizada en SonarQube no se ha integrado con las pruebas de software. Ver el siguiente numeral para analizar las pruebas.

Duplicaciones de código: SonarQube ha detectado un 2.3% de duplicaciones de código en el back-end, y un 7.8% en el front-end. Si bien en parte estos porcentajes son debidos a elementos similares inherentes a los frameworks utilizados, se plantea reducir hasta por lo menos 5% en futuras actualizaciones.

Líneas de código: El Back-end cuenta con ms de 5200 líneas de código y el front-end con mas de 32000 según la estimación de SonarQube (tener en cuenta el uso de librerías externas).

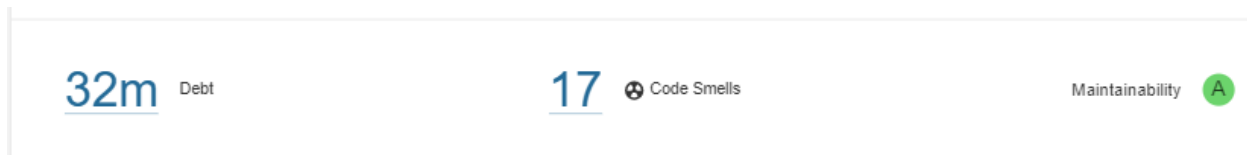


Ilustración 66 Análisis de líneas de código back-end

Para el back-end, SonarQube calcula que los errores en la mantenibilidad del código pueden llevar hasta 32 minutos de “deuda técnica”, es decir, el tiempo de refactorización para corregir estos errores. Es un índice bueno, y sigue manteniendo una calificación ‘A’, la más alta posible. Aún así, posteriores actualizaciones deben intentar reducir el margen en la medida de lo posible.

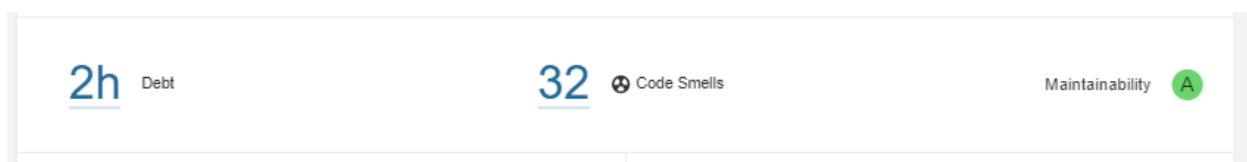


Ilustración 67 Análisis de líneas de código front-end

Para el front-end esta deuda técnica es moderada, de 2 horas.

Por su parte, la complejidad ciclomática del front-end es de 833 mientras la del back-end es de 646.

6.5.10. Pruebas de rendimiento web

Global score



77/100



Ilustración 68 Pruebas de rendimiento web

Yellow Lab Tools reporta una optimización de 77/100, calificada como “buena”. No se analiza la versión móvil pues, aunque la plataforma carga en celulares, no ha sido diseñada pensando en tal fin (Esto porque resultaría incómodo llevar a cabo maratones de programación desde un celular).

Detalle del análisis: En general, los aspectos relacionados al detalle de rendimiento web son bastante aceptables, los contenidos front-end que se descargan en el navegador (HTML, CSS, Js, Json y recursos multimedia) son ligeros, las peticiones al cliente no se demoran más de lo esperado, la complejidad del código fuente es bastante baja, y se están manejando librerías en su versión más actualizada (Ilustración 69).

D	Page weight	● Total weight	3.38 MB
		● Image optimization	6.8 KB
		● Oversized images	2
		● Gzip/Brotli compression	311 KB
		● File minification	861 bytes
A	Requests	● Requests number	75
		● Different domains	6
		● 404 not found	0
		● Identical content	0
		● Empty requests	0
		● Below the fold images	4
		● Hidden images	1
A	DOM complexity	● DOM elements count	255
		● DOM max depth	14
		● Number of iframes	0
		● IDs duplicated	2
A	JS complexity	● Total JS execution time	128 ms
		● DOM access	188
		● Scroll events bound	1
A	Bad JS	● JavaScript errors	0
		● document.write calls	0
		● Synchronous Ajax requests	0
		● Global variables	15
B	jQuery	● jQuery version	2.2.4
		● Several jQuery loaded	1
B	CSS complexity	● Rules count	5877
		● Complex selectors	11
		● Colors count	171
		● Similar colors	7
		● Breakpoints count	8
		● Not mobile-first media queries	12
B	Bad CSS	● CSS syntax error	0
		● Uses of @import	0
		● Duplicated selectors	7
		● Duplicated properties	12
		● Empty rules	0
		● Uses of !important	1052 ⚠
		● Old IE fixes	6
		● Old prefixes	249
		● Redundant body selectors	0
		● Redundant tags selectors	7
C	Web fonts	● Webfonts number	3
		● Overweighted webfonts	201 KB ⚠
		● Unused Unicode ranges	3
		● WOFF 2	4.0 KB
C	Server config	● HTTP protocols	60
		● TLS protocols	0
		● Connections closed	0
		● Caching not specified	62 ⚠
		● Caching disabled	0
		● Caching too short	2

Ilustración 69 Detalle del análisis de rendimiento web.

6.5.11. Pruebas de aceptación de software

Para el desarrollo de estas pruebas se llevó a cabo la prueba del correcto funcionamiento de cada uno de los requerimientos funcionales establecidos en la fase de análisis y que los

clientes o interesados determinen su aceptación. Originalmente diversas pruebas dieron resultados negativos, gracias a lo cual se pudieron corregir a tiempo fallos específicos en el código fuente.

Formato de las pruebas

Tabla 32 Formato de las pruebas

Formato de pruebas	
Identificador	Valores identificadores únicos para la prueba.
Nombre	Nombre de la prueba asociada.
Contexto	Entorno del aplicativo web en donde se realiza la prueba.
Tipo de prueba	Tipo de prueba realizada.
Estado de la prueba	Estado obtenido al realizar la prueba.
Descripción de la prueba	Descripción de forma específica del entorno y realización de la prueba.
Datos de entrada	Datos de entrada que deberá recibir el software para realizar la prueba.
Criterios de	Criterios que permitirán establecer el éxito o no de la prueba.

aceptación	
Resultado esperado	Información que se espera responda el software al momento de realizar la prueba.
Resultado obtenido	Información presentada por el aplicativo como respuesta a la prueba.
Observaciones	Información anexa acerca del desarrollo de la prueba.
Evidencias	
Evidencias (capturas de pantalla) presentadas para justificar la realización de la prueba especificada.	

Tabla 33 Formato de pruebas CP_01

Formato de pruebas	
Identificador	CP_01
Nombre	Implementar Blockly
Contexto	Se realiza la prueba desde el módulo de problemas y competencias de la plataforma.
Tipo de prueba	Estándar

Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Implementar Blockly ”.
Datos de entrada	N/A
Criterios de aceptación	El usuario debe acceder la sección de tablero de Blockly desde el módulo de problemas para colegios y maratones de programación.
Resultado esperado	<ul style="list-style-type: none"> - El sistema debe mostrar el tablero de Blockly en el módulo de problemas para colegios - El sistema debe mostrar el tablero de Blockly en el módulo de competencias para colegios
Resultado obtenido	El sistema carga la página de ver problemas (también en maratones) con el tablero de Blockly.
Observaciones	Ninguna.
Evidencias	

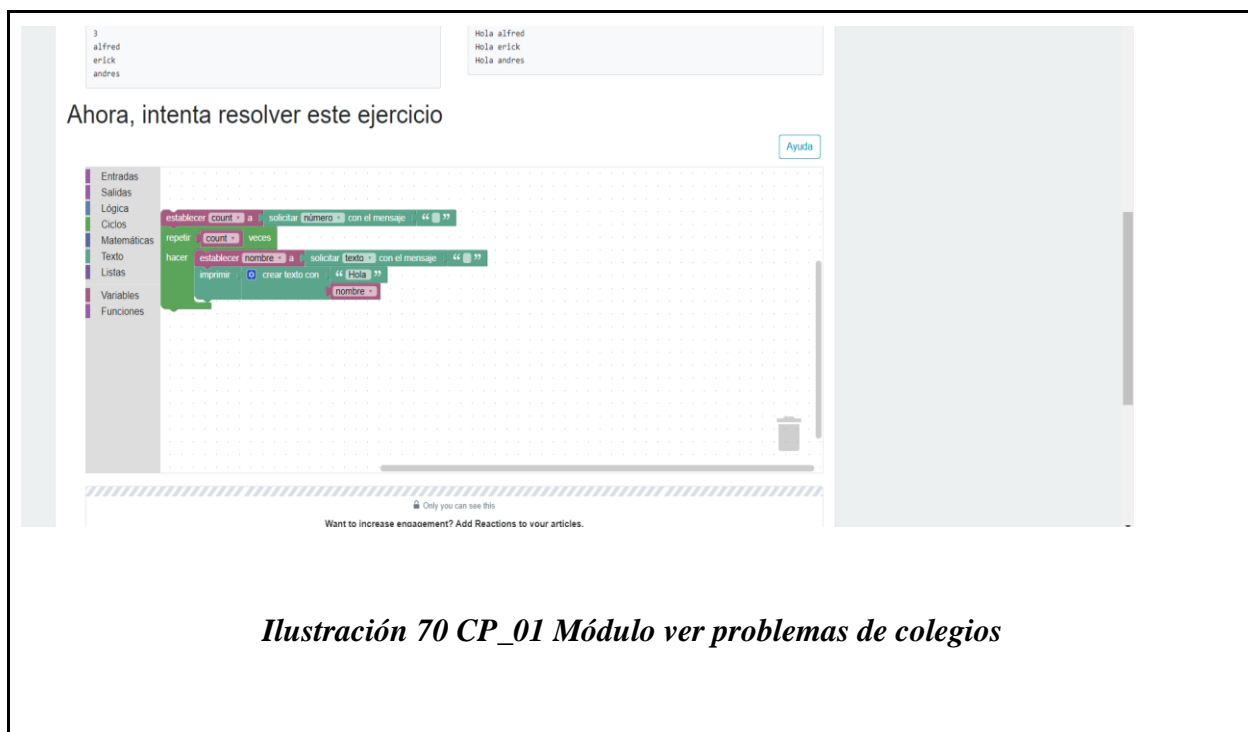


Ilustración 70 CP_01 Módulo ver problemas de colegios

Tabla 34 Formato de pruebas CP_02

Formato de pruebas	
Identificador	CP_02
Nombre	Listar problemas
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.

Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Listar problemas”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe desplegar al usuario la lista de problemas almacenados en la plataforma con la posibilidad de filtrar bajo diferentes criterios.
Resultado esperado	- El sistema debe desplegar al usuario la lista de problemas almacenados en la plataforma con la posibilidad de filtrar bajo diferentes criterios.
Resultado obtenido	El sistema puede desplegar al usuario la lista de problemas almacenados en la plataforma con la posibilidad de filtrar bajo diferentes criterios.
Observaciones	Ninguna.
Evidencias	

Categorías / Problemas de "Introducción Colegios"

Introducción Colegios

Mostrar 10 problemas en Cualquier idioma Ordenar por Id en forma Ascendente

Id	Problema	Dificultad	Idioma
1001	Hola mundo ✓	1	ES EN
1002	Hola Mundo Reloaded	1	ES EN
1003	Multiplicando por 11	1	ES EN
1004	Cociente y Residuo	1	ES EN
1005	Doble de un número	1	ES EN
1006	Mayor y Menor	1	ES EN
1007	Tabla de Multiplicar	1	ES EN

Inicio Anterior 1 Siguiete Final

Ilustración 71 CP_02 Listado de problemas

Tabla 35 Formato de pruebas CP_03

Formato de pruebas	
Identificador	CP_03
Nombre	Agregar problemas
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar

Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Agregar problemas”.
Datos de entrada	<ul style="list-style-type: none"> - Idioma: español o inglés. - Nombre: Nombre del problema. - Dificultad: Nivel de dificultad 1 (Fácil) a 10 (muy difícil). - Tipo de institución: Colegios o universidades. - Categoría: Listado de categorías ofrecidos dependiendo del tipo de institución (colegios o universidades). - Tiempo límite: 1 a 10 segundos. - Descripción del problema: Descripción general, formato de entrada y salida requeridos. - Entradas de ejemplo: Datos que cumplan con la descripción de las entradas. - Salidas de ejemplo: Datos que cumplan con la descripción de las salidas. - Casos de prueba: archivos que contienen datos de prueba muy grandes y sirven para calificar los envíos de los usuarios (en formato .in y .out).
Criterios de	El sistema debe permitir a los docentes y administradores añadir nuevos

aceptación	problemas para que los estudiantes los solucionen en lenguajes de programación (Java, C++, Python, Blockly).
Resultado esperado	El sistema debe permitir a los docentes y administradores añadir nuevos problemas para que los estudiantes los solucionen en lenguajes de programación (Java, C++, Python, Blockly).
Resultado obtenido	El sistema puede permitir a los docentes y administradores añadir nuevos problemas.
Observaciones	Ninguna.
Evidencias	

Nuevo problema

Idioma: Español

Tipo de institución: Elige una institución

Nombre: Nombre del problema

Categoría: Elige una categoría

Dificultad: 1

Tiempo límite: 1

Contenido

Descripción

Reemplaza este texto con la descripción de tu problema. Recuerda que puedes usar la sintaxis de Markdown.

Entradas

Reemplaza este texto con la especificación de la entrada de tu problema. Si no conoces la sintaxis markdown, puedes hacer uso de las opciones de la barra superior.

Salidas

El problema ha sido guardado correctamente

SALIDAS DE EJEMPLO se mostrarán junto al enunciado en la plataforma. Se recomienda que te las entradas y salidas. Por su parte, los casos de prueba **PRIVADOS** son los que utilizará la solución. Los casos de prueba privados pueden ser más extensos y completos (Dado que es posible que estos casos sean muy extensos, de .in)

Ilustración 72 CP_03 Vista agregar problemas

Ilustración 73 CP_03 Confirmación agregar problemas

Tabla 36 Formato de pruebas CP_04

Formato de pruebas	
Identificador	CP_04

Nombre	Corregir problemas
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Corregir problemas”.
Datos de entrada	<ul style="list-style-type: none"> - Idioma: español o inglés. - Nombre: Nombre del problema. - Dificultad: Nivel de dificultad 1 (Fácil) a 10 (muy difícil). - Tipo de institución: Colegios o universidades. - Categoría: Listado de categorías ofrecidos dependiendo del tipo de institución (colegios o universidades). - Tiempo límite: 1 a 10 segundos. - Descripción del problema: Descripción general, formato de entrada y salida requeridos. - Entradas de ejemplo: Datos que cumplan con la descripción de las entradas. - Salidas de ejemplo: Datos que cumplan con la descripción de las salidas.

	Casos de prueba: archivos que contienen datos de prueba muy grandes y sirven para calificar los envíos de los usuarios (en formato .in y .out).
Criterios de aceptación	El sistema debe permitir corregir problemas cuando el usuario lo vea necesario.
Resultado esperado	El sistema debe permitir a los docentes y administradores editar problemas.
Resultado obtenido	El sistema puede permitir a los docentes y administradores editar problemas ya creados y al visualizarlos, poder ver el formato con los nuevos cambios.
Observaciones	Ninguna.
Evidencias	

Nuevo problema

Idioma: Tipo de institución:

Nombre: Categoría:

Dificultad: Tiempo límite:

Contenido

Descripción
Reemplaza este texto con la descripción de tu problema. Recuerda que puedes usar la sintaxis de Markdown.

Entradas
Reemplaza este texto con la especificación de la entrada de tu problema. Si no conoces la sintaxis markdown, puedes hacer uso de las opciones de la barra superior.

Salidas

Ilustración 74 CP_04 Opción editar problema

Tabla 37 Formato de pruebas CP_05

Formato de pruebas	
Identificador	CP_05
Nombre	Eliminar problemas
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar
Estado de la	Finalizada.

prueba	
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Eliminar problemas”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir a los docentes y administradores eliminar problemas.
Resultado esperado	El sistema debe permitir a los docentes y administradores eliminar problemas de forma exitosa.
Resultado obtenido	El sistema puede permitir a los docentes y administradores eliminar problemas.
Observaciones	Ninguna.
Evidencias	

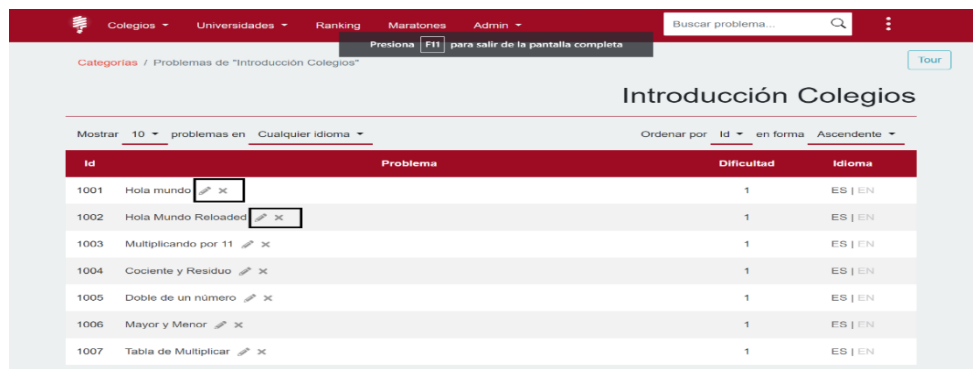


Ilustración 75 CP_05 Opción eliminar problemas

The screenshot shows a web interface for a problem list. At the top, there are filters: 'Mostrar 10 problemas en Cualquier idioma' and 'Ordenar por Id en forma Ascendente'. Below this is a table with columns 'Id', 'Problema', 'Dificultad', and 'Idioma'. A modal dialog box is open in the center, asking '¿Estás seguro de eliminar este problema?' with the subtext 'Esta acción no puede ser revertida' and two buttons: 'Si' and 'No'.

Id	Problema	Dificultad	Idioma
1	Hola mundo	1	ES EN
6	Divisores	1	ES EN
148	El mayor	1	ES EN
468	Echo Problem 20	9	ES EN
472	Doble de un número	1	ES EN
475	Punto y Circunferencia	1	ES EN
476	Tienda	1	ES EN

Ilustración 76 CP_05 Mensaje de advertencia eliminar problema

The screenshot shows the same problem list as in Illustration 75. A teal-colored message box is overlaid on the table, displaying 'Problema' and 'El problema se ha eliminado correctamente'. The message box has a close button (X) in the top right corner. The table below the message shows the 'Dificultad' column with values 1, 1, and 1.

Problema	Dificultad
El problema se ha eliminado correctamente	1
	1
	1

Ilustración 77 CP_05 Confirmación eliminar problema

Tabla 38 Formato de pruebas CP_06

Formato de pruebas	
Identificador	CP_06
Nombre	Acceder al foro de discusión
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Acceder al foro de discusión”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir acceder al foro de discusión de DISQUS para todos los problemas de la plataforma, comentar y leer comentarios de otros usuarios.
Resultado esperado	El sistema debe permitir acceder al foro de discusión para cada problema.
Resultado	El sistema puede permitir a los usuarios acceder al foro de DISQUS.

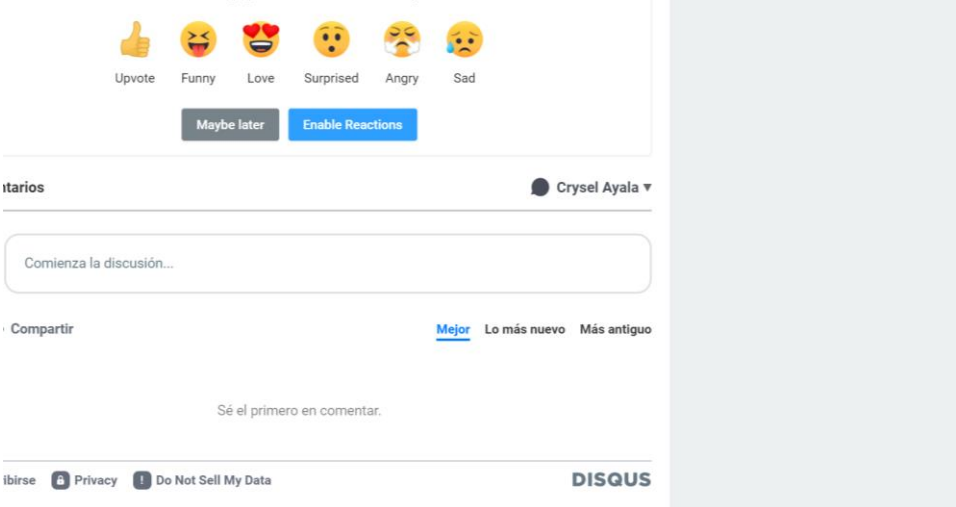
obtenido	
Observaciones	Ninguna.
Evidencias	
 <p><i>Ilustración 78 CP_06 Foro de discusión DISQUS por problema</i></p>	

Tabla 39 Formato de pruebas CP_07

Formato de pruebas	
Identificador	CP_07

Nombre	Filtrar problemas
Contexto	Se realiza la prueba desde el menú principal de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Filtrar problemas”.
Datos de entrada	- Filtro: Texto que contiene el parámetro de búsqueda de problemas.
Criterios de aceptación	El sistema debe permitir que los usuarios filtren los problemas por nombre para módulo colegios y módulo universidad.
Resultado esperado	El sistema debe permitir a usuarios filtrar problemas.
Resultado obtenido	El sistema puede permitir a los usuarios filtrar problemas.
Observaciones	Ninguna.
Evidencias	

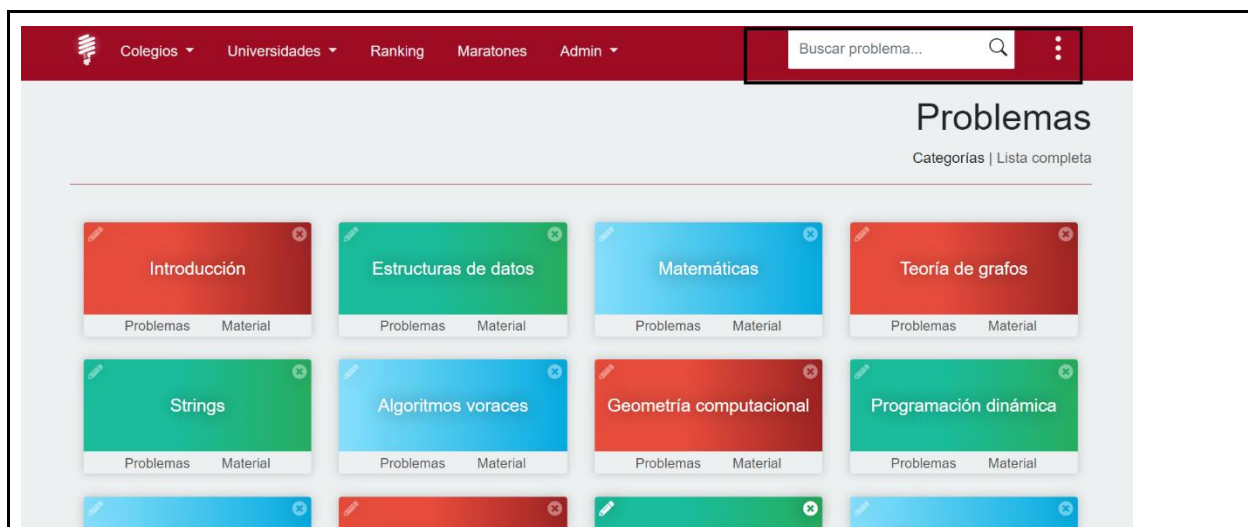


Ilustración 79 CP_07 Opción buscar problemas

The screenshot shows the search results page for the query 'Hola'. The header is dark red and contains the same navigation links as the previous page. The search bar now contains the text 'Hola'. Below the header, the page title 'Resultados para "Hola"' is displayed. The page includes filters for 'Mostrar 10 problemas en Cualquier idioma' and 'Ordenar por Id en forma Ascendente'. A table with five columns (Id, Problema, Tipo, Dificultad, Idioma) displays the search results.

Id	Problema	Tipo	Dificultad	Idioma
0	Hola mundo	Universidad	1	ES EN
1	Hola mundo	Colegio	1	ES EN
392	Hola Mundo 2.0	Universidad	2	ES EN
1001	Hola mundo	Colegio	1	ES EN

Ilustración 80 CP_07 Resultados de la búsqueda

Tabla 40 Formato de pruebas CP_08

Formato de pruebas	
Identificador	CP_08
Nombre	Categorizar problemas
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Categorizar problemas”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir agrupar problemas bajo diferentes categorías globales dependiendo el tipo de categoría (colegio-universidad).
Resultado esperado	El sistema debe permitir a los usuarios listar categorías.
Resultado obtenido	El sistema puede permitir a los usuarios listar categorías.

Observaciones	Ninguna.
----------------------	----------

Evidencias

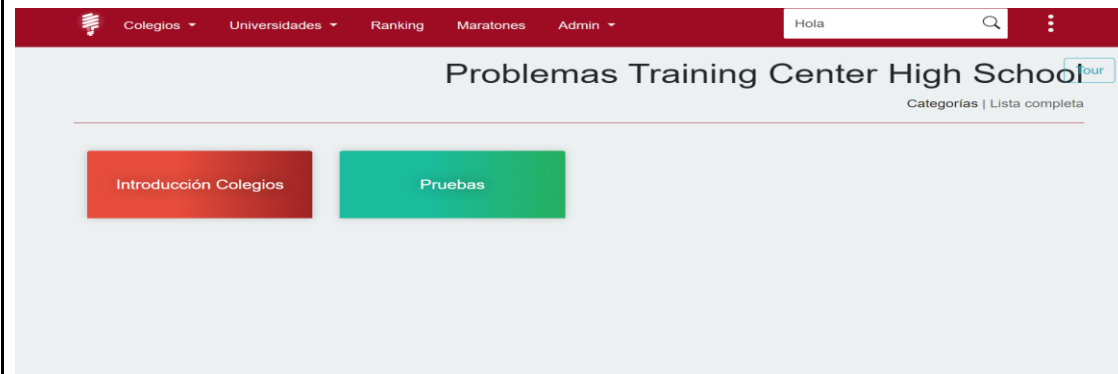


Ilustración 81 CP_08 Categorías de problemas para colegios



Ilustración 82 CP_08 Categorías de problemas para universidades

Tabla 41 Formato de pruebas CP_09

Formato de pruebas	
Identificador	CP_09
Nombre	Ver ranking
Contexto	Se realiza la prueba desde el módulo de ranking de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Categorizar problemas”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe generar ranking de los usuarios según el número de problemas resueltos.
Resultado esperado	El sistema debe permitir listar el ranking de usuarios y realizar filtros.
Resultado obtenido	El sistema puede permitir listar el ranking de usuarios y realizar filtros.

Observaciones	Ninguna.
----------------------	----------

Evidencias

Problemas
Categorías | Lista completa

Ilustración 83 CP_09 Opción ver ranking desde menú

Clasificación general

Filtrar por institución: Elegir: Filtrar por año: Filtrar por categoría:

Puesto	Usuario	Institución	Inst. Tipo	Soluciones correctas	Envíos realizados
1	Edinsson Melo (Mackenzie)	UFPS	Universidad	265	976
2	Marco (MarcoReus)	UFPS	Universidad	251	520
3	Sebastian Vega (C bas)	UFPS	Universidad	244	606
4	Crisel Ayala :D (Tarde pero seguro prros :v)	UFPS	Universidad	224	636

Ilustración 84 CP_09 Listado de ranking con opción de filtrar

Tabla 42 Formato de pruebas CP_10

Formato de pruebas	
Identificador	CP_10
Nombre	Crear competencias
Contexto	Se realiza la prueba desde el módulo de maratones de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Crear competencias”.
Datos de entrada	<ul style="list-style-type: none"> - Nombre: Nombre de la maratón. - Descripción: Descripción de la maratón. - Reglas: Reglas de la maratón. - Tipo: Para colegios o universidades. - Desde el: Fecha y hora de inicio. - Hasta el: Fecha y hora de finalización. - Visibilidad: Pública o privada (necesita contraseña).
Criterios de	El sistema debe permitir la creación de competencias (maratones) de

aceptación	programación en vivo.
Resultado esperado	El sistema debe permitir crear maratones.
Resultado obtenido	El sistema puede permitir crear maratones.
Observaciones	Ninguna.
Evidencias	

Crear Maratón

Mis maratones

Mostrar 10 maratones Ordenar por Id en forma Ascendente

Id	Nombre	Visibilidad	Tipo	Fecha de Inicio	Acción
108	ensayo	Pública	Universidad	4 de Febrero del 2021 - 5:40PM	Editar Ver
122	prueba 23082022	Pública	Universidad	23 de Agosto del 2022 - 10:30PM	Editar Ver
147	prueba 20112022	Pública	Universidad	20 de Noviembre del 2022 - 2:01PM	Editar Ver

Ilustración 85 CP_10 Opción crear maratón

Maratones / Nueva maratón

Nueva Maratón

Nombre:

Descripción:

Reglas:

Tipo:

Desde el: a las:

Ilustración 86 CP_10 Formulario crear maratón

La maratón ha sido creada con éxito

Ilustración 87 CP_10 Notificación maratón creada

Tabla 43 Formato de pruebas CP_11

Formato de pruebas	
Identificador	CP_11
Nombre	Añadir material de ayuda
Contexto	Se realiza la prueba desde el módulo de materiales de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Añadir material de ayuda”.
Datos de entrada	<ul style="list-style-type: none"> - Nombre: Nombre de material. - Descripción: Descripción del material. - Tipo: URL o PDF. - Material: URL o documento PDF adjunto.
Criterios de aceptación	El sistema debe permitir que los usuarios agreguen material de ayuda (PDF, videos).
Resultado esperado	El sistema debe permitir que los usuarios agreguen material de ayuda (PDF, videos).

