

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO

CARRERA:

COMPUTACIÓN

Trabajo de titulación previo a la obtención del título:

Ingenieros en Ciencias de la Computación

TEMA:

**DESARROLLO DE UNA PLATAFORMA INFORMÁTICA PARA EL ANÁLISIS Y
ESTUDIO DE ORIENTACIÓN VOCACIONAL Y PROFESIONAL EN LA
UNIVERSIDAD POLITÉCNICA SALESIANA.**

AUTORES:

ERICK ANTHONY CABEZAS MIRANDA

KEVIN ALEXANDER CHACHA MOLINA

TUTOR:

RODRIGO EFRAÍN TUFÍÑO CÁRDENAS

Quito, noviembre de 2021

CESIÓN DE DERECHOS DE AUTORES

Nosotros, Kevin Alexander Chacha Molina con documento de identificación 1400878532 y Erick Anthony Cabezas Miranda con documento de identificación 1753788890 manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación con el tema: **DESARROLLO DE UNA PLATAFORMA INFORMÁTICA PARA EL ANÁLISIS Y ESTUDIO DE ORIENTACIÓN VOCACIONAL Y PROFESIONAL EN LA UNIVERSIDAD POLITÉCNICA SALESIANA.** Mismo que ha sido desarrollado para optar el título de **INGENIEROS EN CIENCIAS DE LA COMPUTACIÓN** en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada.

En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato digital a la biblioteca de la Universidad Politécnica Salesiana.



Kevin Alexander Chacha Molina
1400878532



Erick Anthony Cabezas Miranda
1753788890

Quito, noviembre de 2021

DECLARATORIA DE COAUTORÍA DEL TUTORES

Yo declaro que bajo mi dirección y asesoría fue desarrollado el Proyecto Técnico con el tema: **DESARROLLO DE UNA PLATAFORMA INFORMÁTICA PARA EL ANÁLISIS Y ESTUDIO DE ORIENTACIÓN VOCACIONAL Y PROFESIONAL EN LA UNIVERSIDAD POLITÉCNICA SALESIANA** realizado por: Kevin Alexander Chacha Molina y Erick Anthony Cabezas Miranda, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerado como trabajo final de titulación.

Quito, noviembre de 2021



Rodrigo Efraín Tufiño Cárdenas
C.I: 1717646390

DEDICATORIA

Este trabajo lo dedicamos con todo nuestro amor y cariño a nuestros padres amados los cuales han sido el pilar fundamental en este camino de aprendizaje ya que sin su apoyo no hubiera sido posible cumplir esta meta.

AGRADECIMIENTO

Principalmente gracias a Dios por brindarnos salud, familia, valor y un amplio conocimiento para poder culminar esta etapa.

A nuestros padres por los buenos consejos en este largo camino, por su amor, su apoyo mediante los cuales nos hemos formado personal y profesionalmente.

A la Universidad Politécnica Salesiana por expandirnos el conocimiento y brindarnos el apoyo necesario para formarnos profesionalmente.

Contenido

RESUMEN	1
ABSTRACT	2
1. INTRODUCCIÓN	3
1.1 ANTECEDENTE	3
1.2 PROBLEMA	3
1.3 JUSTIFICACIÓN	6
1.4 OBJETIVOS	7
1.4.1 Objetivo General	7
1.4.2 Objetivos Específicos	7
1.5 METODOLOGÍA	8
2. ANÁLISIS DEL PROBLEMA	9
2.1 ANÁLISIS DEL PROCESO ACTUAL	9
2.1.1 Grupo de Innovación Educativa	9
2.1.2 Composición de las Formulas	9
2.1.3 Proceso de aplicación de exámenes	12
2.2 ANÁLISIS DE VIABILIDAD	13
2.2.1 Viabilidad técnica	13
2.2.2 Viabilidad económica	15
2.2.3 Viabilidad operacional	15
2.2.4 Conclusiones	16
2.3 ANÁLISIS DE REQUERIMIENTOS	17
2.3.1 Alcance	17
2.3.2 Definiciones, acrónimos y abreviaturas	18
2.3.3 Visión general	18
2.3.4 Descripción general	18
2.3.5 Requerimientos funcionales	20
2.3.6 Requerimientos de rendimiento	24
2.3.7 Requerimientos de diseño	24
3. DISEÑO	25
3.1 DIAGRAMAS DE CASOS DE USO	25
3.1.1 Diagrama de caso de uso Administrador	25
3.1.2 Diagrama de caso de uso Investigador	26
3.1.3 Diagrama de caso de uso Alumno	27
3.2 DIAGRAMA DE BASE DE DATOS	28
3.2.1 DICCIONARIO DE DATOS	31
3.3 DIAGRAMA DE CLASES	36
3.4 MAPA DE NAVEGACIÓN	38
3.4.1 Diagrama de navegación para Actor Investigador	38
3.4.2 Diagrama de navegación para Actor Administrador	38
3.4.3 Diagrama de navegación para Actor Alumno	40

3.5	DISEÑO INTERFAZ ABSTRACTA	40
3.5.1.	Interfaz abstracta general	40
3.5.2.	Interfaz abstracta para login	41
3.5.3.	Interfaz abstracta para Actor Investigador	42
3.5.4	Interfaz abstracta para Actor Alumno	44
4.	CONSTRUCCIÓN Y PRUEBAS	46
4.1	HERRAMIENTAS Y TECNOLOGÍAS	46
4.1.1	Python 3.8.8	46
4.1.2	Django 3.2	46
4.1.3	PostgreSQL 13.3	47
4.2	PRODUCT BACKLOG	47
4.3	SPRINTS	48
4.3.1	Sprint 1: Configuración del entorno de desarrollo.	48
4.3.2.	Sprint 2: Importación, exportación y autenticación.	49
4.3.3	Sprint 3: Plantilla e Interfaces de investigador y alumno	51
4.3.4	Sprint 4: Gestión de encuestas, formulas y asignaciones	53
4.3.5	Sprint 5: Visualizar y Llenar encuesta	60
4.4	PRUEBAS	63
4.4.1	Pruebas contra requerimientos	64
4.4.2	Pruebas de código	65
4.4.3	Pruebas de carga y rendimiento	68
4.4.4	Pruebas con usuarios	72
5.	IMPLEMENTACIÓN	74
5.1	INFRAESTRUCTURA EN AWS	74
5.2	INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE	76
5.2.1	Instalación de paquetes y servicios	77
5.3	DESPLIEGUE	78
6.	CONCLUSIONES	83
7.	RECOMENDACIONES	84
	REFERENCIAS	85
	ANEXOS	87

Índice de Tablas

Tabla 1	Elementos de SCRUM	8
Tabla 2	Campos profesionales.....	10
Tabla 3	ICAV	10
Tabla 4	Equipo 1.....	13
Tabla 5	Equipo 2.....	13
Tabla 6	Detalle del servidor	14
Tabla 7	Información de los participantes del proyecto	14
Tabla 8	Costos totales detallados	15
Tabla 9	Definiciones, acrónimos y abreviaturas	18
Tabla 10	Perfiles, funciones	20
Tabla 11	Requerimiento funcional 1.1	21
Tabla 12	Requerimiento funcional 1.2	21
Tabla 13	Requerimiento funcional 1.3	21
Tabla 14	Requerimiento funcional 2.1	21
Tabla 15	Requerimiento funcional 2.2	22
Tabla 16	Requerimiento funcional 2.3	22
Tabla 17	Requerimiento funcional 2.4	22
Tabla 18	Requerimiento funcional 2.5	22
Tabla 19	Requerimiento funcional 2.6	23
Tabla 20	Requerimiento funcional 3.1	23
Tabla 21	Requerimiento funcional 4.1	23
Tabla 22	Requerimiento funcional 4.2	23
Tabla 23	Detalle tabla encuesta	31
Tabla 24	Detalle tabla alumno.....	31
Tabla 25	Detalle tabla categoría.....	31
Tabla 26	Detalle tabla pregunta.....	32
Tabla 27	Detalle tabla opción.....	32
Tabla 28	Detalle tabla fórmula	32
Tabla 29	Detalle tabla respuesta.....	32
Tabla 30	Detalle tabla relación.....	33
Tabla 31	Detalle tabla tpregunta.....	33
Tabla 32	Detalle tabla detalle	33
Tabla 33	Detalle tabla formula_categoria	33
Tabla 34	Detalle tabla rendimiento	33
Tabla 35	Detalle tabla término	34
Tabla 36	Detalle tabla asignación.....	34
Tabla 37	Detalle tabla estudio	34
Tabla 38	Detalle tabla evaluación	34
Tabla 39	Detalle tabla resultado	35
Tabla 40	Detalle tabla periodo	35
Tabla 41	Detalle tabla registro	35

Tabla 42	Detalle tabla nivel.....	35
Tabla 43	Detalle tabla carrera.....	35
Tabla 44	Detalle tabla parámetro	33
Tabla 45	Product Backlog	47
Tabla 46	Sprint Backlog 1	49
Tabla 47	Sprint backlog 2.....	50
Tabla 48	Sprint Backlog 3.....	52
Tabla 49	Sprint Backlog 4.....	53
Tabla 50	Casos de pruebas de requerimientos funcionales del administrador. 64	
Tabla 51	Casos de pruebas de requerimientos funcionales del psicólogo.	64
Tabla 52	Casos de pruebas de requerimientos funcionales del estudiante.....	65
Tabla 53	Casos de pruebas de requerimientos funcionales comunes de los actores 65	
Tabla 54	Sprint 2	88
Tabla 55	Sprint 3	83

Índice de figuras

Figura 1.	Comparación con el punto de corte.....	11
Figura 2.	Resultados en Excel.....	12
Figura 3.	<i>Caso de uso Administrador</i>	25
Figura 4.	Caso de uso Investigador.....	26
Figura 5.	Caso de uso Alumno.....	27
Figura 6.	<i>Diagrama de base de datos</i>	30
Figura 1.	Diagrama de clases	37
Figura 2.	Diagrama navegacional Investigador	38
Figura 3.	<i>Diagrama navegacional Administrador</i>	39
Figura 4.	<i>Diagrama navegacional Alumno</i>	38
Figura 5.	Interfaz abstracta general.....	39
Figura 6.	Interfaz abstracta login.	39
Figura 7.	Interfaz abstracta Investigador para creación de encuestas	42
Figura 8.	Interfaz abstracta Investigador para la asignación de encuesta.....	42
Figura 9.	Interfaz abstracta fórmula paso 1.	43
Figura 10.	Interfaz abstracta formula paso 2	43
Figura 11.	Interfaz abstracta formula paso 3	43
Figura 12.	Interfaz abstracta formula paso 4	44
Figura 13.	Interfaz abstracta del panel de mando.....	44
Figura 14.	Interfaz abstracta encuesta	45
Figura 15.	Interfaz abstracta para visualización de resultados finales.	45
Figura 16.	Formulario de acceso	50
Figura 17.	Interfaz para el usuario investigador.....	52
Figura 18.	Interfaz para el usuario alumno.....	53
Figura 19.	Gestión de encuestas	55
Figura 20.	Gestión de fórmulas	58
Figura 21.	Asignación de encuestas	59
Figura 22.	Resumen métricas de SonarQube	66
Figura 23.	Métrica fiabilidad	66
Figura 24.	Métrica seguridad.....	67
Figura 25.	Métrica duplicidad.....	67
Figura 26.	Resultados finales.....	68

RESUMEN

El sistema desarrollado en este proyecto es una plataforma informática para exámenes psicométricos, a través de un sistema automatizado que recolecte de forma rápida todos los datos de los encuestados, muestre los resultados de una manera exacta y permita a los investigadores realizar ajustes a sus estudios y exámenes psicométricos. El desarrollo del sistema inició en las fases de análisis, diseño, construcción, culminando con las pruebas y la implementación del sistema en el servidor de AWS. Se utilizó la metodología SCRUM para la construcción del aplicativo web. Como resultado a través de una serie de fórmulas se obtuvo un cálculo más preciso, rápido de la afinidad profesional del estudiante.

ABSTRACT

The system developed in this project is a computer platform for psychometric tests, through an automated system that quickly collects all the data from the respondents, displays the results in an exact way and allows researchers to make adjustments to their studies and Psychometric tests. The development of the system began in the analysis, design, construction phases, culminating in the testing and implementation of the system on the AWS server. The SCRUM methodology was used to build the web application. As a result, through a series of formulas, a more precise and rapid calculation of the professional affinity of the student was obtained.

1. INTRODUCCIÓN

1.1 ANTECEDENTE

La base para el desarrollo de un profesional en la sociedad se fundamenta en la correcta elección de su carrera, en 1999 la Standards for Educational and Psychological propone como definición que una prueba consiste en la selección de un determinado campo académico profesional compuesto de un conjunto de actividades lúdico-laborales que se quieran realizar profesionalmente y para el cual existen una serie de habilidades, aptitudes, conocimientos previos y motivos que llevan a una elección de titulación o profesión futura (Lorenzo Delgado, Rojas Ruiz, & Cepero González, 2009).

En 1936 cuando se conocería el primer artículo propio de la informática, que sería una creación de Alan Turing. Este matemático británico supo entender la manera en que algunas operaciones elementales de cálculo eran universales. Para lo cual la integración de distintas plataformas informáticas en el ámbito académico ha permitido aumentar la disponibilidad de los recursos y mejorar la interacción entre los distintos actores que intervienen logrando así que se conviertan en una de las vías de acceso a bienes y servicios habituales para los ciudadanos y muy especialmente para los jóvenes (Ferrer, 2018).

1.2 PROBLEMA

Para el desarrollo de este proyecto se han identificado dos problemáticas: la primera es la falta de una plataforma informática que se encargue de recopilar toda la información del alumno a través de una serie de encuestas establecidas por los investigadores. Por otro lado, el análisis y estudio de orientación vocacional y preferencias profesionales, es un medio decisivo para dar respuesta a la necesidad de duda de un alumno al escoger una carrera,

ofreciendo información relevante para una segura y mejor selección de carrera (Zumárraga et al., 2019).

A nivel mundial las preferencias profesionales constituyen un factor de especial importancia al momento de elegir una carrera universitaria. Esto se debe a que los intereses profesionales ofrecen señales que pueden ayudar a detectar aquellas capacidades o habilidades del alumno que tienen mayor potencial para lograr un desempeño satisfactorio en el proceso de formación profesional (Zumárraga et al., 2019). Según Bravo Torres y Vergara Tamayo (2018) la falta de una detección clara de las preferencias profesionales, debido en parte a la carencia de procesos oportunos de orientación vocacional, que conlleva a la selección de carreras contrarias a las habilidades académicas de los alumnos, dando lugar a problemáticas como el fracaso o la deserción académica.

A nivel de Latinoamérica, se han identificado una serie de factores académicos, demográficos y socioeconómicos que se conectan con el fracaso y el abandono universitario. En términos académicos, aspectos como la jornada de estudio y el tipo de carrera seleccionada pueden influir en el rendimiento académico (Carrillo Regalado & Ríos Almodóvar, 2013). En relación con las características demográficas, los hombres tienden a presentar un menor rendimiento académico que las mujeres (Gómez Sánchez, Oviedo Marín, & Martínez López, 2011). El estatus socioeconómico opera como un predictor del rendimiento académico, siendo aquellos alumnos de estratos más altos quienes suelen presentar un mayor nivel de calificaciones (Vargas, 2007). Además, existen otros factores como la falta de interés, falta de apoyo familiar, falta de metodologías de aprendizaje, entre otras.