Resultado obtenido	El sistema puede permitir que los usuarios agreguen material de ayuda (PDF, videos).
Observaciones	Ninguna.
Evidencias	

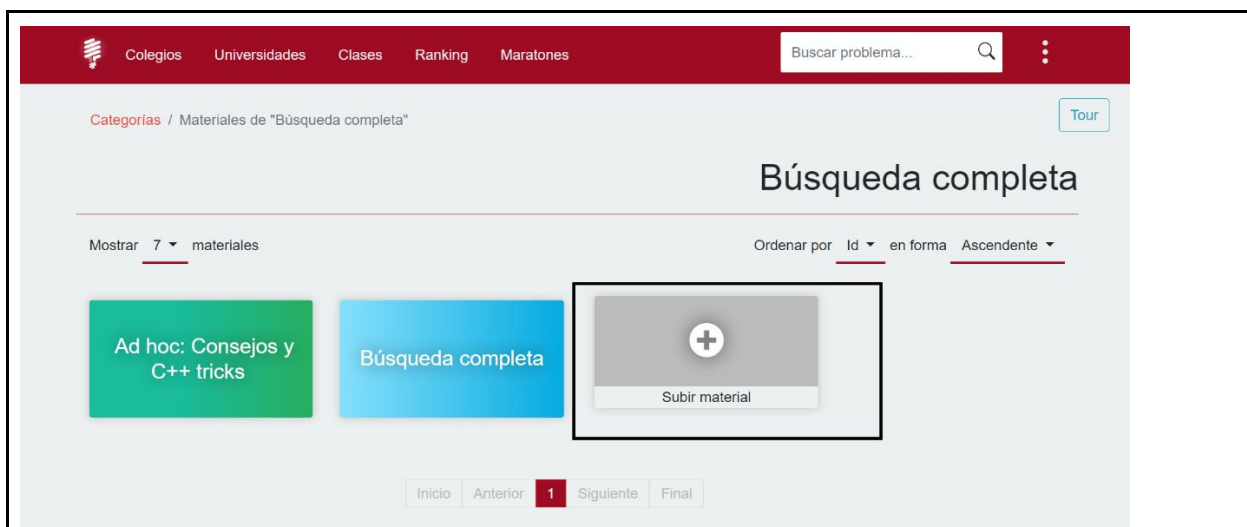


Ilustración 88 CP_11 Opción subir material por categoría

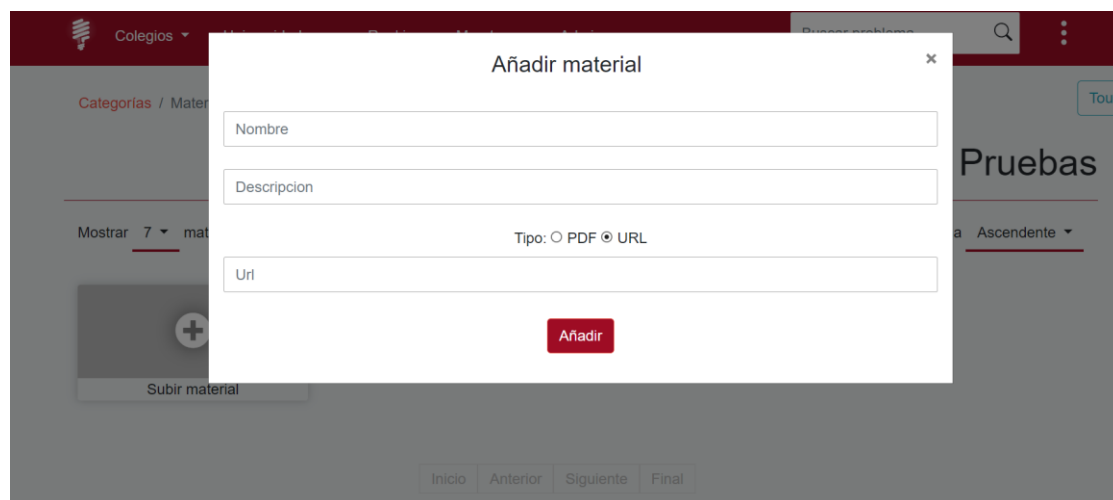


Ilustración 89 CP11 Formulario anexas material a categoría

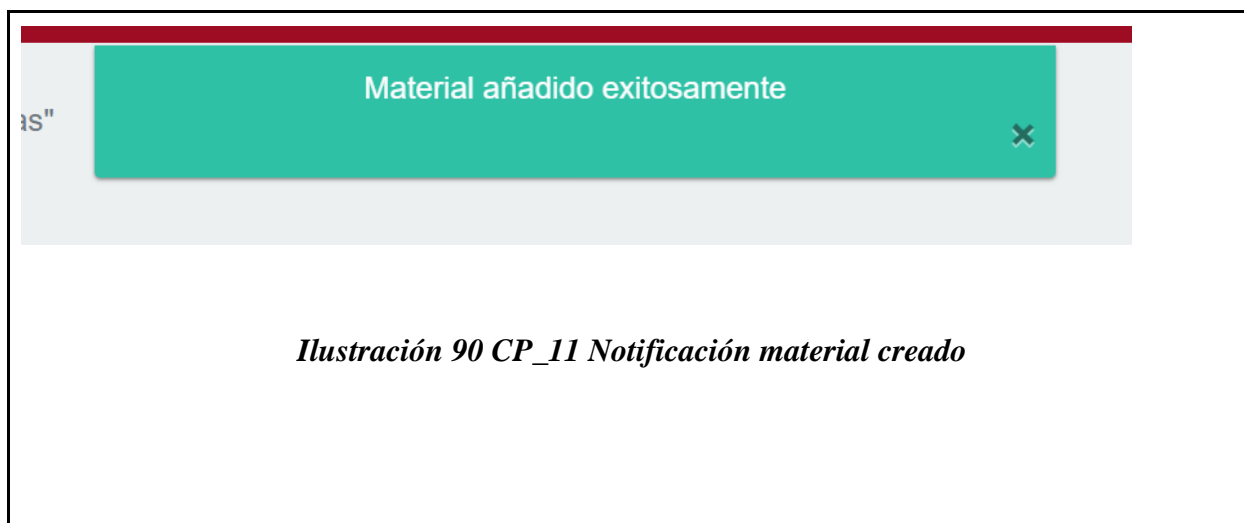


Tabla 44 Formato de pruebas CP_12

Formato de pruebas	
Identificador	CP_12
Nombre	Ver material de ayuda
Contexto	Se realiza la prueba desde el módulo de materiales de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional "Ver material de ayuda".

Datos de entrada	N/A
Criterios de aceptación	El sistema permite cargar el material de ayuda para mejorar su entrenamiento.
Resultado esperado	El sistema permite cargar el material de ayuda.
Resultado obtenido	El sistema puede cargar el material de ayuda.
Observaciones	Ninguna.
Evidencias	

Categorías / Materiales de "Matemáticas" Tour

Matemáticas

Mostrar **7** materiales Ordenar por **Id** en forma **Ascendente**

Matemáticas I

Matemáticas II

Video: MCM y MCD

Video: Sucesión de fibonacci

Video: Números triangulares

Video: Números de catalán

Video: Números primos

Ilustración 91 CP_12 Listado de materiales por categoría

Colegios ▾ Universidades ▾ Ranking Maratones Admin ▾

Categorías / Materiales de "Matemáticas" / Matemáticas I

Matemáticas I

Principales temáticas matemáticas útiles en competencias de programación

MATEMÁTICAS

Ilustración 92 CP_12 Ver detalle de material

Tabla 45 Formato de pruebas CP_13

Formato de pruebas	
Identificador	CP_13
Nombre	Crear Modo guiado (Tour)
Contexto	Se realiza la prueba desde todos los módulos de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Crear Modo guiado”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir a los usuarios de Training Center acceder a un tour que permite el conocimiento de las nuevas funcionalidades de la plataforma.
Resultado esperado	El sistema permite mostrar el tour de ayuda paso a paso.
Resultado	El sistema puede mostrar el tour de ayuda paso a paso.

obtenido	
Observaciones	Ninguna.
Evidencias	

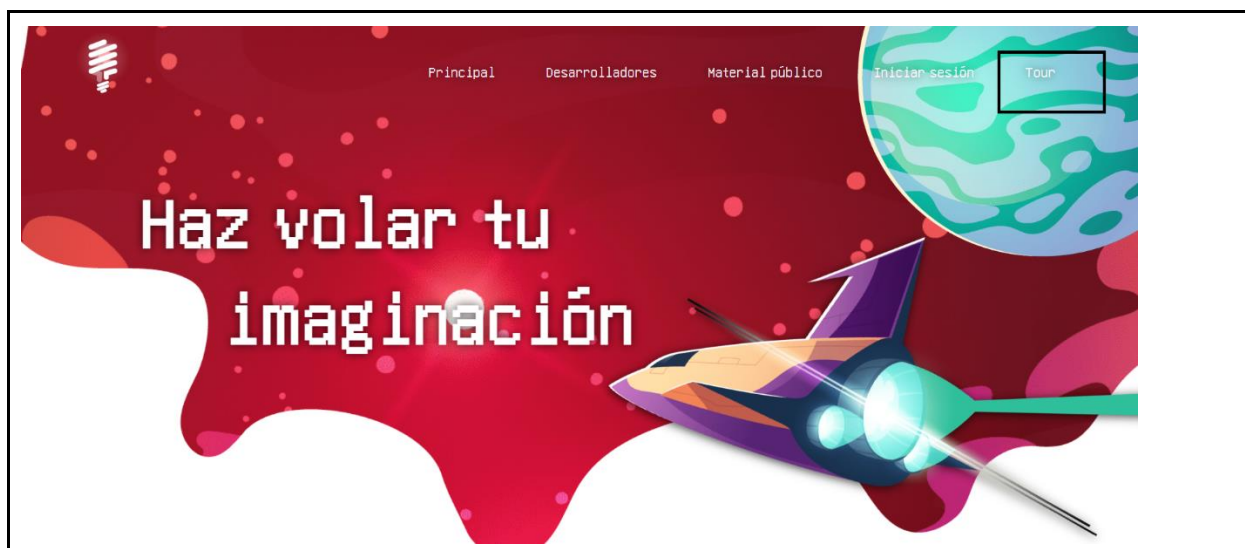


Ilustración 93 CP_13 Ejemplo tour página principal

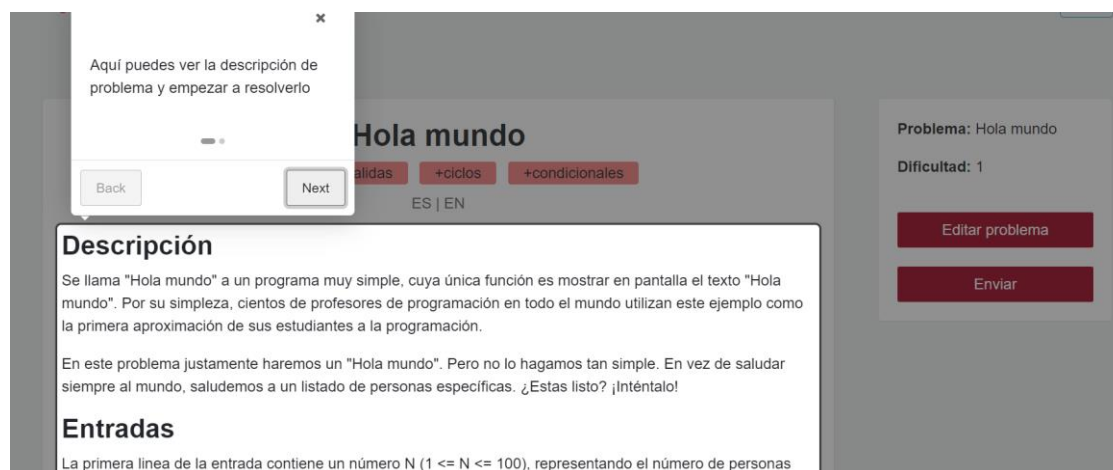
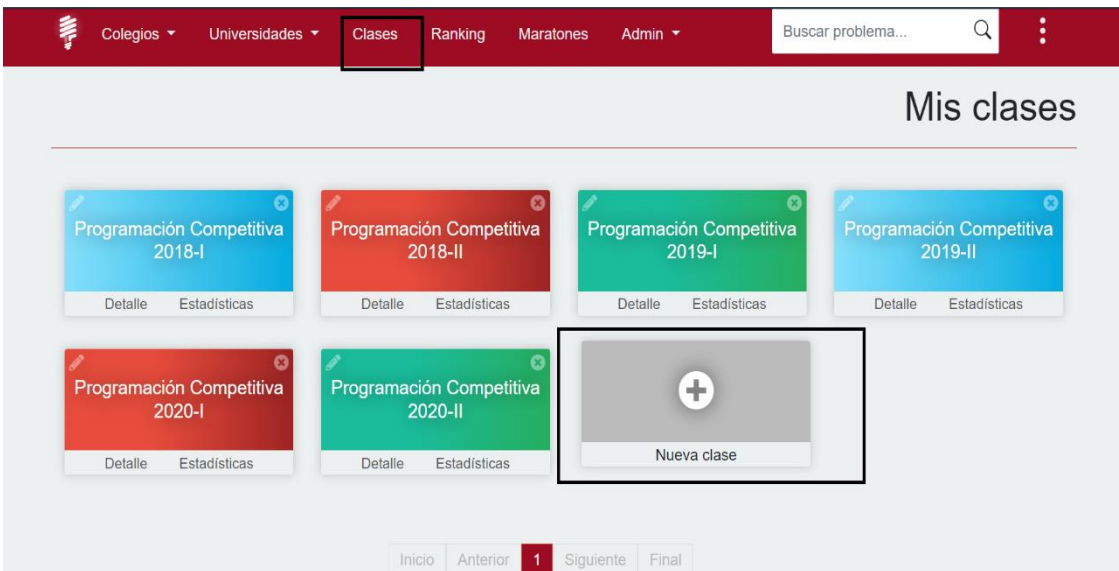


Ilustración 94 CP_13 Ver tour desde página de detalle de problema

Tabla 46 Formato de pruebas CP_14

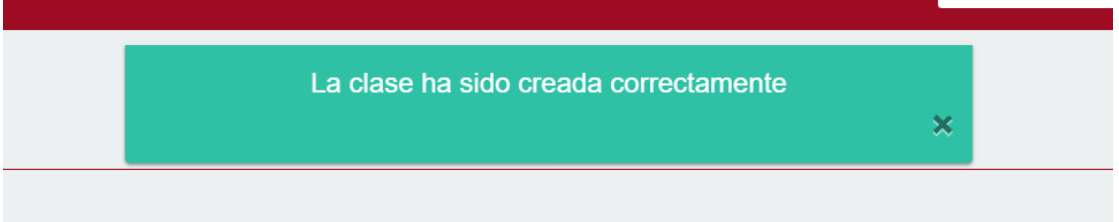
Formato de pruebas	
Identificador	CP_14
Nombre	Resolver tareas de las clases
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	El sistema debe permitir a los coach/docentes crear tareas (clases o syllabus) para que sus estudiantes resuelvan los ejercicios que proponen y ver sus materiales.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir acceder al foro de discusión de DISQUS para todos los problemas de la plataforma, comentar y leer comentarios de otros usuarios.
Resultado esperado	- El sistema debe permitir a los docentes y administradores eliminar problemas de forma exitosa.

Resultado obtenido	El sistema puede permitir a los docentes y administradores eliminar problemas.
Observaciones	Ninguna.
Evidencias	



The screenshot shows a web interface with a dark red navigation bar at the top. The 'Clases' menu item is highlighted with a red box. Below the navigation bar, the page title is 'Mis clases'. There are six class cards displayed in a grid, each with a title and two buttons: 'Detalle' and 'Estadísticas'. The cards are for 'Programación Competitiva' in semesters 2018-I, 2018-II, 2019-I, 2019-II, 2020-I, and 2020-II. A seventh card, labeled 'Nueva clase' with a plus sign icon, is highlighted with a red box. At the bottom of the grid, there are navigation buttons: 'Inicio', 'Anterior', '1', 'Siguiente', and 'Final'.

Ilustración 95 CP_14 Opción crear syllabus



The screenshot shows a green notification message box with the text 'La clase ha sido creada correctamente' and a close button (X) on the right side.

Ilustración 96 CP_14 Notificación tarea creada

Tabla 47 Formato de pruebas CP_15

Formato de pruebas	
Identificador	CP_15

Nombre	Recepcionar solución de tareas
Contexto	Se realiza la prueba desde el módulo de clases de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Recepcionar solución de tareas”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir la recepción de las tareas enviadas por los estudiantes que previamente fueron propuestas por el coach/docente en la plataforma.
Resultado esperado	El sistema debe permitir ver y resolver tareas de una clase.
Resultado obtenido	El sistema puede permitir ver y resolver tareas de una clase.
Observaciones	Ninguna.
Evidencias	

Clases / Programación Competitiva 2018-I

Programación Competitiva 2018-I

Clase del grupo de estudio en programación competitiva de la Universidad Francisco de Paula Santander

Tareas

- Semana I - Introducción (básico)
- Semana II - Arreglos (básico)
- Semana II - Búsqueda completa (avanzado)
- Semana III - Pilas y colas (básico)
- Semana III - Greedy (avanzado)
- Semana IV - Ad-Hoc (básico)
- Semana IV - Divide y vencerás (avanzado)
- Semana V - Sets y maps (básico)

Ilustración 97 CP_15 Opción ver tareas de syllabus

Guardar cambios

Id	Problema	Dificultad	Idioma
1	Hola mundo ✕	1	ES EN
2	Partido amistoso ✕	1	ES EN
6	Divisores ✕	1	ES EN
3	Número de divisores ✕	1	ES EN
4	Corre ✕	2	ES EN
5	Cirene ✕	2	ES EN

Ilustración 98 CP_15 Edición de tarea

Usuario	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje	Código
Angie Gonzalez (Andeley)	17 de Mayo del 2018 - 6:55PM	Respuesta equivocada	0.126s	Java	Ver código
Crisel Ayala :D (Tarde pero seguro prros :v)	13 de Mayo del 2018 - 4:50PM	Correcto	0.119s	Java	Ver código
Crisel Ayala :D (Tarde pero seguro prros :v)	13 de Mayo del 2018 - 4:50PM	Error de compilación	0.000s	C++	Ver código
Hernandez p (joder tio)	13 de Mayo del 2018 - 3:14PM	Correcto	0.123s	Java	Ver código
david alejandro tolosa zabala (EL TIO)	13 de Mayo del 2018 - 2:05PM	Correcto	0.094s	Java	Ver código
Julian Navarro-Najiluc (Najiluc)	12 de Mayo del 2018 - 10:48AM	Correcto	0.131s	Java	Ver código

Ilustración 99 CP_15 Estadísticas de envíos hechos por tarea

Tabla 48 Formato de pruebas CP_16

Formato de pruebas	
Identificador	CP_16
Nombre	Eliminar usuarios
Contexto	Se realiza la prueba desde el módulo de administración de la plataforma.
Tipo de prueba	Estándar

Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Eliminar usuarios”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir al administrador eliminar usuarios no deseados.
Resultado esperado	El sistema debe permitir eliminar usuarios.
Resultado obtenido	El sistema puede eliminar usuarios.
Observaciones	Ninguna.
Evidencias	

Contraseña: Contraseña:

[Crear usuario](#)

Filtrar por institución:

Elegir:

Mostrar usuarios Ordenar por en forma


Id	Nombre	Usuario	Código	Tipo	Email	Institución	Inst. Tipo	Acciones
1	Angie Melissa Delgado León	Meyito	1150990	Administrador	amelisdl@gmail.com	UFPS		Revocar
2	Gerson Lázaro	GersonLazaro	1150972	Administrador	gersonyesidlc@ufps.edu.co	UFPS		Revocar

Ilustración 100 CP_16 Opción eliminar usuario desde usuario administrador

Tabla 49 Formato de pruebas CP_17

Formato de pruebas	
Identificador	CP_17
Nombre	Calificar envíos
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar

Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Calificar envíos”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe calificar automáticamente las soluciones enviadas por los estudiantes, a los diferentes problemas e indicarles su calificación.
Resultado esperado	El sistema debe permitir calificar problemas.
Resultado obtenido	El sistema puede permitir calificar problemas.
Observaciones	Ninguna.
Evidencias	



The screenshot shows a submission record for the problem 'Hola mundo' with a difficulty of 1, submitted on December 21, 2022, at 3:13 PM. The verdict is 'Correcto' and the execution time is 0.020s. A modal window titled 'Descargar código (Python)' displays the following code:

```

establecer count a solicitar número con el mensaje “ ”
repetir count veces
hacer establecer nombre a solicitar texto con el mensaje “ ”
imprimir * crear texto con “ Hola ”
nombre
  
```

Below the code, there is a 'Ver código' link for each submission entry in the background table.

Ilustración 101 CP_17 Ver calificación de envío (Blockly)

Tabla 50 Formato de pruebas CP_18

Formato de pruebas	
Identificador	CP_18
Nombre	Mostrar categorías, problemas y clases separados por TCHS y TCU
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar

Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Mostrar categorías, problemas y clases separados por TCHS y TCU”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe mostrarle al usuario las categorías, problemas y clases separadas dependiendo si está en módulo colegios o módulo universidades.
Resultado esperado	El sistema debe permitir separar módulos por categorías.
Resultado obtenido	El sistema puede permitir separar módulos por categorías.
Observaciones	Ninguna.
Evidencias	

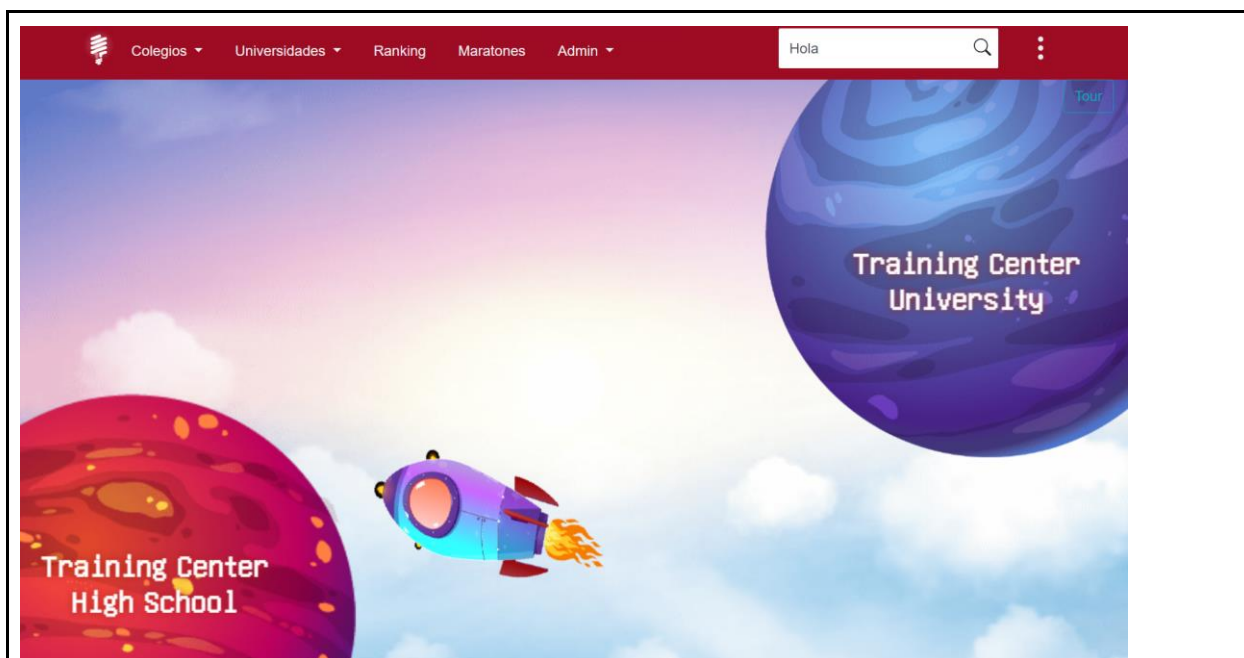


Ilustración 102 CP_18 Separación de categorías (Por tipo colegio y universidad)

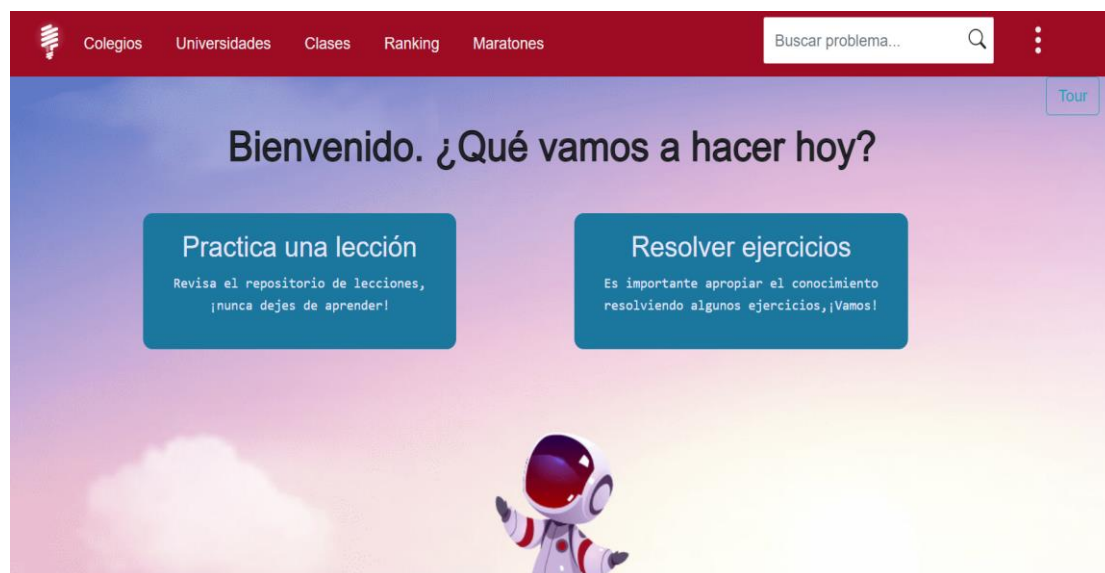


Ilustración 103 CP_18 Página principal Training Center High School

Tabla 51 Formato de pruebas CP_19

Formato de pruebas	
Identificador	CP_19
Nombre	Inspeccionar códigos en competencias (como usuario administrador).
Contexto	Se realiza la prueba desde el módulo de problemas de la plataforma.
Tipo de prueba	Estándar
Estado de la prueba	Finalizada.
Descripción de la prueba	Con la ejecución de la prueba se comprueba el correcto funcionamiento del requerimiento funcional “Inspeccionar códigos en competencias”.
Datos de entrada	N/A
Criterios de aceptación	El sistema debe permitir a un administrador inspeccionar códigos en competencias en vivo, para evitar posibles plagios.
Resultado esperado	El sistema debe permitir inspeccionar códigos de usuarios.
Resultado obtenido	El sistema puede permitir inspeccionar códigos de usuarios.

Observaciones	Ninguna.
Evidencias	

Clasificación general

Filtrar por institución: Elegir: Filtrar por año: Filtrar por categoría:

Puesto	Usuario	Institución	Inst. Tipo	Soluciones correctas	Envíos realizados	Ver envíos
1	Edinsson Melo (Mackenzie)	UFPS	Universidad	265	976	Ver
2	Marco (MarcoReus)	UFPS	Universidad	251	520	Ver
3	Sebastian Vega (C bas)	UFPS	Universidad	244	606	Ver
4	Crisel Ayala :D (Tarde pero seguro prros :v)	UFPS	Universidad	224	636	Ver

Ilustración 104 CP_19 Opción ver envíos de usuario desde ranking (opción disponible para administrador)

Maratones / Copa Programación Cuartos de final 1 / Resultados

Copa Programación Cuartos de final 1
Competencia finalizada

Inicio / Problemas / Resultados

Posición	Usuario	A	B	C	D	E	F	G	Total	Envíos Realizados
1	mavd09 (mavd09)	1/7	1/27	1/43		2/111	1/69	1/74	6 (351)	Ver Envíos
2	Juan José Suárez (jjsuarestra99)	1/110	2/156			1/93	1/37	1/11	5 (427)	Ver Envíos
3	Crisel Ayala :D (Tarde pero seguro prros :v)								0 (0)	Ver Envíos

Ilustración 105 CP_19 Opción ver envíos en competencias (opción disponible para administrador)

Usuario	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje	Código
Paola Bird (Binomio cuadrado perfecto)	6 de Abril del 2018 - 2:35PM	Correcto	0.207s	Java	Ver código
Paola Bird (Binomio cuadrado perfecto)	6 de Abril del 2018 - 2:27PM	Error en tiempo de ejecución	0.243s	Java	Ver código
Paola Bird (Binomio cuadrado perfecto)	6 de Abril del 2018 - 1:54PM	Respuesta equivocada	0.227s	Java	Ver código
Paola Bird (Binomio cuadrado perfecto)	6 de Abril del 2018 - 1:53PM	Respuesta equivocada	0.252s	Java	Ver código
Brayan Godoy (BrayanGodoy)	6 de Abril del 2018 - 8:43AM	Correcto	0.221s	Java	Ver código

Ilustración 106 CP_19 Opción ver envíos de estudiantes en tareas (disponible para usuario coach)

6.6. Documentación de software

La documentación de software es uno de los aspectos importantes del proceso de desarrollo de software; la documentación técnica y la definición de los diferentes componentes de alto nivel, y de más bajo nivel, como variables, listas, funciones, y demás estructuras de código sean de fácil acceso incluso para quienes quieran hacer cambios nuevos en el software y no conozcan muy bien acerca de su funcionamiento y también esta documentación reducirá la curva de aprendizaje. Por lo tanto, se implementó jsdoc para documentar los aspectos más técnicos del software Training Center UFPS; jsdoc es una librería que ofrece una sintaxis, en

distintos formatos, que sirve para agregar documentación al código fuente de javascript. En la siguiente ilustración, se puede ver una sección de la documentación de software del proyecto Training Center. Para ver más a detalle la documentación de software, revisar anexos de código fuente del proyecto.

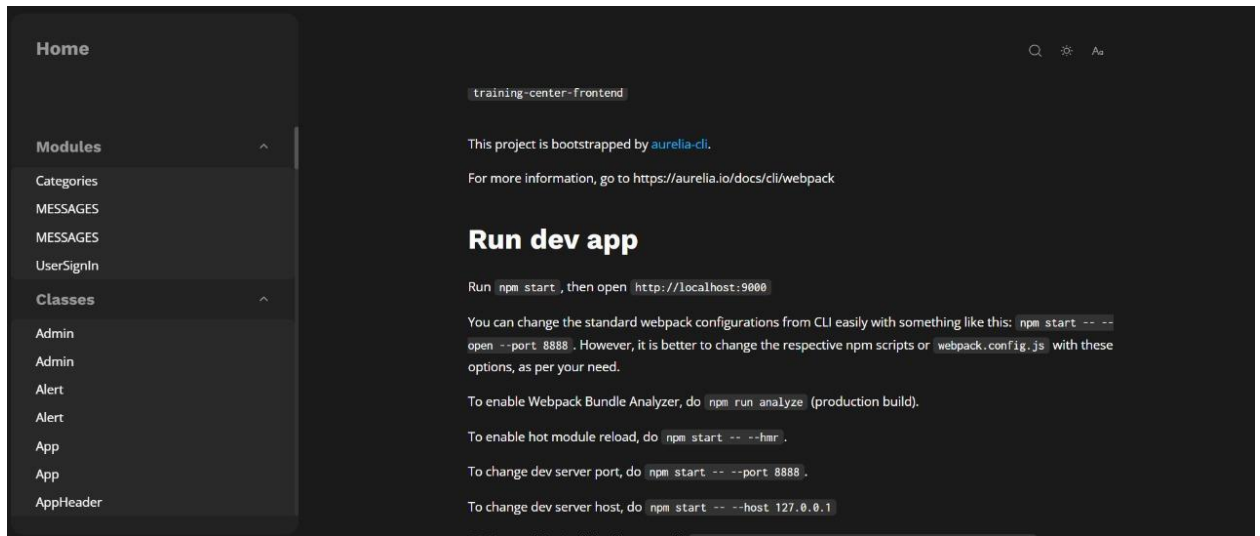


Ilustración 107 Documentación de software

7. Prueba piloto en colegios

Según la ruta previamente establecida, detallada en la Ilustración 37 de este proyecto, se dio inicio con la ejecución de la prueba piloto y así dar cumplimiento con el último objetivo específico del presente proyecto.

Inicio planificación prueba piloto

El día 5 de octubre del 2022 se realizó una reunión de manera virtual con la profesora Deisy YURLey con el objetivo de definir la fecha para capacitar a los profesores de los colegios acerca de la plataforma Training Center y su funcionamiento con enfoque en colegios. En esta reunión se realizó la presentación del marco de trabajo de Training Center High School (Ilustración 70) con el fin de generar entendimiento general acerca del trabajo realizado por el grupo de estudio de programación competitiva y el marco de trabajo que se plantea para el trabajo con colegios, tomando como base la metodología trabajada hasta ahora.

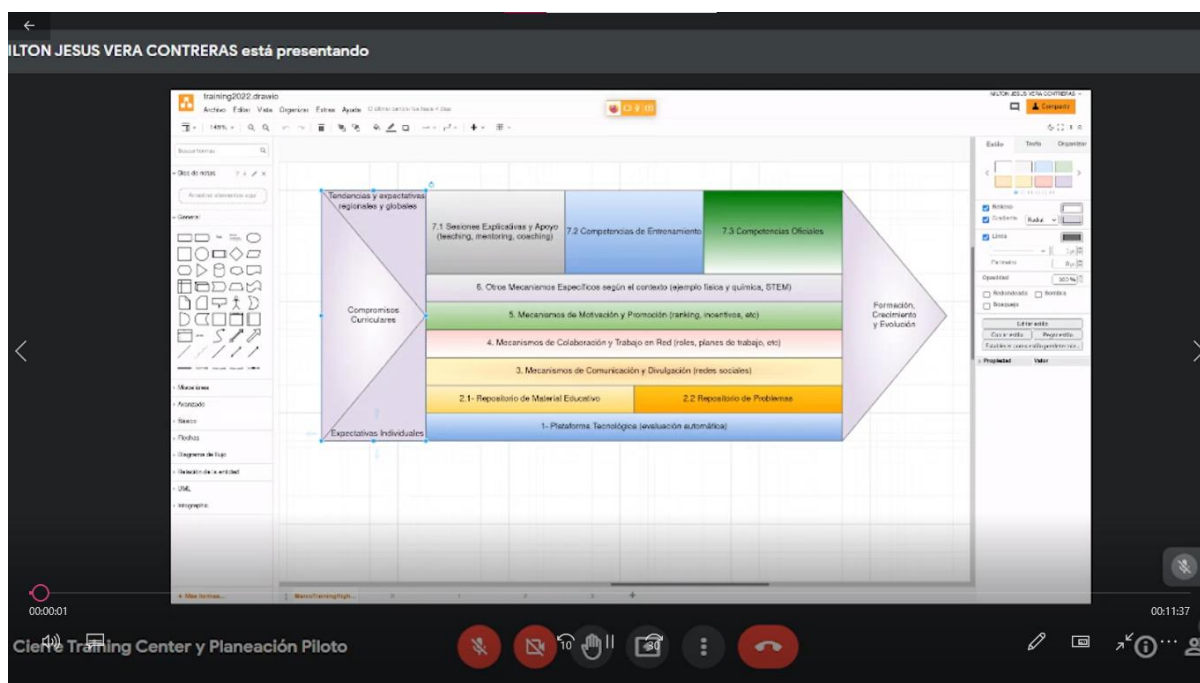


Ilustración 108 Reunión Inicial planeación prueba piloto.

La reunión concluyó agendando fecha de reunión con los profesores de colegio para el día 9 de noviembre del 2022 en horas de la mañana en la universidad Francisco de Paula Santander.

Inducción Training Center High School para Profesores:

El 9 de noviembre a las 8 de la mañana se realizó de forma presencial en las instalaciones de la universidad Francisco de Paula Santander la reunión agendada, con la compañía de la profesora Deisy Yurley Rolón y el profesor Jimmy Alexander Gelvez Jaimes. Esta reunión tuvo como objetivo presentar la plataforma Training center en su más reciente actualización la cual es

su módulo para colegios “Training Center High School”. Se mostraron y explicaron las funcionalidades de la plataforma indicando así también el trabajo realizado por medio de grupo de estudio de programación competitiva desde años anteriores y el cual se quiere seguir realizando y ahora extendiéndose a estudiantes de colegio. Se socializó también la ruta propuesta para la ejecución de la prueba piloto, con el fin de avanzar al siguiente ítem y acordar una fecha para la prueba piloto con los estudiantes de las diferentes sedes del colegio. Luego de la socialización de la plataforma, sus funcionalidades y el trabajo realizado a lo largo de estos años por medio del grupo de estudio, los docentes expresaron su interés por involucrar a sus alumnos en este proceso y también usar la plataforma como herramienta para presentar de una forma diferente y novedosa diferentes áreas y temas de formación que ellos imparten. La reunión concluyó acordando como fecha para la inducción y primera maratón de programación para estudiantes de colegios de la ciudad de Cúcuta, el jueves 17 de noviembre de 8 a 11 am. Según lo acordado en esta reunión, para el día 17 se realizaría la actividad en las instalaciones de la universidad Francisco de Paula Santander, comprometiéndose así los profesores con el traslado de los estudiantes hacia la universidad. Esto debido a la facilidad de equipos tecnológicos presentes en la universidad que permiten mayor facilidad para la realización de dicha actividad.



Ilustración 109 Reunión Inducción sobre Training Center High School a profesores de colegios.



Ilustración 110 Reunión Inducción sobre Training Center High School a profesores de colegios.

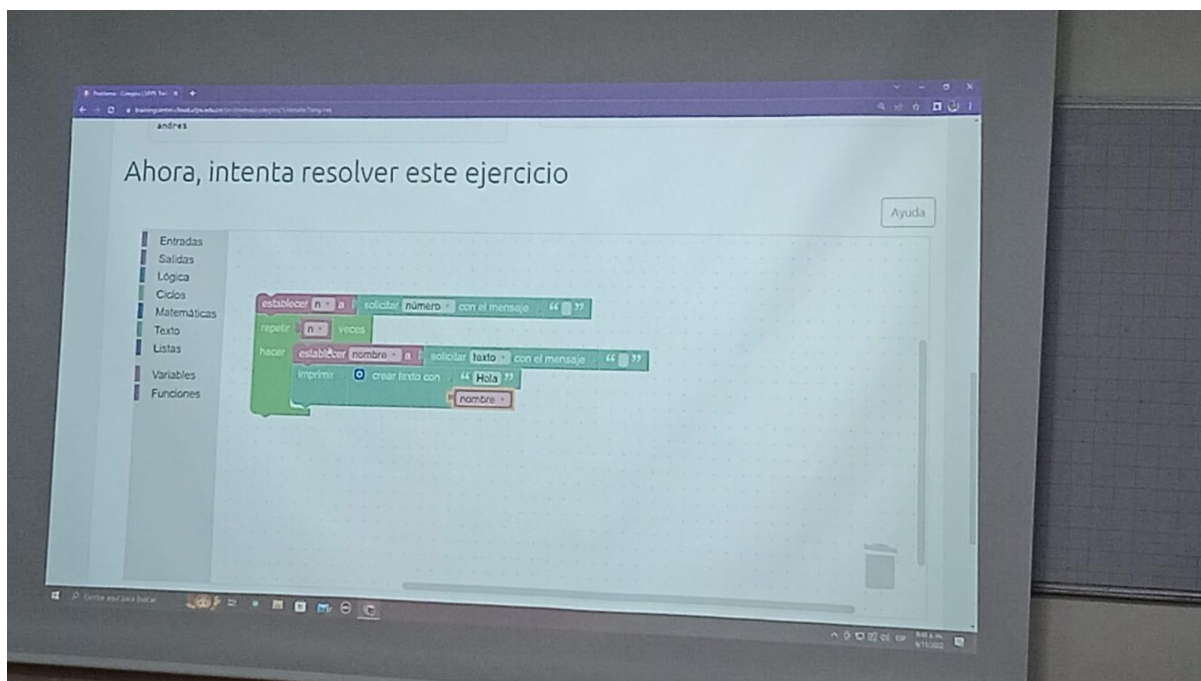


Ilustración 111 Reunión Inducción sobre Training Center High School a profesores de colegios.



Ilustración 112 Reunión Inducción sobre Training Center High School a profesores de colegios.

El día 17 de noviembre se realizó según lo programado la primera prueba piloto con estudiantes de colegios de la ciudad de Cúcuta.

Inducción Training Center High School para estudiantes de colegios:

En las instalaciones de la universidad Francisco de Paula Santander se inició a las 8:00 am con la inducción a 21 estudiantes de colegio sobre Training Center High School. Se realizó una introducción en la que se expuso acerca del grupo de programación competitiva y sobre sus avances en los últimos años. También se habló del Training Center, su objetivo y posteriormente se les expuso a los estudiantes el deseo que se tiene con este proyecto de llevar todo el trabajo que se ha venido realizando tanto a nivel de universidad como también a nivel de colegios. Se realizó la explicación de la plataforma Training Center, indicando el link de ingreso a la plataforma, al cual cada estudiante ingresó ya que cada uno contaba con un computador en sala.

Seguidamente se indicó a los estudiantes cómo realizar el registro a la plataforma para tener acceso a sus funcionalidades. Los estudiantes realizaron exitosamente su registro en la plataforma.

Teniendo a todos los estudiantes registrados en la plataforma, se realizó un warm-up, el cual es un espacio que se les da en competencias para probar envíos, saber como funciona este procedimiento y resolver dudas o problemas presentados. En este espacio se le indicó a los estudiantes cómo se podrían registrar a la maratón creada específicamente para esta prueba piloto. Se les proporcionó la contraseña para realizar el registro a la maratón ya que la maratón

era de acceso privado. Los estudiantes se registraron exitosamente a la maratón y se les indicó que el primer problema en la lista sería el problema con el que se iba a probar envíos, siendo este el problema más sencillo de la maratón. Dicho problema fue el famoso “Hola mundo”. Teniendo claro las especificaciones inició el conteo regresivo para el inicio de la maratón. La maratón inició y como ya antes se les había indicado a los estudiantes, solo debían dar clic en enviar, ya que la solución de “Hola mundo” se carga por defecto en Blockly. El primer estudiante en enviar el primer problema según el ranking, obtuvo un globo contramarcado con el icono de programación competitiva. Posteriormente se dió explicación de la solución en bloques del “Hola mundo” que todos acababan de enviar, se explicó el funcionamiento de los bloques más representativos y se resolvieron dudas del porqué algunos tuvieron respuesta diferente a “correcto”. Casos que se presentaron porque algunos estudiantes modificaron los bloques y por lo tanto la solución que enviaron ya no era la correcta. Todo esto sirvió para que ellos intuitivamente fueran aprendiendo el funcionamiento de envíos de soluciones con Blockly. Inmediatamente después de esto se les indicó que podían seguir creando la solución a los diferentes problemas planteados en la maratón. Realizando un seguimiento y acompañamiento ante las dudas.



Ilustración 113 Reunión Inducción sobre Training Center High School a estudiantes de colegios.

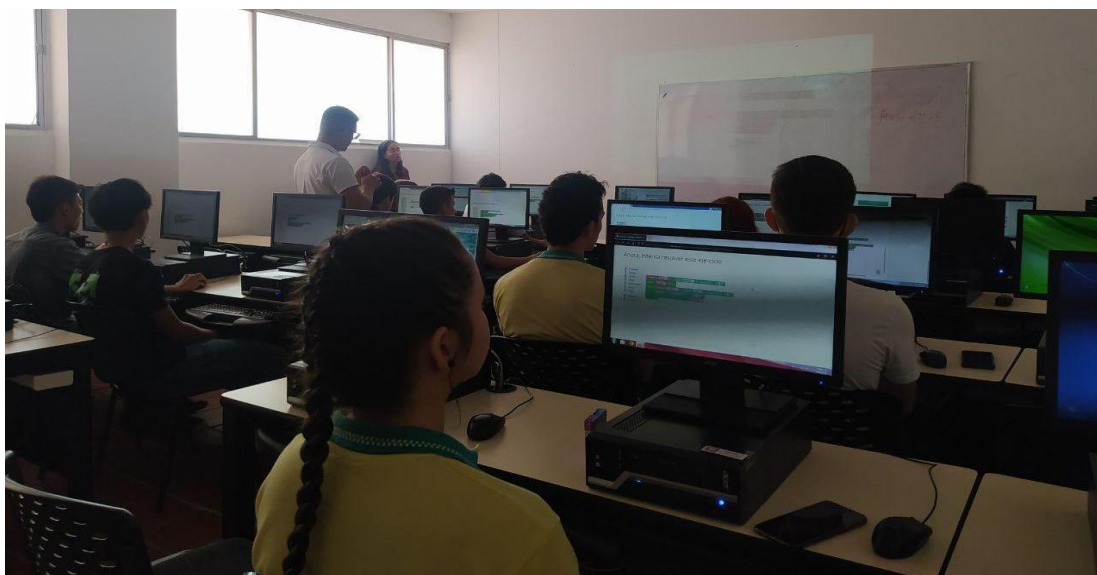


Ilustración 114 Reunión Inducción sobre Training Center High School a estudiantes de colegios.



Ilustración 115 Reunión Inducción sobre Training Center High School a estudiantes de colegios.

Maratón de entrenamiento:

La maratón constó de un conjunto de 7 problemas de dificultad 1.

Mostrar 10 problemas en Cualquier idioma		Ordenar por Id en forma Ascendente	
Id	Problema	Dificultad	Idioma
1001	Hola mundo ✓	1	ES EN
1002	Hola Mundo Reloaded ✓	1	ES EN
1003	Multiplicando por 11 ✓	1	ES EN
1004	Cociente y Residuo ✓	1	ES EN
1005	Doble de un número	1	ES EN
1006	Mayor y Menor	1	ES EN
1007	Tabla de Multiplicar	1	ES EN

Ilustración 116 Problemas maratón de entrenamiento, prueba piloto Training Center High School.

Inicialmente se había planteado realizar la maratón en equipos de 3 estudiantes como suelen ser las maratones oficiales. Pero al tener la posibilidad de un computador para cada estudiante se concluyó que para mayor apropiación de la plataforma la competencia fuese de manera individual.

La maratón tuvo una duración de 1 hora, tiempo durante el cuál los estudiantes podían enviar sus soluciones planteadas y recibir un globo por cada solución correcta.



Ilustración 117 Maratón de entrenamiento, prueba piloto Training Center High School.

En total fueron enviadas a la maratón 135 soluciones de las cuales 46 fueron soluciones correctas.

×

Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje
Tabla de Multiplicar	1	16 de Noviembre del 2022 - 11:19PM	Correcto	0.020s	Blockly

Descargar código (Python)

```

establecer count a solicitar número con el mensaje “ ”
repetir count veces
hacer
  establecer a a redondear solicitar número con el mensaje “ ”
  establecer counti a 1
  repetir 10 veces
  hacer
    establecer p a redondear counti × a
    imprimir * crear texto con a
    “ * ”
    counti
    “ = ”
    p
  establecer counti a redondear counti + 1
  
```

Ilustración 118 Solución de Ejercicio en lenguajes de bloques, en maratón de entrenamiento, prueba piloto Training Center High School.

La clasificación final de la maratón quedó de la siguiente manera:

Maratones / Piloto Training Center High School Julio Pérez Ferrero / Resultados

Piloto Training Center High School Julio Pérez Ferrero

Competencia finalizada

Inicio / Problemas / Resultados

Posición	Usuario	A	B	C	D	E	F	G	Total	Envíos Realizados
1	Darcy Yuliana Mantilla Villamizar (yuli25)	1/1	2/27	3/58					3 (146)	Ver Envíos
2	Laudy (Laudy_6)	1/1	2/30	5/49					3 (180)	Ver Envíos
3	lauris yulleth nuñez quintero (Lalisa kim)	1/1	2/21	6/58					3 (200)	Ver Envíos
4	Ángel Beltrán (acbeldbg)	1/1	4/20	6/55					3 (236)	Ver Envíos
5	Luis Fernando Martínez Pabon (luis martinez)	1/1	1/23						2 (24)	Ver Envíos
6	luz (esther)	1/1	1/24				1/-		2 (25)	Ver Envíos
7	daíra monsalve (fernanda)	1/1	1/25						2 (26)	Ver Envíos
8	MARY (YELITZA)	1/2	1/25						2 (27)	Ver Envíos
9	SHARICT HERNANDEZ (SHARICT)	1/1	2/21	2/-					2 (42)	Ver Envíos
10	carlos rojas caicedo (sorlax)	1/1	2/24						2 (45)	Ver Envíos

Ilustración 119 Clasificación general maratón de entrenamiento, prueba piloto Training Center High School.

Terminada la maratón se tomó la foto con el conjunto de estudiantes participantes y sus globitos obtenidos y se realizó la divulgación en redes sociales del evento realizado.



Ilustración 120 Estudiantes participantes en maratón de entrenamiento, prueba piloto Training Center High School.

Cada estudiante recibió un certificado de participación en la prueba piloto.

Retroalimentación:

Se logró realizar con éxito la prueba piloto del Software Training Center, con su iniciativa Training Center High School, con el fin de promover el pensamiento computacional y la programación de computadores desde el ámbito de la programación competitiva. Por medio de

la integración de Blockly en la plataforma, se logró una extensión que permitió el acceso a colegios. Se pudo observar el gran talento que tienen los estudiantes y la capacidad rápida de integración con una herramienta como Blockly, además de su entusiasmo por conocer e involucrarse más a la programación competitiva.

Se logró por medio de este piloto promover y dar a conocer el mundo de la programación competitiva, esta vez de una manera creativa y llamativa para los estudiantes de colegio.

Como retroalimentación técnica acerca del comportamiento de la plataforma durante la maratón en la prueba piloto se presentan los siguientes datos que nos proporciona google cloud.

The screenshot shows the Google Cloud console interface for editing a VM instance. The left sidebar contains navigation options like 'Máquinas virtuales', 'Almacenamiento', and 'VM Manager'. The main content area is titled 'Editar instancia trainingcenter2' and includes sections for 'Protección contra la eliminación', 'Configuración de la máquina' (with details for vCPU: 1 and Memory: 3.75 GB), 'Plataforma de CPU' (Intel Haswell), 'Dispositivo de visualización', and 'Redes'. At the bottom, there are 'GUARDAR' and 'CANCELAR' buttons.

Ilustración 121 Recopilación de datos de servidor de Google cloud 1

The screenshot displays the monitoring dashboard for the 'trainingcenter2' VM instance. The top navigation bar includes 'COMPUTE ENGINE' and 'trainingcenter2'. The dashboard features several monitoring widgets: 'MÉTRICAS' (Metrics), 'REGISTROS' (Logs), 'Resumen' (Summary), 'Conexiones nuevas con VM/Recurso extern...' (New connections with VM/external resource), 'Envío a VM/Recurso externo/Google' (Send to VM/external resource/Google), 'Recibido de VM/recurso externo/Google' (Received from VM/external resource/Google), 'Paquetes entrantes de firewall rechazados' (Rejected incoming firewall packets), and 'Tamaño medio de los paquetes (bytes)' (Average packet size in bytes). Each widget contains a line graph showing data over time from 08:30 to 10:30 UTC-5. The right sidebar offers additional tools like 'Network Intelligence Center' and 'ESTADÍSTICAS DE FIREWALL'.

Ilustración 122 Recopilación de datos de servidor de Google cloud 2

El reto siguiente es mantener la continuidad de este proyecto. Eso se puede lograr con el trabajo realizado por medio del semillero SILUX en conjunto con el compromiso de los profesores de colegios para mantener la continuidad de entrenamiento con sus estudiantes. Con el propósito de llegar en 2023 a tener maratones de programación oficiales en los colegios y aún más, también maratones intercolegiales.

8. Divulgación científica y evaluación por pares

Se realizó ponencia del proyecto Training Center High School en el VI Encuentro Regional de Semilleros de Investigación.



Ilustración 123 Certificado Ponencia VII Encuentro Regional Semilleros de Investigación 1



Ilustración 124 Certificado Ponencia VII Encuentro Regional Semilleros de Investigación 2

Ponencia en XXIII ENCUENTRO NACIONAL Y XVII ENCUENTRO INTERNACIONAL DE SEMILLEROS DE INVESTIGACIÓN – FUNDACIÓN REDCOLSI (ENISI 2020).



Training Center High School

Angie González - Crisel Ayala - Ing. Milton Vera

Universidad Francisco de Paula Santander -
Programa de Ingeniería de Sistemas

Grupo de estudio de Programación Competitiva del
Semillero SILUX - 2020

Ilustración 125 Presentación Training Center High School como ponencia



TRAINING CENTER HIGH SCHOOL



Crisel Ayala, Angie González, Milton Vera, criseljazminal@ufps.edu.co,
angiemadeleynegb@ufps.edu.co, miltonjesusvc@ufps.edu.co Universidad Francisco de Paula
 Santander, Norte de Santander.

Introducción

Este proyecto busca extender el semillero de Programación Competitiva y su plataforma "Training Center UFPS" a nivel de colegios. Se trata de darle continuidad al trabajo colaborativo de los estudiantes del grupo de estudio de Programación Competitiva, del semillero SILUX, del programa "Ingeniería de Sistemas" de la Universidad Francisco de Paula Santander. Con el objetivo de adaptar la plataforma, para ofrecer nuevos lenguajes de programación y mejorar la estrategia, a fin de generar motivación y despertar interés por la Programación Competitiva en los estudiantes de bachillerato de los colegios de la ciudad de Cúcuta.

Marco teórico








Objetivo General

Desarrollar una extensión de la plataforma Training Center UFPS para promover la Programación Competitiva en colegios de la ciudad de Cúcuta.

Objetivos Específicos

1. Hacer una revisión sobre las plataformas y lenguajes de programación usadas en colegios y en programación competitiva para seleccionar el lenguaje más apropiado a usar en colegios de la ciudad de Cúcuta.
2. Identificar los requerimientos y adaptaciones de la Plataforma Training Center UFPS para su utilización en colegios de la ciudad de Cúcuta.
3. Realizar el análisis, diseño, implementación, pruebas y despliegue necesarios para que la plataforma Training Center UFPS se pueda usar en colegios de la ciudad de Cúcuta.

Situación Problema

Para el año 2020 se requieren 150.000 programadores que aún no se han formado; en toda Latinoamérica sigue siendo insuficiente el talento humano en programación y posiblemente en el futuro será indispensable saber programar computadores para emplearse (Iván Duque Márquez - Presidente de Colombia, 2019). Esto es una necesidad que incluye a nuestro departamento y la cual se quiere abordar a través de la implementación de la programación competitiva en colegios, de manera atractiva y divertida. Con el fin de generar interés en los estudiantes.

Metodología



Referencias:

Andriessen, M. (2011). Why Software Is Eating the World—WSJ. THE WALL STREET JOURNAL. <https://www.wsj.com/article/SB1100144495311903489304676312203919629466>

Buitrago Pérez, F., Casallas, R., Hernández, M., Reyes, A., Restrepo, S., & Davies, G. (2017). Changing a Generation's Way of Thinking: Teaching Computational Thinking Through Programming. Review of Educational Research, 87(4), 634-660. <https://doi.org/10.3102/0034654317130096>

Gorgio, H. R. D., & Mon, A. (2018). Niveles de productos software en la industria 4.0. International Journal of Information Systems and Software Engineering for Big Companies: IJISESB, 5(2 (Diciembre/December)), 53-62.

Fuentes electrónicas

FedeSoft. (2019). Concurso Nacional de Programación para Estudiantes de Colegios 2019. FedeSoft. <https://fedesoft.org/concurso-nacional-de-programacion-para-estudiantes-de-colegios-2019/>

MinTIC. (2018). De cretos—Ministerio de Tecnologías de la Información y las Comunicaciones. MinTIC. <https://www.mintic.gov.co/post/nacion/NormalasdiDecretos/>

Resultados



Conclusiones

Se finalizó la primera etapa de desarrollo del proyecto.

Se están evaluando requerimientos con los colegios para la realización de la respectiva prueba piloto, en la que se capacitará acerca del uso de la plataforma.

El trabajo realizado por el semillero SILUX luego de la ejecución total del proyecto, será de gran impacto para lograr resultados favorables y confirmar la sostenibilidad del proyecto.

Ilustración 126 Poster Training Center High School



Ilustración 127 Certificado ponencia XXIII Encuentro Nacional y XVII Encuentro Internacional de Semilleros de Investigación 1



RedCOLSI hace constar que,

ANGIE MADELEYNE GONZÁLEZ BARRERA

Con número de documento de identidad 1098761044

Participó en el **XXIII ENCUENTRO NACIONAL Y XVII ENCUENTRO INTERNACIONAL DE SEMILLEROS DE INVESTIGACIÓN**, realizado de manera virtual para el año 2020, en calidad de

PONENTE

Con el proyecto:

Training center high school


HOLME HARRINSO GUTIERREZ BAUTISTA
Coordinador Nacional
Diciembre 10 de 2020

Ilustración 128 Certificado ponencia XXIII Encuentro Nacional y XVII Encuentro Internacional de Semilleros de Investigación 2

Participación en entrevista acerca de capacitaciones para mujeres emprendedoras en TIC.



Ilustración 129 Capacitaciones para mujeres emprendedoras en TIC. Fuente (Oriente noticias Canal TRO)

Ponencia en Encuentro Internacional de Educación en Ingeniería ACOFI 2020.



Ilustración 130 Certificado ponencia en Encuentro Internacional de Educación en Ingeniería ACOFI 2020 1



Ilustración 131 Certificado Ponencia en Encuentro Internacional de Educación en Ingeniería ACOFI 2020 2

9. Productos

Documento acerca de las tecnologías seleccionadas para adaptar Training Center a los colegios.

En el apartado 3.5 del presente documento se especifica la tecnología elegida para esta adaptación y el porqué de esta elección.

Documentación de la integración de Training Center a las tecnologías escogidas en este proyecto.

En el apartado 3 del presente documento se encuentra a detalle esta información.

Documento con la descripción de las tecnologías ya utilizadas.

En los anexos de este documento se puede encontrar el documento de Arquitectura que detalla la descripción de las tecnologías antes utilizadas y su actualización y adaptación para el presente proyecto.

Documento de Requerimientos del sistema

El documento de especificación de requerimientos se puede encontrar dentro de los documentos anexos a este documento.

Documento de Gestión de cambios en Training Center.

En el apartado 5.2 se encuentra la especificación de la gestión de cambios en Training Center.

Documento de análisis y diseño del sistema:

En el apartado 5.3 de este documento se encuentra la especificación del diseño y arquitectura de este proyecto. Adicionalmente en los documentos anexos se encuentran los siguientes documentos:

- Documentos de mockups de este proyecto.
- Material de ayuda Tablero Blockly, material entradas y salidas Blockly

Documento de pruebas de Software.

Modulo Training Center High School

Como producto de software se entrega la extensión de la plataforma Training Center con su nuevo módulo denominado Training Center High School.

URL de la plataforma desplegada : <https://trainingcenter.cloud.ufps.edu.co/bienvenido>

En el siguiente mapa de navegación se puede detallar cómo queda la estructuración de la plataforma con su nuevo módulo y nuevas funcionalidades.

Para ver esta imagen más a detalle, revisar diagramas adjuntos.

10. Conclusiones

Considerando la necesidad que surge para realizar este proyecto, la plataforma Training Center UFPS ahora también será un medio de apoyo para promover el aprendizaje por medio de la programación competitiva en los colegios de la ciudad de Cúcuta. Muchos de los colegios encuestados afirmaron que en sus aulas de clase, no se promueve enseñar programación de computadores o lo que se enseña es muy poco; por eso también surge la oportunidad de hacer estas implementaciones. Teniendo una base más sólida a partir de los resultados del marco de trabajo del Grupo de Estudio, se hizo una natural implementación inicial en colegios de la ciudad de Cúcuta.

Con apoyo del semillero SILUX, se espera que Training Center High School sea una puerta de entrada, para que la mayoría de colegios puedan incentivar a sus estudiantes a aprender a programar, compitiendo y resolviendo problemas sencillos usando lenguajes visuales.

Con la implementación de Blockly, se deja a la mano una herramienta de más alto nivel, para hacer envíos de soluciones, eso demuestra una vez más lo efectiva que ya es la arquitectura planteada en el sistema, hacer estas adaptaciones, no modificaron radicalmente los flujos que ya se venían trabajando y de paso, esto constituye una herramienta altamente ajustable, fácil de aprender y muy intuitiva para el usuario final.

Con la presentación de este proyecto, se pudo revisar a profundidad lo planteado durante la primera versión de la plataforma, y se hicieron mejoras y nuevas implementaciones en pro del apoyo a los integrantes del semillero y de los estudiantes de primeros semestres y estudiantes de colegios de Cúcuta, siempre buscando la mejora continua que permita a la UFPS alcanzar aún mejores resultados en competencias de programación.

Es importante mencionar que después de 5 años, la plataforma Training Center UFPS aún se mantiene vigente, ha sido usada por multitud de usuarios dentro y fuera de Colombia, ha sido una herramienta de apoyo importante para el trabajo del grupo de estudio, y eso demuestra que el trabajo en conjunto, y la implementación del marco de trabajo, ha dado muy buenos resultados; ahora, con la nueva implementación de la plataforma, se podrá llevar estos esfuerzos en instituciones educativas de Cúcuta, esperando que los estudiantes puedan aprender a programar desde más temprana edad.

En resumen, la plataforma Training Center UFPS, con su nueva extensión para colegios, permite ser puerta de entrada, para que estudiantes de colegios pueden entrar al mundo de la programación competitiva, el apoyo del grupo de estudio junto a la implementación del marco de trabajo del semillero han sido muy importantes para hacer estos nuevos esfuerzos, la divulgación de este proyecto por medio de las pruebas piloto y los eventos, ha sido bien recibida, porque, esto también puede responder a la necesidad común de extender la programación competitiva a un rango más amplio de personas.

11. Recomendaciones

Training Center High School ha sido creado basado en el marco de trabajo del grupo de estudio del semillero SILUX, realizado por Gerson Lázaro y Melissa Delgado. Inspiradas por este trabajo realizado, decidimos extender la plataforma agregando las adaptaciones necesarias para implementar la iniciativa “Training Center High School”, pensando en darle continuidad a un arduo trabajo que ha traído muchos resultados positivos a nuestra universidad.

Dicho esto, nuestras recomendaciones se basan en impulsar la continuidad de este proyecto, por medio del apoyo dado al Grupo de Estudio de Programación Competitiva, quienes son los que mantienen un trabajo constante y comprometido por mejorar el nivel de la UFPS, en las diferentes competencias de programación competitiva a nivel, regional, nacional y mundial.

A su vez, también se recomienda lograr nuevas y mantener las ya existentes conexiones con colegios interesados en que sus estudiantes se involucren en el mundo de la programación competitiva.

Para el éxito futuro de los objetivos planteados como grupo de estudio se recomienda a los ya integrantes líderes del grupo de estudio, incentivar y animar a los nuevos integrantes para que ellos puedan ser futuros líderes que mantengan la continuidad del grupo y el uso de la plataforma. Así como también mantener material actualizado dentro de la plataforma.

Se recomienda también a los profesores de colegios ayudar a poblar Training Center High School de nuevo material y problemas con temas pertinentes para estudiantes de colegios. Realizando así un trabajo colaborativo que ayuda a mantener la continuidad de nuestra plataforma.

Por último y no menos importante se recomienda, con apoyo del grupo de estudio, y de la plataforma Training Center, futuros eventos de maratones de programación en los colegios las cuales ayuden a implementar una maratón oficial anual por cada colegio involucrado, las cuales en consecuencia servirán para preparar y llevar equipos de representación a las olimpiadas colombianas de computación, y con trabajo constante lograr llegar a ámbitos internacionales como la IOI (Olimpiada Internacional Informática).

12. Trabajo futuro

En esta sección se deja el planteamiento de futuros cambios en la plataforma, los cuáles han sido identificados como importantes e influyentes para el crecimiento, permanencia y uso de Training Center en el futuro. Todos los cambios aquí planteados no se encontraban dentro del alcance de este proyecto, pero hacen parte de una lista de cosas deseables para la plataforma.

Para este proyecto se deja una vista “Niveles” realizada con el propósito de integrar niveles en los problemas de Training Center High School; clasificando los problemas según su dificultad y así organizarlos en los diferentes niveles: Básico, Intermedio y avanzado.

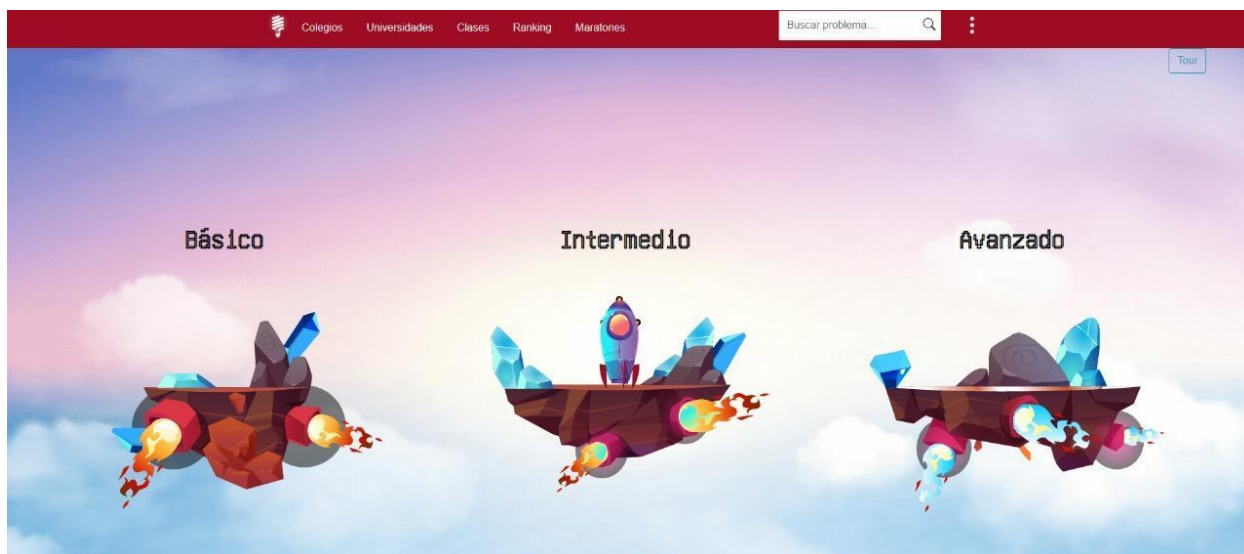


Ilustración 133 Vista niveles Training Center High School

Esto abriría una puerta para llevar a Training Center High School a un modelo de gamificación. El cuál en nuestro concepto podría integrar muchas cosas que hagan posible que el interés por el crecimiento en programación competitiva desde edades tempranas sea aún mayor.

13. Referencias

Carrillo, G. Y. L., León, A. M. D., Contreras, M. J. V., & Vera-Rivera, F. H. (2019). Development of a technological platform for teaching and training in competitive programming contests. *Journal of Physics: Conference Series*, 1386, 012145.