En Ecuador, en la Universidad Politécnica Salesiana los rasgos de personalidad han ganado creciente interés como un factor explicativo del rendimiento académico y la deserción en

entornos universitarios para lo cual la Universidad Politécnica Salesiana ha desarrollado una estrategia orientada a fortalecer sus procesos de orientación vocacional y profesional mediante la incorporación de un diagnóstico temprano de personalidad. Posibilitando, de esta manera, evaluar y mejorar el grado de compatibilidad entre las características personales de los aspirantes universitarios y la carrera elegida, con el propósito de asegurar mejores niveles de rendimiento académico y reducir el riesgo de abandono (Zumárraga et al., 2019).

El presente proyecto de investigación responderá a la siguiente problemática ¿Existe una plataforma informática que permita realizar el análisis y estudio de orientación vocacional y profesional en alumnos aspirantes de la Universidad Politécnica Salesiana? La Universidad Politécnica Salesiana realiza gran cantidad exámenes psicométricos que son considerados como un activo de suma importancia y utilidad, sin embargo, se presentan algunos problemas como un elevado tiempo al realizar cada examen, resultados pocos confiables los cuales luego de un tiempo tienen como consecuencia la deserción académica (Torres & Vergara, 2018), elevado gasto en papeles poco reciclables, etc. Con la plataforma informática a implementarse se obtendrán beneficios como la automatización de procesos en los exámenes y la obtención de resultados más exactos para unas próximas investigaciones.

La razón por la cual es importante desarrollar una plataforma informática para exámenes psicométricos es el elevado porcentaje de tiempo en cada encuesta, pérdida de exámenes psicométricos, resultados finales erróneos, falta de automatización, etc. La falta de automatización es el problema con mayor impacto en esta área debido a que todos los procesos se realizan de forma manual. Para brindar una solución a los problemas previamente identificadas se propone la implementación de una plataforma informática de fácil uso.

Los efectos que puede traer la falta de una plataforma informática son muchos, entre los que hemos detectado están; un alto porcentaje de error en la tabulación en los exámenes, pérdida de información, un elevado tiempo de respuesta, etc. El alto porcentaje de error en cada examen ha sido el efecto con más impacto debido a que no se tiene el conocimiento ni la claridad de la ubicación de esta información, por lo cual no asegura la integridad ni la disponibilidad de estos datos.

Por ello, el presente proyecto pretende desarrollar una plataforma informática para exámenes psicométricos a través de un sistema automatizado que recolecte de forma rápida todos los datos de los encuestados, muestre los resultados y permita a los investigadores realizar ajustes a sus estudios y exámenes psicométricos.

1.3 JUSTIFICACIÓN

La deserción de estudios superiores incide directamente en el desarrollo de la sociedad, por esto la importancia de resolver factores que influyan en la decisión del alumno de abandonar los estudios (Lázaro et al., 2017). La vocación es un factor fundamental para explicar el abandono universitario, por lo cual es necesario fortalecer los procesos de orientación vocacional y profesional (Zumárraga et al., 2019).

Las plataformas informáticas aportan una gestión eficiente y ágil en una organización; por esta razón el presente proyecto tiene como propósito optimizar los procesos de orientación vocacional que se dan mediante exámenes psicométricos en la Universidad Politécnica Salesiana.

La obtención, registro y análisis de exámenes psicométricos, es una problemática a la cual se enfrenta el Grupo de Innovación Educativa (GIE) de Orientación Profesional y Vocacional de

la UPS. Pues, existen procesos manuales durante el registro de exámenes, lo que dificulta y ralentiza la obtención de resultados.

Es así, que utilizar herramientas que faciliten almacenar y resguardar correctamente los datos de cada proceso de orientación vocacional permitirá implementar nuevos métodos de estudio en la investigación.

Por consiguiente, el desarrollo del proyecto permitirá al Grupo de Innovación Educativa de Orientación Profesional y Vocacional de la UPS apoyarse en herramientas de software, que aseguren la calidad e integridad de los datos y puedan acceder a estos idóneamente. De igual forma, permitirá a los alumnos de la Universidad Politécnica Salesiana obtener una puntuación inmediata sobre su aptitud vocacional.

1.4 OBJETIVOS

1.4.1 Objetivo General

Desarrollar una plataforma informática que permita el análisis y estudio de orientación vocacional y profesional en alumnos de la Universidad Politécnica Salesiana.

1.4.2 Objetivos Específicos

- Analizar y modelar el proceso por el cual se rige la actual implementación de exámenes psicométricos para aptitud profesional.
- Diseñar una plataforma informática que permita gestionar el almacenamiento y el acceso oportuno a los datos.
- Construir la plataforma mediante metodología SCRUM.
- Evaluar la plataforma informática y el uso de recursos computacionales utilizando métricas.
- Implementar la plataforma informática para el Grupo de Innovación Educativa de Orientación Profesional y Vocacional de la UPS-Sede Quito.

1.5 METODOLOGÍA

SCRUM

SCRUM es un marco de metodología ágil que permite el desarrollo incremental de un producto, apoyando a personas, equipos y organizaciones a generar valor mediante soluciones adaptables a problemas complejos (Schwaber & Sutherland, 2020).

Esta metodología se basa en el pensamiento Lean y el empirismo. El pensamiento Lean describe el enfocarse en lo esencial y reducir los desperdicios. Mientras que el empirismo asegura que el conocimiento se da de la experiencia y las decisiones tomadas en base a lo observado (Schwaber & Sutherland, 2020).

Sprint es el nombre que se le da a un tiempo de duración en el que se desarrolla un objetivo claro. Los equipos en scrum son generalmente pequeños y de menos de 10 personas, son auto gestionables y multifuncionales. Cada equipo tiene la habilidad para completar valor en un sprint (Schwaber & Sutherland, 2020).

SCRUM define elementos que proporcionan una guía clara para su implementación los cuales son:

Tabla 1 Elementos de SCRUM

Nombre	Función
Product Owner	Es el dueño del producto, encargado de sacar el mayor beneficio a su producto y al equipo de desarrollo.
Development Team	Son los encargados de entregar el producto terminado de forma incremental.
SCRUM Máster	Líder del equipo asegura que el equipo cumpla con la metodología y reglas.
El Sprint	Es un tiempo generalmente de un mes o menos, en el que se define el incremento que tendrá el producto terminado.
Product Backlog	Una lista que contiene todo lo que necesita el producto, esta consta como la única fuente de requisitos
Sprint Backlog	Contiene los pendientes que tiene el sprint que se asignan de la lista completa de requisitos.

Nota: listado general de los elementos de SCRUM. Elaborado por: los autores.

2. ANÁLISIS DEL PROBLEMA

Este capítulo introducirá al grupo de beneficiarios del proyecto y sus labores actuales, también proporciona un estudio de viabilidad para determinar la factibilidad de este trabajo. Y se establecen los requerimientos de la plataforma.

2.1 ANÁLISIS DEL PROCESO ACTUAL

2.1.1 *Grupo de Innovación Educativa*

El Grupo de Innovación Educativa (GIE) de Orientación Profesional y Vocacional de la UPS-Sede Quito, según (Zumárraga et al., 2019) estudia las principales y más preocupantes causas de deserción estudiantil. El grupo asumió la función de apoyar el fortalecimiento de procesos de orientación profesional-vocacional de la UPS. El objetivo principal fue, generar instrumentos psicométricos propios y mantener en constante mejora los mismos.

Para mejorar y crear instrumentos psicométricos de mayor calidad el grupo GIE estudió los diferentes factores que afectan en la elección de una carrera, se puede evidenciar esto en “¿Los rasgos de personalidad importan? Medición y análisis de su correlación con el rendimiento académico como aporte a los procesos institucionales de orientación vocacional” (Zumárraga, Luzuriaga, Armas, González, & Peñaherrera, 2020), con esto, se generaron dos encuestas que son: “Cuestionario de intereses profesionales” (CIPRO) y “Cuestionario de Rasgos de personalidad” (CUPER).

2.1.2 *Composición de las Formulas*

Para la creación de fórmulas que favorezcan los resultados presentados frente a las encuestas de vocación y orientación profesional se especificaron diferentes campos profesionales.

Actualmente están definidos 13 campos profesionales y se los puede evidenciar en la siguiente tabla 2.

Tabla 2 Campos profesionales

Campos profesionales [10-50]	COD
Ciencias Físicas	CF
Mecánica	M
Eléctrica-Electrónica	EE
Tecnologías de la Información	TI
Ingeniería Civil	IC
Administración	A
Ciencias Biológicas	CB
Educación	ED
Comunicación Social	CS
Científico-Experimental	CE
Persuasivo	P
Arte y Música	AM
Servicio Social	SS

Nota: Equipo portátil para desarrollo del proyecto. Elaborado por: los autores.

2.1.3.1 Punto de corte

Es un valor numérico decimal establecido por los estudios antes mencionados y que sirve para aplicarse con los diferentes componentes de las fórmulas.

2.1.3.2 Índice compuesto de ajuste vocacional

Para este estudio, las fórmulas están asociadas a una carrera ofertada por la UPS, además cada formula puede contener uno o más campos profesionales. Para los investigadores, la agrupación de campos profesionales con valores numéricos se denomina “Índice compuesto de ajuste vocacional” (ICAV). A continuación, se muestra un ejemplo del ICAV.

$$0.58 A - 0.44 AM - 0.111 IC$$

2.1.3.3 Criterios de reescalado

Son valores estáticos conformados por un máximo y por un mínimo que son parte de la formula ICAV para definir la afinidad o no afinidad del resultado.

2.1.3.4 Afinidad y no afinidad

Tanto la afinidad como la no afinidad están conformadas por un rendimiento riesgoso y un rendimiento satisfactorio; que son valores numéricos porcentuales establecidos por los estudios antes mencionados. A este componente se lo compara con el punto de corte para establecer la afinidad o no afinidad, dependiendo del resultado.

2.1.3.5 Resultado

Para obtener un resultado aplicando todos los anteriores componentes se aplica la siguiente formula.

$$ICAV[0 - 100] = \frac{ICAV - MIN}{MAX - MIN} \times 100$$

Con el resultado de esta fórmula y la comparación con el punto de corte, se conoce la afinidad del alumno frente a la carrera. Finalmente, con estas comparaciones se puede estimar el rendimiento que se espera del alumno. A continuación, se presenta el pseudocódigo que permite la validación entre afinidad y no afinidad.

Figura 1. ICAV

<p>Si ICAV mayor o igual que PUNTO DE CORTE Es AFIN Si no Es NO AFIN</p>
--

Nota: Pseudo código que representa el cálculo de la afinidad. Elaborado por: Los autores.

2.1.3 Proceso de aplicación de exámenes

Las encuestas son realizadas por los psicólogos investigadores en base a los factores más importantes que sus estudios hayan demostrado. Posteriormente se imprimen hojas para que cada uno de los alumnos las llenen a mano. Después, existe un proceso de tabulación de las hojas de respuesta de los alumnos. Actualmente los investigadores poseen un prototipo hecho en Microsoft Excel, que pretende simular la respuesta que un alumno debe obtener tras realizar una encuesta.

Figura 2. Resultados en Excel

UNIVERSIDAD POLITÉCNICA SALESIANA (UPS)			
CUESTIONARIO DE INTERESES PROFESIONALES (CIPRO-UPS)			
PORTAL DE DIAGNÓSTICO VOCACIONAL AUTOMATIZADO			
Nombre	Anonimizado		
Documento de identidad	XXXXXXXXXX		
BIENVENIDO			
Tus puntajes obtenidos por campo profesional [10 - 50]		Carrera de interés	<i>Seleccione por favor</i>
Ciencias Físicas	10		Psicología
Mecánica	20		
Eléctrica-Electrónica	10	Tu Índice de Ajuste Vocacional [0 - 100]	
Tecnologías de la Información	25	87.56	
Ingeniería Civil	20	Diagnóstico de ajuste vocacional	
Administración	20	PERFIL VOCACIONAL	AFÍN
Ciencias Biológicas	45	Si eliges esta carrera:	
Educación	15	Probabilidad de rendimiento satisfactorio	69.9%
Comunicación Social	20	Probabilidad de rendimiento riesgoso (promedio de calificaciones por debajo de 75/100)	30.1%
Científico-Experimental	40		
Persuasivo	25		
Arte y Música	15		
Servicio Social	50		<i>Probar con otra carrera?</i>
<p>Nota: Recuerda que este diagnóstico es de carácter informativo y tiene un margen de error. Un diagnóstico favorable no garantiza buenas calificaciones, un diagnóstico desfavorable tampoco asegura malas calificaciones. Evalúa tu afinidad con el resto de carreras ofertadas por la UPS, consulta sobre sus respectivos perfiles de egreso y perspectivas laborales. Reflexiona y elige tu carrera de manera informada.</p>			

Nota: Resultado que se obtiene de los datos tabulados. Elaborado por: Zumárraga Luzuriaga, Armas, González, Peñaherrera.

2.2 ANÁLISIS DE VIABILIDAD

Para identificar las fortalezas y debilidades del presente proyecto se ha llevado a cabo un estudio de viabilidad técnica, económica y operacional.

2.2.1 Viabilidad técnica

La plataforma informática será desarrollada en base a los recursos que se listan a continuación:

- **Hardware**

Para el desarrollo del proyecto se cuentan con los siguientes equipos:

Tabla 3 Equipo 1

Dispositivo	Computador Portátil
Marca	Lenovo Legion 5
Procesador	I7 103700H
Memoria	8 GB en RAM
Almacenamiento	256 GB SSD
Arquitectura	64 bits

Nota: Equipo portátil para desarrollo del proyecto. Elaborado por: los autores.

Tabla 4 Equipo 2

Dispositivo	Computador Portátil
Marca	Dell Inspiron 14
Procesador	I7 103700H
Memoria	8 GB en RAM
Almacenamiento	1 Tb HDD
Arquitectura	64 bits

Nota: Equipo 2 portátil para desarrollo del proyecto. Elaborado por: los autores.

- **Software**

El software utilizado en el desarrollo será:

- ✓ Python 3
- ✓ Django web framework 3.2
- ✓ Javascript
- ✓ PostgreSQL

- **Infraestructura**

La plataforma informática será implementada bajo un servidor *t2.micro* de Amazon Web Services (AWS), las especificaciones del servidor están detallados a continuación:

Tabla 5 Detalle del servidor

Detalle del Servidor			
Nombre	VCPU	RAM (GiB)	Almacenamiento (GiB)
t2.micro	1	1	25

Nota: Detalle del servidor que se utilizara. Elaborado por: los autores.

- **Personal**

El personal implicado en el desarrollo del Proyecto son dos alumnos del último nivel de la carrera de Ingeniería en Ciencias de la Computación junto con la guía de un docente Tutor.

Tabla 6 Información de los participantes del proyecto

Nombre	Cargo
Erick Anthony Cabezas Miranda	Desarrollador
Kevin Alexander Chacha Molina	Desarrollador
Rodrigo Efrain Tufiño Cardenas	Docente Tutor

Nota: Participantes para desarrollo del proyecto. Elaborado por: los autores.

2.2.2 Viabilidad económica

Se han recopilado los costes del personal implicado y las herramientas para el desarrollo del proyecto. Tomando en cuenta que el desarrollo tomará un total de seis meses se ha detallado los costos en la siguiente tabla:

Tabla 7 Costos totales detallados

Ítem	Cantidad	Meses	Costo	Total
Recursos Humanos				
Desarrollador	2	6	500	6000
Gastos operativos	-	6	100	600
Recursos Hardware				
Computador	2	-	800	1600
Estimación de Amazon EC2				
Amazon Elastic Block Storage (EBS) pricing (monthly)	1	12	-	-
Amazon EC2 Instance Savings Plans instances (monthly)	1	12	-	-
Total				8200

Nota: Resumen de gastos totales en el proyecto. Elaborado por: los autores.