<https://doi.org/10.1088/1742-6596/1386/1/012145>

Kölling, M., Utting, I., mccall, D., Brown, N., Stevens, P., & Berry, M. (2022). *About Greenfoot*. Greenfoot. Retrieved November 13, 2022, from <https://www.greenfoot.org/overview>

Carnegie Mellon University. (2008). *ALICE*. Alice – Tell Stories. Build Games. Learn to Program. Retrieved November 13, 2022, from <http://www.alice.org/>

Minecraft, un videojuego educativo para aprender a programar | Escuela de programación, robótica y pensamiento computacional. (2020, December 7). Codelearn. Retrieved November 8, 2022, from <https://codelearn.es/blog/minecraft-un-videojuego-educativo-para-aprender-a-programar/>

Tynker. (2013). *TYNKER*. Tynker: Coding For Kids, Kids Online Programming Classes & Games. Retrieved November 13, 2022, from <https://www.tynker.com>

Qué es pseudocódigo. (2019, June 18). Openwebinars. Retrieved November 8, 2022, from <https://openwebinars.net/blog/que-es-pseudocodigo/>

Acepta el reto. (2022). *Preguntas frecuentes*. ¡Acepta el reto! Retrieved November 22, 2022, from <https://www.aceptaelreto.com/doc/faq.php>

AWS. (2022). *¿Qué es Python? / Guía de Python para principiantes de la nube.*

AWS. Retrieved November 22, 2022, from <https://aws.amazon.com/es/what-is/python/>

Beecrowd. (2022). *Beecrowd*. Beecrowd. Retrieved November 22, 2022, from

<https://www.beecrowd.com.br/judge/en/login>

Blockly. (2022). *Blockly*. Google Developers. Retrieved November 22, 2022,

from <https://developers.google.com/Blockly>

Blockly. (2022, October 13). *Add Custom Blocks / Blockly*. Google Developers.

Retrieved November 22, 2022, from

<https://developers.google.com/Blockly/guides/configure/web/custom-blocks>

BOCA. (2022). [*@redprogramacion*]. Red de Programación Competitiva.

Retrieved November 22, 2022, from

<https://redprogramacioncompetitiva.com/contests/2022/03/>

CCPL. (2022). Colombian Collegiate Programming League 2022: Home.

Retrieved November 22, 2022, from <https://www.programmingleague.org>

CODEFORCES. (2022). *Codeforces*. Codeforces. Retrieved November 22, 2022,

from <https://codeforces.com>

CODE.ORG. (2022). *About*. Code.org. Retrieved November 22, 2022, from

<https://code.org/international/about>

FODA-DAFO. (2020). ▷ Analisis PESTEL - El mejor y más completo análisis.

FODA-DAFO. <https://foda-dafo.com/analisis-pestel/>

Donetic. (2022). Diccionario Scrum | Términos metodología ágil | donetonic.

Retrieved November 22, 2022, from <https://donetonic.com/es/diccionario-scrum/>

Experto en Negocios Online. (2019, November 29). ▷ *Trello. Qué es, Para Qué Sirve y Cómo Funciona*. Experto en Negocios Online. Retrieved November 22, 2022, from <https://www.expertosnegociosonline.com/que-es-trello-para-que-sirve/>

GUAPA. (2022). *GUAPA. JUEZ GUAPA*. [Https://juezguapa.com](https://juezguapa.com)

ICPC. (2022). *The ICPC International Collegiate Programming Contest*. The ICPC International Collegiate Programming Contest. Retrieved November 22, 2022, from <https://icpc.global/regionals/abouticpc>

IOI. (2022, August 22). *Locations*. International Olympiad in Informatics. Retrieved November 22, 2022, from <https://ioi.te.lv/history.shtml>

Martin, E. (2022). *Plataforma de desarrollo en tiempo real de Unity | Motor de VR y AR en 3D y 2D*. Retrieved November 22, 2022, from <https://unity.com/es>

MDN. (2022, November 7). *Javascript | MDN*. MDN Web Docs. Retrieved November 22, 2022, from <https://developer.mozilla.org/es/docs/Web/javascript>

Microsoft. (2022). *Documentos de Visual Basic: inicio, tutoriales y referencias*. Microsoft Learn. Retrieved November 22, 2022, from <https://docs.microsoft.com/es-es/dotnet/visual-basic/>

MINTIC. (2022). *Misión TIC 2022*. Misión TIC 2022. Retrieved November 22, 2022, from <https://www.misiontic2022.gov.co/portal/Secciones/Mision-TIC-2022/>

OCC. (2022). *Generalidades*. Olimpiadas Colombianas. Retrieved November 22, 2022, from <http://oc.uan.edu.co/olimpiada-colombiana-de-computacion>

OCI. (2022). *Olimpiada Chilena de Informática*. Olimpiada Chilena de Informática | OCI. Retrieved November 22, 2022, from <https://www.olimpiada-informatica.cl>

ODI. (2022). *Olimpiada Dominicana de Informática*. Olimpiada Dominicana de Informática: ODI. Retrieved November 22, 2022, from <http://olimpiadasdeinformatica.org.do>

OIA. (2022). *Olimpiada Informática Argentina*. Olimpiada Informática Argentina: OIA. Retrieved November 22, 2022, from <https://www.oia.unsam.edu.ar>

OIE. (2022). *Olimpiada Informática Española*. Olimpiada Informática Española | Olimpiada Informática Española (OIE). Retrieved November 22, 2022, from <https://olimpiada-informatica.org>

Olimpiada Mexicana de Informática. (2022). Olimpiada Mexicana de Informática: OMI. Retrieved November 22, 2022, from <https://www.olimpiadadeinformatica.org.mx/OMI/OMI/Inicio.aspx>

Pájaro, P., & Romero, F. (2022). *Implantación de una Red Social Académica (ASN, Academic Social Network) para la Universidad Francisco de Paula Santander*.

The PC² CCS. (2022). The PC² CCS | Home. Retrieved November 22, 2022, from <https://pc2ccs.github.io>

Pseint. (2022). Pseint. Retrieved November 22, 2022, from <https://pseint.sourceforge.net>

VIRTUAL JUDGE. (2022). Virtual Judge. Retrieved November 22, 2022, from <https://vjudge.net>

HOUR OF CODE. (2015). *La Hora del Código está aquí*. Code.org.

<https://hourofcode.com/>

ARAQUE, M. (2017, febrero 8). *Metodología Scrum: Qué es y cómo funciona*. We are marketing. <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html>

SCRATCH. (2019). *Scratch—About* [Crea historias, juegos y animaciones. Comparte con otros alrededor del mundo]. Scratch. Recuperado de <https://scratch.mit.edu/about>

Contreras, M. J. V., & Peña Reyes, J. I. (2013). *Rigor y Relevancia: Un Marco Teórico y Metodológico para estudiar sistemas de información*. (p.22). Presentado en Primera Conferencia Colombiana en Gestión de Sistemas de Información y de TIC, Universidad de Manizales.

ACIS. (2019). *Consolidado de Maratones de Programación / ACIS*. Asociación Colombiana de Ingenieros de Sistemas. <https://acis.org.co/portal/content/consolidado-de-maratones-de-programacion>.

Akcaoglu, M. (2016). Design and Implementation of the Game-Design and Learning Program. *TechTrends*, 60(2), 114-123. <https://doi.org/10.1007/s11528-016-0022-y>

Andreessen, M. (2011). *Why Software Is Eating the World—WSJ*. THE WALL STREET JOURNAL. <https://www.wsj.com/articles/SB10001424053111903480904576512250915629460>

14. Lista de Anexos

Anexo 1. Manual de Usuario - Estudiante

Manual de Usuario para estudiantes

Plataforma de software: Training Center UFPS

Proyecto: Training Center High School

V. 2.0.0

Training Center UFPS

Training Center UFPS es una plataforma para el entrenamiento en programación competitiva. A través de ella puedes mejorar tus habilidades algorítmicas y de programación, mientras te mides con tus compañeros y contigo mismo.

La plataforma está diseñada de forma intuitiva y con múltiples detalles para hacer más fácil el acceso a la información solicitada.

A continuación, se presentan las principales funciones de la plataforma, y te invitamos a recorrerla y familiarizarte con cada uno de sus componentes.

1. Bienvenido (Home)

En la página inicial encontrarás una vista primeramente con un menú en la parte superior:



En este Menú podrás acceder a:

- **Desarrolladores:** La cual te redirecciona a una vista que presenta todas las personas que han colaborado en la elaboración, actualización y mejoras de esta plataforma:
- **Material público:** Aquí podrás acceder al material (diapositivas, videos, lecturas) que el grupo de estudio comparte de manera pública. Para esto no hay necesidad de haberte registrado o iniciado sesión

The screenshot shows the 'Training Center' interface. At the top, there is a navigation bar with the logo and the text 'Training Center'. To the right of the logo are links for 'Principal', 'Material público', 'Iniciar sesión', and 'Registrarse'. Below the navigation bar, there are two dropdown menus: 'Mostrar 8 materiales' and 'Ordenar por Id en forma Ascendente'. The main content area displays a grid of eight course cards with the following titles: 'Introducción a la programación competitiva', 'Tips en programación competitiva', 'Entradas y salidas', 'Estructuras de datos I', 'Matemáticas I', 'Matemáticas II', 'Manejo de strings', and 'Algoritmos voraces'. At the bottom of the grid, there is a pagination control with buttons for 'Inicio', 'Anterior', '1', '2', '3', '4', '5', 'Siguiente', and 'Final'.

- **Iniciar sesión:** Se despliega un modal en el que podrás iniciar sesión con tu correo y contraseña si ya tienes una cuenta lo contrario te da la opción de registrarte, mostrando así un formulario de registro el cuál te permite crear una cuenta.

The screenshot shows the 'Iniciar Sesión' modal form. The modal has a red header with the logo and the text 'Iniciar Sesión'. Below the header, there are two input fields: 'Correo Electrónico' with the value 'programacioncompetitivaufps+1@gmail.c' and 'Contraseña' with a masked password '.....'. Below the input fields is a red button labeled 'Iniciar Sesión'. At the bottom of the modal, there are two links: '¡Regístrate!' and '¿Olvidaste tu contraseña?'. The background of the modal is a dark red color with a stylized graphic of a globe and a rocket.

Regístrate

Nombre

Username
programacioncompetitivaups+1@gmail.com

Código (0 si no eres estudiante UFPS) ✓
0

Tipo de institución

Correo Electrónico

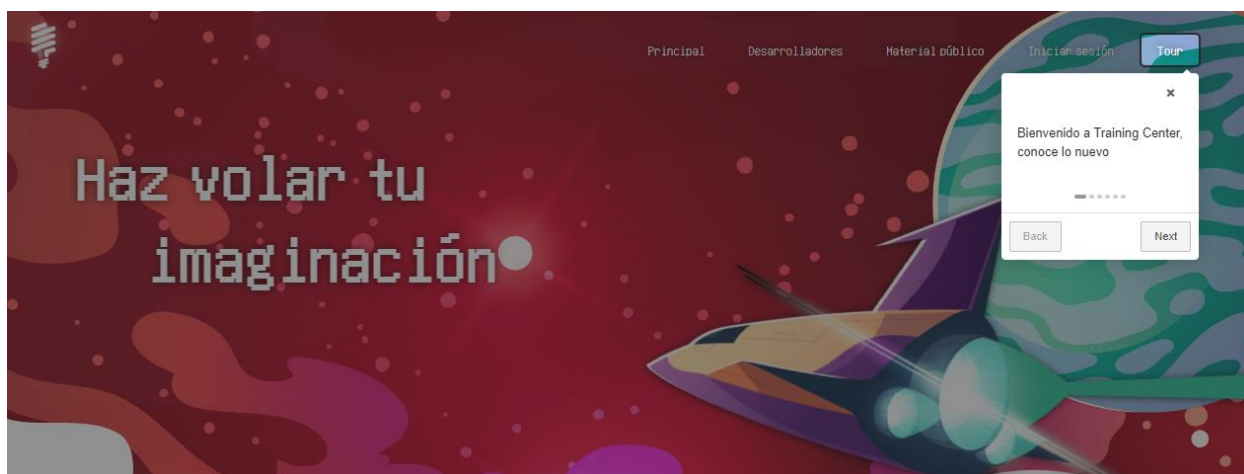
Contraseña

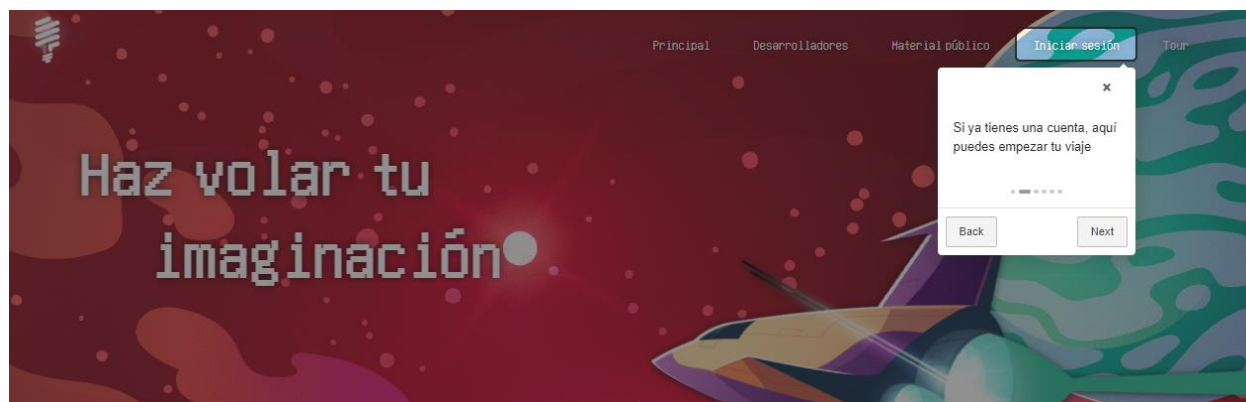
Repite la contraseña

Regístrate

¿Ya tienes una cuenta? ¡Inicia Sesión!

- Tour: Por medio de esta opción podrás hacer un recorrido por todo lo nuevo que se ha integrado en cada vista.





En la parte de abajo encontrarás la sección “Novedades”, en la cual podrás ver todo lo nuevo relacionado a la programación competitiva.

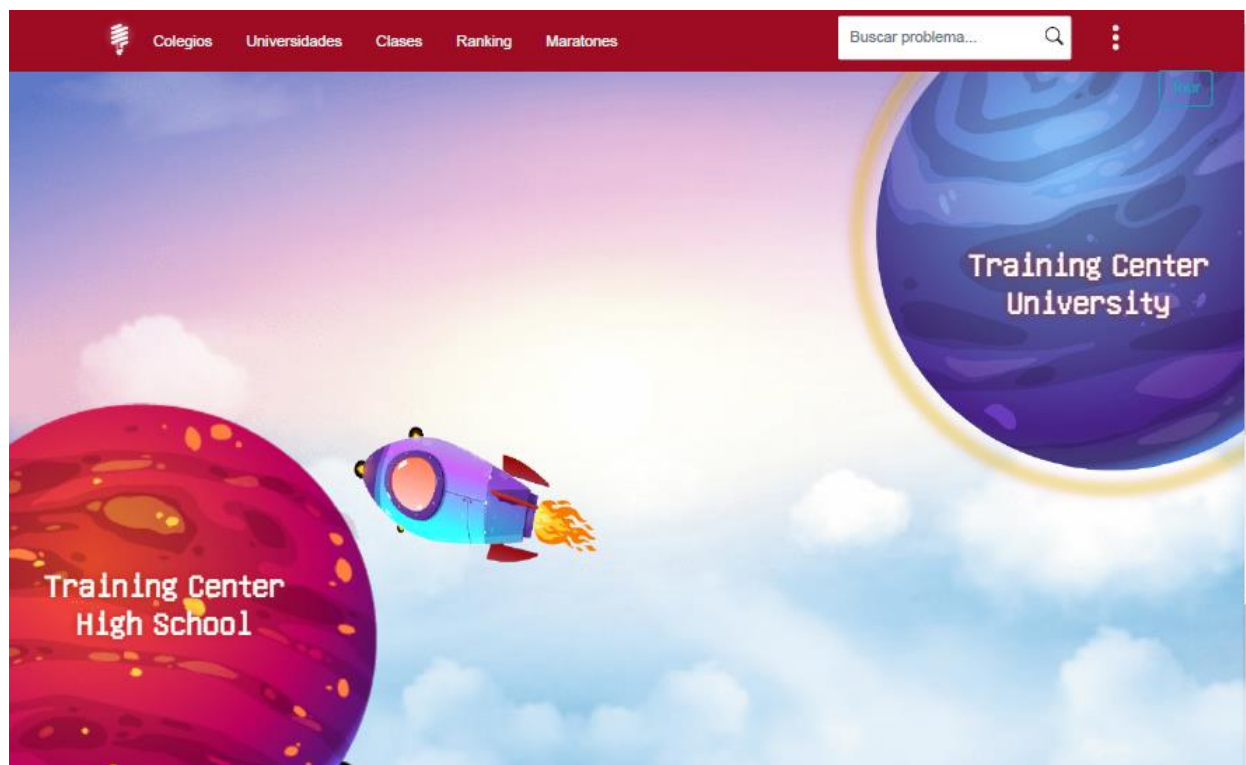


Al seguir bajando encontrarás la sección “Temas de interés” donde podrás visualizar temas relevantes.



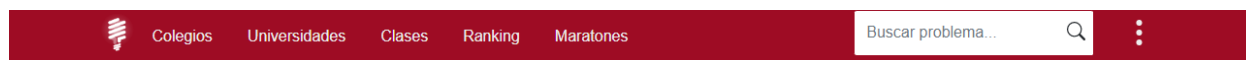
2. Principal

Al iniciar sesión te encontrarás con nuestra página “principal”



Aquí podrás elegir entre las 2 opciones (mundos) “Training Center High School” el cual es nuestro nuevo modulo enfocado para estudiantes de colegio, o “Training Center University” el cuál conserva lo que tenia la plataforma en su versión inicial más algunos ajustes y nuevas funcionalidades.

En la parte superior encuentras el menú de navegación que te permite moverte por los diferentes módulos y opciones disponibles.



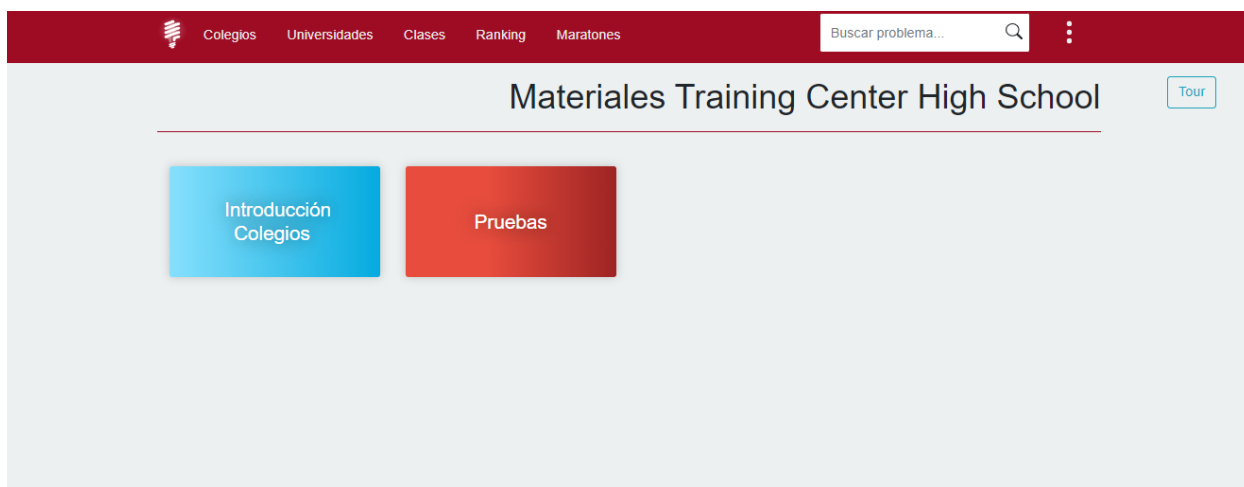
- Colegios, donde podrás acceder a todas las funcionalidades del módulo colegios
- Universidades, donde podrás acceder a todas las funcionalidades del módulo Universidades
- Clases, en donde puedes realizar cursos guiados, bien sea como parte de una materia o del grupo de estudio.
- Ranking, para comparar tus resultados en la institución con respecto a los demás usuarios.
- Maratones, donde encuentras las competencias de maratones de programación en las que puedes participar o también puedes crear una competencia de programación.

3. Training Center High School (Colegios)

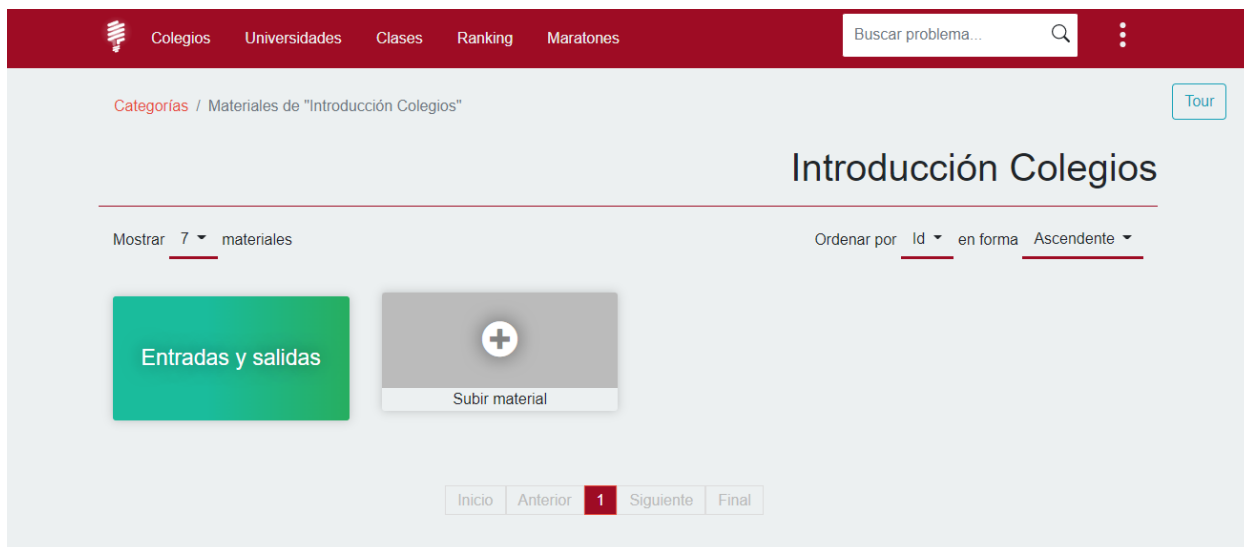


Aquí encontrarás 2 opciones:

Practica una lección: Dónde encontrarás todo el material disponible para que puedas ir practicando temas y lecciones que te ayudarán a resolver ejercicios.



El material lo encontrarás clasificado por categorías.



The screenshot shows a web application interface with a dark red header. The header contains navigation links: 'Colegios', 'Universidades', 'Clases', 'Ranking', 'Maratones', and 'Admin'. A search bar with the text 'Buscar problema...' and a magnifying glass icon is on the right. Below the header, there is a breadcrumb trail: 'Categorías / Materiales de "Introducción Colegios" / Entradas y salidas'. A 'Tour' button is visible in the top right corner. The main content area is titled 'Entradas y salidas' and includes a sub-header: 'Este material te servirá para conocer que papel cumplen las entradas y salidas de los problemas y como puedes manejarlas por medio de blockly.' Below this is a preview of a Blockly workspace. The workspace has a title bar 'Entradas' and a toolbar with icons for undo, redo, and zoom. The main canvas contains a yellow box with the text 'ENTRADAS' and two paragraphs of text: 'Las entradas son los datos que nos proporcionan con los cuales debemos generar una salida para solucionar nuestro problema. Pero ¿Cómo podemos leer esas entradas?' and 'Los siguientes bloques nos ayudarán a leer nuestras entradas para poder crear nuestra solución con ellas.' The workspace is decorated with colorful shapes and icons.

Cada categoría contiene el material relacionado. Además de ver el material, también tienes la opción de añadir material, el cual pasará a ser revisado por el administrador para posteriormente ser aprobado y que se pueda visualizar en la plataforma.

Resolver ejercicios: Aquí encontrarás la sección de problemas en la cuál podrás ver las diferentes categorías que hacen parte de la plataforma, y por cada una de ellas los problemas.

Colegios Universidades Clases Ranking Maratones

Buscar problema...

Problemas Training Center High School

Tour

Categorías | Lista completa

Introducción Colegios Pruebas

Colegios Universidades Clases Ranking Maratones

Buscar problema...

Categorías / Problemas de "Introducción Colegios"

Introducción Colegios

Tour

Mostrar 10 problemas en Cualquier idioma

Ordenar por Id en forma Ascendente

Id	Problema	Dificultad	Idioma
1001	Hola mundo ✓	1	ES EN
1002	Hola Mundo Reloaded ✓	1	ES EN
1003	Multiplicando por 11 ✓	1	ES EN
1004	Cociente y Residuo ✓	1	ES EN
1005	Doble de un número	1	ES EN
1006	Mavor v Menor	1	ES EN

Cuando abres un problema, verás la siguiente interfaz:

Colegios Universidades Clases Ranking Maratones

Q
⋮

Categorías / Problemas en esta categoría / Problema "Hola mundo" Tour

Hola mundo

+Entradas y salidas
+ciclos
+condicionales

ES | EN

Descripción

Se llama "Hola mundo" a un programa muy simple, cuya única función es mostrar en pantalla el texto "Hola mundo". Por su simpleza, cientos de profesores de programación en todo el mundo utilizan este ejemplo como la primera aproximación de sus estudiantes a la programación.

En este problema justamente haremos un "Hola mundo". Pero no lo hagamos tan simple. En vez de saludar siempre al mundo, saludemos a un listado de personas específicas. ¿Estas listo? ¡Inténtalo!

Entradas

La primera línea de la entrada contiene un número N ($1 \leq N \leq 100$), representando el número de personas que debe saludar tu programa. A continuación aparecen N líneas, cada una con el nombre de una persona a saludar. Todos los

Problema: Hola mundo

Dificultad: 1

Crea tu solución utilizando el tablero de Blockly.

Enviar

¡Felicitaciones!
Ya has resuelto este problema anteriormente.

saludar tu programa. A continuación aparecen N líneas, cada una con el nombre de una persona a saludar. Todos los nombres están en minúscula, conteniendo como máximo 10 caracteres entre la 'a' y la 'z', sin espacios.

Salidas

Por cada nombre de la entrada debes imprimir una línea saludando a esa persona. Si por ejemplo, la persona se llama "alfred", tu programa debe imprimir "Hola alfred". "Hola" siempre debe iniciar con H mayúscula, y debe haber un espacio entre "Hola" y el nombre. El nombre debe estar en minúscula, tal como se leyó desde la entrada. Para mas detalles, mira las salidas de ejemplo.

Entrada de ejemplo

3
alfred
erick
andres

Salida de ejemplo

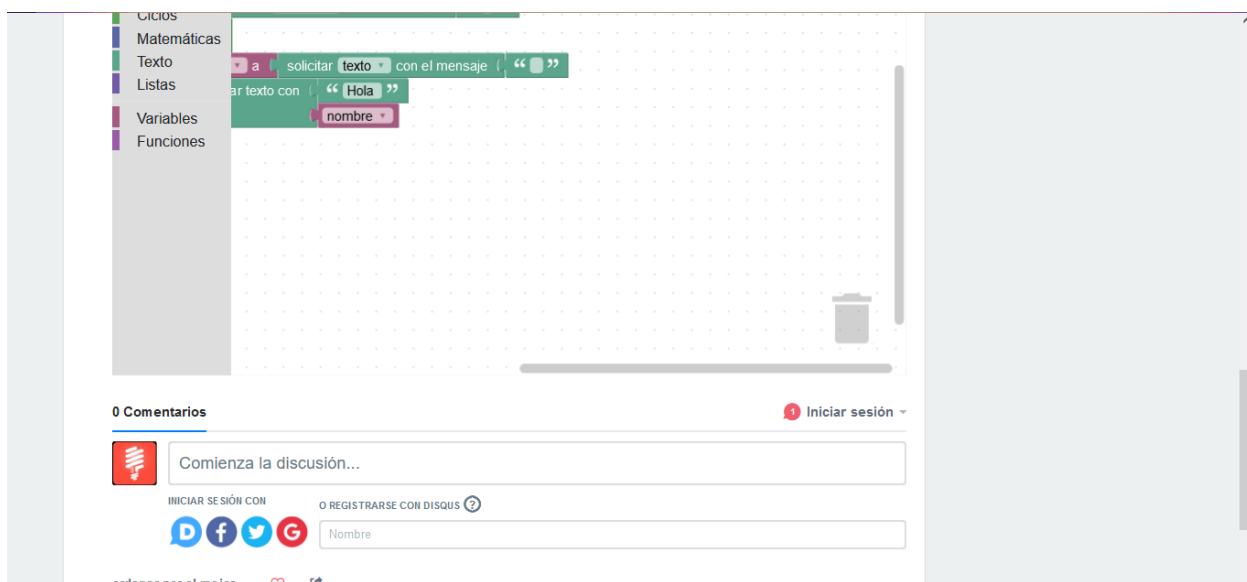
Hola alfred
Hola erick
Hola andres

Ahora, intenta resolver este ejercicio

Ayuda

- Entradas
- Salidas
- Lógica

Solicitar número con el mensaje. “ ”



Desde aquí puedes ver la descripción del problema, la definición de las entradas y las salidas, así como ejemplos de entrada y salida. Si el problema está disponible en múltiples idiomas, puedes cambiarlo haciendo clic en el idioma respectivo justo debajo del título. En la parte de abajo puedes visualizar el espacio donde puedes crear tu solución en lenguaje de bloques (Blockly). Cuando tengas lista tu solución solo debes dar clic en “Enviar”.

Justo después del tablero de Blockly podrás encontrar nuestro foro, en el cual puedes dejar tus dudas o comentarios relevantes acerca del problema.

Nota: Si ya has resuelto el problema de manera satisfactoria, cuando ingreses al problema verás la etiqueta que aparece en la imagen.



4. Training Center University

Problemas

En la sección problemas, podrás ver las diferentes categorías que hacen parte de la plataforma, y por cada una de ellas los problemas y el material disponible (Este material es el mismo que puedes ver desde la sección pública sin iniciar sesión).

The screenshot shows the 'Problemas' section of the Training Center University website. The navigation bar includes a search bar with the text 'Buscar problema...' and a magnifying glass icon, along with menu items: 'Colegios', 'Universidades', 'Clases', 'Ranking', and 'Maratones'. The main content area is titled 'Problemas' and includes a link for 'Categorías | Lista completa'. Below this, there is a grid of 12 category boxes, each with a title and two sub-links: 'Problemas' and 'Material'.

Category	Problemas	Material
Introducción	Problemas	Material
Estructuras de datos	Problemas	Material
Matemáticas	Problemas	Material
Teoría de grafos	Problemas	Material
Strings	Problemas	Material
Algoritmos voraces	Problemas	Material
Geometría computacional	Problemas	Material
Programación dinámica	Problemas	Material
Otros	Problemas	Material
Divide y vencerás	Problemas	Material
Implementación	Problemas	Material
Búsqueda completa	Problemas	Material

Categorías / Problemas de "Introducción"

Introducción

Mostrar 10 problemas en Cualquier idioma Ordenar por Id en forma Ascendente

Id	Problema	Dificultad	Idioma
0	Hola mundo	1	ES EN
2	Partido amistoso ✓	1	ES EN
25	Estación de buses ✓	1	ES EN
26	Conectores ✓	1	ES EN
30	Torre de CDs ✓	1	ES EN
36	Operadores relacionales ✓	1	ES EN

Cuando ingresa un problema, verás la siguiente interfaz:

Categorías / Problemas en esta categoría / Problema "Partido amistoso"

Partido amistoso

ES | EN

Descripción

En su preparación para el mundial de fútbol, Colombia enfrenta a Francia en un partido amistoso. Todo el país está frente a su televisor esperando el partido, excepto un grupo de talentosos jóvenes programadores que están frente a sus computadores mejorando sus habilidades en programación competitiva.

Una vez termina el partido, en el videobeam del salón donde se encuentran se proyectan dos números. El primero de ellos, la cantidad de goles que hizo Francia, equipo local. El segundo, la cantidad de goles que hizo Colombia. Con estos dos resultados, ¿Puedes avisarle al grupo de jóvenes quien ganó el partido?

Entradas

La primera línea contiene un número entero N ($1 \leq N \leq 100$), la cantidad de casos de prueba. A continuación aparecen N líneas, cada una con dos números enteros entre 0 y 10 separados por un espacio. La cantidad de goles de Francia y Colombia respectivamente.

Problema: Partido amistoso

Dificultad: 1

Selecciona el archivo con tu código, y el lenguaje a utilizar.

Lenguaje...