El costo del servidor Amazon EC2 t2.micro se encuentra en la capa gratuita de 12 meses que ofrece Amazon a sus nuevos usuarios. Los demás costos estarán a cuenta de los alumnos encargados del proyecto.

2.2.3 Viabilidad operacional

En la actualidad la Universidad Politécnica Salesiana no cuenta con una plataforma que tenga la capacidad de ofrecer un examen psicométrico y rápidamente mostrar la afinidad profesional de un alumno, el cual debe estar en la capacidad de generar un reporte de todas las materias con sus respectivos porcentajes de afinidad.

Entre los principales riesgos detectados están:

El proyecto realizado por tesisistas es un riesgo que afecta de forma significativa, debido a que estos alumnos una vez culminados su etapa de estudios, deciden abandonar el proyecto y dejar a cargo a los responsables, y por otra parte el hecho de que el proyecto no tenga un seguimiento adecuado por parte de los encargados podría dar como consecuencia un proyecto obsoleto.

2.2.4 Conclusiones

De acuerdo con el análisis de viabilidad realizado, el sistema se constituye operativo debido a las siguientes razones:

El presente proyecto quedará en continuo desarrollo y a cargo del grupo de investigación y de los docentes encargados, lo que mitigará el riesgo de abandono una vez que los alumnos terminen sus estudios. Se cuenta con hardware y las tecnologías necesarias para el desarrollo e implementación del proyecto. Se cuenta con el presupuesto para cubrir los gastos detallados.

Se estima que la plataforma tendrá una gran acogida por parte de los investigadores, debido a su alta certeza en los resultados obtenidos al finalizar la encuesta. Se evitará la confusión o aburrimiento por parte de los usuarios ya que la plataforma será de fácil acceso y con interfaces funcionales. La plataforma estará disponible Amazon Web Services (AWS).

Por ende, se puede evidenciar que existe apoyo por parte de la Universidad Politécnica Salesiana y del grupo de investigación GIE. Por lo que, el desarrollo de este proyecto propuesto es viable.

2.3 ANÁLISIS DE REQUERIMIENTOS

Este capítulo presenta el detalle de los requerimientos de la plataforma informática y una visión general de lo que se va a desarrollar. Se especifican funcionalidades, rendimiento y requisitos de diseño de la plataforma.

2.3.1 Alcance

Para alcanzar el desarrollo de la plataforma se realizará entrevistas con el personal interesado, para recabar requisitos y posteriormente enfocarlos a la metodología en la que se trabajará. Se modelará el proceso actual que realizan manualmente los investigadores, de forma que bajo este modelo se puede empezar a diseñar la plataforma.

Se pretende diseñar la plataforma conforme los requisitos recabados y modelados, para lo cual se utilizarán diagramas UML y de bases de datos. Con estos diseños se procederá a planificar la arquitectura. De acuerdo con los requisitos planteados, el diseño y diagramas realizados se iniciarán con el levantamiento del entorno de desarrollo mediante la arquitectura planteada. Se utilizará GitHub con el propósito de mantener un control continuo del desarrollo. En concordancia con lo descrito la plataforma informática constará de los siguientes módulos: i) Formularios de exámenes psicométricos, ii) Interfaz de administración y gestión de formularios, iii) Interfaz de administración para formulas, iv) Visualización de datos. Adicionalmente se efectuarán las pruebas necesarias para el correcto funcionamiento de la plataforma. Para garantizar un rendimiento satisfactorio de la plataforma informática se realizarán evaluaciones de: capacidad, disponibilidad, escalabilidad, velocidad, estabilidad, fiabilidad, seguridad y concurrencia.

Se desplegará la plataforma en un servidor en la nube de Amazon Web Services para su disponibilidad y acceso inmediato a sus interesados.

2.3.2 Definiciones, acrónimos y abreviaturas

Tabla 8 Definiciones, acrónimos y abreviaturas

NOMBRE	DESCRIPCIÓN
UPS	Universidad Politécnica Salesiana.
GIE	Grupo de Innovación Educativa.
SCRUM	Es una forma de trabajo para desarrollo ágil de software.
Product Backlog	Una lista que contiene todo lo que necesita el producto, esta consta como la única fuente de requisitos.
Encuesta	Examen psicométrico con preguntas de aptitud profesional
AWS	Amazon Web Services.
HTTP	Protocolo de transferencia de hipertexto.
CRUD	Create, Read, Update, Delete
OSI	Open Source Initiative
IDE	Entorno de desarrollo integrado
BSD	Berkley Software Distribution

Nota: Resumen general de definiciones, acrónimos y abreviaturas. Elaborado por: los autores.

2.3.3 Visión general

A continuación, se describe la funcionalidad del proyecto, este apartado permitirá establecer un panorama claro sobre el proyecto y definirá los requisitos.

2.3.4 Descripción general

2.2.4.1 Perspectivas del producto.

La plataforma que se propone es independiente de otros sistemas, requiere de una base de datos para almacenar la información. Cabe mencionar que el registro de calificaciones de alumnos en seguimiento deberá obtenerse de un sistema externo y almacenarlo en la

plataforma a desarrollar. Los usuarios podrán acceder a través de un navegador web y permitirá realizar encuestas y reflejar resultados rápidos a alumnos.

2.2.4.2 Funciones del producto.

La plataforma se desarrollará para que funcione como una aplicación web, en la cual los alumnos, los investigadores y el administrador accederán mediante un navegador y deberán pasar por un sistema de autenticación que dará permiso a los siguientes módulos dependiendo de los permisos de cada usuario:

Módulo de administración

Este módulo tiene como funcionalidad el manejo de todas las configuraciones del sistema y la posibilidad de crear usuarios, cambiar claves, cambiar tiempos de exámenes, colores, tipo de letras, etc.

Módulo de encuestas

En este módulo todos los alumnos deberán responder una serie de preguntas y posterior a eso observarán un resultado por carrera.

Módulo de investigadores

En este módulo se llevarán a cabo diferentes funcionalidades como la creación de categorías, cambio de fórmulas, creación de nuevas encuestas, añadir preguntas, colocar las notas del seguimiento del alumno.

2.2.4.3 Características de usuario.

Para esta plataforma se tiene tres perfiles con diferentes funciones las cuales son:

Tabla 9 Perfiles, funciones

PERFILES	FUNCIONES
Administrador	Persona encargada de acceder al módulo de administración en el cual realizara funciones como la creación de usuarios y configuración del sistema.
Investigadores	Persona encargada de acceder al módulo de investigadores en el cual realizara funciones como la gestión de encuestas, generación de resultados y actualización de fórmulas establecidas.
Alumno	Esta persona es la que se encargara de responder todas las preguntas y de visualizar los resultados mediante gráficos.

Nota: Resumen perfiles con sus respectivas funciones. Elaborado por: los autores.

2.2.5 Requerimientos específicos

2.2.5.1 Requerimientos de interfaz externo

Interfaces de usuario. Las interfaces de usuario serán minimalistas procurando una interacción eficaz entre el usuario y la interfaz, estarán adaptadas para la web.

Interfaces de hardware. La plataforma trabajara bajo arquitecturas de 64 y 32 bits y en una resolución estándar de 1280 x 1024 píxeles.

Interfaces de software. La plataforma estará disponible en navegadores basados en Chrome versión 22 o superior y Firefox versión 66.0 o superior.

Interfaces de comunicación. La plataforma podrá trabajar sobre el protocolo HTTP.

2.3.5 Requerimientos funcionales

En esta sección se listarán los requerimientos por cada usuario mediante las siguientes tablas:

1. Administrador

Tabla 10 Requerimiento funcional 1.1

Descripción	Gestión de usuarios
Precondición	Administrador autenticado correctamente.
Entrada	Información de los usuarios.
Proceso	Agregar, modificar, eliminar y actualización de usuarios.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para la gestión de usuarios. Elaborado por: los autores.

Tabla 11 Requerimiento funcional 1.2

Descripción	Gestión de parámetros
Precondición	Administrador autenticado correctamente.
Entrada	Carreras, periodos, niveles, tipos de pregunta.
Proceso	Agregar, modificar, eliminar y actualizar registros del sistema.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para la gestión de parámetros. Elaborado por: los autores.

Tabla 12 Requerimiento funcional 1.3

Descripción	Carga de información
Precondición	Administrador autenticado correctamente.
Entrada	Información para subir al sistema (csv).
Proceso	Carga la información en las tablas correspondientes.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para la carga de información de los alumnos. Elaborado por: los autores.

2. Investigador

Tabla 13 Requerimiento funcional 2.1

Descripción	Gestión de estudios
Precondición	Investigador autenticado correctamente.
Entrada	Información de los alumnos:
Proceso	Crear estudios que se aplican a los alumnos.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para la carga de información de los alumnos. Elaborado por: los autores.

Tabla 14 Requerimiento funcional 2.2

Descripción	Gestión de encuestas
Precondición	Investigador autenticado correctamente.
Entrada	Información de encuestas, categorías, preguntas, opciones de respuesta.
Proceso	Agregar, modificar, eliminar y actualizar encuestas del sistema.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para la gestión de encuestas. Elaborado por: los autores.

Tabla 15 Requerimiento funcional 2.3

Descripción	Gestión de fórmulas
Precondición	Investigador autenticado correctamente, carreras creadas, encuestas creadas.
Entrada	Información de fórmulas: Valor nuevo de las fórmulas, porcentaje nuevo de las fórmulas.
Proceso	Agregar, modificar, eliminar y actualizar formulas del sistema.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para la gestión de fórmulas. Elaborado por: los autores.

Tabla 16 Requerimiento funcional 2.4

Descripción	Asigna encuestas a alumnos
Precondición	Investigador autenticado correctamente, alumnos creados, encuestas creadas, estudios creados.
Entrada	Información de las encuestas.
Proceso	Asignar una encuesta a un alumno.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para la asignación de encuestas a los alumnos. Elaborado por: los autores.

Tabla 17 Requerimiento funcional 2.5

Descripción	Seguimiento a alumnos
Precondición	Investigador autenticado correctamente, periodo creado, alumnos creados, carreras creadas.
Entrada	Información del alumno: Notas de los primeros niveles.
Proceso	Registro de notas de alumnos.
Salida	Mensaje de éxito o error.

Nota: Requerimiento para el seguimiento de alumno en los primeros niveles. Elaborado por: los autores.

Tabla 18 Requerimiento funcional 2.6

Descripción	Resultados
Precondición	Investigador autenticado correctamente.
Entrada	Seleccionar nombre del alumno.
Proceso	Visualizar los resultados aplicados por alumno, estudio y período.
Salida	Creación de gráficos con información resumida del alumno.

Nota: Requerimiento para la visualización de resultados de los alumnos. Elaborado por: los autores.

3. Alumno

Tabla 19 Requerimiento funcional 3.1

Descripción	Responde a encuestas
Precondición	Alumno autenticado correctamente.
Entrada	Datos de autenticación del alumno.
Proceso	Llena las preguntas.
Salida	Visualiza resultados.

Nota: Requerimiento para llenado de encuesta por el alumno. Elaborado por: los autores.

4. Requerimientos comunes a todos los usuarios.

Tabla 20 Requerimiento funcional 4.1

Descripción	Autenticación en el sistema
Precondición	Usuario se encuentre creado en el sistema.
Entrada	Credenciales de acceso: usuario y contraseña
Proceso	Ingresar credenciales de acceso.
Salida	Mensaje de éxito o error.

Nota: Requerimiento común de todos los actores para la autenticación en el sistema. Elaborado por: los autores.

Tabla 21 Requerimiento funcional 4.2

Descripción	Cierre de sesión (logout)
Precondición	Usuario se encuentre autenticado.
Proceso	Cierre de la aplicación.
Salida	Mensaje despedido.

*Nota: Requerimiento común de todos los actores para el cierre de sesión en el sistema.
Elaborado por: los autores.*

2.3.6 *Requerimientos de rendimiento*

La plataforma estará disponible 24 horas al día y siete horas a la semana

Estará disponible para al menos 100 usuarios concurrentes.

2.3.7 *Requerimientos de diseño*

Las herramientas y software utilizado para el desarrollo e implementación de la plataforma serán software libre.

3. DISEÑO

En este capítulo se detallan los casos de uso, las clases y las funcionalidades con las que interactúan los diferentes actores de la plataforma; también el diseño de base de datos que se utilizará para la plataforma.

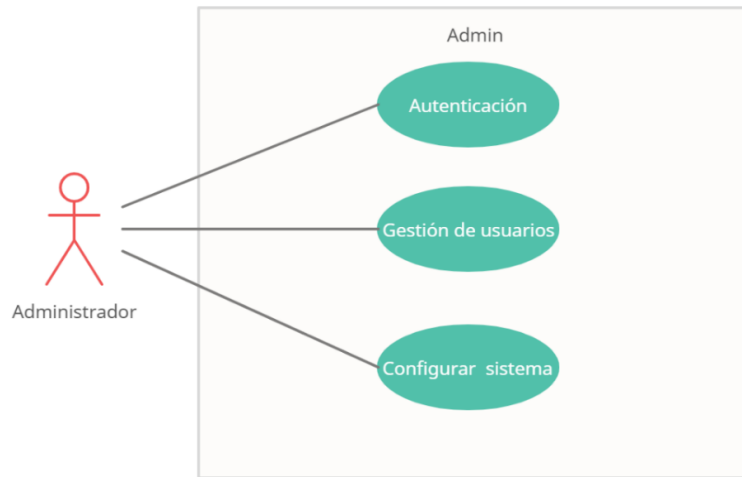
3.1 DIAGRAMAS DE CASOS DE USO

Los siguientes diagramas detallan los actores y funciones con las que interactúa en la plataforma. Existen tres tipos de actores que son: administrador, investigador y alumno. Todos los actores del sistema pueden autenticarse y acceder a las funciones definidas para cada uno.

3.1.1 *Diagrama de caso de uso Administrador*

La gestión de usuarios es una función a la cual solo el administrador puede acceder, esta permite la creación de diferentes usuarios para la plataforma asignando sus diferentes permisos y roles; pueden ser administrador, alumno e investigador. La función de configuración del sistema permite gestionar niveles, periodos y carreras nuevas para la plataforma.

Figura 3. *Caso de uso Administrador*

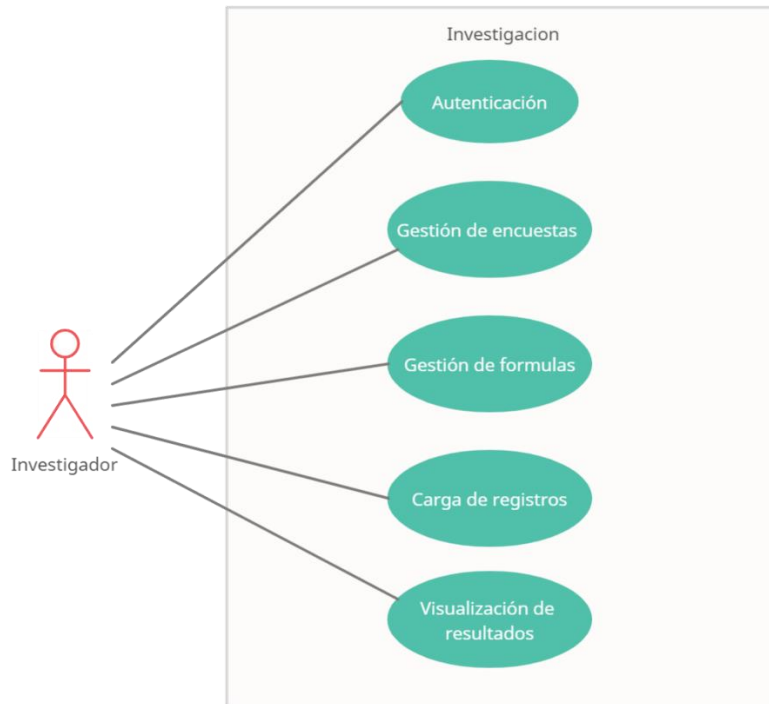


Nota: Flujo de pantallas a las que el actor Administrador accede. Elaborado por: los autores.

3.1.2 Diagrama de caso de uso Investigador

Mediante la gestión de encuestas el investigador obtiene la capacidad para generar nuevas encuestas con sus respectivas preguntas y opciones de respuesta. La gestión de fórmulas permite que el investigador asigne valores numéricos asociados a diferentes carreras, para aplicarlos a las respuestas de las encuestas. Además, puede cargar registros mediante archivos de tipo csv y permiten agregar alumnos y notas de los alumnos a la plataforma. Finalmente, puede visualizar gráficos en base a las respuestas de las encuestas.

Figura 4. Caso de uso Investigador

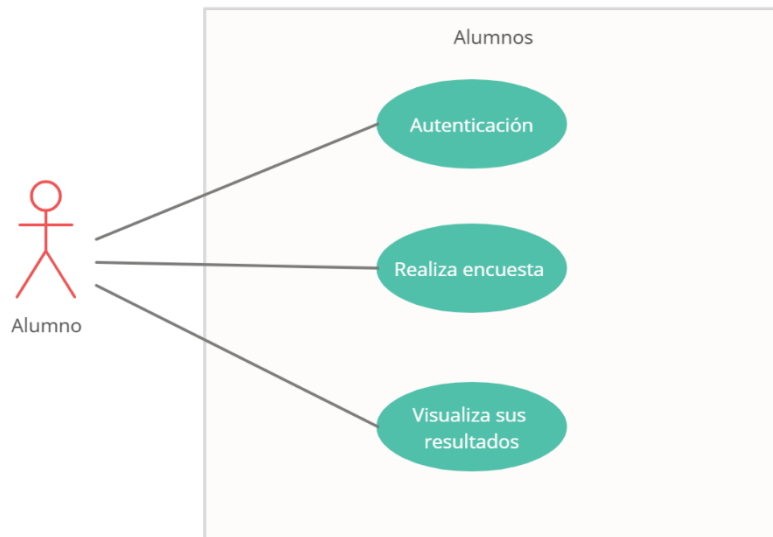


Nota: Flujo de pantallas a las que el actor Investigador accede. Elaborado por: los autores.

3.1.3 Diagrama de caso de uso Alumno

El alumno tiene acceso a visualizar la encuesta y dar respuesta a las preguntas planteadas. Al finalizar la encuesta, se presenta una pantalla donde puede ver su puntaje para cada una de las carreras.

Figura 5. Caso de uso Alumno



Nota: Flujo de pantallas a las que el actor Alumno accede. Elaborado por: los autores.

3.2 DIAGRAMA DE BASE DE DATOS

A continuación, se presenta el diagrama de la base de datos junto con el diccionario de datos que contiene una explicación detallada de cada una de las tablas, como de sus campos. La base de datos diseñada contiene un total de 22 tablas, en las cuales se ignoran tablas relacionadas a autenticación o manejo de usuarios, esto debido a que el framework de desarrollo proporciona estas funcionalidades.

Las principales tablas que componen la base de datos son: formula, encuesta, alumno y detalle. De estas tablas se derivan las siguientes, tres tablas para fórmula, siete tablas para encuesta, cinco tablas para alumno y dos para detalle.

La sección de las encuestas (tablas de color verde), hace referencia a todas las encuestas que se pueden cargar dentro del sistema, este diseño ha sido considerado para que se pueda cargar cualquier tipo de encuesta con cualquier tipo de pregunta.

Cada encuesta tiene sus preguntas las cuales están categorizadas. Por otra parte, cada pregunta tiene diferentes tipos como: si/no, opción múltiple y abierta. Cada tipo de pregunta tiene diferentes opciones las cuales contienen todas las posibles respuestas del usuario.

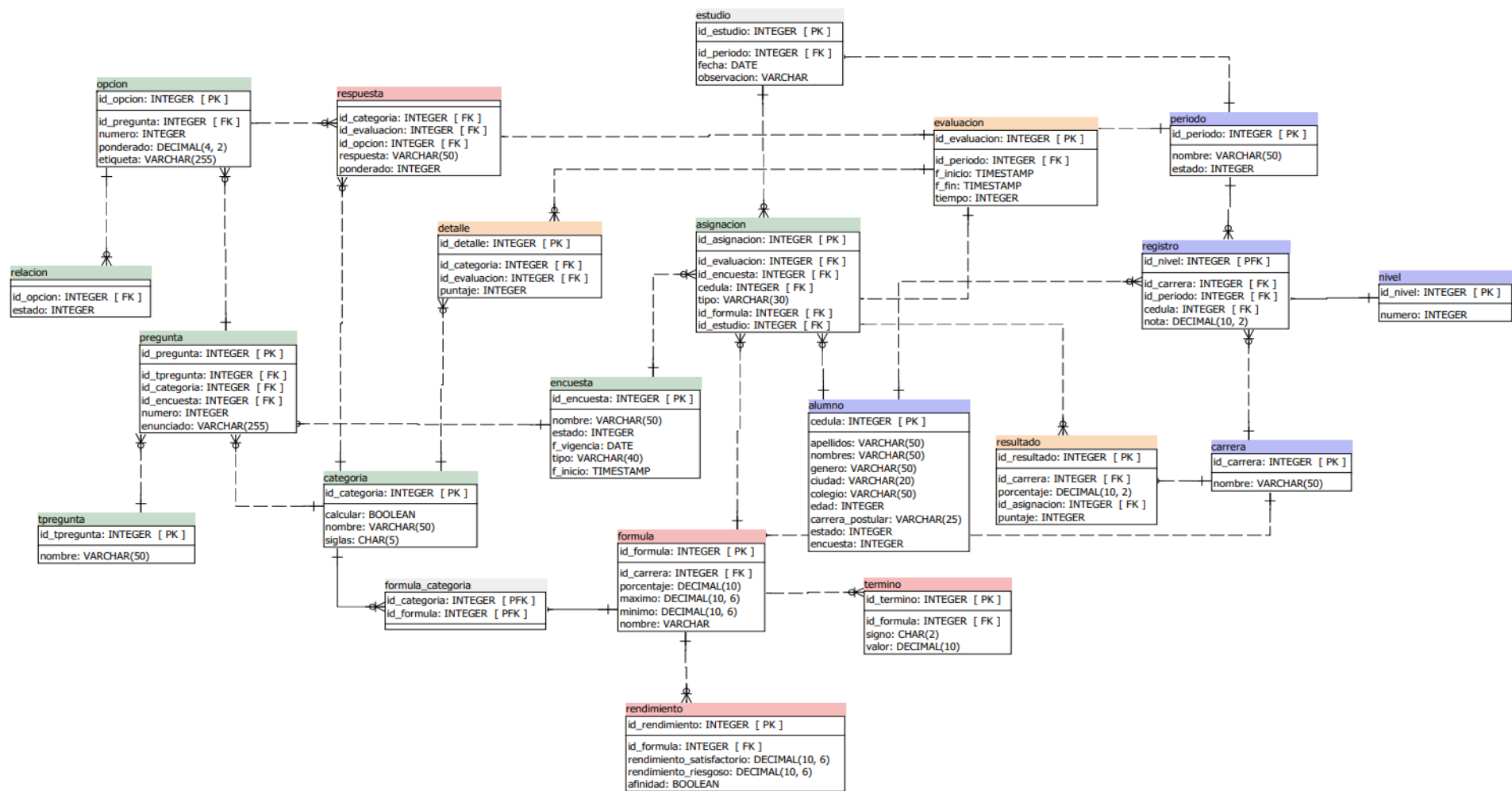
La sección del alumno (tablas de color morado), hace referencia a los datos del alumno la cual está relacionado con un registro general el cual contendrá datos como el periodo, nivel y carrera.

La sección de las fórmulas (tablas de color rojo), hace referencia al ingreso detallado de las fórmulas con sus respectivos términos, dando como resultado un rendimiento riesgoso o satisfactorio, en la sección ANÁLISIS DEL PROCESO ACTUAL se detalla cada uno de los campos más importantes de esta sección.

La sección del detalle (tablas de color amarillo crema), se hace referencia a las características de la evaluación la cual contiene las respectivas fechas de inicio, fin y tiempo para realizar. Por otro lado, la tabla resultada contendrá el porcentaje de la afinidad de estudiante a las carreras con su respectiva asignación.

El siguiente grafico presenta la estructura de la base de datos de la plataforma:

Figura 6. Diagrama de base de datos



Nota: Tablas que contiene la plataforma elaborada. Elaborado por: los autores.

3.2.1 DICCIONARIO DE DATOS

En esta sección se dará a conocer el diccionario de datos de la base establecida, describiendo cada una de sus tablas y campos.

Tabla 22 Detalle tabla encuesta

Tabla Encuesta	Esta tabla agrupa preguntas con sus respectivas opciones y categorías			
Campo	Tipo	PK	FK	Decripción
id_encuesta	integer	x		Campo identificador de la tabla
nombre	varchar			Campo para la asignación de un nombre a la encuesta
estado	integer			Campo que permite manejar la eliminación lógica de la encuesta
f_vigencia	date			Campo que permite asignar una fecha desde la cual la encuesta está disponible
tipo	varchar			Campo que permite asignar el nombre de una categoría a la encuesta
f_inicio	date			Campo para manejar la creación de la encuesta

Tabla 23 Detalle tabla alumno

Tabla Alumno	Esta tabla contiene los datos de alumnos			
Campo	Tipo	PK	FK	Decripción
cedula	integer	x		Campo identificador de la tabla
apellidos	varchar			Campo para la asignación de apellidos del alumno
nombres	varchar			Campo para la asignación de nombres del alumno
genero	varchar			Campo para la asignación de un género del alumno
ciudad	varchar			Campo que permite asignar el nombre de una categoría a la encuesta
edad	date			Campo para manejar la creación de la encuesta
carrera_postula	varchar			Campo para el manejo de la carrera de preferencia del alumno
estado	Integer			Campo para verificar si alumno esta activo , inactivo o en curso.
Encuesta	Integer			Campo para verificar si el estudiante ya realizo la encuesta.

Tabla 24 Detalle tabla categoria

Tabla Categoria	Esta tabla contiene la categoria a la cual pertenece una pregunta			
Campo	Tipo	PK	FK	Decripción
id_categoria	integer	x		Campo identificador de la tabla
calcular	boolean			Campo que permite controlar las categorías en las que se debe aplicar una fórmula
nombre	varchar			Campo que permite asignar un nombre a la categoría
siglas	char			Campo que permite asignar siglas a la categoría

Tabla 25 Detalle tabla pregunta

Tabla Pregunta	Esta tabla contiene las preguntas			
Campo	Tipo	PK	FK	Descripción
id_pregunta	integer	x		Campo identificador de la tabla
id_tpregunta	integer		x	Campo que agrupa la pregunta por tipos de pregunta
id_categoria	integer		x	Campo que agrupa la pregunta por categorías
id_encuesta	integer		x	Campo que agrupa la pregunta por encuesta
numero	integer			Campo que permite asignar un número a la pregunta
enunciado	varchar			Campo que permite asignar el texto de la pregunta.

Tabla 26 Detalle tabla opción

Tabla Opción	Esta tabla contiene las opciones que se asignan a cada pregunta			
Campo	Tipo	PK	FK	Descripción
id_opcion	integer	x		Campo identificador de la tabla
id_pregunta	integer		x	Campo que agrupa la opción por pregunta
numero	integer			Campo que permite asignar un número a la opción
ponderado	integer			Campo que permite asignar un valor a la opción
etiqueta	integer			Campo que permite asignar un texto a la opción

Tabla 27 Detalle tabla fórmula

Tabla Fórmula	Esta tabla contiene las fórmulas de cada una de las carreras			
Campo	Tipo	PK	FK	Descripción
id_formula	integer	x		Campo identificador de la tabla
id_carrera	integer		x	Campo que relaciona las fórmulas con sus respectivas carreras
porcentaje	decimal		x	Campo que agrupa la opción por pregunta
máximo	decimal			Campo que permite asignar un número a la opción
mínimo	decimal			Campo que permite asignar un valor a la opción
nombre	varchar			Campo que permite asignar un texto a la opción

Tabla 28 Detalle tabla respuesta

Tabla Respuesta	Esta tabla contiene las respuestas dadas por el alumno			
Campo	Tipo	PK	FK	Descripción
id_categoria	integer		x	Campo que relaciona las respuestas con las categorías a las que pertenecen
id_evaluación	integer		x	Campo que relaciona las respuestas con la evaluación
id_opción	integer		x	Campo que relaciona las respuestas con las opciones a las que pertenecen
respuesta	varchar			Campo que almacena la respuesta dada por el alumno
Ponderado	Integer			Campo que almacena el ponderado de cada opción.

Tabla 29 Detalle tabla relación

Tabla Relación	Esta tabla contiene las respuestas dadas por el alumno			
Campo	Tipo	PK	FK	Descripción
id_opción	integer		x	Campo que agrupa la relación con las opciones
estado	varchar			Campo que controla la eliminación lógica de la relación

Tabla 30 Detalle tabla tpregunta

Tabla Tpregunta	Esta tabla contiene los tipos de pregunta para agrupar las preguntas			
Campo	Tipo	PK	FK	Descripción
id_tpregunta	integer	x		Campo identificador de la tabla
nombre	varchar		x	Campo que permite asignar un texto al tipo de pregunta

Tabla 31 Detalle tabla detalle

Tabla Detalle	Esta tabla contiene el puntaje obtenido en la encuesta			
Campo	Tipo	PK	FK	Descripción
id_detalle	integer	x		Campo identificador de la tabla
id_categoria	integer		x	Campo que relaciona el detalle con la categoría
id_evaluacion	integer		x	Campo que relaciona el detalle con la evaluación a la que pertenece
puntaje	integer			Campo que almacena el puntaje obtenido en la evaluación

Tabla 32 Detalle tabla formula_categoria

Tabla Formula_categoria	Esta tabla contiene la relación entre formulas y categorías.			
Campo	Tipo	PK	FK	Descripción
id_categoria	integer	x		Campo identificador de la tabla
id_formula	integer		x	Campo identificador de la tabla

Tabla 33 Detalle tabla rendimiento

Tabla Rendimiento	Esta tabla contiene campos de rendimiento correspondientes a la fórmula			
Campo	Tipo	PK	FK	Descripción
id_rendimiento	integer	x		Campo identificador de la tabla
id_formula	integer		x	Campo que relaciona el rendimiento con la fórmula a la que pertenece
rendimiento_satisfactorio	integer			Campo que permite realizar el cálculo del rendimiento del alumno
rendimiento_riesgoso	integer			Campo que permite realizar el cálculo del rendimiento del alumno
afinidad	boolean			Campo que permite controlar la afinidad del alumno

Tabla 34 Detalle tabla término

Tabla Término	Esta tabla contiene campos términos de los cuales se compone la fórmula			
Campo	Tipo	PK	FK	Descripción
id_termino	integer	x		Campo identificador de la tabla
id_formula	integer		x	Campo que relaciona los términos con la fórmula a la que pertenecen
signo	char			Campo que permite asignar un símbolo - o + al término
valor	decimal			Campo que permite asignar un valor numérico para la fórmula

Tabla 35 Detalle tabla asignación

Tabla Asignación	Esta tabla contiene el alumno y su evaluación			
Campo	Tipo	PK	FK	Descripción
id_asignacion	integer	x		Campo identificador de la tabla
id_evaluacion	integer		x	Campo que relaciona la asignación con la evaluación
id_encuesta	integer		x	Campo relaciona la asignación con la encuesta a la que pertenece
cedula	integer		x	Campo relaciona la asignación con el alumno
id_formula	integer		x	Campo que permite aplicar las fórmulas en la encuesta asignada
id_estudio	integer		x	Campo que agrupa las asignaciones con los estudios que realiza el Grupo GIE

Tabla 36 Detalle tabla estudio

Tabla Estudio	Esta tabla contiene los datos generales de la aplicación de la encuesta a los alumnos			
Campo	Tipo	PK	FK	Descripción
id_estudio	integer	x		Campo identificador de la tabla
id_periodo	integer		x	Campo que relaciona el estudio con el periodo en el que se realiza
fecha	integer			Campo que permite asignar una fecha en la que se empieza el estudio
observación	integer			Campo permite agregar un texto según se requiera

Tabla 37 Detalle tabla evaluación

Tabla Evaluación	Esta tabla contiene los datos generales de configuración de la evaluación			
Campo	Tipo	PK	FK	Descripción
id_evaluación	integer	x		Campo identificador de la tabla
id_periodo	integer		x	Campo que relaciona el estudio con el periodo en el que se realiza
f_inicio	Date			Campo que permite asignar una fecha en la que se empieza el estudio
f_fin	Date			Campo permite agregar un texto según se requiera
tiempo	integer			Campo que contiene el tiempo máximo e el cual se puede llenar la encuesta

Tabla 38 Detalle tabla resultado

Tabla Resultado	Esta tabla contiene el resultado general de un alumno			
Campo	Tipo	PK	FK	Decripción
id_resultado	integer	x		Campo identificador de la tabla
id_carrera	integer		x	Campo que relaciona el resultado con la carrera a la que pertenece
id_asignacion	Date			Campo que contiene la asignación a la que pertenece el resultado
porcentaje	integer			Campo que contiene el porcentaje de rendimiento para una carrera
puntaje	integer			Campo que contiene el puntaje final de los resultados.