¡Felicitaciones!
Ya has resuelto este

Salidas

Por cada caso de prueba imprime "ganamos" si ganó Colombia, "perdimos" si ganó Francia, o "casi ganamos" si el partido termina en empate.

Entrada de ejemplo

```
3
2 1
1 2
2 2
```




Salida de ejemplo



```
perdimos
ganamos
casi ganamos
```

1 Comentario Iniciar sesión ▾

Únete a la discusión...

INICIAR SESIÓN CON O REGISTRARSE CON DISQUS ?

ordenar por el mejor ▾  

Ya has resuelto este problema anteriormente.

Desde aquí puedes ver la descripción del problema, la definición de las entradas y las salidas, así como ejemplos de entrada y salida. Si el problema está disponible en múltiples idiomas, puedes cambiarlo haciendo clic en el idioma respectivo justo debajo del título. Para enviar tu solución a un problema basta con adjuntar su código fuente en la sección derecha de la interfaz y elegir el lenguaje en el que resolviste el problema.

Nota: Si ya has resuelto el problema de manera satisfactoria, cuando ingreses al problema verás la etiqueta que aparece en la imagen.



5. Clases

Desde la sección clases puedes ver las clases en las que estás inscrito, o inscribirte a nuevas clases. Si las clases son públicas, basta hacer clic en registrarte. Si son privadas, necesitas previamente contar con la contraseña que te debe compartir el profesor o coach que creó dicha clase.

The screenshot shows a web interface for managing classes. At the top, there is a navigation bar with links for 'Colegios', 'Universidades', 'Clases', 'Ranking', and 'Maratones'. A search bar labeled 'Buscar problema...' is also present. The main content is divided into two sections: 'Mis clases' and 'Clases disponibles'.

Mis clases: This section displays three class cards:

- Programación Competitiva 2018-I:** Public class, buttons for 'Ver clase' and 'Detalle'.
- Programación Competitiva 2018-II:** Private class, buttons for 'Ver clase' and 'Detalle'.
- Programación Competitiva 2019-I:** Public class, buttons for 'Ver clase' and 'Detalle'.

Clases disponibles: This section displays eight class cards:

- Programación Competitiva 2018-I:** Public, Registered.
- Silux:** Private, Register.
- Programación Competitiva 2018-II:** Public, Registered.
- Probando cadenas:** Private, Register.
- Programación Competitiva 2019-I:** Public, Registered.
- Programación Competitiva 2019-II:** Public, Register.
- Programación Competitiva 2020-I:** Public, Register.
- Programación Competitiva 2020-II:** Public, Register.

At the bottom, there is a pagination control with buttons for 'Inicio', 'Anterior', '1' (selected), '2', 'Siguiente', and 'Final'.

Una vez estés en una clase, podrás ver las diferentes tareas asignadas por el profesor o coach, y los materiales que él te haya compartido. Ten en cuenta que las tareas tienen un plazo establecido, ¡no dejes que ese plazo se venza sin haberlas terminado!

Clases / Programación Competitiva 2018-I

Programación Competitiva 2018-I

Clase del grupo de estudio en programación competitiva de la Universidad Francisco de Paula Santander

Salir de esta clase

Tareas

- Semana I - Introducción (básico) - Tarea cerrada
- Semana II - Arreglos (básico) - Tarea cerrada
- Semana II - Búsqueda completa (avanzado) - Tarea cerrada
- Semana III - Pilas y colas (básico) - Tarea cerrada
- Semana III - Greedy (avanzado) - Tarea cerrada
- Semana IV - Ad-Hoc (básico) - Tarea cerrada
- Semana IV - Divide y vencerás (avanzado) - Tarea cerrada
- Semana V - Sets y maps (básico) - Tarea cerrada

Clases / Clase actual / Tarea "Semana I - Introducción (básico)"

Semana I - Introducción (básico)

Ejercicios introductorios a la Programación Competitiva para los estudiantes que están iniciando en el grupo. Ante cualquier inquietud, puede acercarse a los monitores del grupo.

Disponible desde el 23 de Marzo del 2018 a las 2:00PM hasta el 6 de Abril del 2018 a las 5:00PM

Ordenar por Id en forma Ascendente

Id	Problema	Dificultad	Idioma
1	Hola mundo	1	ES EN
2	Partido amistoso	1	ES EN
6	Divisores	1	ES EN
3	Número de divisores	1	ES EN

Si por error te has registrado en la clase equivocada también tienes la opción de desvincularte de la clase.

En el detalle de la clase puedes ver la clasificación de la clase y medir tus resultados en esa clase.

Puesto	Usuario	Soluciones correctas	Envios realizados
1	Crisel Ayala :D	67	112
2	Manuel Salazar	66	94
3	Hernández.	61	116
4	William Moreno	51	86
5	Carlos Calderón	49	65
6	Sebastian Vega	48	110
7	david alejandro tolosa zabala	47	62

6. Ranking

Puesto	Usuario	Institución	Inst. Tipo	Soluciones correctas	Envios realizados
1	Edinsson Melo (Mackenzie)	UFPS	Universidad	265	976
2	Marco (MarcoReus)	UFPS	Universidad	248	510
3	Sebastian Vega (C bas)	UFPS	Universidad	244	606
4	Crisel Ayala :D (Tarde pero seguro prros :v)	UFPS	Universidad	224	632
5	Coach Grupo (coachGrupo)	UFPS	Universidad	221	453
6	Edid (Yoshuab)	UFPS	Universidad	216	207

Aquí puedes encontrar la clasificación general con respecto a todos los usuarios de la plataforma y sus soluciones enviadas, para así comparar tus resultados. Esta información la puedes filtrar según tu interés; por tipo de institución, año o categoría

7. Maratones

No solo puedes competir en maratones, sino que puedes crear las tuyas propias con ejercicios de la plataforma, y compartirla con tus compañeros.

Crear Maratón

Maratones en ejecución y futuras

Filtrar por institución: Seleccione

Mostrar 10 maratones

Ordenar por Id en forma Ascendente

Id	Nombre	Visibilidad	Fecha de inicio	
8	Training 27/04/2018	Pública	27 de Abril del 2018 - 7:30PM	Abrir
9	Training 04/05/2018	Pública	4 de Mayo del 2018 - 7:30PM	Abrir
16	Maratón Interna UFPS 05/05	Pública	5 de Mayo del 2018 - 1:00PM	Abrir
18	Training 13/05/2018	Pública	13 de Mayo del 2018 - 7:30PM	Abrir

[Colegios](#)
[Universidades](#)
[Clases](#)
[Ranking](#)
[Maratones](#)

[Maratones](#) / Nueva maratón

Nueva Maratón

Nombre:

Descripción:

Reglas:

Tipo:

Desde el: a las:

Hasta el: a las:

Hasta el: a las:

Esta maratón será:

 Pública Privada

Mis maratones

Mostrar maratones
 Ordenar por en forma

Id	Nombre	Visibilidad	Tipo	Fecha de inicio	Acción
146	Maratón Colegios	Privada	Universidad	20 de Noviembre del 2022 - 1:44PM	<input type="button" value="Editar"/> <input type="button" value="Ver"/>
148	Maratón TCHS	Privada	Colegio	21 de Noviembre del 2022 - 2:07PM	<input type="button" value="Editar"/> <input type="button" value="Ver"/>

Al crear una maratón tienes la opción de editar y ver. Al editar te aparecerá la siguiente vista

Desde el: a las:

Hasta el: a las:

Esta maratón será:
 Pública Privada

Nueva Clave:

[Editar Maratón](#)

Problemas

Añadir: [Añadir](#)

Id	Problema	Idioma
No hay problemas actualmente en esta maratón.		

Aquí podrás añadir problemas a la maratón por medio del id del problema.

Desde el: a las:

Hasta el: a las:

Esta maratón será:
 Pública Privada

Nueva Clave:

[Editar Maratón](#)

Problemas

Añadir: [Añadir](#)

Id	Problema	Idioma
1001	Hola mundo	ES EN

Al añadir problemas se debe tener en cuenta que, si la maratón ha sido creada de tipo colegio, solo se pueden ingresar problemas de tipo colegio, que son aquellos que su solución es en lenguaje de bloques. Si la maratón fue creada de tipo universidad, de la misma manera solo se pueden agregar problemas de tipo universidad.

The screenshot displays the 'Maratón TCHS' page. At the top left, there is a breadcrumb 'Maratones / Maratón TCHS'. On the right, the title 'Maratón TCHS' is shown with a countdown 'Inicia en 24:2:10'. Below this is a horizontal navigation bar with three tabs: 'Inicio' (highlighted in red), 'Problemas', and 'Resultados'. The main content area is divided into two columns. The left column contains a white box with the heading 'Descripción' and the text 'TCHS', 'Reglas', and 'No rendirse'. The right column shows the start time 'Inicio: 21 de Noviembre del 2022 - 2:07PM' and the end time 'Finalización: 21 de Noviembre del 2022 - 3:07PM', with a red 'Regístrate' button below.

Cuando la maratón inicia aquellos que se hayan registrado a la maratón podrán acceder a los problemas de la maratón y resolverlos y también podrán ir viendo los resultados de la maratón y quienes han resuelto más problemas.

Maratones / Training 27/04/2018 / Problemas

Training 27/04/2018

Competencia finalizada

Inicio / Problemas / Resultados


















Problema		Idioma
A	Múltiplos AB	ES EN
B	Rectángulos de madera	ES EN
C	Robot recolector de stickers	ES EN
D	Reina	ES EN
E	Serie de tribonacci	ES EN

Maratones / Training 27/04/2018 / Resultados

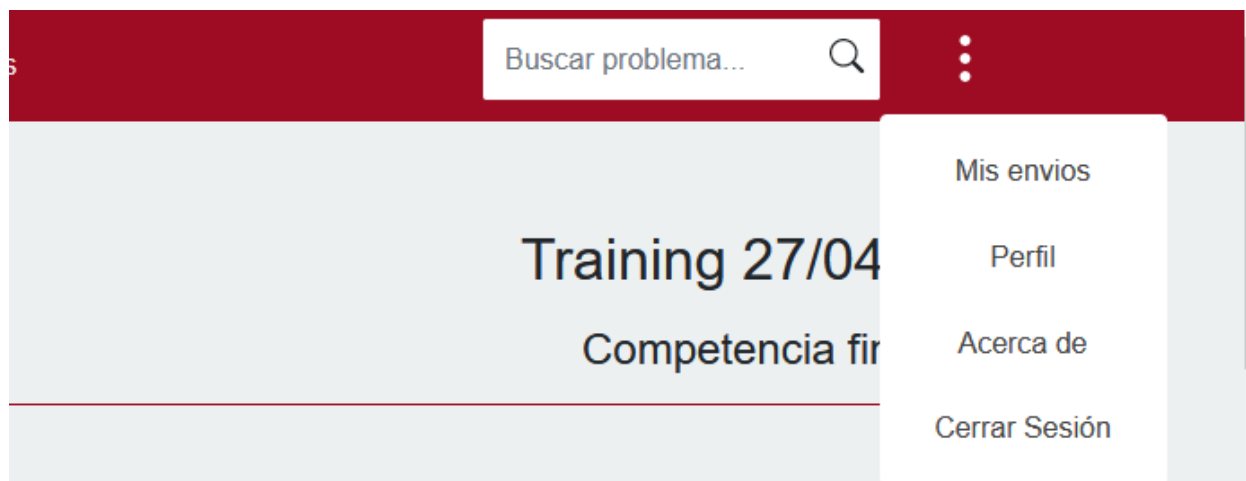
Training 27/04/2018

Competencia finalizada

Inicio / Problemas / Resultados

Posición	Usuario	A	B	C	D	E	Total
1	Crisel Ayala :D (Tarde pero seguro prros :v)	 1/4	 3/151	 1/108	 1/17	 1/23	5 (343)
2	William Moreno (Abracadabra)	 2/41			 1/61	 1/31	3 (153)
3	Julian Navarro- Najiluc (Najiluc)	 1/55			 1/63	 1/48	3 (166)
4	Karen Beltrán (karen24)	 1/21			 2/77	 1/96	3 (214)
5	Sebastian Vega (C						3 (227)

8. Menú final



En este menú final se encuentran las siguientes opciones

- **Mis envíos:** Permite visualizar tus envíos realizados, con la opción de ver el código realizado.

Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje	Código
Cociente y Residuo	1	17 de Noviembre del 2022 - 0:59AM	Correcto	0.019s	Blockly	Ver código
Multiplicando por 11	1	17 de Noviembre del 2022 - 0:46AM	Correcto	0.022s	Blockly	Ver código
Multiplicando por 11	1	17 de Noviembre del 2022 - 0:43AM	Respuesta equivocada	0.020s	Blockly	Ver código
Multiplicando por 11	1	17 de Noviembre del 2022 - 0:33AM	Error en tiempo de ejecución	0.020s	Blockly	Ver código
Hola Mundo		17 de Noviembre del 2022 -				Ver

Al dar en la opción de ver código muestra el siguiente modal:

Si el problema es de tipo colegio se muestra la imagen de la solución realizada, con la opción de descargar el código en lenguaje Python.

Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje
Cociente y Residuo	1	17 de Noviembre del 2022 - 0:59AM	Correcto	0.019s	Blockly

[Descargar código \(Python\)](#)

```

establecer count a solicitar número con el mensaje " "
repetir count veces
hacer
  establecer dividendo a Convertir a entero solicitar número con el mensaje " "
  establecer divisor a Convertir a entero solicitar número con el mensaje " "
  establecer cociente a Convertir a entero dividendo ÷ divisor
  establecer residuo a Convertir a entero dividendo - divisor × cociente
imprimir * crear texto con dividendo
" "
" = "
" "
divisor
" "
" * "

```

Si la solución enviada es de tipo universidad muestra el código fuente enviado.

Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje
Maxi Descuentos	1	12 de Mayo del 2019 - 6:40PM	Respuesta equivocada	0.008s	C++

[Descargar código](#)

```

#include <bits/stdc++.h>

using namespace std;

int main() {
    // your code goes here
    int n;

    while(cin>>n){
        vector<int> cajas;
        int vec [n];
        int vec4 [n];
        int ci;
        int menor= 1001;
        for(int i=0;i<n;i++){
            cin>>ci;

```

- Perfil: Permite ver tu información personal registrada
- Acerca de: Te permite conocer la información acerca de la plataforma, cuando se desarrolló y con qué objetivo
- Cerrar Sesión: Te saca de la sesión y redirige a la pantalla de inicio de la plataforma.

9. Recomendaciones

Después de ver este manual, es momento de crear tu cuenta y empezar a entrenar.

Aprovecha todas las opciones que UFPS Training Center provee.

Y no dudes en asistir al grupo de estudio en programación competitiva para potenciar tus habilidades y representar a la UFPS en competencias nacionales e internacionales.

Anexo 2. Manual de Usuario – Coach-Docente

Manual de Usuario para docentes y coach

Plataforma de Software: Training Center UFPS

Proyecto: Training Center High School

V. 2.0.0

UFPS Training Center

UFPS Training Center es una plataforma para el entrenamiento en programación competitiva. Originalmente, en las maratones de programación los equipos cuentan con un “coach” que puede ser un docente o estudiante y está a cargo del entrenamiento del equipo. Un coach puede crear problemas y tareas para que sus equipos entrenen y mejoren. Pero esta herramienta puede usarse más allá del entorno de programación competitiva y saltar al aula de clase, y por eso se invita a los docentes a que integren Training Center UFPS como parte de sus clases.

La plataforma está diseñada de forma intuitiva y con múltiples detalles para hacer más fácil el acceso a la información solicitada.

A continuación, se presentan las principales funciones de la plataforma, y te invitamos a recorrerla y familiarizarte con cada uno de sus componentes.

1. Registro

Solo los estudiantes pueden crear cuentas a través de la interfaz gráfica. Si eres un docente o un coach, anima a tus estudiantes a que se registren. Para crear tu cuenta, basta que la

solicites a un administrador. Si eres estudiante, pero quieres entrenar con tu equipo y asignar tareas internamente, adelante, comunícate con un administrador para crear la cuenta. No solo profesores pueden ser coach.

2. Bienvenido (Home)

En la página inicial encontrarás una vista primeramente con un menú en la parte superior:



En este Menú podrás acceder a:

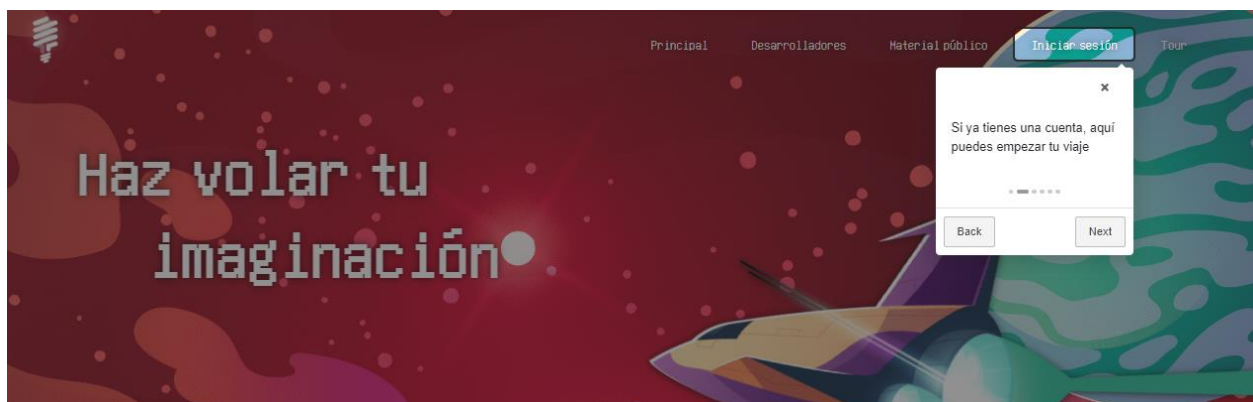
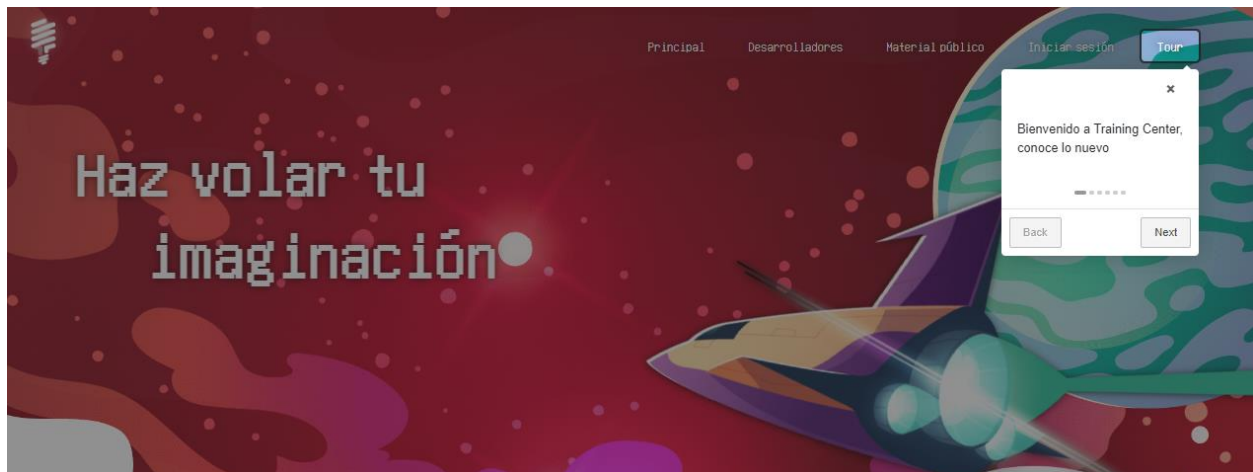
- **Desarrolladores:** La cual te redirecciona a una vista que presenta todas las personas que han colaborado en la elaboración, actualización y mejoras de esta plataforma:
- **Material público:** Aquí podrás acceder al material (diapositivas, videos, lecturas) que el grupo de estudio comparte de manera pública. Para esto no hay necesidad de haberte registrado o iniciado sesión

The screenshot shows the 'Training Center' interface. At the top, there is a navigation bar with the logo and the text 'Training Center'. To the right of the logo are links for 'Principal', 'Material público', 'Iniciar sesión', and 'Registrarse'. Below the navigation bar, there are filters for 'Mostrar 8 materiales' and 'Ordenar por Id en forma Ascendente'. The main content area displays a grid of eight course cards: 'Introducción a la programación competitiva', 'Tips en programación competitiva', 'Entradas y salidas', 'Estructuras de datos I', 'Matemáticas I', 'Matemáticas II', 'Manejo de strings', and 'Algoritmos voraces'. At the bottom of the grid, there is a pagination control with buttons for 'Inicio', 'Anterior', '1', '2', '3', '4', '5', 'Siguiente', and 'Final'.

Iniciar sesión: Cuando un administrador haya creado tu cuenta puedes iniciar sesión a través de tu correo y la contraseña que te fue asignada. En caso de olvidar la contraseña, la misma interfaz te proporciona un mecanismo rápido y seguro de recuperación.

The screenshot shows the 'Iniciar Sesión' (Login) form. The form is centered on a dark red background with a space-themed illustration. The form has a white background and a red header with the logo and the text 'Iniciar Sesión'. Below the header, there are two input fields: 'Correo Electrónico' (Email) and 'Contraseña' (Password). Below the input fields is a red button labeled 'Iniciar Sesión'. At the bottom of the form, there are two links: '¡Regístrate!' and '¿Olvidaste tu contraseña?'. The background features a large globe, a rocket, and the text 'Haz volar tu imaginación'.

- Tour: Por medio de esta opción podrás hacer un recorrido por todo lo nuevo que se ha integrado en cada vista.



En la parte de abajo encontrarás la sección “Novedades”, en la cual podrás ver todo lo nuevo relacionado a la programación competitiva.

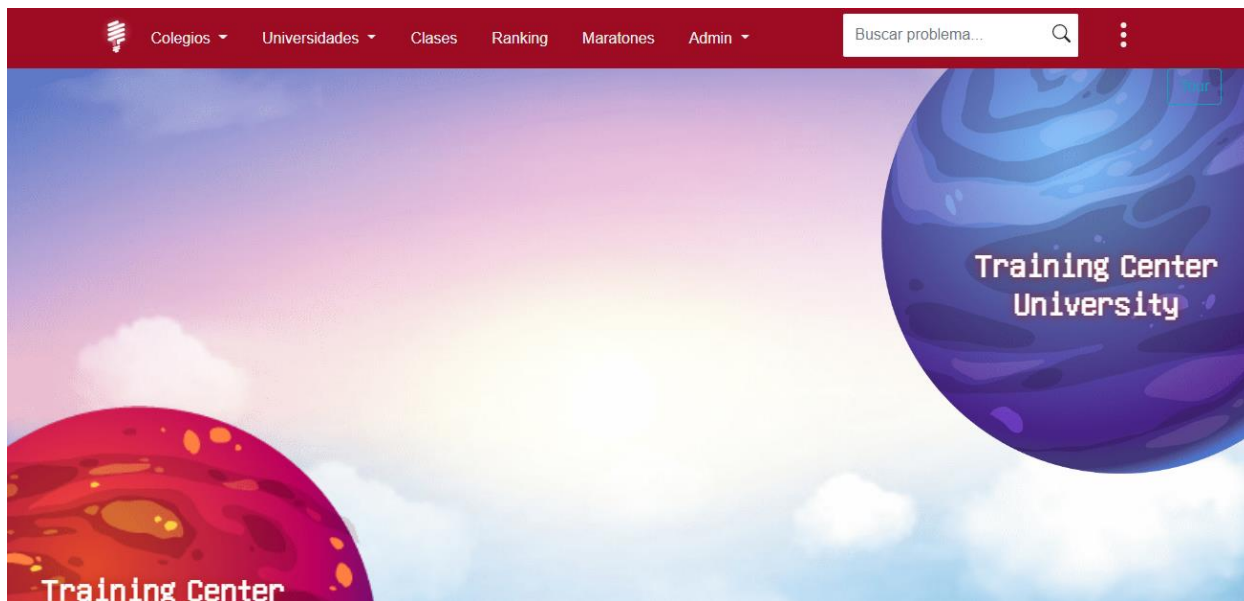


Al seguir bajando encontrarás la sección “Temas de interés” donde podrás visualizar temas relevantes.



3. Principal

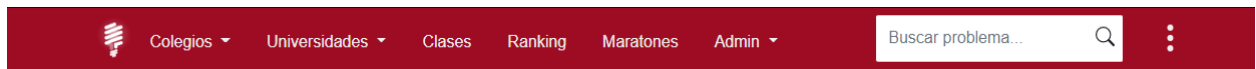
Al iniciar sesión te encontrarás con nuestra página “principal”



Aquí podrás elegir entre las 2 opciones (mundos) “Training Center High School” el cual es nuestro nuevo módulo enfocado para estudiantes de colegio, o “Training Center University” el cuál conserva lo que tenía la plataforma en su versión inicial más algunos ajustes y nuevas funcionalidades.

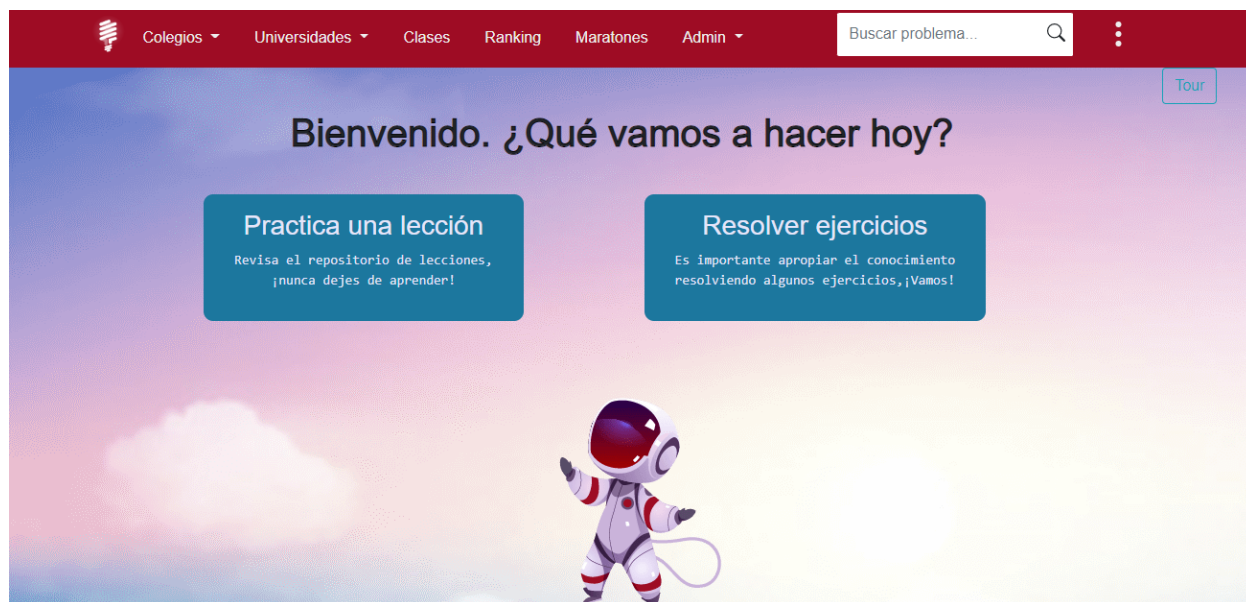
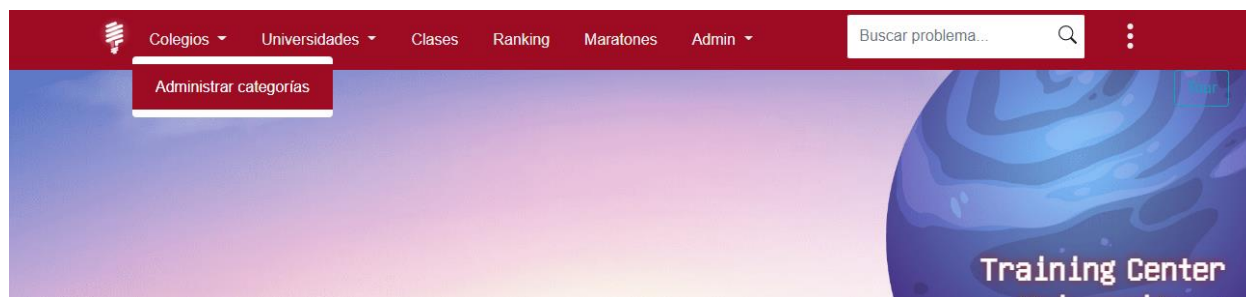
4. Acceso a las opciones

En la parte superior encuentras el menú de navegación que te permite moverte por los diferentes módulos y opciones disponibles.



- Colegios, donde podrás acceder a todas las funcionalidades del módulo colegios
- Universidades, donde podrás acceder a todas las funcionalidades del módulo Universidades
- Clases, en donde puedes realizar cursos guiados, bien sea como parte de una materia o del grupo de estudio.
- Ranking, para comparar tus resultados en la institución con respecto a los demás usuarios.
- Maratones, donde encuentras las competencias de maratones de programación en las que puedes participar o también puedes crear una competencia de programación.
- Admin, Despliega la opción que permite añadir problemas.

5. Training Center High School (Colegios)



Aquí encontrarás 2 opciones:

Practica una lección: Dónde encontrarás todo el material disponible para que puedas ir practicando temas y lecciones que te ayudaran a resolver ejercicios.

The screenshot shows the homepage of the Materials Training Center High School. At the top, there is a dark red navigation bar with a logo on the left and menu items: 'Colegios', 'Universidades', 'Clases', 'Ranking', 'Maratones', and 'Admin'. A search bar with the placeholder text 'Buscar problema...' and a magnifying glass icon is on the right. Below the navigation bar, the main heading is 'Materiales Training Center High School' with a 'Tour' button to its right. Two large buttons are displayed: a blue one labeled 'Introducción Colegios' and a red one labeled 'Pruebas'.

El material lo encontrarás clasificado por categorías.

Cada categoría contiene el material relacionado. Además de ver el material, también tienes la opción de añadir material, el cual pasará a ser revisado por el administrador para posteriormente ser aprobado y que se pueda visualizar en la plataforma.

The screenshot shows the 'Introducción Colegios' category page. The navigation bar is identical to the homepage. Below it, the breadcrumb trail reads 'Categorías / Materiales de "Introducción Colegios"'. The main heading is 'Introducción Colegios' with a 'Tour' button. Below the heading, there are two controls: 'Mostrar 7 materiales' and 'Ordenar por Id en forma Ascendente'. A green button labeled 'Entradas y salidas' is on the left, and a grey button with a plus sign and the text 'Subir material' is on the right. At the bottom, there is a pagination bar with buttons for 'Inicio', 'Anterior', '1' (highlighted), 'Siguiete', and 'Final'.

The screenshot shows a website interface with a dark red header. The header contains a logo on the left, navigation links for 'Colegios', 'Universidades', 'Clases', 'Ranking', 'Maratones', and 'Admin', a search bar with the text 'Buscar problema...', and a menu icon on the right. Below the header, there is a breadcrumb trail: 'Categorías / Materiales de "Introducción Colegios" / Entradas y salidas'. A 'Tour' button is located in the top right corner. The main content area features the title 'Entradas y salidas' and a subtext: 'Este material te servirá para conocer que papel cumplen las entradas y salidas de los problemas y como puedes manejarlas por medio de blockly.' Below this is a preview of a Blockly workspace. The workspace has a title bar 'Entradas' and a toolbar with icons for undo, redo, zoom (100%), and other functions. The workspace content includes a yellow box with the title 'ENTRADAS' and two paragraphs of text: 'Las entradas son los datos que nos proporcionan con los cuales debemos generar una salida para solucionar nuestro problema. Pero ¿Cómo podemos leer esas entradas?' and 'Los siguientes bloques nos ayudarán a leer nuestras entradas para poder crear nuestra solución con ellas.'

Resolver ejercicios: Aquí encontrarás la sección de problemas en la cuál podrás ver las diferentes categorías que hacen parte de la plataforma, y por cada una de ellas los problemas.

The screenshot shows a website interface with a dark red header. The header contains a logo on the left, navigation links for 'Colegios', 'Universidades', 'Clases', 'Ranking', 'Maratones', and 'Admin', a search bar with the text 'Buscar problema...', and a menu icon on the right. Below the header, there is a title 'Problemas Training Center High School' and a 'Tour' button. A breadcrumb trail reads 'Categorías | Lista completa'. Below the title, there are two buttons: a red button labeled 'Introducción Colegios' and a green button labeled 'Pruebas'.