Tabla 39 Detalle tabla periodo

Tabla Periodo	Esta tabla contiene datos generales del periodo			
Campo	Tipo	PK	FK	Decripción
id_periodo	integer	x		Campo identificador de la tabla
nombre	varchar		x	Campo que permite asignar un nombre al periodo
estado	integer			Campo que controla en qué periodo se encuentra

Tabla 40 Detalle tabla registro

Tabla Registro	Esta tabla contiene notas de alumnos en varios periodos			
Campo	Tipo	PK	FK	Decripción
id_nivel	integer	x		Campo identificador de la tabla
id_carrera	varchar		x	Campo que relaciona el registro con la carrera a la que pertenece
id_periodo	integer		x	Campo que relaciona el registro con el periodo al que pertenece
cedula	integer		x	Campo que relaciona el registro con el alumno al que pertenece
nota	decimal			Campo que contiene la nota alcanzada por el alumno

Tabla 41 Detalle tabla nivel

Tabla Nivel	Esta tabla contiene datos generales del nivel			
Campo	Tipo	PK	FK	Decripción
id_nivel	integer	x		Campo identificador de la tabla
numero	integer			Campo que contiene el número del nivel

Tabla 42 Detalle tabla carrera

Tabla Carrera	Esta tabla contiene datos generales de la carrera			
Campo	Tipo	PK	FK	Decripción
id_carrera	integer	x		Campo identificador de la tabla
nombre	integer			Campo que permite asignar un nombre a la carrera

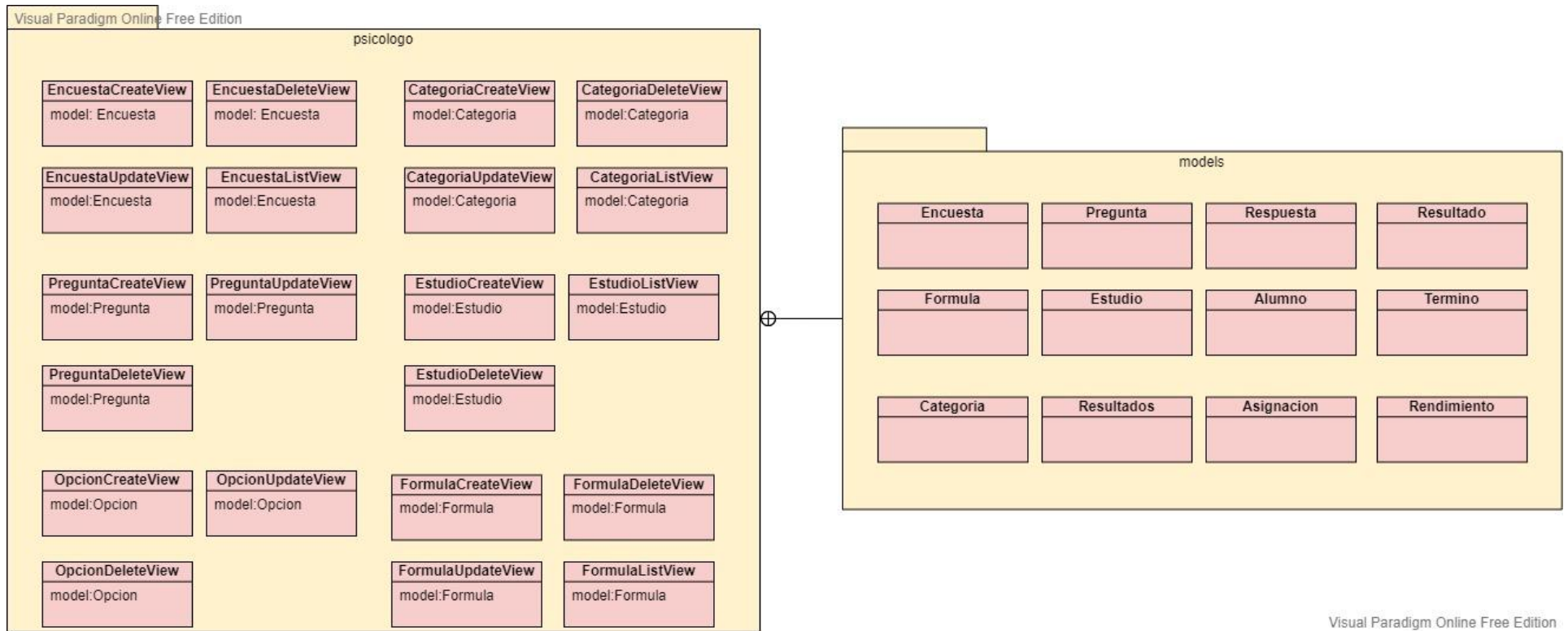
3.3 DIAGRAMA DE CLASES

El siguiente diagrama muestra cada una de las clases que contendrá el proyecto, con sus respectivos atributos que describen cada instancia de la clase y métodos que permitirán especificar el comportamiento de cada clase.

En Django, las vistas abstraen varios métodos de control del *crud* de una aplicación. En el proyecto se trabajó con Class Based Views de Django, por lo que no se necesitó implementar métodos para manejar la creación, actualización y eliminación de registros en la base de datos. A continuación, se muestra el diagrama de clases, en el cual se agruparon la vista del psicólogo con las clases que manejan el *crud* y los modelos que son manipulados a través de estas clases.

- **NombreCreateView:** Son las clases que permiten crear un registro en la base de datos mediante la abstracción
- **NombreUpdateView:** Estas clases permiten realizar la actualización de registros en la base de datos
- **NombreDeleteView:** Este tipo de clases permiten eliminar registros de la base de datos.
- **NombreListView:** Estas clases permiten traer una lista de todos los elementos pertenecientes al modelo desde la base de datos.

Figura 1. Diagrama de clases



Visual Paradigm Online Free Edition

Nota: Clases, atributos y métodos generados en Django Framework. Elaborado por: los autores.

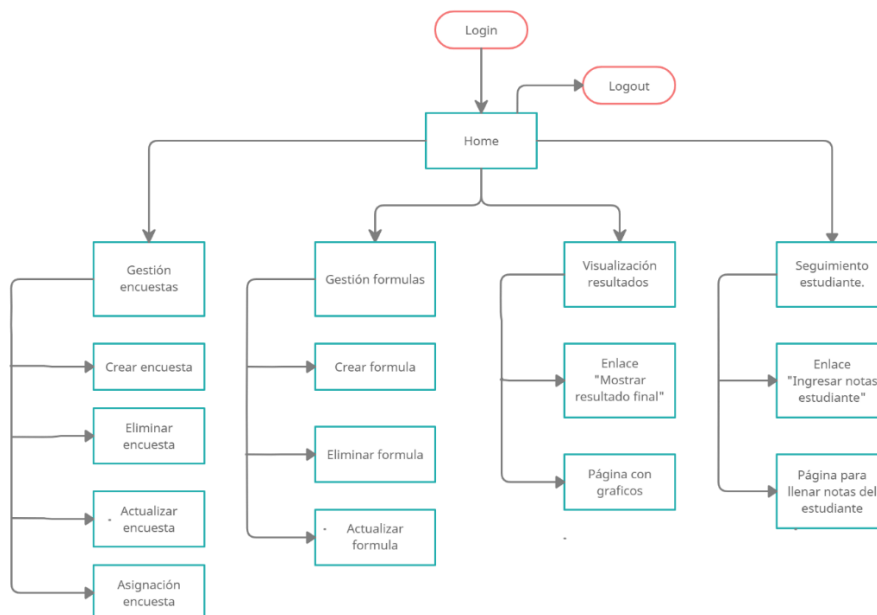
3.4 MAPA DE NAVEGACIÓN

A continuación, se va a presentar los diferentes mapas de navegación para los diferentes actores del sistema.

3.4.1 Diagrama de navegación para Actor Investigador

El actor investigador tiene acceso a gestionar fórmulas, gestionar encuestas, visualizar un resumen de las encuestas aplicadas y agregar notas de los alumnos de diferentes niveles para realizar un seguimiento del mismo.

Figura 2. Diagrama navegacional Investigador

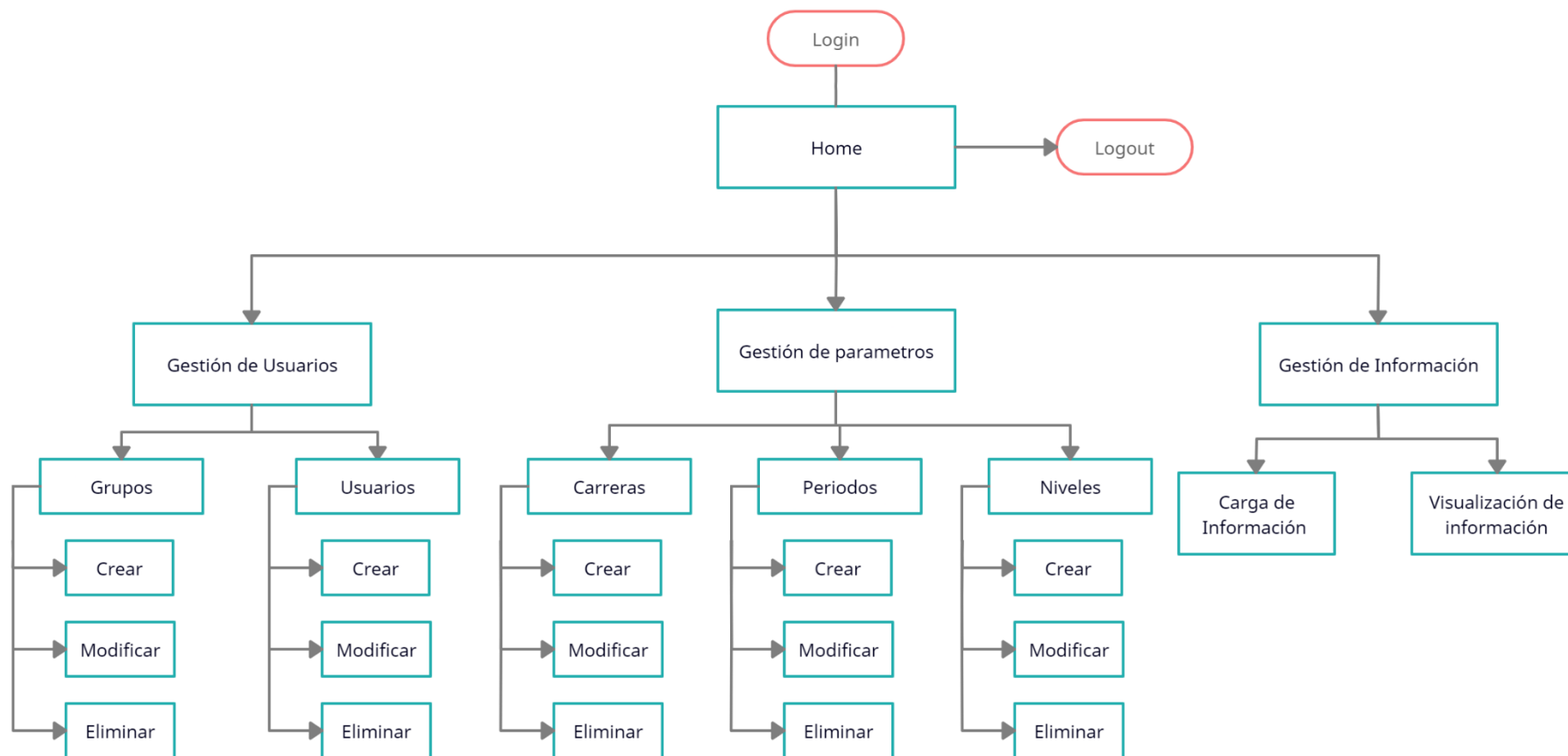


Nota: Funcionalidades a las que tiene acceso el Investigador. Elaborado por: los autores

3.4.2 Diagrama de navegación para Actor Administrador

El administrador de la plataforma tiene acceso a crear, modificar y eliminar usuarios. De la misma forma tiene acceso a crear, modificar y eliminar grupos, además puede agrupar los usuarios y asignarles diferentes permisos.

Figura 3. Diagrama navegacional Administrador

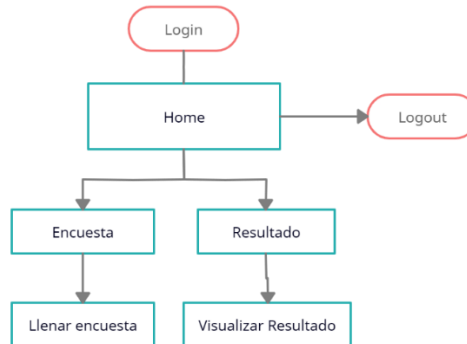


Nota: Funcionalidades a las que tiene acceso el Administrador. Elaborado por: los autores

3.4.3 Diagrama de navegación para Actor Alumno

El alumno tiene acceso a visualizar y llenar la encuesta que le ha sido asignada, además de visualizar resultados al completar la encuesta.

Figura 4. Diagrama navegacional Alumno



Nota: Funcionalidades a las que tiene acceso el Alumno. Elaborado por: los autores.

3.5 DISEÑO INTERFAZ ABSTRACTA

A continuación, se va a presentar los diferentes diseños de interfaz abstracta para los diferentes actores del sistema.