Categorías / Problemas de "Introducción Colegios" Tour

Introducción Colegios

Mostrar 10 problemas en Cualquier idioma Ordenar por Id en forma Ascendente

Id	Problema	Dificultad	Idioma
1001	Hola mundo	1	ES EN
1002	Hola Mundo Reloaded	1	ES EN
1003	Multiplicando por 11	1	ES EN
1004	Cociente y Residuo	1	ES EN
1005	Doble de un número	1	ES EN
1006	Mayor y Menor	1	ES EN

Cuando abres un problema, verás la siguiente interfaz:

Categorías / Problemas en esta categoría / Problema "Hola mundo" Tour

Hola mundo

+Entradas y salidas +ciclos +condicionales

ES | EN

Descripción

Se llama "Hola mundo" a un programa muy simple, cuya única función es mostrar en pantalla el texto "Hola mundo". Por su simpleza, cientos de profesores de programación en todo el mundo utilizan este ejemplo como la primera aproximación de sus estudiantes a la programación.

En este problema justamente haremos un "Hola mundo". Pero no lo hagamos tan simple. En vez de saludar siempre al mundo, saludemos a un listado de personas específicas. ¿Estas listo? ¡Inténtalo!

Entradas

La primera línea de la entrada contiene un número N ($1 \leq N \leq 100$), representando el número de personas que debe

Problema: Hola mundo

Dificultad: 1

Crea tu solución utilizando el tablero de Blockly.

Enviar

saludar tu programa. A continuación aparecen N líneas, cada una con el nombre de una persona a saludar. Todos los nombres están en minúscula, conteniendo como máximo 10 caracteres entre la 'a' y la 'z', sin espacios.

Salidas

Por cada nombre de la entrada debes imprimir una línea saludando a esa persona. Si por ejemplo, la persona se llama "alfred", tu programa debe imprimir "Hola alfred". "Hola" siempre debe iniciar con H mayúscula, y debe haber un espacio entre "Hola" y el nombre. El nombre debe estar en minúscula, tal como se leyó desde la entrada. Para más detalles, mira las salidas de ejemplo.

Entrada de ejemplo

```
3
alfred
erick
andres
```

Salida de ejemplo

```
Hola alfred
Hola erick
Hola andres
```

Ahora, intenta resolver este ejercicio

[Ayuda](#)

0 Comentarios

ANGIE MADELEY...

Comienza la discusión...

ordenar por el mejor

Sé el primero en comentar.

Desde aquí puedes ver la descripción del problema, la definición de las entradas y las salidas, así como ejemplos de entrada y salida. Si el problema está disponible en múltiples idiomas, puedes cambiarlo haciendo clic en el idioma respectivo justo debajo del título. En la

parte de abajo puedes visualizar el espacio donde se puede crear la solución en lenguaje de bloques (Blockly). Cuando la solución esté lista solo debes dar clic en “Enviar”.

Justo después del tablero de Blockly podrás encontrar nuestro foro, en el cual puedes dejar tus dudas o comentarios relevantes acerca del problema o alguna aclaración adicional que como coach quieras dejarle a tus alumnos.

Nota: Si ya has resuelto el problema de manera satisfactoria, cuando ingreses al problema verás la etiqueta que aparece en la imagen.



6. Training Center University

En la sección problemas, podrás ver las diferentes categorías que hacen parte de la plataforma, y por cada una de ellas los problemas y el material disponible (Este material es el mismo que puedes ver desde la sección pública sin iniciar sesión).

Colegios ▾ Universidades ▾ Clases Ranking Maratones Admin ▾ Buscar problema... 🔍 ⋮

Administrar categorías

Problemas

Categorías | Lista completa

Introducción

Problemas Material

Estructuras de datos

Problemas Material

Matemáticas

Problemas Material

Teoría de grafos

Problemas Material

Strings

Problemas Material

Algoritmos voraces

Problemas Material

Geometría computacional

Problemas Material

Programación dinámica

Problemas Material

Otros

Problemas Material

Divide y vencerás

Problemas Material

Implementación

Problemas Material

Búsqueda completa

Problemas Material

Colegios ▾ Universidades ▾ Clases Ranking Maratones Admin ▾ Buscar problema... 🔍 ⋮

Categorías / Problemas de "Estructuras de datos"

Estructuras de datos

Mostrar 10 ▾ problemas en Cualquier idioma ▾ Ordenar por Id ▾ en forma Ascendente ▾

Id	Problema	Dificultad	Idioma
18	Invertido	1	ES EN
20	Invertido II	2	ES EN
21	Matriz loca	2	ES EN
23	Chris Challenge ✓	6	ES EN
24	Drought in nlogonia ✓	4	ES EN
27	Parentesis ✓	1	ES EN

The screenshot shows a web interface for a programming problem. At the top, there is a dark red navigation bar with a logo on the left and menu items: 'Colegios', 'Universidades', 'Clases', 'Ranking', 'Maratones', and 'Admin'. A search bar on the right contains the text 'Buscar problema...'. Below the navigation bar, a breadcrumb trail reads 'Categorías / Problemas en esta categoría / Problema "Invertido"'. The main content area is divided into two columns. The left column contains the problem details: the title 'Invertido' with 'ES | EN' below it, a 'Descripción' section with the text 'Escribe un programa que lee n números enteros, y luego los prime en orden invertido.', an 'Entradas' section with the text 'La primera línea contiene el entero n (1 <= n <= 100), el número de valores a leer. A continuación aparecen n líneas que contienen cada una un entero i (1 <= i <= 100).', and a 'Salidas' section with the text 'La salida debe contener los n valores leídos, impresos en orden invertido, es decir, imprime primero el último valor, después el penúltimo, y así sucesivamente hasta llegar al primero. Cada número debe estar en una línea.' Below these sections are two buttons: 'Entrada de ejemplo' and 'Salida de ejemplo'. The right column contains a control panel with the text 'Problema: Invertido', 'Dificultad: 1', and 'Selecciona el archivo con tu código, y el lenguaje a utilizar.' It features a 'Seleccionar' button, a language selection dropdown menu with 'Lenguaje...' and a downward arrow, and an 'Enviar' button.

Cuando abres un problema, verás la siguiente interfaz:

This screenshot is identical to the one above, showing the same web interface for the 'Invertido' programming problem. It includes the same navigation bar, breadcrumb trail, problem description, input/output specifications, and control panel.

Desde aquí puedes ver la descripción del problema, la definición de las entradas y las salidas, así como ejemplos de entrada y salida. Si el problema está disponible en múltiples idiomas, puedes cambiarlo haciendo clic en el idioma respectivo justo debajo del título. Para enviar tu solución a un problema basta con adjuntar su código fuente en la sección derecha de la interfaz.

7. Clases

Desde la sección clases puedes ver las clases que has creado previamente, crear otras nuevas, o administrar las ya existentes.

The screenshot displays the 'Mis clases' (My classes) section of a website. At the top, there is a dark red navigation bar with a logo on the left and a search bar on the right. The navigation bar contains the following items: 'Colegios', 'Universidades', 'Clases', 'Ranking', 'Maratones', and 'Admin'. The search bar contains the text 'Buscar problema...'. Below the navigation bar, the main content area is titled 'Mis clases'. It features a grid of class cards. Each card represents a class and includes a title, a 'Detalle' button, and an 'Estadísticas' button. The cards are: 'Programación Competitiva 2018-I' (blue), 'Programación Competitiva 2018-II' (red), 'Programación Competitiva 2019-I' (green), 'Programación Competitiva 2019-II' (blue), 'Programación Competitiva 2020-I' (red), and 'Programación Competitiva 2020-II' (green). A 'Nueva clase' button with a plus sign is also visible. At the bottom of the grid, there is a pagination bar with buttons for 'Inicio', 'Anterior', '1', 'Siguiente', and 'Final'.

Una vez estés en una clase, podrás ver las diferentes tareas y crear nuevas, así como compartir materiales con tus estudiantes, o ver un seguimiento de su rendimiento.

Colegios ▾ Universidades ▾ Clases Ranking Maratones Admin ▾

Clases / Programación Competitiva 2018-I

Programación Competitiva 2018-I

Clase del grupo de estudio en programación competitiva de la Universidad Francisco de Paula Santander

Tareas

Semana I - Introducción (básico)

[Editar tarea](#) [Estadísticas](#)

Semana II - Arreglos (básico)

[Editar tarea](#) [Estadísticas](#)

Semana II - Búsqueda completa (avanzado)

[Editar tarea](#) [Estadísticas](#)

Semana III - Pilas y colas (básico)

[Editar tarea](#) [Estadísticas](#)

Semana III - Greedy (avanzado)

[Editar tarea](#) [Estadísticas](#)

Semana IV - Ad-Hoc (básico)

[Editar tarea](#) [Estadísticas](#)

Semana IV - Divide y vencerás (avanzado)

[Editar tarea](#) [Estadísticas](#)

Semana V - Sets y maps (básico)

[Editar tarea](#) [Estadísticas](#)

Colegios ▾ Universidades ▾ Clases Ranking Maratones Admin ▾

Clases / Clase actual / Detalle de la tarea "Semana I - Introducción (básico)"

Semana I - Introducción (básico)

Detalles de la tarea

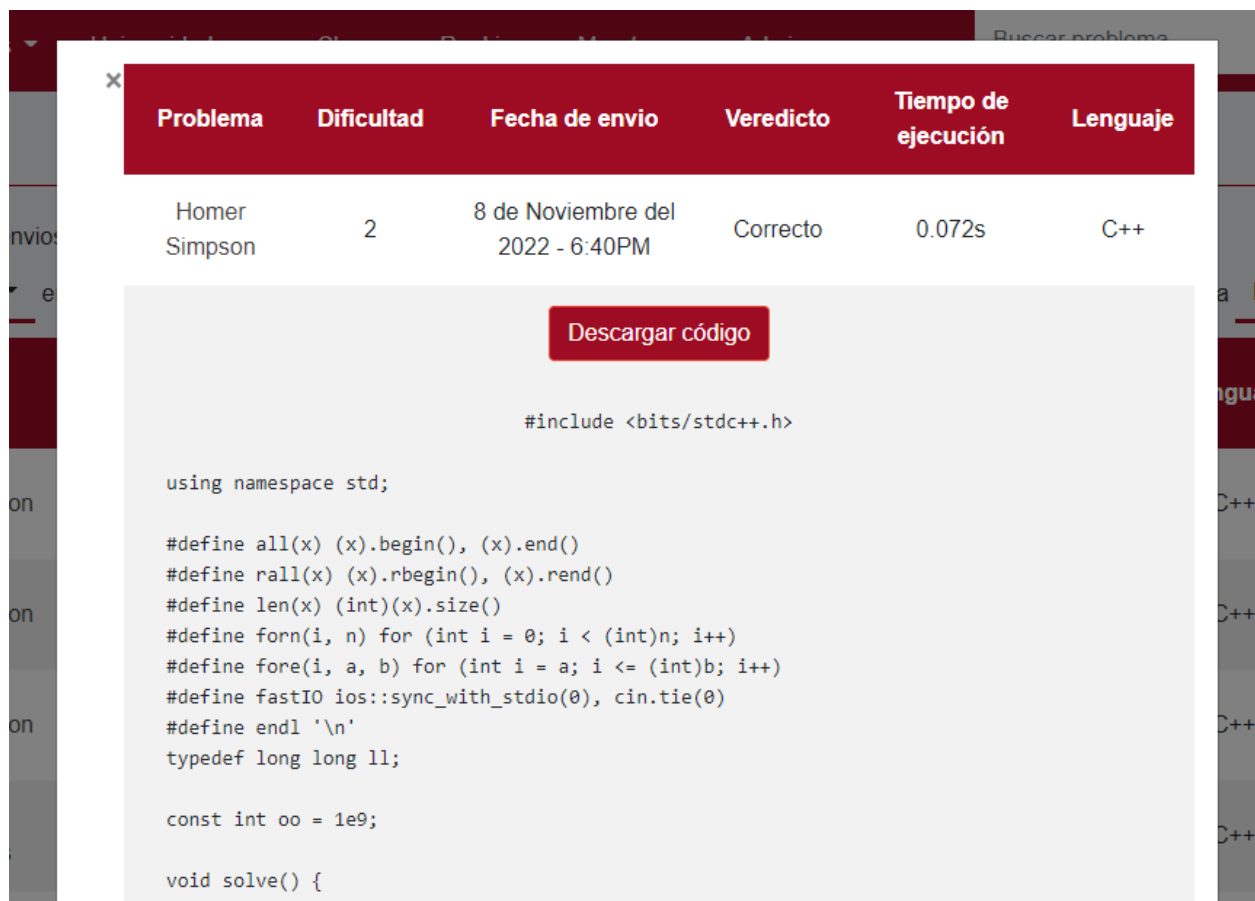
Indice	Problema	Dificultad	Idioma	Detalle
A	Hola mundo	1	ES EN	Detalle
B	Partido amistoso	1	ES EN	Detalle
C	Divisores	1	ES EN	Detalle
D	Número de divisores	1	ES EN	Detalle
E	Corra	2	ES EN	Detalle

8. Ranking

Puesto	Usuario	Institución	Inst. Tipo	Soluciones correctas	Envíos realizados	Ver envíos
1	Edinsson Melo (Mackenzie)	UFPS	Universidad	265	976	Ver
2	Marco (MarcoReus)	UFPS	Universidad	248	510	Ver
3	Sebastian Vega (C bas)	UFPS	Universidad	244	606	Ver
4	Crisel Ayala :D (Tarde pero seguro prros :v)	UFPS	Universidad	224	632	Ver
5	Coach Grupo (coachGrupo)	UFPS	Universidad	221	453	Ver

Aquí puedes encontrar la clasificación general con respecto a todos los usuarios de la plataforma y sus soluciones enviadas, para así comparar sus resultados. Esta información la puedes filtrar según tu interés; por tipo de institución, año o categoría. Como coach puedes ver los envíos de tus alumnos y supervisar sus códigos.

Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje	Código
Homer Simpson	2	8 de Noviembre del 2022 - 6:40PM	Correcto	0.072s	C++	Ver código
Homer Simpson	2	8 de Noviembre del 2022 - 6:40PM	Respuesta equivocada	0.075s	C++	Ver código
Homer Simpson	2	8 de Noviembre del 2022 - 6:39PM	Respuesta equivocada	0.004s	C++	Ver código
Maxi Descuentos	1	13 de Noviembre del 2020 - 5:08PM	Respuesta equivocada	0.008s	C++	Ver código



The screenshot shows a submission record for a problem named 'Homer Simpson'. The submission was made on November 8, 2022, at 6:40 PM, and was successful. The execution time was 0.072s and the language used was C++.

Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje
Homer Simpson	2	8 de Noviembre del 2022 - 6:40PM	Correcto	0.072s	C++

Below the submission record, there is a 'Descargar código' button and a code editor showing the following C++ code:

```
#include <bits/stdc++.h>

using namespace std;


#define all(x) (x).begin(), (x).end()
#define rall(x) (x).rbegin(), (x).rend()
#define len(x) (int)(x).size()
#define forn(i, n) for (int i = 0; i < (int)n; i++)
#define fore(i, a, b) for (int i = a; i <= (int)b; i++)
#define fastIO ios::sync_with_stdio(0), cin.tie(0)
#define endl '\n'
typedef long long ll;

const int oo = 1e9;

void solve() {
```

9. Maratones

Puedes crear maratones de programación y compartirlas con tus estudiantes.



[Colegios](#)
[Universidades](#)
[Clases](#)
[Ranking](#)
[Maratones](#)
[Admin](#)

[Crear Maratón](#)

Mis maratones

Mostrar 10 maratones Ordenar por Id en forma Ascendente

Id	Nombre	Visibilidad	Tipo	Fecha de inicio	Acción
8	Training 27/04/2018	Pública	Universidad	27 de Abril del 2018 - 7:30PM	Editar Ver
9	Training 04/05/2018	Pública	Universidad	4 de Mayo del 2018 - 7:30PM	Editar Ver
16	Maratón Interna UFPS 05/05	Pública	Universidad	5 de Mayo del 2018 - 1:00PM	Editar Ver
18	Training 13/05/2018	Pública	Universidad	13 de Mayo del 2018 - 7:30PM	Editar Ver


[Colegios](#)
[Universidades](#)
[Clases](#)
[Ranking](#)
[Maratones](#)
[Admin](#)

[Maratones](#) / Nueva maratón

Nueva Maratón

Nombre:

Descripción:

Reglas:

Tipo:

Desde el:

Hasta el:

Esta maratón será:

Pública
 Privada

[Crear Maratón](#)

10. Admin

Como coach puedes añadir problemas. Para eso, en el menú superior pulsa sobre Admin y en el menú desplegable selecciona añadir problema. Encontrarás la siguiente interfaz:

The screenshot shows the 'Nuevo problema' (New problem) form. At the top, there is a dark red navigation bar with a logo on the left and a search bar on the right containing the text 'Buscar problema...'. The navigation bar also contains several menu items: 'Colegios', 'Universidades', 'Clases', 'Ranking', 'Maratones', and 'Admin'. Below the navigation bar, the page title 'Nuevo problema' is displayed in a large, bold font. The form itself is divided into two main sections. The first section contains several input fields: 'Idioma' (Language) with a dropdown menu set to 'Español'; 'Nombre' (Name) with a text input field containing 'Nombre del problema'; 'Dificultad' (Difficulty) with a text input field containing '1'; 'Tipo de institución' (Institution type) with a dropdown menu set to 'Elige una institución'; 'Categoría' (Category) with a dropdown menu set to 'Elige una categoría'; and 'Tiempo límite' (Time limit) with a text input field containing '1'. The second section is titled 'Contenido' (Content) and features a rich text editor with a toolbar containing icons for bold, italic, heading, list, link, unlink, image, and help. Below the toolbar, the content area is divided into three sections: '# Descripción' (Description), '# Entradas' (Inputs), and '# Salidas' (Outputs). Each section has a placeholder text: 'Reemplaza este texto con la descripción de tu problema. Recuerda que puedes usar la sintaxis de Markdown.' for the description, and 'Reemplaza este texto con la especificación de la entrada de tu problema. Si no conoces la sintaxis markdown, puedes hacer uso de las opciones de la barra superior.' for the inputs.

Salidas

Reemplaza este texto con la especificación de la salida de tu problema.

Importante: LAS ENTRADAS Y SALIDAS DE EJEMPLO se mostrarán junto al enunciado en la plataforma. Se recomienda que estos sean algunos casos simples que expliquen brevemente las entradas y salidas. Por su parte, los casos de prueba PRIVADOS son los que utilizará la plataforma para evaluar los ejercicios. Por tanto, estos deberían ser mas extensos y completos (Dado que es posible que estos casos sean muy extensos, deben subirse en formato .txt o .in)

Entradas de ejemplo

Salidas de ejemplo

Seleccione los casos de prueba privados:

Entradas Salidas

Añadir versión en Inglés Guardar problema

En un par de minutos puedes agregar tus propios problemas a la plataforma para que sean solucionados por cualquier usuario.

11. Recomendaciones

Utiliza la plataforma, explora sus opciones a fondo, ayúdanos a cargar material útil y ejercicios retadores, y entre todos construyamos un Training Center UFPS que explote al máximo las capacidades de los estudiantes tanto de universidades como de colegios.

Anexo 3. Manual de Usuario – Administrador

Manual de Usuario para administradores

Plataforma de software: Training Center UFPS

Proyecto: Training Center High School

V. 2.0.0

UFPS Training Center

Training Center UFPS es una plataforma para el entrenamiento en programación competitiva. Esta herramienta es el centro de aprendizaje en el grupo de estudio en Programación Competitiva, y por tanto se invita a los líderes de dicho grupo a tomar la iniciativa de administrar la plataforma. Es una labor relativamente sencilla, y ayudará a mantener la herramienta al alcance de todos.

La plataforma está diseñada de forma intuitiva y con múltiples detalles para hacer más fácil el acceso a la información solicitada.

A continuación, se presentan las principales funciones de la plataforma, y te invitamos a recorrerla y familiarizarte con cada uno de sus componentes.

1. Registro

Solo los estudiantes pueden crear cuentas a través de la interfaz gráfica. Para crear una cuenta de administrador, es necesario que otro administrador la apruebe. Por tanto, dirígete a un administrador actual o a los líderes del grupo de estudio para solicitar la cuenta. Se respeta el criterio de los administradores actuales para aceptar o no a nuevos administradores.

2. Bienvenido (Home)

En la página inicial encontrarás una vista primeramente con un menú en la parte superior:

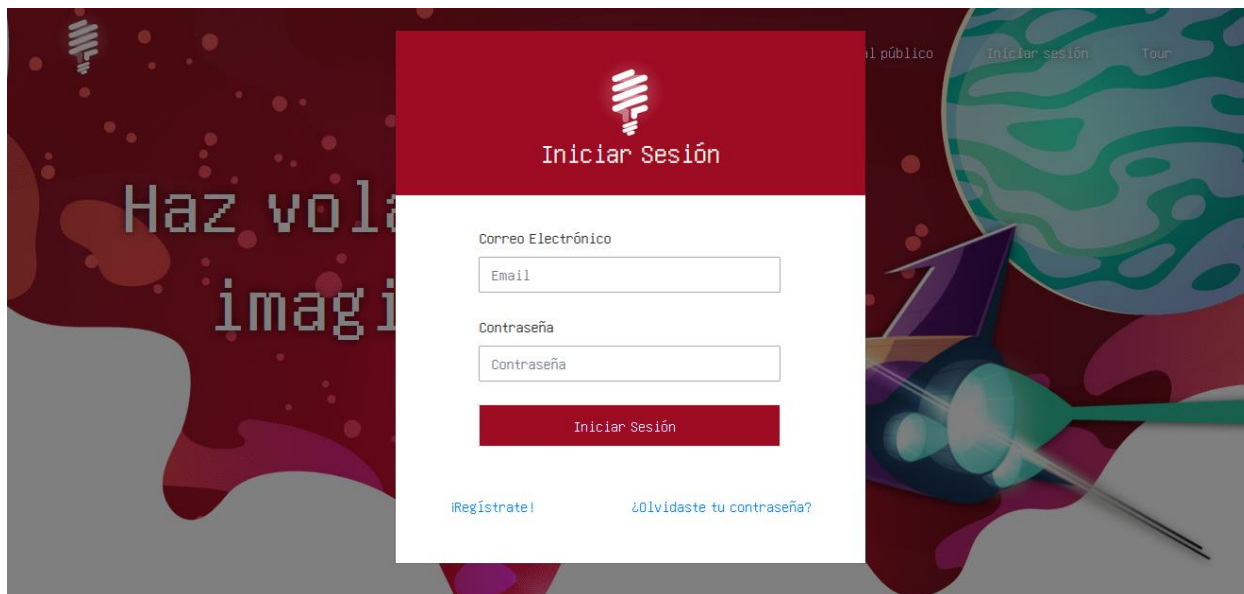


En este Menú podrás acceder a:

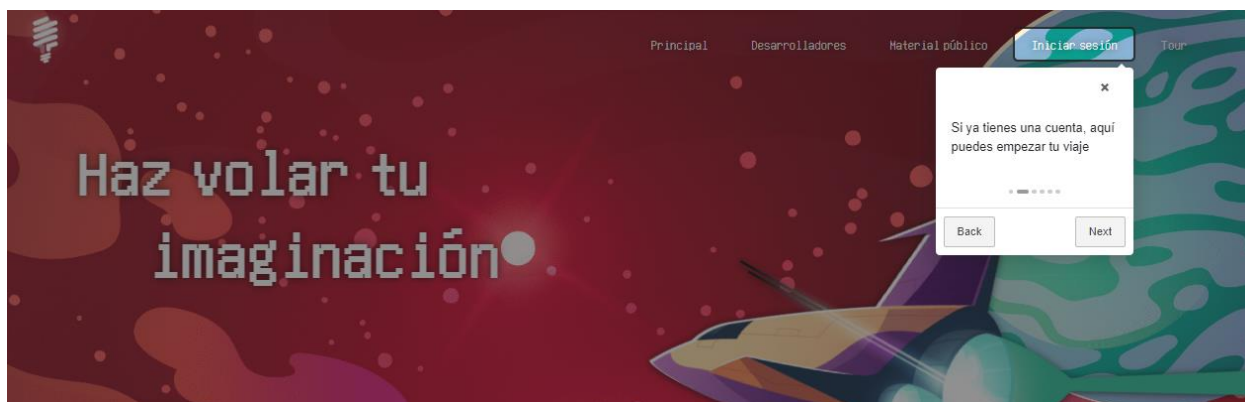
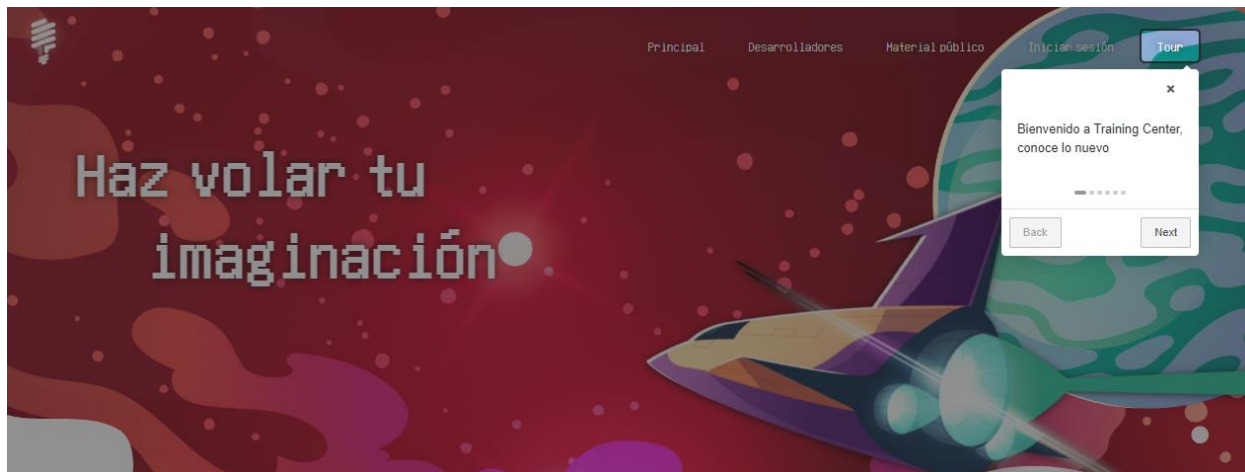
- **Desarrolladores:** La cual te redirecciona a una vista que presenta todas las personas que han colaborado en la elaboración, actualización y mejoras de esta plataforma:
- **Material público:** Aquí podrás acceder al material (diapositivas, videos, lecturas) que el grupo de estudio comparte de manera pública. Para esto no hay necesidad de haberte registrado o iniciado sesión.



Iniciar sesión: Cuando un administrador haya creado tu cuenta puedes iniciar sesión a través de tu correo y la contraseña que te fue asignada. En caso de olvidar la contraseña, la misma interfaz te proporciona un mecanismo rápido y seguro de recuperación.



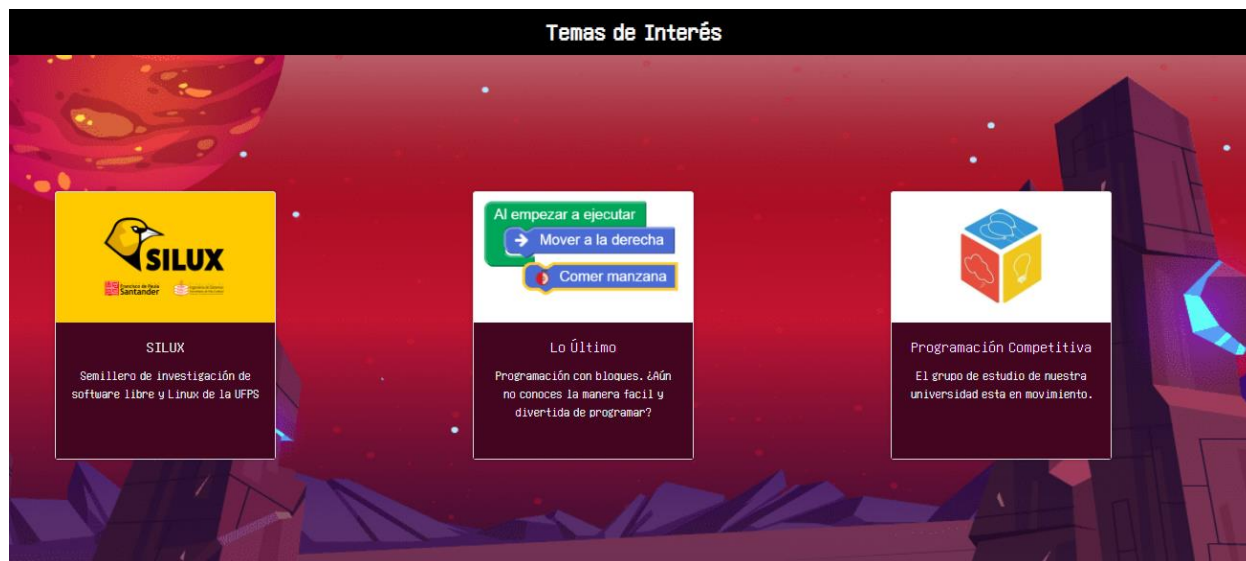
- Tour: Por medio de esta opción podrás hacer un recorrido por todo lo nuevo que se ha integrado en cada vista.



En la parte de abajo encontrarás la sección “Novedades”, en la cual podrás ver todo lo nuevo relacionado a la programación competitiva.

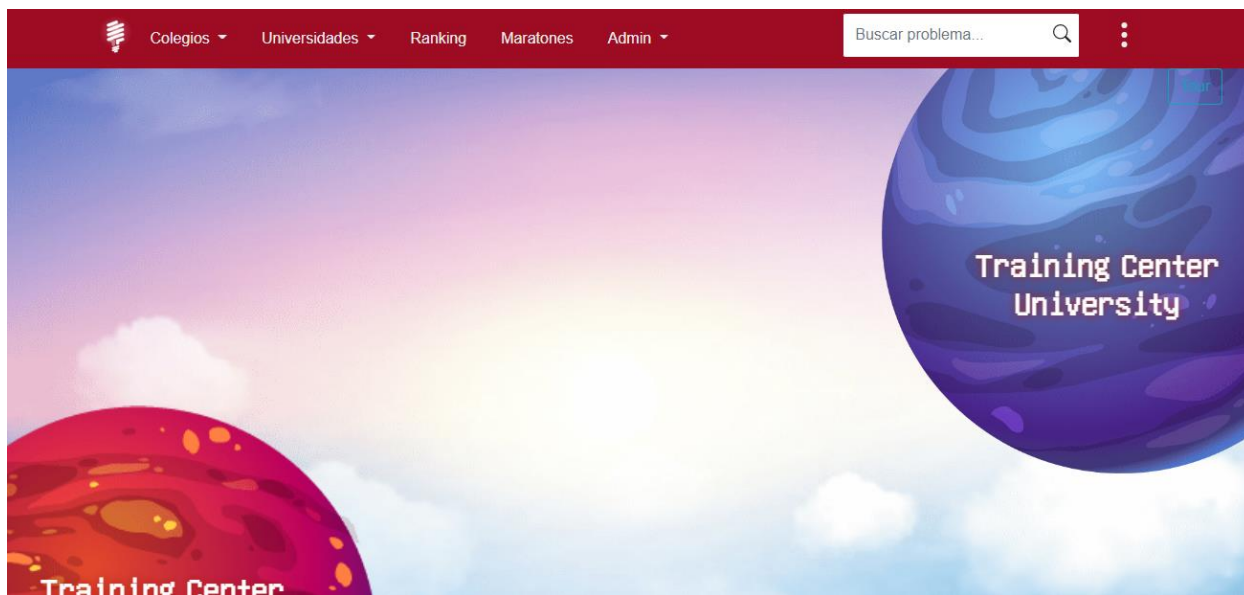


Al seguir bajando encontrarás la sección “Temas de interés” donde podrás visualizar temas relevantes.



3. Principal

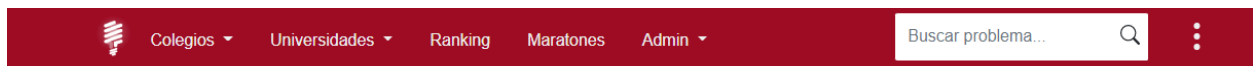
Al iniciar sesión te encontrarás con nuestra página “principal”



Aquí podrás elegir entre las 2 opciones (mundos) “Training Center High School” el cual es nuestro nuevo módulo enfocado para estudiantes de colegio, o “Training Center University” el cuál conserva lo que tenía la plataforma en su versión inicial más algunos ajustes y nuevas funcionalidades.

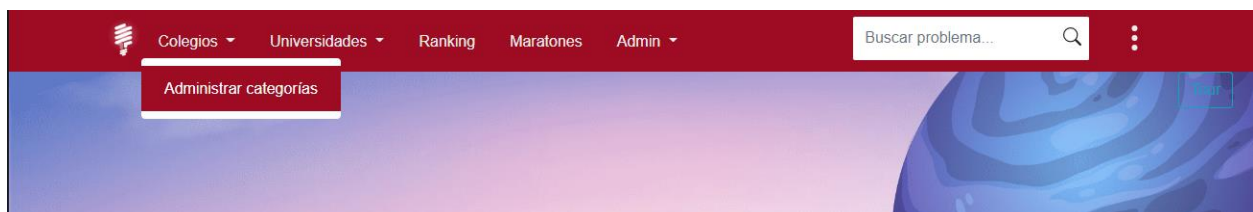
4. Acceso a las opciones

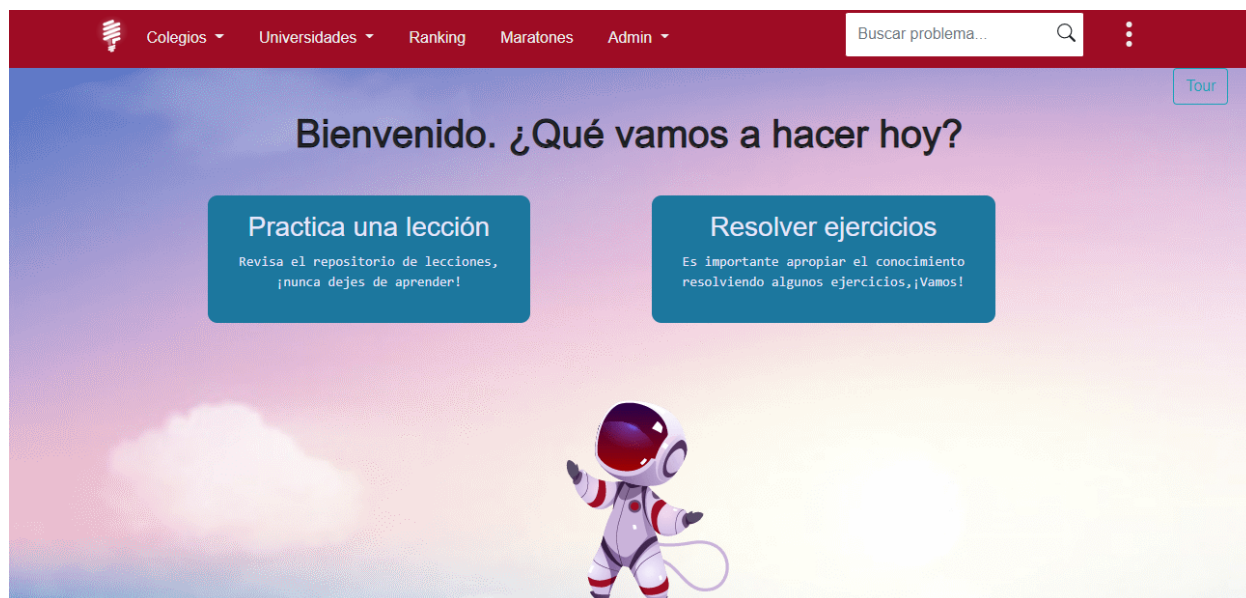
En la parte superior encuentras el menú de navegación que te permite moverte por los diferentes módulos y opciones disponibles.



- Colegios, donde podrás acceder a todas las funcionalidades del módulo colegios
- Universidades, donde podrás acceder a todas las funcionalidades del módulo Universidades
- Ranking, para comparar tus resultados en la institución con respecto a los demás usuarios.
- Maratones, donde encuentras las competencias de maratones de programación en las que puedes participar o también puedes crear una competencia de programación.
- Admin, despliega las opciones que permiten gestionar administración y añadir problemas.

5. Training Center High School (Colegios)





Aquí encontrarás 2 opciones:

Practica una lección: Dónde encontrarás todo el material disponible y como administrador no sólo puedes visualizar esta información, sino que cuentas también con la opción de crear nuevas categorías, editar las existentes (a través del lápiz mostrado en la esquina superior izquierda de cada una de ellas) o eliminarlas (a través de la X en la esquina superior derecha).

The screenshot shows the main dashboard of the Training Center High School materials management system. At the top, there is a dark red navigation bar with a logo on the left and menu items: "Colegios", "Universidades", "Ranking", "Maratones", and "Admin". A search bar labeled "Buscar problema..." is on the right. Below the navigation bar, the page title "Materiales Training Center High School" is displayed, followed by a "Tour" button. The main content area features three large, rounded rectangular buttons: a blue button labeled "Introducción Colegios", a red button labeled "Pruebas", and a grey button with a plus sign labeled "Nueva categoría".

Cada categoría contiene el material relacionado. Además de ver el material, también tienes la opción de añadir material y eliminar material.

The screenshot shows the "Introducción Colegios" category page. The navigation bar is identical to the previous screenshot. Below it, the breadcrumb "Categorías / Materiales de 'Introducción Colegios'" is visible, along with a "Tour" button. The main heading is "Introducción Colegios". Below the heading, there are two dropdown menus: "Mostrar 7 materiales" and "Ordenar por Id en forma Ascendente". The main content area contains two buttons: a green button labeled "Entradas y salidas" and a grey button with a plus sign labeled "Subir material". At the bottom, there is a pagination bar with buttons for "Inicio", "Anterior", "1" (highlighted), "Siguiete", and "Final".

Colegios ▾ Universidades ▾ Ranking Maratones Admin ▾

Buscar problema...

Categorías / Materiales de "Introducción Colegios" / Entradas y salidas

Entradas y salidas

Este material te servirá para conocer que papel cumplen las entradas y salidas de los problemas y como puedes manejarlas por medio de blockly.

Entradas

1 / 2 | - 100% +

ENTRADAS

Las entradas son los datos que nos proporcionan con los cuales debemos generar una salida para solucionar nuestro problema. Pero ¿Cómo podemos leer esas entradas?.

Los siguientes bloques nos ayudarán a leer nuestras entradas para poder crear nuestra solución con ellas.

Resolver ejercicios: Aquí podrás ver las diferentes categorías que hacen parte de la plataforma, y por cada una de ellas los problemas.

Colegios ▾ Universidades ▾ Ranking Maratones Admin ▾

Buscar problema...

Problemas Training Center High School

Categorías | Lista completa

Introducción Colegios

Pruebas

Categorías / Problemas de "Introducción Colegios" Tour

Introducción Colegios

Mostrar 10 problemas en Cualquier idioma Ordenar por Id en forma Ascendente

Id	Problema	Dificultad	Idioma
1001	Hola mundo	1	ES EN
1002	Hola Mundo Reloaded	1	ES EN
1003	Multiplicando por 11	1	ES EN
1004	Cociente y Residuo	1	ES EN
1005	Doble de un número	1	ES EN

Cuando abres un problema, verás la siguiente interfaz:

Categorías / Problemas en esta categoría / Problema "Hola mundo" Tour

Hola mundo

+Entradas y salidas
+ciclos
+condicionales

ES | EN

Descripción

Se llama "Hola mundo" a un programa muy simple, cuya única función es mostrar en pantalla el texto "Hola mundo". Por su simpleza, cientos de profesores de programación en todo el mundo utilizan este ejemplo como la primera aproximación de sus estudiantes a la programación.

En este problema justamente haremos un "Hola mundo". Pero no lo hagamos tan simple. En vez de saludar siempre al mundo, saludemos a un listado de personas específicas. ¿Estas listo? ¡Inténtalo!

Entradas

La primera línea de la entrada contiene un número N ($1 \leq N \leq 100$), representando el número de personas que debe saludar. Las siguientes N líneas de la entrada continúan apareciendo N líneas, cada una con el nombre de una persona a saludar. Todos los nombres...

Problema: Hola mundo

Dificultad: 1

Editar problema

Enviar

saludar tu programa. A continuación aparecen N líneas, cada una con el nombre de una persona a saludar. Todos los nombres están en minúscula, conteniendo como máximo 10 caracteres entre la 'a' y la 'z', sin espacios.

Salidas

Por cada nombre de la entrada debes imprimir una línea saludando a esa persona. Si por ejemplo, la persona se llama "alfred", tu programa debe imprimir "Hola alfred". "Hola" siempre debe iniciar con H mayúscula, y debe haber un espacio entre "Hola" y el nombre. El nombre debe estar en minúscula, tal como se leyó desde la entrada. Para más detalles, mira las salidas de ejemplo.

Entrada de ejemplo

```
3
alfred
erick
andres
```

Salida de ejemplo

```
Hola alfred
Hola erick
Hola andres
```

Ahora, intenta resolver este ejercicio

[Ayuda](#)

0 Comentarios

ANGIE MADELEY...

Comienza la discusión...

ordenar por el mejor

Sé el primero en comentar.

Desde aquí puedes ver la descripción del problema, la definición de las entradas y las salidas, así como ejemplos de entrada y salida. Si el problema está disponible en múltiples idiomas, puedes cambiarlo haciendo clic en el idioma respectivo justo debajo del título.

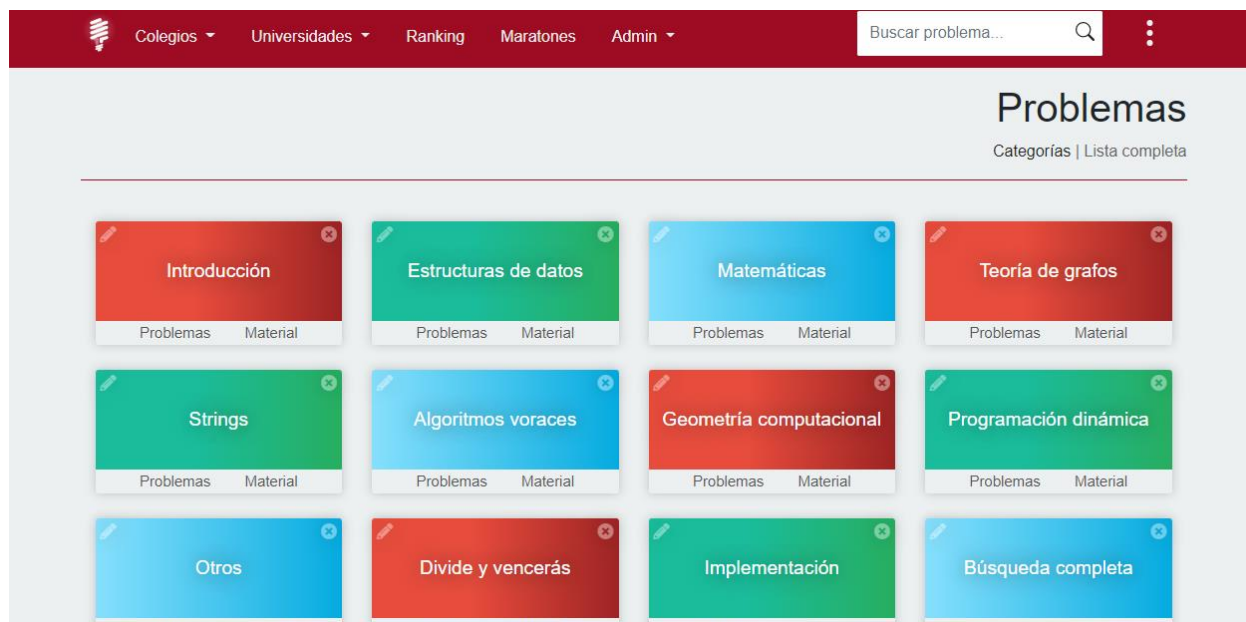
Como administrador al entrar al problema encontrarás la opción de editar problema.

Como administrador no puedes enviar soluciones de problemas, por lo que al usar la opción enviar muestra el siguiente mensaje:

The screenshot shows a web application interface for a programming problem. At the top, there is a dark red navigation bar with a logo on the left and menu items: 'Colegios', 'Universidades', 'Ranking', 'Maratones', and 'Admin'. A search bar on the right contains the text 'Buscar problema...'. Below the navigation bar, a breadcrumb trail reads 'Categorías / Problemas en esta categoría / Problema "Hola mundo"'. A red error message box is displayed, stating 'Usted no tiene permisos para realizar esta acción'. To the right of the error message is a 'Tour' button. The main content area is divided into two columns. The left column features the title 'Hola mundo' in large bold letters, followed by three red buttons: '+Entradas y salidas', '+ciclos', and '+condicionales'. Below these is the text 'ES | EN'. The 'Descripción' section explains that 'Hola mundo' is a simple program that displays the text 'Hola mundo'. It notes that many professors use this as a first example for students. The text continues: 'En este problema justamente haremos un "Hola mundo". Pero no lo hagamos tan simple. En vez de saludar siempre al mundo, saludemos a un listado de personas específicas. ¿Estas listo? ¡Inténtalo!'. The 'Entradas' section states: 'La primera línea de la entrada contiene un número N (1 <= N <= 100), representando el número de personas que debe'. The right column contains a white box with the text 'Problema: Hola mundo' and 'Dificultad: 1'. Below this are two red buttons: 'Editar problema' and 'Enviar'.

Los administradores no tienen acceso a las clases, ya que son una herramienta de comunicación exclusiva entre docentes y estudiantes.

6. Training Center University



En la sección problemas, podrás ver las diferentes categorías que hacen parte de la plataforma, y por cada una de ellas los problemas y el material disponible (Este material es el mismo que puedes ver desde la sección pública sin iniciar sesión).

Como administrador no sólo puedes visualizar esta información, sino que cuentas también con la opción de crear nuevas categorías, editar las existentes (a través del lápiz mostrado en la esquina superior izquierda de cada una de ellas) o eliminarlas (a través de la X en la esquina superior derecha).

Problemas

Aquí puedes ver, editar y elimina los problemas por cada categoría

Categorías / Problemas de "Introducción"

Introducción

Mostrar 10 problemas en Cualquier idioma Ordenar por Id en forma Ascendente

Id	Problema	Dificultad	Idioma
0	Hola mundo	1	ES EN
2	Partido amistoso	1	ES EN
25	Estación de buses	1	ES EN
26	Conectores	1	ES EN
30	Torre de CDs	1	ES EN
36	Operadores relacionales	1	ES EN

Cuando abres un problema, verás la siguiente interfaz:

Categorías / Problemas en esta categoría / Problema "Hola mundo"

Hola mundo

ES | EN

Descripción

Se llama "Hola mundo" a un programa muy simple, cuya única función es mostrar en pantalla el texto "Hola mundo". Por su simpleza, cientos de profesores de programación en todo el mundo utilizan este ejemplo como la primera aproximación de sus estudiantes a la programación.

En este problema justamente haremos un "Hola mundo". Pero no lo hagamos tan simple. En vez de saludar siempre al mundo, saludemos a un listado de personas específicas. ¿Estas listo? ¡Inténtalo!

Entradas

La primera línea de la entrada contiene un número N ($1 \leq N \leq 100$), representando el número de personas que debe saludar tu programa. A continuación aparecen N líneas, cada una con el nombre de una persona a saludar. Todos los nombres están en minúscula, conteniendo como máximo 10 caracteres entre la 'a' y la 'z', sin espacios.

Problema: Hola mundo

Dificultad: 1

[Editar problema](#)

Salidas

Por cada nombre de la entrada debes imprimir una línea saludando a esa persona. Si por ejemplo, la persona se llama "alfred", tu programa debe imprimir "Hola alfred". "Hola" siempre debe iniciar con H mayúscula, y debe haber un espacio entre "Hola" y el nombre. El nombre debe estar en minúscula, tal como se leyó desde la entrada. Para más detalles, mira las salidas de ejemplo.

Entrada de ejemplo

```
3
alfred
erick
andres
```

Salida de ejemplo

```
Hola alfred
Hola erick
Hola andres
```

1 Comentario

ANGIE MADELEY...



Únete a la discusión...

ordenar por el mejor  

Una de tus tareas principales es corregir problemas con errores. Los coaches tienen permiso para añadir problemas, pero no pueden editar los problemas de los demás. En cambio los administradores pueden editar cualquier problema para corregir errores en la redacción, los ejemplos o las entradas.

Idioma: Español

Nombre: Hola mundo

Dificultad: 1

Tipo de institución: Universidades

Categoría: Introducción

Tiempo límite: 1

Contenido

Descripción

Se llama "Hola mundo" a un programa muy simple, cuya única función es mostrar en pantalla el texto "Hola mundo". Por su simpleza, cientos de profesores de programación en todo el mundo utilizan este ejemplo como la primera aproximación de sus estudiantes a la programación.

En este problema justamente haremos un "Hola mundo". Pero no lo hagamos tan simple. En vez de saludar siempre al mundo, saludemos a un listado de personas específicas. ¿Estas listo? ¡Inténtalo!

Entradas

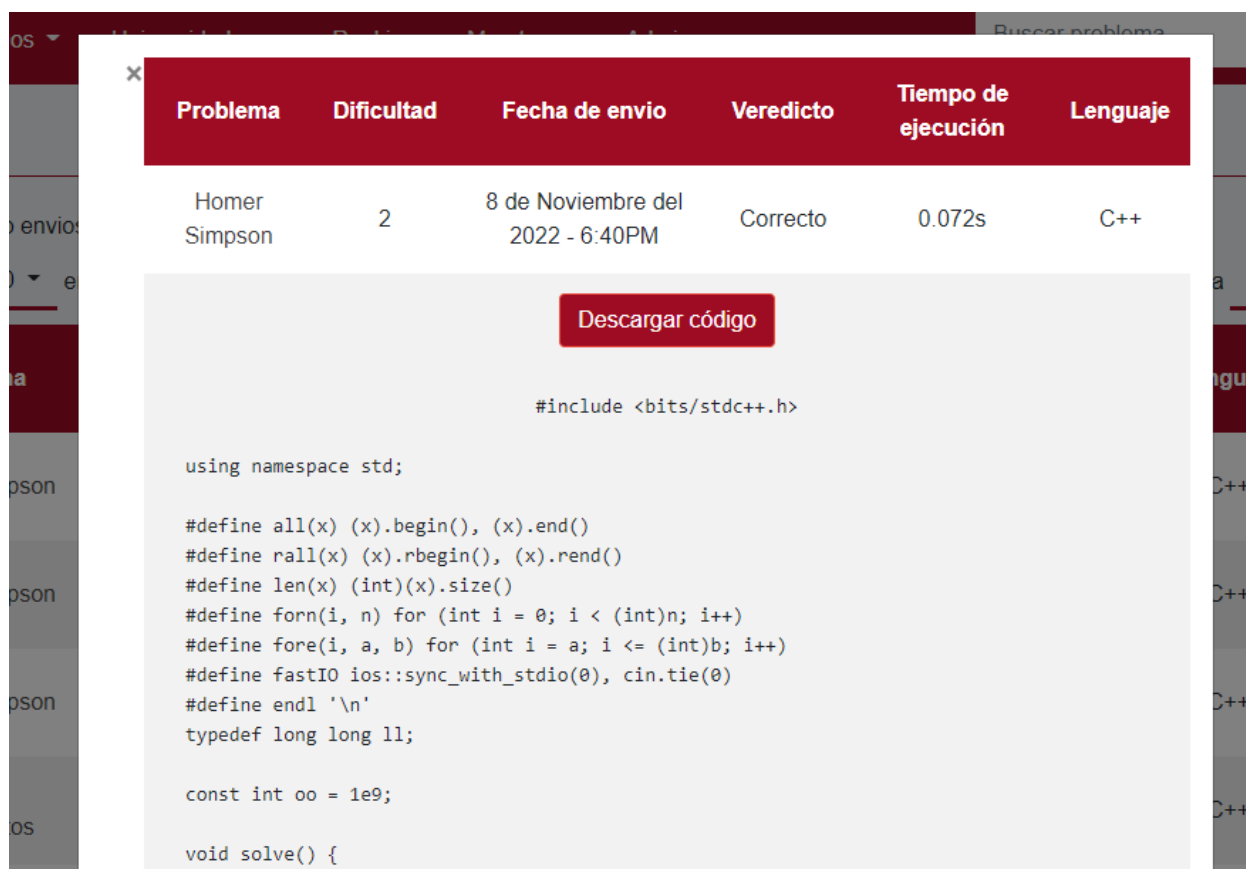
Se espera de buena fe que los estudiantes que se ofrezcan como coach no realicen ningún tipo de acciones vandálicas con los problemas. Sin embargo, si detectas problemas groseros, inconclusos o con fallas premeditadas, puedes eliminarlo inmediatamente.

7. Ranking

Clasificación general						
Puesto	Usuario	Institución	Inst. Tipo	Soluciones correctas	Envíos realizados	Ver envíos
1	Edinsson Melo (Mackenzie)	UFPS	Universidad	265	976	Ver
2	Marco (MarcoReus)	UFPS	Universidad	248	510	Ver
3	Sebastian Vega (C bas)	UFPS	Universidad	244	606	Ver
4	Crisel Ayala :D (Tarde pero seguro prros :v)	UFPS	Universidad	224	632	Ver
5	Coach Grupo (coachGrupo)	UFPS	Universidad	221	453	Ver

Aquí puedes encontrar la clasificación general con respecto a todos los usuarios de la plataforma y sus soluciones enviadas, para así comparar sus resultados. Esta información la puedes filtrar según tu interés; por tipo de institución, año o categoría. Como administrador puedes ver los envíos de los usuarios y supervisar sus códigos.

Envíos						
Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje	Código
Homer Simpson	2	8 de Noviembre del 2022 - 6:40PM	Correcto	0.072s	C++	Ver código
Homer Simpson	2	8 de Noviembre del 2022 - 6:40PM	Respuesta equivocada	0.075s	C++	Ver código
Homer Simpson	2	8 de Noviembre del 2022 - 6:39PM	Respuesta equivocada	0.004s	C++	Ver código
Maxi Descuentos	1	13 de Noviembre del 2020 - 5:08PM	Respuesta equivocada	0.008s	C++	Ver código



The screenshot shows a submission result for a problem named 'Homer Simpson'. The submission was made on November 8, 2022, at 6:40 PM, and was marked as 'Correcto' (Correct). The execution time was 0.072s and the language used was C++. Below the result table is a 'Descargar código' (Download code) button and a code editor showing the C++ code used for the submission.

Problema	Dificultad	Fecha de envío	Veredicto	Tiempo de ejecución	Lenguaje
Homer Simpson	2	8 de Noviembre del 2022 - 6:40PM	Correcto	0.072s	C++

[Descargar código](#)

```
#include <bits/stdc++.h>

using namespace std;


#define all(x) (x).begin(), (x).end()
#define rall(x) (x).rbegin(), (x).rend()
#define len(x) (int)(x).size()
#define forn(i, n) for (int i = 0; i < (int)n; i++)
#define fore(i, a, b) for (int i = a; i <= (int)b; i++)
#define fastIO ios::sync_with_stdio(0), cin.tie(0)
#define endl '\n'
typedef long long ll;



const int oo = 1e9;

void solve() {
```

8. Maratones

Puedes crear maratones de programación y compartirlas con los estudiantes.


[Colegios](#) ▾
 [Universidades](#) ▾
 [Ranking](#)
[Maratones](#)
[Admin](#) ▾







[Crear Maratón](#)

Mis maratones

Mostrar **10** ▾ maratones Ordenar por **Id** ▾ en forma **Ascendente** ▾


Id	Nombre	Visibilidad	Tipo	Fecha de inicio	Acción
38	Copa Programación Clasificatoria 1	Privada	Universidad	22 de Noviembre del 2018 - 7:30PM	Editar Ver
39	Copa Programación Clasificatoria 2	Privada	Universidad	23 de Noviembre del 2018 - 7:30PM	Editar Ver
40	Copa Programación Clasificatoria 3	Privada	Universidad	26 de Noviembre del 2018 - 7:30PM	Editar Ver
41	Copa Programación Clasificatoria 4	Privada	Universidad	27 de Noviembre del 2018 - 7:30PM	Editar Ver



[Colegios](#) ▾
 [Universidades](#) ▾
 [Clases](#)
[Ranking](#)
[Maratones](#)
[Admin](#) ▾






[Maratones](#) / Nueva maratón





Nueva Maratón





Nombre: 

Descripción: 

Reglas: 

Tipo: 

Desde el:  
a las:  

Hasta el:  
a las:  

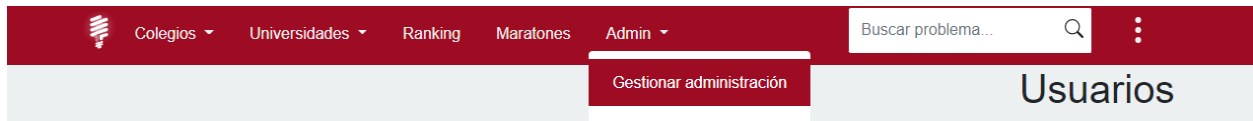
Esta maratón será:

Pública
 Privada

[Crear Maratón](#)

9. Administración

Gestionar administración:



Existen diferentes labores administrativas que puedes realizar. Por ejemplo, tanto docentes como administradores pueden añadir materiales a la plataforma. Los estudiantes por su parte pueden sugerir materiales, pero estos deben ser aprobados por un administrador. Así que puedes ver la lista de materiales sugeridos por los estudiantes, y aceptarlos o rechazarlos (Ten cuidado con los derechos de autor: Aún si un material es útil para el grupo no debe ser aprobado si ha sido obtenido de forma irregular o si quien lo comparte no tiene los derechos para compartirlo).

Colegios ▾
Universidades ▾
Ranking
Maratones
Admin ▾

🔍
⋮

Usuarios

Desde esta sección puede añadir docentes (coachs) o administradores. Los estudiantes pueden crear sus propias cuentas desde la página inicial.

Administrador Coach o docente

Nombre: ?

Email: ?

Contraseña: ?

Código: ?

Username: ?

Contraseña: ?

Crear usuario

Filtrar por institución: Seleccione ▾

Elegir : Seleccione ▾

Mostrar 10 ▾ usuarios Ordenar por Id ▾ en forma Ascendente ▾

Id	Nombre	Usuario	Código	Tipo	Email	Institución	Inst. Tipo	Acciones
1	Angie Melissa Delgado León	Meyito	1150990	Administrador	amelisd@gmail.com	UFPS		Revocar
2	Gerson Lázaro	GersonLazaro	1150972	Administrador	gersonyesidlc@ufps.edu.co	UFPS		Revocar
3	Gerson Lazaro	TinkyWinky	1150972	Estudiante	gersonlazaroc@gmail.com	UFPS		Revocar
4	Gerson Lázaro	GersonLazaroCC	1150972	Coach	gersonlazaro@outlook.com	UFPS		Revocar
5	Angie Melissa Delgado León	amdelgado	1150990	Estudiante	amelisd+1@gmail.com	UFPS		Revocar
6	Coach Grupo	coachGrupo	1150972	Coach	programacioncompetitivaufps@gmail.com	UFPS		Revocar

9	Manuel Salazar	manuel30	1151382	Estudiante	manuelsalazar1030@gmail.com	UFPS	Revocar
10	Johan Hernandez	hernandezjohan27	1151713	Estudiante	johansaacha@ufps.edu.co	UFPS	Revocar

[Inicio](#)
[Anterior](#)
[1](#)
[2](#)
[3](#)
[4](#)
[5](#)
[Siguinte](#)
[Final](#)

Materiales

Los siguientes materiales han sido añadidos por estudiantes. Para que aparezcan en la plataforma, deben ser aprobados por un administrador.

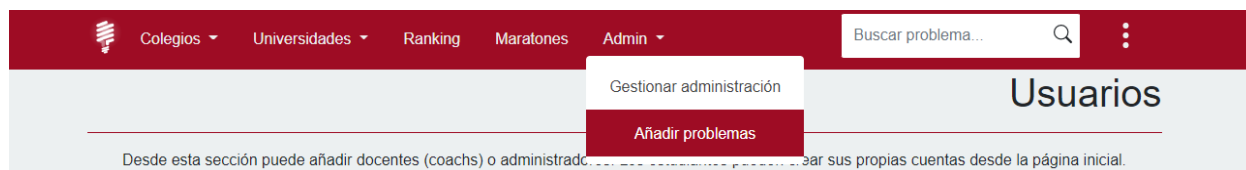
Mostrar 10 materiales Ordenar por Id en forma Ascendente

Id	Nombre	Categoría	Acción
130	Entradas y salidas	Introducción Colegios	Abrir Aprobar Descartar

Del mismo modo un administrador puede añadir nuevas cuentas a docentes o coaches cada vez que alguien la solicite, o cuentas de administrador a usuarios que la soliciten. Para crear cuentas de administrador dejamos a tu criterio la decisión de aceptar o no las solicitudes.

La plataforma no está exenta de acciones no deseadas: Usuarios que envían códigos con comportamientos abusivos, o que comparten materiales que no tienen cabida en la plataforma. Como administrador puedes dar de baja a estos usuarios, pero se cuidadoso. Antes de eliminarlo, asegurate que en realidad el usuario esté realizando acciones vandálicas, y no sean simples errores (En la programación es posible crear programas con comportamientos indeseados por error o desconocimiento).

Añadir problemas:



Como administrador puedes añadir nuevos problemas (pulsando en la barra superior “Admin”, y seleccionando en el menú desplegable “Añadir problemas”), así como eliminar o editar los problemas existentes.

10. Recomendaciones

Utiliza la plataforma, explora sus opciones a fondo, ayúdanos a cargar material útil y ejercicios retadores, y entre todos construyamos un UFPS Training Center que explote al máximo las capacidades de los estudiantes UFPS.

Anexo 4. Documento Mockups

Documento de Mock-up's de Requerimientos.

Plataforma de software: Training Center UFPS

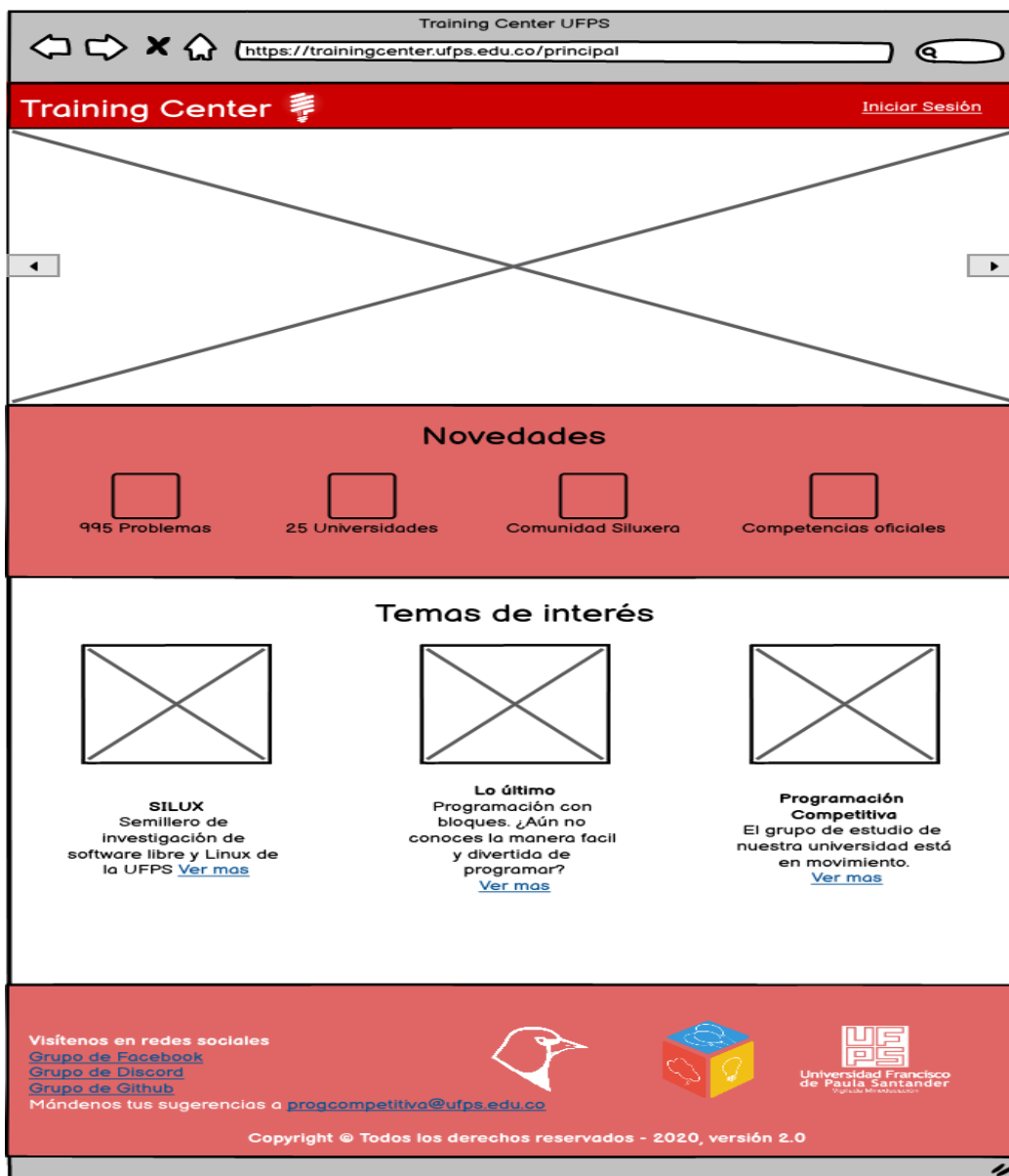
Proyecto: Training Center High School

V. 2.0.0

1. Introducción

El presente documento presenta los mockups iniciales realizados para el presente proyecto. Los cuales se usaron como base para la realización de las vistas necesarias para la extensión de la plataforma.