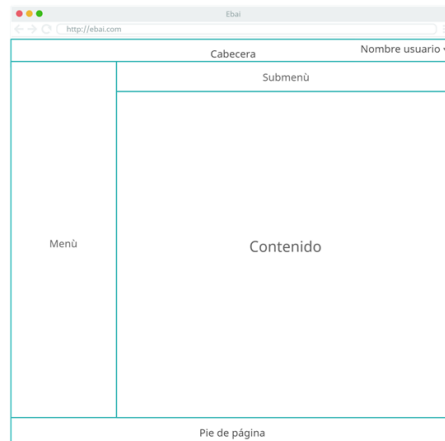
3.5.1. Interfaz abstracta general

Esta interfaz general se divide en diferentes partes como:

- **Cabecera:** la cual contendrá el nombre de la página oficial.
- **Nombre usuario:** la cual contendrá el nombre del actor que ha ingresado.
- **Menú:** en esta parte se mostrará las diferentes funcionalidades principales dependiendo del actor ingresado.
- **Submenú:** en esta parte se habilitará las diferentes funciones relacionadas a la función principal se haya escogido en el menú.

- **Contenido:** esta parte mostrara todo el contenido relacionado con lo seleccionado en el submenú.
- **Pie de página:** esta parte contendrá los nombres de los desarrolladores y el año.

Figura 5. *Interfaz abstracta general*

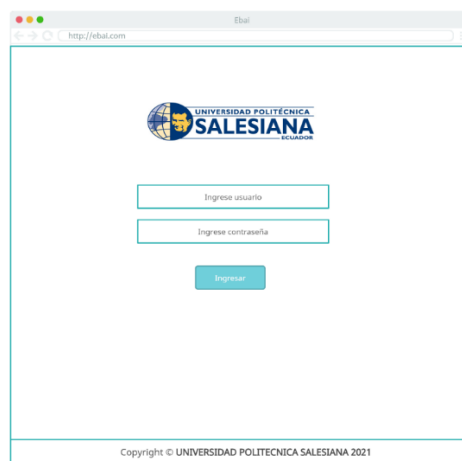


Nota: Distribución de elementos en la plataforma web. Elaborado por: los autores.

3.5.2. *Interfaz abstracta para login*

En esta interfaz el actor deberá colocar su usuario y contraseña que se le haya establecido y seguidamente presionar el botón ingresar el cual lo llevará a la interfaz principal.

Figura 6. *Interfaz abstracta login.*



Nota: Distribución de elementos para la autenticación en la plataforma web. Elaborado por: los autores.

3.5.3. Interfaz abstracta para Actor Investigador

En esta interfaz el investigador tendrá habilitado el menú con sus diferentes funcionalidades principales establecidas, las cuales contendrán diferentes funciones que se habilitarán según lo seleccionado en el menú.

Una vez seleccionado el investigador procederá a llenar el contenido dependiendo de la funcionalidad que haya escogido y seguidamente confirmará la acción mediante un botón.

Figura 7. Interfaz abstracta Investigador para creación de encuestas

The screenshot shows a web browser window with the URL 'http://ebai.com'. The page title is 'Ebai'. The user is logged in as 'Jorge C'. The main navigation menu includes 'Gestión encuestas', 'Gestión formulas', 'Visualización resultados', and 'Seguimiento estudiante'. The 'Gestión encuestas' menu is expanded, showing four options: 'Crear encuesta', 'Eliminar encuesta', 'Actualizar encuesta', and 'Asignación encuesta'. The 'Crear encuesta' option is selected. The main content area contains a form with the following fields: 'Ingrese nombre encuesta' (text input), 'Seleccione fecha vigencia' (dropdown menu), and a 'Crear encuesta' button. The footer of the page reads 'Copyright ©UNIVERSIDAD POLITECNICA SALESIANA 2021'.

Nota: Interfaz para la creación de encuestas. Elaborado por: los autores.

Figura 8. Interfaz abstracta Investigador para la asignación de encuesta

The screenshot shows a web browser window with the URL 'http://eibaie.com'. The page title is 'Ebaie'. The user is logged in as 'Jorge C'. The main navigation menu includes 'Gestión encuestas', 'Gestión formulas', 'Visualización resultados', and 'Seguimiento estudiante'. The 'Gestión encuestas' menu is expanded, showing four options: 'Crear encuesta', 'Eliminar encuesta', 'Actualizar encuesta', and 'Asignación encuesta'. The 'Asignación encuesta' option is selected. The main content area contains a form with the following fields: 'Seleccione Encuesta' (dropdown menu), 'Elegir alumnos' (table), and an 'Asignar encuesta' button. The table has two columns: 'Alumnos-' and 'Asignar-'. The table contains three rows of data: 'Rodrigo Tuffino', 'Daniel Diaz', and 'Holger Ortega'. The 'Asignar-' column has checkboxes for each row, with the first and third rows checked. The footer of the page reads 'Copyright ©UNIVERSIDAD POLITECNICA SALESIANA 2021'.

Alumnos-	Asignar-
Rodrigo Tuffino	<input checked="" type="checkbox"/>
Daniel Diaz	<input type="checkbox"/>
Holger Ortega	<input checked="" type="checkbox"/>

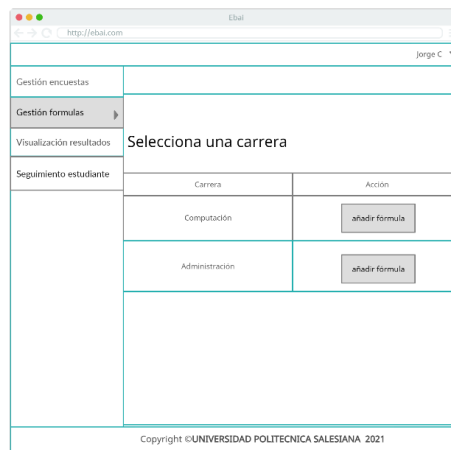
Nota: Interfaz para la asignación de encuestas. Elaborado por: los autores.

Figura 9. *Interfaz abstracta fórmula paso 1.*







Nota: Paso 1 para la creación de fórmulas. Elaborado por: los autores.

Figura 10. *Interfaz abstracta formula paso 2*



Nota: Paso 2 para la creación de fórmulas. Elaborado por: los autores.

Figura 11. *Interfaz abstracta formula paso 3*

Punto de corte	ICAV	Acciones
13.2	$0.118 A - 0.123 B + 1.256 C$	 
13.2	$0.118 A - 0.123 B + 1.256 C$	 

Nota: Paso 3 para la creación de fórmulas. Elaborado por: los autores.

Figura 12. Interfaz abstracta formula paso 4

Agrega una formula

Punto de corte	
<input type="text"/>	
Criterios de Reescalado	
Minimo	Maximo
<input type="text"/>	<input type="text"/>

Rendimientos	
A fin	
Satisfactorio	Riesgoso
<input type="text"/>	<input type="text"/>
No a fin	
Satisfactorio	Riesgoso
<input type="text"/>	<input type="text"/>

Campo profesional	Valor
Ciencias Fisicas	0.325

Campo profesional	Valor
Ciencias Fisicas	0.325

Guardar

Nota: Paso 4 para la creación de fórmulas. Elaborado por: los autores.

Figura 13. Interfaz abstracta del panel de mando



Nota: Interfaz para la visualización de resultados. Elaborado por: los autores.

3.5.4 Interfaz abstracta para Actor Alumno

El actor alumno visualizará una interfaz en la que se le solicitan sus diferentes datos personales. Posteriormente observará la encuesta que se le ha asignado y podrá responder todas las preguntas en una única pantalla. Y finalmente confirmar el envío de la encuesta a través de un botón.

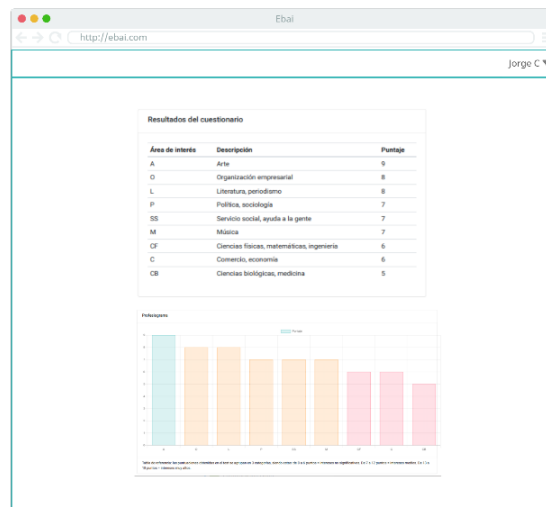
Figura 14. Interfaz abstracta encuesta

The screenshot shows a web browser window titled 'Ebal' with the URL 'http://ebal.com'. The user 'Jorge C.' is logged in. The main content area contains a form with the following sections:

- Datos Generales:** A text input field for 'Nombres, Apellidos, Genero del alumno'.
- Tipo de Cuestionario:** A dropdown menu.
- Questions:** Four 'Pregunta 1' items, each with two radio button options labeled 'Opción 1'.
- Submit:** A blue 'Enviar' button.
- Footer:** Copyright ©Kevin Chacha - Anthony Cabezas 2020.

Nota: Distribución de elementos para la visualización de la encuesta en la plataforma informática. Elaborado por: los autores.

Figura 15. Interfaz abstracta para visualización de resultados finales.



Nota: Interfaz de resultados finales del alumno. Elaborado por: los autores.

4. CONSTRUCCIÓN Y PRUEBAS

En este capítulo se detallan las herramientas y tecnologías que se utilizarán durante el proceso de desarrollo de la plataforma informática. También se presentan fragmentos del código esencial para el funcionamiento del mismo, además se presentan elementos de la metodología utilizada y planificaciones.

Para la construcción del proyecto se utilizará el framework *Django*, que facilita la configuración de la base de datos, además de proveer seguridad y un panel de administración de los modelos creados.

4.1 HERRAMIENTAS Y TECNOLOGÍAS

A continuación, se presentan las diferentes herramientas y tecnologías que permiten a la plataforma web funcionar adecuadamente.

4.1.1 *Python 3.8.8*

Python es un lenguaje de programación de tipo interpretado, multiplataforma, multiparadigma y multipropósito; soporta orientación a objetos, programación imperativa, programación funcional y tipado dinámico (Pythones, 2021). Es de código abierto y posee una licencia aprobada por la Open Source Initiative (OSI), por lo que está disponible y puede ser usado en cualquier proyecto. (Python Org, 2021).

4.1.2 *Django 3.2*

Django es un framework para plataformas web gratuito, de código abierto publicado bajo la licencia Berkley Software Distribución y escrito en Python (Django Girls, 2021). Django

posee diferentes elementos comunes en las plataformas web, que pueden ser utilizados para disminuir el tiempo de desarrollo. Es completamente escalable y proporciona una forma muy efectiva de mantener la seguridad de las plataformas web (Django Project, 2021).

4.1.3 PostgreSQL 13.3

PostgreSQL es un sistema gestor de base de datos relacional de código abierto publicado bajo la licencia PostgreSQL. Utiliza el lenguaje de consultas SQL, permite la orientación a objetos mediante el mapeo objeto-relacional. Brinda confiabilidad, integridad de datos, robustez, extensibilidad y alto rendimiento (PostgreSQL, 2021).

4.2 PRODUCT BACKLOG

La siguiente tabla contiene todas las tareas necesarias para el desarrollo del proyecto. Está ordenada secuencialmente, se han establecido fechas de inicio y tiempos estimados para cada tarea con su respectivo estado y prioridad.

Tabla 43 Product Backlog

Product Backlog					
ID	Tarea	Tiempo estimado (días)	Estado	Fecha de inicio	Prioridad
1	Determinar herramientas de desarrollo	2	Completo	09/04/2021	Baja
2	Configurar el entorno de desarrollo	3	Completo	09/04/2021	Baja
3	Creación de la base de datos en Django	3	Completo	09/04/2021	Alta
4	Acceso de usuarios	10	Completo	23/04/2021	Alta
5	Importación / exportación de usuarios	2	Completo	23/04/2021	Media
6	Importación / exportación de registros	2	Completo	23/04/2021	Media
7	Desarrollo de interfaz alumno	2	Completo	07/05/2021	Alta

8	Desarrollo de interfaz psicologo	2	Completo	07/05/2021	Alta
9	Gestión de encuestas	5	Completo	21/05/2021	Alta
10	Gestión de formulas	5	Completo	21/05/2021	Alta
11	Asignación encuestas	4	Completo	21/05/2021	Media
12	Visualizar y Llenar encuesta	5	Completo	20/08/2021	Alta
13	Cálculo de resultados y fórmulas	5	Completo	26/08/2021	Alta
14	Visualización de resultados del alumno	5	Completo	01/09/2021	Alta
15	Dashboard psicologo	3	Completo	07/09/2021	Alta

Nota: Lista de tareas a elaborar para la construcción de la plataforma. Elaborado por: los autores.

4.3 SPRINTS

Los Sprints están configurados para realizarlos en el periodo de dos semanas, para lo que se estableció un calendario mensual donde se pueden observar las actividades. Al finalizar cada semana se realizó una reunión con el tutor, donde se evaluó el entregable, se definió y planeó el nuevo alcance, además de plantear una retrospectiva del entregable.

La plataforma tiene el nombre “Elise” cuyo código generado, está ubicado en un repositorio de Github perteneciente al tutor del proyecto, el cual es accesible mediante el siguiente enlace:

<https://github.com/rtufino/elise>

4.3.1 *Sprint 1: Configuración del entorno de desarrollo.*

Para el Sprint Backlog 1 se determinaron los siguientes entregables: el entorno de desarrollo y la configuración de la base de datos. Siendo estas PYCHARM como IDE de desarrollo y SQLITE como base de datos para desarrollo.

Tabla 44 Sprint 1

Fechas	Desde: 09/04/2021		Hasta: 16/04/2021		
Días			10/04	11/04	14/04
Tarea	Estado	Responsable	Esfuerzo (hrs)		
Determinar herramientas de desarrollo	Completo	Kevin, Anthony	2		
Configurar el entorno de desarrollo	Completo	Kevin, Anthony	2		
Creación de la base de datos en Django	Completo	Kevin, Anthony	3	3	3

Nota: Tareas del Sprint 1 de la elaboración de la plataforma. Elaborado por: los autores.

Configuración del entorno

Antes de empezar con el desarrollo se planificaron las herramientas y configuraciones con las que se iba a llevar a cabo el proyecto, siendo las que se detallan: *pycharm*, *sqlite* y *brave browser*. Cada una de las mencionadas como: ide de programación, base de datos y entorno de prueba, respectivamente.

Entorno de Django.

Se implementó un entorno virtual con *venv* en el que se instalaron todas las dependencias. De igual forma el archivo de nombre *requirements.txt* contiene las librerías y dependencias que se utilizan en el proyecto. Finalmente se utiliza *sqlite* agilizar el proceso de desarrollo y mantener una consistencia de datos entre los desarrolladores.

4.3.2. Sprint 2: Importación, exportación y autenticación.

El entregable del Sprint Backlog 2 fue la importación, exportación de datos de alumnos y registros de notas mediante el *admin* en *Django*, además de la autenticación al sitio web de la plataforma tanto para investigadores como para alumnos.

Tabla 45 Sprint 2

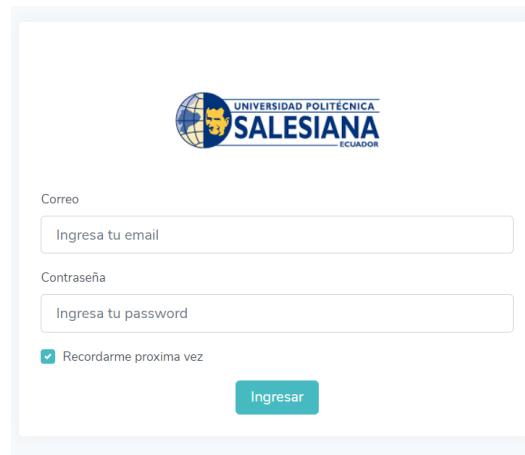
Fechas	Desde: 23/04/2021		Hasta: 30/04/2021		
Días			26/04	28/04	29/04
Tarea	Estado	Responsable	Esfuerzo		
Acceso de usuarios	Completo	Kevin	5	5	5
Importación / exportación de usuarios	Completo	Anthony	4	4	
Importación / exportación de registros	Completo	Anthony		3	2

Nota: Tareas del Sprint 2 de la elaboración de la plataforma. Elaborado por: los autores.

Acceso de usuarios

La plataforma tiene 3 tipos de usuarios, investigador, administrador y alumno. Los usuarios investigador y alumno acceden a la plataforma mediante el formulario creado en HTML, mientras que el usuario administrador accede mediante el formulario que proporciona *Django*.

Figura 16. *Formulario de acceso*



The image shows a login form for the Universidad Politécnica Salesiana Ecuador. At the top center is the university's logo, which consists of a globe and the text 'UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR'. Below the logo, there are two input fields: 'Correo' with the placeholder text 'Ingresa tu email' and 'Contraseña' with the placeholder text 'Ingresa tu password'. Under the password field, there is a checked checkbox labeled 'Recordarme proxima vez'. At the bottom center of the form is a blue button with the text 'Ingresar'.

Nota: Formulario de Autenticación de usuarios en la plataforma. Elaborado por: los autores.

Importación y Exportación de usuarios y registros

La librería *django-import-export* (Read the docs, 2020) de *Django*, permite importar y exportar archivos hacia los modelos, en diferentes formatos desde el *admin* de *Django*. La misma, permitió la configuración de importación y exportación de datos, facilitando su carga

en formato *csv* correspondiente a los alumnos y registros de notas en la base de datos de la plataforma. A continuación, se puede observar la implementación de la librería mediante código de programación:

Figura 17. Clase Resource de Import Export

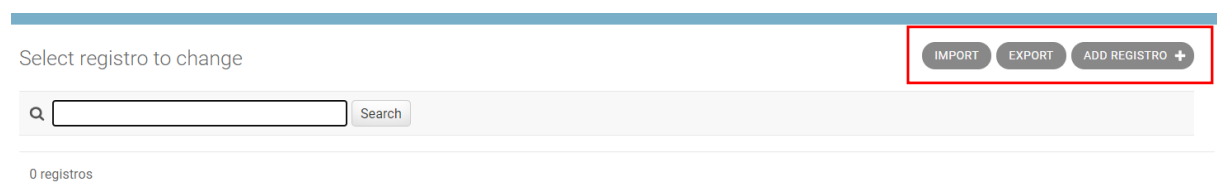
```
class RegistroResource(resources.ModelResource):
    nota = fields.Field(
        column_name='nota',
        attribute='nota'
    )
    class Meta:
        model = Registro
```

Nota: Código que permite implementar import-export. Elaborado por: los autores.

Esta clase hereda de *ModelResource* que contiene el paquete *django-import-export*, para modelos que poseen claves foráneas se utiliza *widgets* de la propia librería, de esta forma se logra importar archivos con nombres en lugar de claves foráneas de la base de datos.

Al finalizar la implementación se puede observar que el admin de Django posee dos botones nuevos para importar y exportar archivos.

Figura 18. Admin de Django con botones nuevos.



Nota: Botones de importación y exportación. Elaborado por: los autores.

4.3.3 Sprint 3: Plantilla e Interfaces de investigador y alumno

Para las tareas del Sprint 3 se definió la plantilla a utilizar, además se empezó la aplicación de la misma en las pantallas de la plataforma contemplando a todos los usuarios.

Tabla 46 Sprint 3

Fechas	Desde: 07/05/2021		Hasta: 14/05/2021		
Días			10/05	12/05	13/05
Tarea	Estado	Responsable	Esfuerzo		
Desarrollo de interfaz alumno	Completo	Kevin	5	5	5
Desarrollo de interfaz psicologo	Completo	Anthony	4	4	5

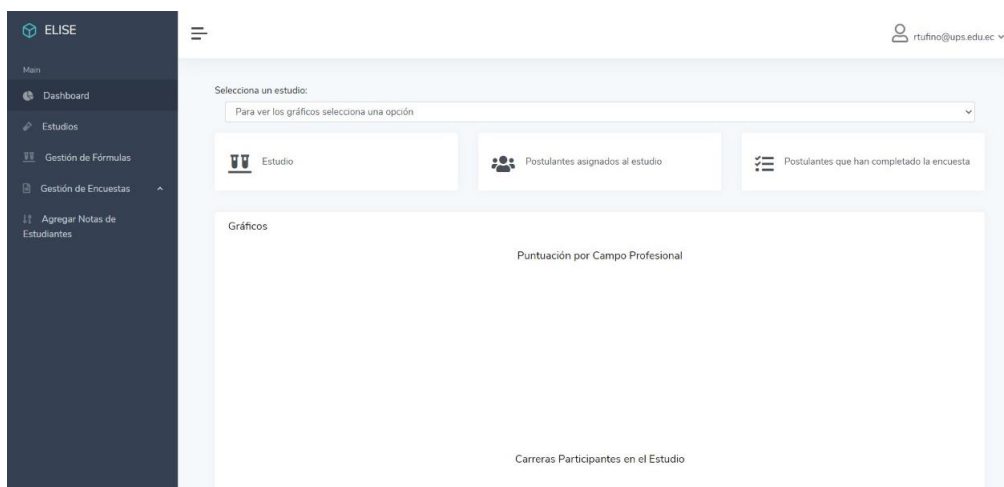
Nota: Tareas del Sprint Backlog 3 de la elaboración de la plataforma. Elaborado por: los autores.

Plantilla para las interfaces

La plantilla utilizada para la plataforma fue *AppStack* que proporciona componentes y un diseño amigable y responsive, tiene entre sus utilidades integrado Bootstrap 4.4.1 y JQuery 3.6.0. Está disponible en este enlace.

A continuación, se muestra la interfaz del investigador la cual contiene un Dashboard en donde se mostrarán los gráficos de los estudios, número postulantes asignados, postulantes que han completado la encuesta, entre otros. Adicionalmente contiene las opciones de estudios en donde se creará un estudio, la gestión de fórmulas en donde se podrá asignar los valores de ICAV, gestión de encuestas en donde se crearán las preguntas, opciones y la agregación de notas de estudiantes en curso.

Figura 19. *Interfaz para el usuario investigador*



Nota: Interfaz web de la plataforma para el investigador. Elaborado por: los autores.

La interfaz siguiente es la del usuario , en la cual únicamente se mostrará la encuesta asignada la cual está distribuida en 5 tipos de preguntas en las cuales consta preguntas de interés, preguntas si /no , preguntas abiertas, preguntas múltiple y preguntas tipo Likert.

Figura 20. Interfaz para el usuario alumno

Nota: Interfaz web de la plataforma para el alumno. Elaborado por: los autores.

4.3.4 Sprint 4: Gestión de encuestas, formulas y asignaciones

El presente Sprint Backlog tuvo como entregables: la creación, eliminación y actualización de encuestas y fórmulas. Complementándose con la asignación de encuestas a los estudiantes registrados, todo mediante la interfaz del investigador.

Tabla 47 Sprint 4

Fechas	Desde: 21/05/2021		Hasta: 04/06/2021		
Días			21/05-25/05	27/05-31/06	01/06-04/06
Tarea	Estado	Responsable	Esfuerzo		
Gestión de encuestas	Completo	Anthony	5	10	5
Gestión de formulas	Completo	Anthony		10	5
Asignación encuestas	Completo	Anthony		5	5

Nota: Tareas del Sprint 4 de la elaboración de la plataforma. Elaborado por: los autores.

Gestión de Encuestas

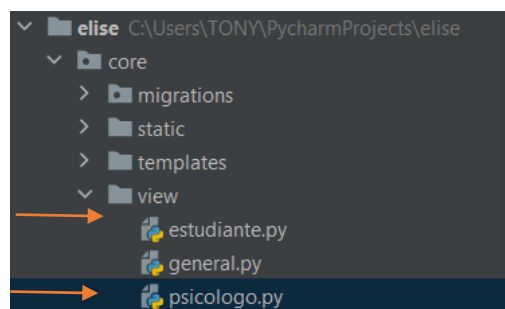
Para crear, listar y actualizar registros mediante formularios HTML, se utilizaron Class Based Views de Django, todas estas clases están en el archivo de vistas de cada usuario. Este tipo de vistas permite una abstracción muy simple de los modelos creados, únicamente necesita asociarse a un modelo para abstraer los objetos del mismo. Para nombrar vistas basadas en clases dentro de este proyecto, se mantiene el siguiente estándar:

Figura 21.

```
class NombredelModeloTipodeVista():
```

Nota: Estándar del nombre de las clases en el proyecto. Elaborado por: los autores.

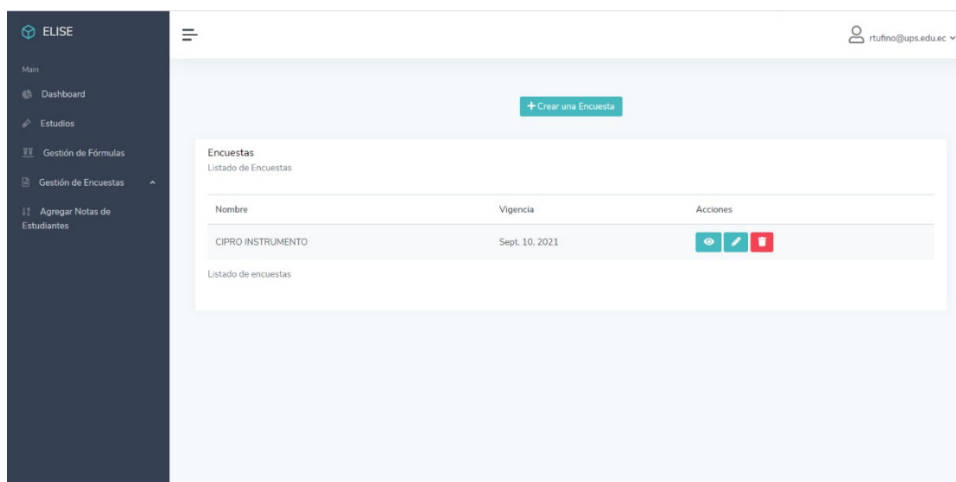
Figura 22. Estructura proyecto



Nota: Vistas de los actores. Elaborado por: los autores.

La interfaz de gestión de encuestas se compone de la creación de preguntas y opciones para la misma, así como también, su respectiva modificación y eliminación. Tanto las preguntas como las opciones tienen acceso desde sus respectivos componentes padres, es decir la pregunta solo es accesible entrando a la encuesta y la opción solo es accesible entrando a la pregunta.

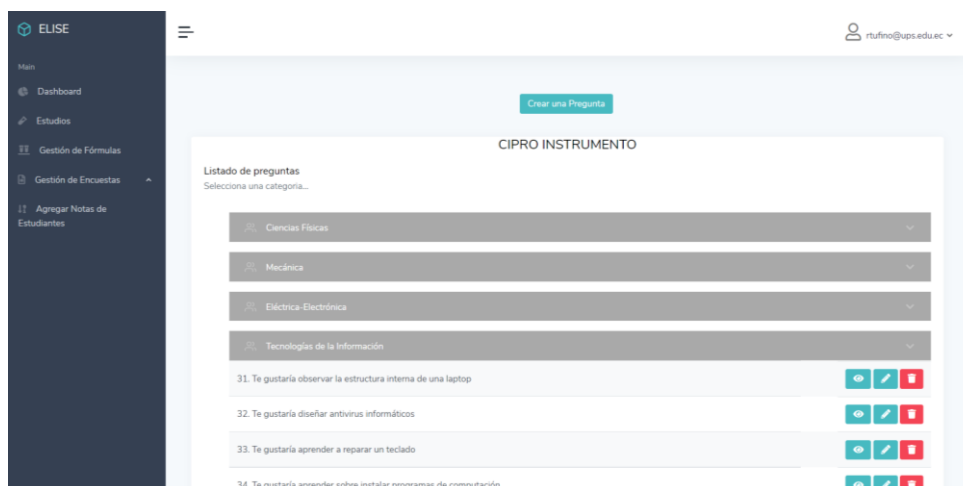
Figura 23. Gestión de encuestas.



Nota: Interfaz web de la plataforma para la creación eliminación y edición de encuestas.
Elaborado por: los autores.

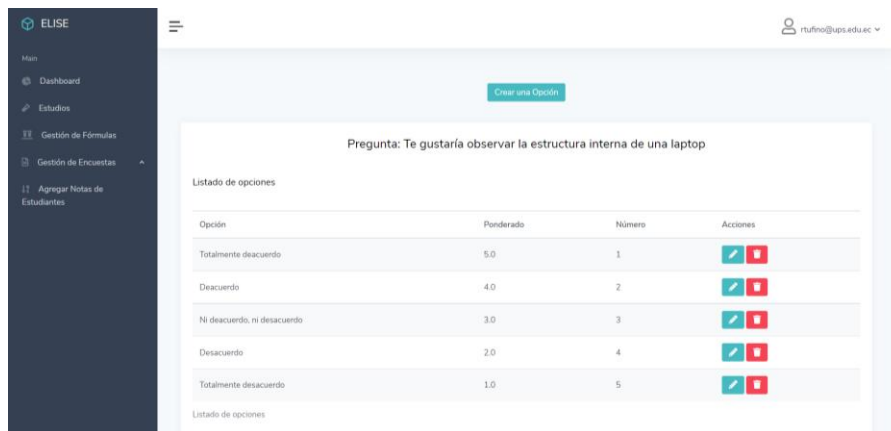
La interfaz de gestión de preguntas nos listará primeramente todas las preguntas creadas en las cuales se tendrá acceso al llenado de sus respectivas opciones.

Figura 24. Gestión de preguntas



Nota: Interfaz web de la plataforma para la creación eliminación y edición de preguntas.
Elaborado por: los autores.

Figura 25. Gestión de opciones



Nota: Interfaz web de la plataforma para la creación eliminación y edición de preguntas.
Elaborado por: los autores.

Cabe mencionar que se mantiene un diseño similar para agregar modificar y eliminar registros de todos los modelos, de esta forma también se logra una interfaz amigable, ya que todas las pantallas siguen el mismo flujo. A continuación, se muestran las pantallas de modificación y eliminación de un registro:

Figura 26. Modificación de registros

Esta opción pertenece a la pregunta: "Te gustaría observar la estructura interna de una laptop"

Edita los parametros de la opción

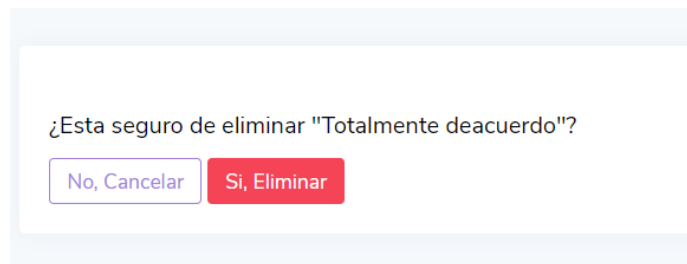
Numero:

Ponderado:

Etiqueta:

Nota: Interfaz para la edición de opciones. Elaborado por: los autores.

Figura 27. Interfaz de eliminación de registros



Nota: Interfaz para la eliminación de opciones. Elaborado por: los autores.