Home



Principal

The image shows a browser window with the title "Training Center UFPS - página principal". The address bar contains the URL "https://trainingcenterufps.edu.co/principañ". The website has a red header with a logo on the left, navigation links for "Problemas", "Clases", and "Maratones", a search bar labeled "Buscar problema", and a user profile icon. The main content area is split into two columns: the left column says "Ir a Training Center High School" and features a cartoon owl wearing a graduation cap and holding a diploma; the right column says "Ir a Training Center University". A red footer contains the text "Copyright © Todos los derechos reservados - 2020, versión 2.0".

Training Center UFPS - página principal

https://trainingcenterufps.edu.co/principañ

Problemas | Clases | Maratones

Buscar problema

Ir a Training Center High School

Ir a Training Center University

Copyright © Todos los derechos reservados - 2020, versión 2.0

Home Training Center High School

Training Center High School - Principal

https://trainingcenterufps.edu.co/high_school

Problemas | Clases | Maratones

Buscar problema

Bienvenido. ¿Que vamos a hacer hoy?

Practicar una lección

Revise el repositorio de lecciones,
¡Nunca dejes de aprender!

Resolver problemas

Es importante apropiar el conocimiento
resolviendo algunos ejercicios, ¡vamos!

Alguna frase aleatoria que resalte la
importancia de aprender programación desde
temprana edad
Fuentes: alguna fuente confiable



Copyright © Todos los derechos reservados - 2020, versión 2.0

Problemas

Training Center High School - Problemas

← → × 🏠

[Problemas](#) |
 [Clases](#) |
 [Maratones](#)

Repositorio de problemas

X

X

X

Entradas y salidas

Arrays

Sentencias lógicas

Buscar por etiquetas: recursividad +entradas y salidas +condicionales +arrays

Id	Problema	Etiquetas	Estado
508	Hola mundo	entradas y salidas - condicionales - arrays	¡Ya resolviste el problema!
509	El dilema de Pinball	Recursividad	Aún puedes resolver este problema

Copyright © Todos los derechos reservados - 2020, versión 2.0

Training Center High School - Hola mundo

← → × 🏠

📖 🎓 Problemas | Clases | Maratones

😊

Problemas > problemas de esta categoría > problema "Hola mundo"

Hola mundo

ES | EN

+entradas y salidas
+condicionales
+arrays

Descripción
A por ello!
Mis envíos
Foro

Soluciona el ejercicio!

Tiempo max de procesamiento: 1 segundo

Veamos como se soluciona

Variables

Sentencias lógicas

Ciclos

Operadores matemáticos

Listas

Manejo de texto

Funciones

```

set Count to 1
repeat while Count ≤ 3
do
  print "Hello World!"
  set Count to Count + 1

```

Ahora intentenmos probar tu solución

Crea tu caso de entrada

```

3
alfred
erick
andres

```

Mostrar salida generada

```

Hola alfred
Hola erick
Hola andres

```

Enviar solución. Estoy segur@ de que todo va correctamente

Copyright © Todos los derechos reservados - 2020, versión 2.0

Foro

Training Center High School - Hola mundo

← → ✕ 🏠

Problemas | Clases | Maratones

Problemas > problemas de esta categoría > problema "Hola mundo"

Hola mundo

ES | EN

+entradas y salidas
+condicionales
+arrays

Descripción
A por ello!
Mis envíos
Foro

Foro

Déjanos tu pregunta acá!

B I U
style
🔗
🔗
🔗
🔗
🔗
🔗
🔗
🔗

Pepito Perez ha preguntado:

15 de abril del 2019, 2:05 pm

Lorem ipsum dolor sit amet consectetur adipiscing elit turpis ac viverra porta, netus vivamus nostra quam purus velit integer metus ad. Nam iaculis facilisis habitant id nulla felis volutpat ultrices diam, purus condimentum malesuada magna magnis ridiculus eros cras, cubilia nec bibendum justo augue semper inceptos quis. Purus ante mollis netus elementum platea risus sollicitudin per justo nullam, primis ultricies vivamus fames imperdiet tortor tempus quis arcu faucibus, nam id lobortis blandit scelerisque parturient duis natoque cras.

Camilito Rojas responde:

15 de abril del 2019, 9:58 pm

Lorem ipsum dolor sit amet consectetur adipiscing elit turpis ac viverra porta, netus vivamus nostra quam purus velit integer metus ad.

+5

Pepito Perez responde:

15 de abril del 2019, 9:58 pm

Lorem ipsum dolor sit amet consectetur adipiscing elit turpis ac viverra porta, netus vivamus nostra quam purus velit integer metus ad.

+1

Deja tu respuesta aquí:

Lorem ipsum dolor sit amet consectetur adipiscing elit turpis ac viverra porta, netus vivamus nostra quam purus velit integer metus ad.

Enviar

Esta pregunta ha sido resuelta, por lo tanto ya no se reciben más respuestas.

Copyright © Todos los derechos reservados - 2020, versión 2.0

Categorías de Material

Training Center High School - Lecciones

https://trainingcenterufps.edu.co/high_school/lecciones

Problemas | Clases | Maratones

Buscar problema

Repositorio de lecciones

Un poco de historia	Entradas y salidas	Sentencias lógicas
Tipos de variables	Ciclos	Manejo de arrays

Copyright © Todos los derechos reservados - 2020, versión 2.0

Ver material

The image shows a browser window with the following elements:

- Address Bar:** Training Center High School - Lección "Entradas y salidas"
https://trainingcenterufps.edu.co/high_school/lecciones/14
- Navigation Bar (Red):**
 - Icons for home, back, forward, and search.
 - Menu items: [Problemas](#) | [Clases](#) | [Maratones](#)
 - Search box:
 - User profile icon.
- Section Header:**

Entradas y Salidas
- Content Area:** Three placeholder boxes, each with a diagonal 'X' and a label below:
 - [Entradas y salidas - lectura](#)
 - [Slides - entradas y salidas](#)
 - [Video - entradas y salidas](#)
- Footer (Red):** Copyright © Todos los derechos reservados - 2020, versión 2.0

Anexo 5. Especificación de requerimientos

Especificación de Requerimientos del Software

Plataforma de software: Training Center UFPS

Proyecto: Training Center High School

Versión 2.0.0

Especificación de requerimientos

1. Introducción

El presente documento presenta los requerimientos iniciales en la construcción de la extensión de la plataforma de software Training Center UFPS, la cual hace parte del proyecto de grado “Training Center High School”.

1. Alcance

El presente documento cubre únicamente los aspectos relacionados con la especificación de los requerimientos. Como tal, el documento es parte de una documentación más completa que cubre las demás fases de desarrollo, y se distribuyen de manera digital como un anexo al informe final del proyecto.

2. Definiciones, acrónimos y abreviaturas

- ICPC: International Collegiate Programming Contest, Maratón de Programación Oficial Universitaria más importante a nivel mundial.
- Maratón de Programación: Competencia donde se ponen en prueba las habilidades algorítmicas y de trabajo en equipo de los estudiantes en un entorno controlado.
- UFPS Training Center: Plataforma de entrenamiento en Programación Competitiva para los estudiantes de universidades y colegios.
- Training Center University (TCU): Módulo de la plataforma Training Center enfocado al entrenamiento en Programación Competitiva para estudiantes de universidad.
- Training Center High School (TCHS): Nuevo módulo de la plataforma Training Center enfocado al entrenamiento en Programación Competitiva para estudiantes de colegios.

2. Listado de requerimientos funcionales

Id.	Nombre
RF-01	Implementar Blockly
RF-02	Listar Problemas
RF-03	Agregar problemas
RF-04	Corregir problemas
RF-05	Eliminar problemas
RF-06	Acceder al foro de discusión (por problema)
RF-07	Filtrar problemas
RF-08	Categorizar problemas
RF-09	Ver ranking
RF-10	Crear competencias
RF-11	Añadir material de ayuda

RF-12	Ver material de ayuda
RF-13	Crear Modo guiado (Tour)
RF-14	Resolver tareas de las clases
RF-15	Recepcionar solución de tareas
RF-16	Eliminar usuarios
RF-17	Calificar problemas (Bloques)
RF-18	Mostrar categorías, problemas y clases separados por TCU y TCHS
RF-19	Inspeccionar códigos en competencias (como usuario administrador)

3. Especificación de requerimientos funcionales

ID del Requerimiento:	<i>RF-01</i>
Nombre del Requerimiento:	Implementar Blockly

<i>El sistema debe permitir la creación de solución de problemas por medio de lenguaje de bloques (Blockly) en Training Center High School.</i>	
<i>Prioridad: Muy alta</i>	

ID del Requerimiento:	<i>RF-02</i>
Nombre del Requerimiento:	Listar problemas
<i>El sistema debe desplegar al usuario la lista de problemas almacenados en la plataforma con la posibilidad de filtrar bajo diferentes criterios.</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-03</i>
Nombre del Requerimiento:	Agregar problemas
<i>El sistema debe permitir a los docentes y administradores añadir nuevos problemas para que los estudiantes los solucionen en lenguajes de programación (Java, C++, Python, Blockly).</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-04</i>
------------------------------	--------------

Nombre del Requerimiento:	Corregir problemas
<i>Los administradores tienen la potestad de corregir problemas si tienen errores en su formato, estructura o entradas y salidas.</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-05</i>
Nombre del Requerimiento:	Eliminar problemas
<i>El administrador puede borrar problemas de la plataforma si no cumple con las características que debería contar un problema de programación.</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-06</i>
Nombre del Requerimiento:	Acceder al foro de discusión
<i>El sistema debe permitir las discusiones de los estudiantes en cada problema.</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-07</i>
Nombre del Requerimiento:	Filtrar problemas
<i>El sistema debe permitir que los estudiantes filtren los problemas por nombre para módulo colegios y módulo universidad dependiendo del módulo en que se encuentre.</i>	
<i>Prioridad: Baja</i>	

ID del Requerimiento:	<i>RF-08</i>
Nombre del Requerimiento:	Categorizar problemas
<i>Los problemas en Training Center pueden agruparse bajo diferentes categorías globales dependiendo el tipo de categoría (colegio-universidad).</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-09</i>
Nombre del Requerimiento:	Ver ranking
<i>El sistema debe generar ranking de los usuarios según el número de problemas resueltos.</i>	

<i>Prioridad: Bajo</i>

ID del Requerimiento:	<i>RF-10</i>
Nombre del Requerimiento:	Crear competencias
<i>El sistema debe permitir la creación de competencias (maratones) de programación en vivo.</i>	
<i>Prioridad: Alto</i>	

ID del Requerimiento:	<i>RF-11</i>
Nombre del Requerimiento:	Añadir material de ayuda
<i>El sistema debe permitir que los usuarios agreguen material de ayuda (PDF, videos).</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-12</i>
Nombre del Requerimiento:	Ver material de ayuda

<i>Los estudiantes pueden ver el material de ayuda para mejorar su entrenamiento.</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-13</i>
Nombre del Requerimiento:	Crear Modo guiado (Tour)
<i>El sistema debe permitir a los usuarios de Training Center acceder a un tour que permite el conocimiento de las nuevas funcionalidades de la plataforma.</i>	
<i>Prioridad: Bajo</i>	

ID del Requerimiento:	<i>RF-14</i>
Nombre del Requerimiento:	Resolver tareas de las clases
<i>El sistema debe permitir a los coach/docentes crear tareas (clases o syllabus) para que sus estudiantes resuelvan los ejercicios que proponen y ver sus materiales.</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-15</i>
Nombre del Requerimiento:	Recepcionar solución de tareas
<i>El sistema debe permitir la recepción de las tareas enviadas por los estudiantes que previamente fueron propuestas por el coach/docente en la plataforma.</i>	
<i>Prioridad: Media</i>	

ID del Requerimiento:	<i>RF-16</i>
Nombre del Requerimiento:	Eliminar usuarios
<i>El sistema debe permitir al administrador eliminar usuarios no deseados.</i>	
<i>Prioridad: Bajo</i>	

ID del Requerimiento:	<i>RF-17</i>
Nombre del Requerimiento:	Calificar envíos
<i>El sistema debe calificar automáticamente las soluciones enviadas por los estudiantes, a los diferentes problemas e indicarles su calificación.</i>	
<i>Prioridad: Muy alta</i>	

ID del Requerimiento:	RF-18
Nombre del Requerimiento:	Mostrar categorías, problemas y clases separados por TCHS y TCU
<i>El sistema debe mostrarle al usuario las categorías, problemas y clases separadas dependiendo si está en módulo colegios o módulo universidades.</i>	
<i>Prioridad: Alta</i>	

ID del Requerimiento:	RF-19
Nombre del Requerimiento:	Inspeccionar códigos en competencias (como usuario administrador)
<i>El sistema debe permitir a un administrador inspeccionar códigos en competencias en vivo, para evitar posibles plagios.</i>	
<i>Prioridad: Media</i>	

4. Requerimientos no funcionales

Id	Nombre	Tipo	Descripción
RNF-01	El sistema debe mantenerse estable aún en	Eficiencia	Durante una sesión de entrenamiento o una competencia hay en promedio 30 estudiantes. En una clase práctica puede haber hasta 50

	momentos de afluencia de público.		estudiantes. El sistema debe mantenerse estable aún si todos estos usuarios están conectados al tiempo.
RNF-02	Los envíos de soluciones no deben afectar la estabilidad del sistema.	Eficiencia	La calificación de una solución enviada por un estudiante implica su ejecución. Algunos problemas no tienen soluciones de complejidad algorítmica baja, y la ejecución de cada solución puede tardar varios segundos. Aún si la calificación presenta congestión por esta razón, el resto de la plataforma debe mantenerse estable y rápida.
RNF-03	Las soluciones enviadas por los usuarios deben ejecutarse en modo “sandbox”.	Seguridad	Los estudiantes envían soluciones en forma de código de programación que se califican automáticamente en la plataforma. Estos códigos deben ejecutarse en un modo “sandbox” que impida que un código malicioso realice daños en la plataforma o demás datos alojados en el servidor.
RNF-04	El tiempo de aprendizaje de un usuario con la	Usabilidad	La interfaz gráfica de la plataforma debe ser lo suficientemente simple para que los usuarios logren entenderla y utilizarla

	plataforma debe ser menor a 20 minutos.		completamente en, como máximo, 20 minutos.
RNF-05	Manuales de usuario	Usabilidad	La plataforma debe contar con manuales de usuario debidamente documentados y claros.
RNF-06	El sistema debe funcionar en cualquier plataforma moderna.	Usabilidad/Producto	La plataforma debe funcionar de forma similar en sistemas Windows/Linux/Mac OS. Esta debe operar a través de cualquier navegador web moderno (Google Chrome, Mozilla Firefox, Internet Explorer 10+, Microsoft Edge, Safari, Opera, Vivaldi, Maxthon o Brave Browser). En cualquier navegador no mencionado en esta lista la aplicación podría funcionar si utiliza tecnologías de HTML/CSS/Javascript modernas.

Anexo 6. Material de apoyo Blockly

ENTRADAS

Las entradas son los datos que nos proporcionan con los cuales debemos generar una salida para solucionar nuestro problema. Pero ¿Cómo podemos leer esas entradas?

Los siguientes bloques nos ayudarán a leer nuestras entradas para poder crear nuestra solución con ellas.

Puedes solicitar un texto o números, depende de lo que te indiquen en las entradas del problema.

Para el tema de recibir entradas y asignarle valores te servirá mucho la sección "Variables"



Quando el problema te indique lo siguiente:

La primera línea contiene un número entero N ($1 \leq N \leq 100$), la cantidad de casos de prueba.

Debes hacer la lectura de la cantidad de casos de prueba. Para ellos debes ir a variables y usar el siguiente bloque:



Luego por medio de un bloque de ciclo repites la acción a realizar que resuelve tu problema, dependiendo del número de casos de prueba que te hayan dado en la entrada.



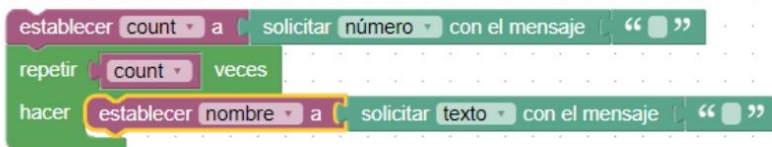
SALIDAS

Las salidas son las respuestas esperadas para nuestro problema. Para mostrar nuestras salidas contamos con el bloque "imprimir"



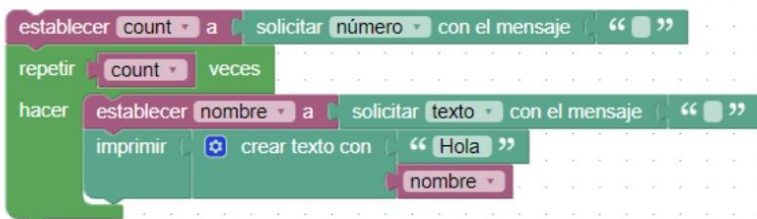
EJEMPLO

Vamos a leer las entradas para nuestro ejercicio "Hola mundo"



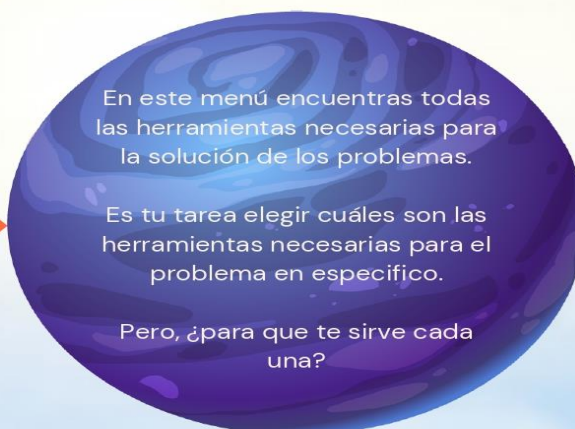
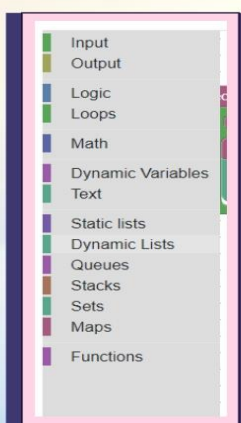
Establecemos un count para el número de casos de prueba. Por cada caso de prueba recibimos la entrada.

Por último generamos nuestra salida y así queda nuestra solución completa:



¿Cómo hacerlo?

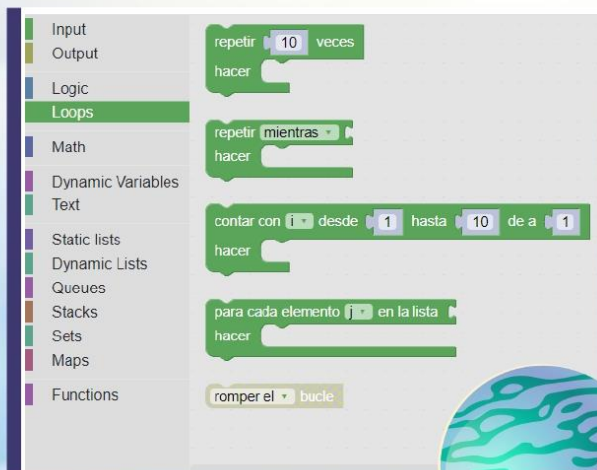
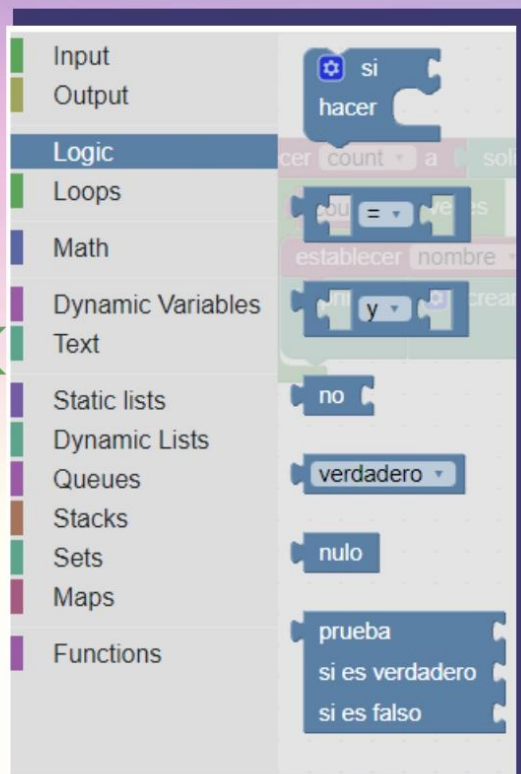
En tu espacio para resolver el problema encontrarás el siguiente menú



En output encontraras el bloque que te permite imprimir tus respuestas o la solución de tu problema



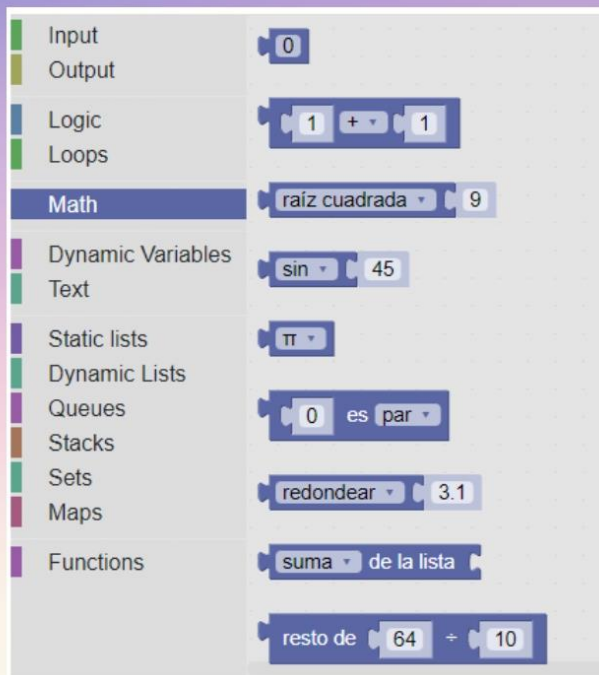

En logic podrás encontrar todas las herramientas para crear condicionales según el problema lo requiera. (Si no tienes claro el tema de condicionales, lo puedes repasar en el material de estudio).



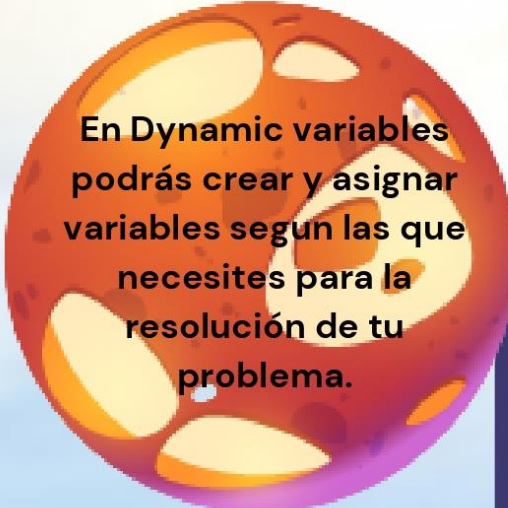
En loops encuentras todos los ciclos que necesitas para resolver tu problema. Esta herramienta te ayuda a realizar operaciones o tareas de forma repetitiva según como lo configures y necesites.



Si tienes dudas acerca de cómo usar los ciclos revisa dentro de nuestro material y conviértete en experto.

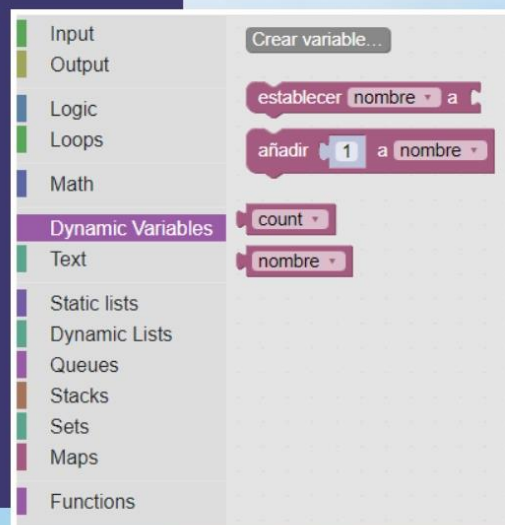



En Math podrás encontrar todo lo relacionado con operaciones matemáticas. Es tu tarea descubrir si el problema dado se puede solucionar de forma matemática. ¿Será?

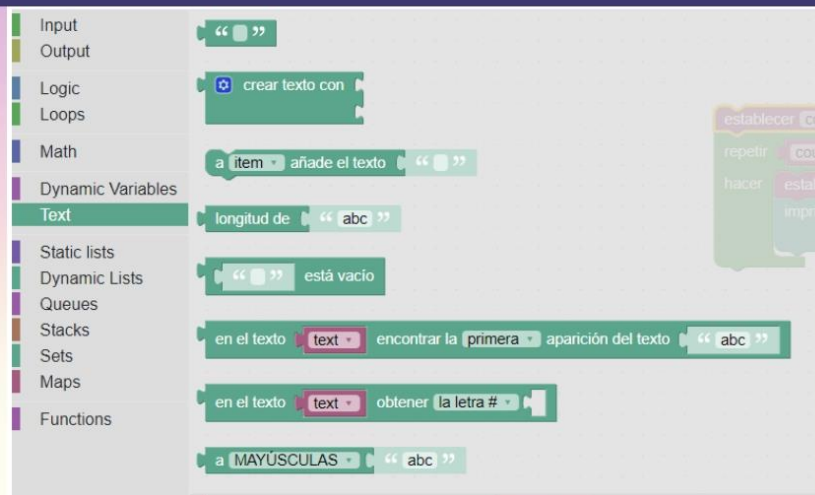


En Dynamic variables podrás crear y asignar variables según las que necesites para la resolución de tu problema.

Recuerda que la claridad del nombre que le coloques a tus variables te ayudará a entender mejor lo que haces.



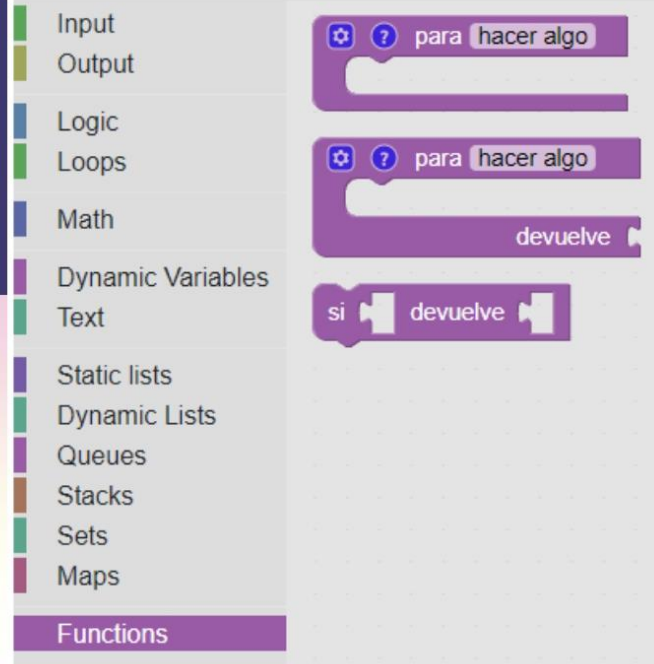
En text encontrarás todo lo relacionado con cadenas de texto (Concatenación, búsqueda en cadenas etc).
 Recuerda que si necesitas saber sobre como usar adecuadamente los Strings o cadenas de texto, puedes revisar el material de estudio.



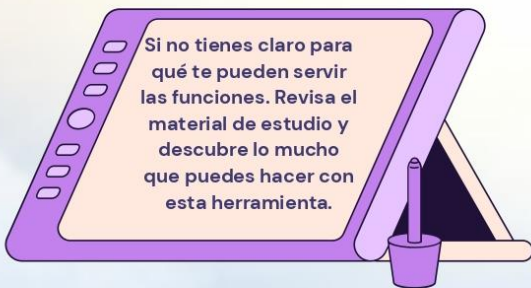
En Static list se encuentra todo lo relacionado con creación y manejo de listas estaticas, inserción, y ordenamiento de las mismas, entre otras cosas.



En functions encontrarás los bloques con los que podrás crear e invocar funciones o métodos .



Si no tienes claro para qué te pueden servir las funciones. Revisa el material de estudio y descubre lo mucho que puedes hacer con esta herramienta.



Anexo 7. Ficha Técnica

PRODUCTO			
CARACTERÍSTICAS DEL PRODUCTO			
Nombre del Producto	UFPS Training Center		
Proyecto padre	Training Center High School		
Versiones Anteriores	V. 1.0.0	Versión Actual	2.0.0
DESCRIPCIÓN DEL PRODUCTO			
Descripción General del Producto	Plataforma para el entrenamiento de programación competitiva de los estudiantes del grupo de estudio en programación competitiva y colegios de la ciudad de Cúcuta.		
Objetivo	Desarrollar una extensión de la plataforma Training Center UFPS para promover la Programación Competitiva en colegios de la ciudad de Cúcuta.		
ARQUITECTURA			
Descripción	Aplicación de cliente /servidor. El cliente se realiza siguiendo un enfoque multicapa y el cliente bajo un enfoque MVVM		
REQUERIMIENTOS DEL PRODUCTO			
Requisitos del Sistema (Servidor)			
Hardware	Requisitos mínimos: Memoria RAM: 1GB CPU: 1GHz Disco duro: 20 GB Transferencia: 20GB/Mes		
Software	Sistema operativo Linux (En windows la plataforma funciona pero no el calificador)		

	Node.js NPM MariaDB Docker Bash
Requisitos del Sistema (Cliente)	
Hardware	Requisitos mínimos: Memoria RAM: 512mb CPU: 512MHz
Software	Navegador web moderno. Garantizado el funcionamiento en: Microsoft Edge, Mozilla Firefox, Google Chrome, Opera, Maxthon y Vivaldi en sus versiones más recientes. Los navegadores modernos con soporte para HTML5 deben funcionar correctamente. No funciona en Internet Explorer 8 y versiones anteriores. En versiones más recientes funciona, con algunos detalles de visualización de interfaz. Se recomienda su utilización en computadores en lugar de dispositivos móviles.
Otros	Es necesaria una conexión a internet.
REQUERIMIENTOS	
Requerimientos Funcionales Generales	<ul style="list-style-type: none"> • Listar problemas • Agregar problemas • Implementar Blockly en problemas • Implementar Blockly en Maratones • Calificar problemas • Corregir problemas • Eliminar problemas • Filtrar problemas (por nombre, por colegio y universidad) • Categorizar problemas • Ver ranking • Crear competencias • Añadir material de ayuda • Ver material de ayuda • Crear modo guiado • Recibir solución de problemas

	<ul style="list-style-type: none"> • Ver estadísticas • Ver modo guiado (Tour) • Gestionar usuarios • Acceder a foro de discusión • Inspeccionar códigos en ranking • Inspeccionar códigos en competencias
Requerimientos Adicionales	<ul style="list-style-type: none"> • Los envíos de soluciones no deben afectar la estabilidad del sistema. • Las contraseñas de los usuarios deben almacenarse encriptadas. • Las soluciones enviadas por los usuarios deben ejecutarse en modo “sandbox”. • El sistema debe funcionar en cualquier plataforma moderna.
Requisitos Legales Aplicables	<ul style="list-style-type: none"> • Cualquier usuario puede eliminar su cuenta y toda su información a la plataforma realizando esta petición al administrador cumpliendo con la ley de habeas data.
Clientes del producto	<p>Grupo de estudio en programación competitiva UFPS Estudiantes de instituciones de educación superior Estudiantes de instituciones educativas</p>