Gestión de fórmulas

Para agregar varios formularios de un modelo, en Django, se utilizó Formsets. Los Formsets permiten configurar una clase padre y una hija, además de permitir seleccionar el número de formularios que queremos que se adicionen. Para este caso el modelo *Formula*, actúa como clase padre de los modelos *Termino* y *Rendimiento*, ya que estos últimos dependen del modelo *Formula*. En el siguiente gráfico podemos observar el código que se utilizó para conseguir esto:

Figura 28. Inline Formsets de Django

```
RendimientoFormset = inlineformset_factory(  
    Formula, Rendimiento, fields=('__all__'), max_num=2, extra=2  
)  
  
TerminoFormset = inlineformset_factory(  
    Formula, Termino, fields=('__all__'), extra=1  
)
```

Nota: Código en Python que permite agregar formularios. Elaborado por: los autores.

Para la creación de coeficientes de las fórmulas se generó un componente dinámico mediante JavaScript, que permite agregar más de un formulario y configurar una fórmula con varios coeficientes. A continuación, se presenta el código:

Figura 29. Seudocódigo para agregar formularios

```
let terminoForm = document.querySelectorAll(".inline-form")
let container = document.querySelector("#form-container")
let addButton = document.querySelector("#add-form")
let totalForms = document.querySelector("#id_termino_set-TOTAL_FORMS")
let formNum = terminoForm.length - 1
addButton.addEventListener('click', addForm)











function addForm(e) {
  e.preventDefault()
  let newForm = terminoForm[0].cloneNode(true)
  let formRegex = RegExp(`termino_set-(\\d){1}-`, 'g')

  formNum++
  newForm.innerHTML = newForm.innerHTML.replace(formRegex,
`termino_set-${formNum}-`)
  container.insertBefore(newForm, addButton)

  totalForms.setAttribute('value', `${formNum + 1}`)
}
```

Nota: Código en JavaScript que permite agregar formularios dinámicamente. Elaborado por: los autores.

Figura 30. Gestión de fórmulas.

Carrera	Punto de corte	ICAV	Acciones
COMPUTACIÓN	71.63	+ 0.194 TI -0.14 A -0.149 CB -0.212 ED -0.156 CE -0.266 AM -0.22 SS	 
MECATRÓNICA	63.65	+ 0.003 M + 0.423 CF + 0.096 EE + 0.049 TI -0.03 A -0.054 CB -0.064 ED -0.067 CS -0.224 AM -0.086 SS	 
ELECTRICIDAD	60.67	+ 0.049 EE + 0.094 CF + 0.094 TI -0.131 A -0.014 CB -0.006 ED -0.029 CS -0.149 AM -0.104 SS	 
EDUCACIÓN BÁSICA	49.48	+ 0.022 ED + 0.31 SS -0.283 CF -0.341 TI -0.399 IC -0.463 CE -0.424 CS	 
INGENIERÍA AMBIENTAL	56.94	+ 0.277 CB + 0.056 CE -0.278 M -0.48 TI -0.444 A -0.175 CS -0.492 ED -0.265 AM	 

Nota: Interfaz web de la plataforma para la creación eliminación y edición de fórmulas. Elaborado por: los autores.

La creación de fórmulas es una de las partes fundamentales del proyecto ya que es necesario configurar fórmulas con diferente cantidad de términos. La interfaz de creación y actualización se presentan en la siguiente figura:

Figura 31. Creación y actualización de fórmulas.

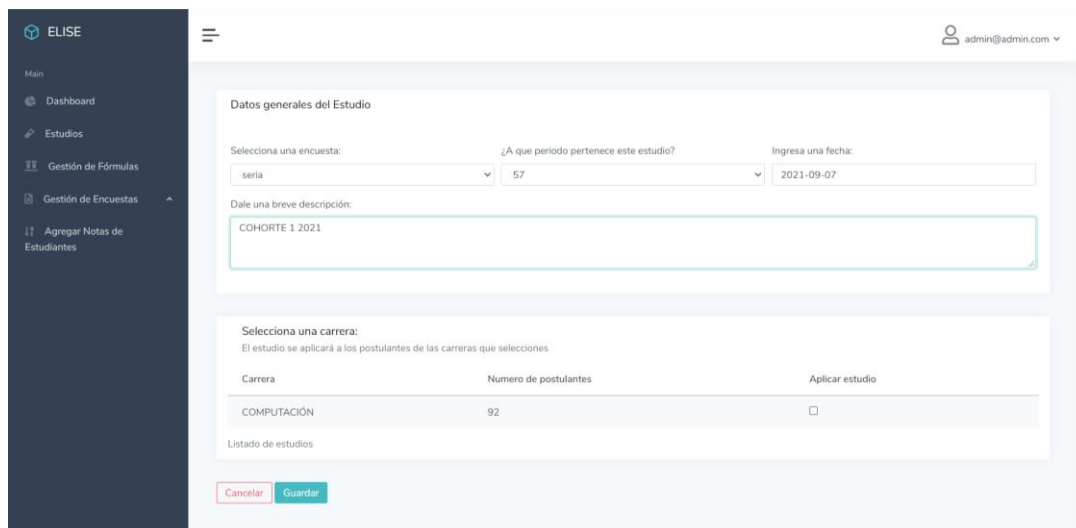
The screenshot displays a web interface for managing formulas, organized into three main sections:

- Datos generales de la Formula:** This section includes a dropdown menu for "¿A que carrera pertenece esta formula?", a text input for "Punto de corte", two text inputs for "Máximo" and "Mínimo" under "Criterios de reescalado", and a text input for "Asigne un nombre a esta formula".
- Rendimientos para la formula:** This section contains two identical pairs of text inputs for "Satisfactorio" and "Riesgoso", each with a checkbox labeled "¿Es a fin?" below it.
- Término:** This section includes a text input for "Valor", a dropdown for "Selecciona un campo profesional", and a checkbox for "Eliminar este termino".

Nota: Interfaz web de la plataforma para la creación y edición de fórmulas. Elaborado por: los autores.

Como se puede observar la interfaz contiene tres apartados, donde se configuran la fórmula, los rendimientos de la fórmula y los términos. Cada uno de los mencionados puede crearse individualmente, es decir, no es necesario que todos los apartados estén llenos.

Figura 32. Asignación de encuestas



Nota: Interfaz web de la plataforma para la creación de estudios y asignación de encuestas. Elaborado por: los autores.

4.3.5 Sprint 5: Visualizar y Llenar encuesta

Tabla 48 Sprint 5

Fechas	Desde: 30/08/2021		Hasta: 06/09/2021		
Días			1/09	5/09	6/09
Tarea	Estado	Responsable	Esfuerzo (hrs)		
Visualizar y Llenar encuesta	Completo	Kevin	6	14	8
Dashboard psicologo	Completo	Anthony	6	7	8

Nota: Tareas del Sprint Backlog 5 de la elaboración de la plataforma. Elaborado por: los autores.

Visualizar y Llenar encuesta

Para la visualización de la encuestas se capturo los datos del alumno , la información de la encuestas y su respectiva asignación ,con el fin de filtrar las preguntas y opciones de la encuesta asignada a dicho estudiante.

La interfaz para la visualización y llenado de la encuesta está distribuida en 5 tipos de preguntas en las cuales consta preguntas de interés, preguntas si /no, preguntas abiertas, preguntas múltiple y preguntas tipo Likert, las cuales contienen sus propias opciones asignadas.

Figura 33. Interfaz para el llenado de la encuesta.

CIPRO INSTRUMENTO

1. Te gustaría conocer la dirección de la fuerza de gravedad

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo, ni desacuerdo
- Desacuerdo
- Totalmente desacuerdo

2. Te gustaría trabajar matemáticamente en sistemas de gravedad

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo, ni desacuerdo
- Desacuerdo
- Totalmente desacuerdo

3. Te gustaría aplicar principios vectoriales para determinar el comportamiento del electrón

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo, ni desacuerdo
- Desacuerdo
- Totalmente desacuerdo

4. Te gustaría determinar la proyección de proyectiles en sentido parabólicos

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo, ni desacuerdo
- Desacuerdo
- Totalmente desacuerdo

5. Te gustaría entender el equilibrio de fuerzas

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo, ni desacuerdo

6. Te gustaría calcular el punto de apoyo para elevar pesos con palancas

- Totalmente de acuerdo
- De acuerdo
- Ni de acuerdo, ni desacuerdo

Nota: Interfaz para el llenado de la encuesta con sus diferentes tipos de preguntas. Elaborado por: los autores.

Al enviarse la encuesta primero se verifica que todas opciones de las preguntas estén completadas para esto se utilizó un atributo booleano de HTML llamado “required” el cual especifica que debe seleccionar el campo antes de enviar el formulario.

Figura 34. Atributo required para las opciones de cada pregunta.

```
<label class="custom-control custom-radio">  
  <input class="custom-control-input" type="radio" value="{i.ponderado}"  
    name="{i.pregunta.id}" id="{i.id}" required>  
  <span class="custom-control-label">{i}</span>  
</label>
```

Nota: Atributo required utilizado para la validación de una encuesta completada . Elaborado por: los autores.

Finalmente, el Alumno contiene tres tipos de estados:

- 0 Estudiante inactivo
- 1 Estudiante postulante
- 2 Estudiante en carrera

Los estudiantes con estado 0 y 2 no podrán acceder a la encuesta ni a la visualización de resultados. Por otra parte, el estudiante con estado 1 podrá acceder a realizar la encuesta y en

el caso de que ya lo haya realizado el programa le redireccionará a la interfaz de los resultados.

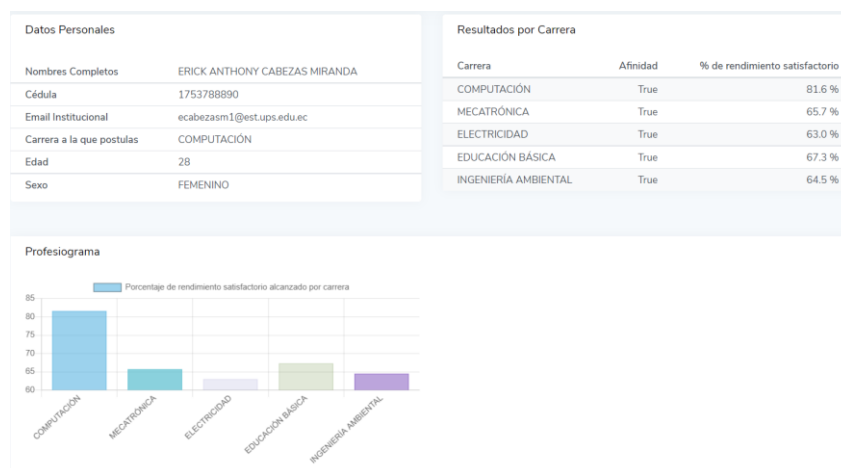
Figura 35. Mensaje de error.



Nota: Mensaje de error para estudiantes que se encuentre en estado 0 o 2 . Elaborado por: los autores.

Los resultados para el alumno se presentan en tres secciones, la primera contiene la información básica del estudiante como: sus nombres, apellidos, sexo, etc. La segunda sección se compone de las carreras con el porcentaje de rendimiento satisfactorio que se ha obtenido aplicando las fórmulas y finalmente un gráfico, en el que de igual forma se muestran los porcentajes alcanzados.

Figura 36. Pantalla Resultados.

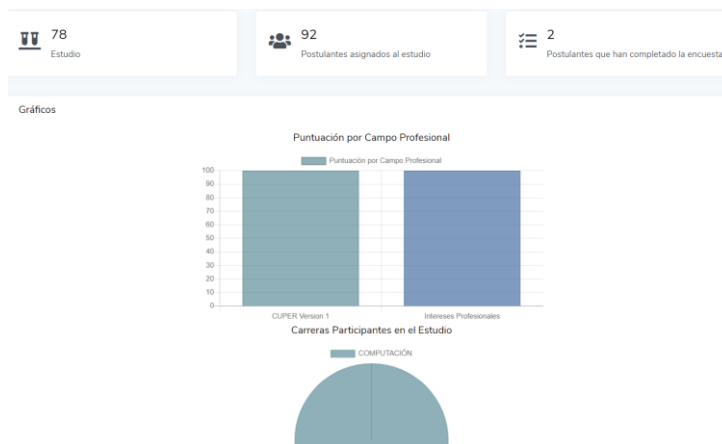


Nota: Pantalla de resultados a estudiantes con estado 1 y que previamente han realizado la encuesta. Elaborado por: los autores.

Dashboard psicologo

El investigador tiene acceso a la pantalla de dashboard al iniciar sesión en la plataforma, esta pantalla contiene información sobre los estudios realizados. Si no existen estudios la pantalla no contendrá ningún gráfico y no se mostrarán resultados. El investigador deberá escoger de una lista el estudio que desea que se muestre en el dashboard.

Figura 37. Dashboard psicologo



Nota: Dashboard del actor investigador. Elaborado por: los autores.

Al inicio de la pantalla se encuentra el listado de estudios, donde se selecciona uno para cargar sus datos, seguido encontramos tres tarjetas que representan el nombre del estudio, la cantidad de postulantes en el estudio y finalmente la cantidad de postulantes que han completado la encuesta. Debajo de las tarjetas mencionadas encontramos un gráfico de barras que está obteniendo las puntuaciones totales para cada categoría o campo profesional. Y Finalmente encontramos un gráfico de pastel el cual contiene la cantidad de postulantes separados por carrera a la que postulan.

4.4 PRUEBAS

Este subcapítulo presenta las diferentes pruebas que se realizaron, a fin de conocer el estado de la plataforma en cuanto a código, seguridad y vulnerabilidades.

4.4.1 Pruebas contra requerimientos

Tabla 49 Casos de pruebas de requerimientos funcionales del administrador.

Aplicación	Caso de Prueba	Req.	Pasos	Estado
WEB	Gestión de usuarios	1.1	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. Seleccionar opción usuaria. 	OK
	Gestión de parámetros	1.2	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. Seleccionar opción parámetros. 	OK
	Carga de información	1.3	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. 4. Ingresar a opción de la columna. 5. Presionar botón “Importar”. 6. Seleccionar y subir archivo xlsx. 	OK

Nota: Esta tabla contiene los casos de pruebas de requerimientos funcionales por parte del administrador.

Tabla 50 Casos de pruebas de requerimientos funcionales del psicólogo.

Aplicación	Caso de Prueba	Req.	Pasos	Estado
WEB	Gestión de estudios	2.1	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. 4. Seleccionar opción Estudios. 5. Crear y asignar estudio. 	OK
	Gestión de encuestas	2.2	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. 4. Seleccionar opción gestión encuesta. 5. Crear encuesta. 	OK
	Gestión de fórmulas	2.3	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. 4. Seleccionar opción gestión fórmulas. 5. Crear e ingresar datos generales formula. 	OK
	Resultados	2.6	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. 	OK

			4. Seleccionar opción Dashboard.	
--	--	--	----------------------------------	--

Nota: Esta tabla contiene los casos de pruebas de requerimientos funcionales por parte del administrador.

Tabla 51 Casos de pruebas de requerimientos funcionales del estudiante.

Aplicación	Caso de Prueba	Req.	Pasos	Estado
WEB	Responde a encuestas	3.1	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 3. Presionar botón “Ingresar”. 	OK

Nota: Esta tabla contiene los casos de pruebas de requerimientos funcionales por parte del estudiante.

Tabla 52 Casos de pruebas de requerimientos funcionales comunes de los actores

Aplicación	Caso de Prueba	Req.	Pasos	Estado
WEB	Autenticación en el sistema	4.1	<ol style="list-style-type: none"> 1. Ingresar a la aplicación. 2. Digitar usuario y contraseña. 	OK
	Cierre de sesión (logout)	4.2	<ol style="list-style-type: none"> 1. Presionar botón Sing -out. 	OK

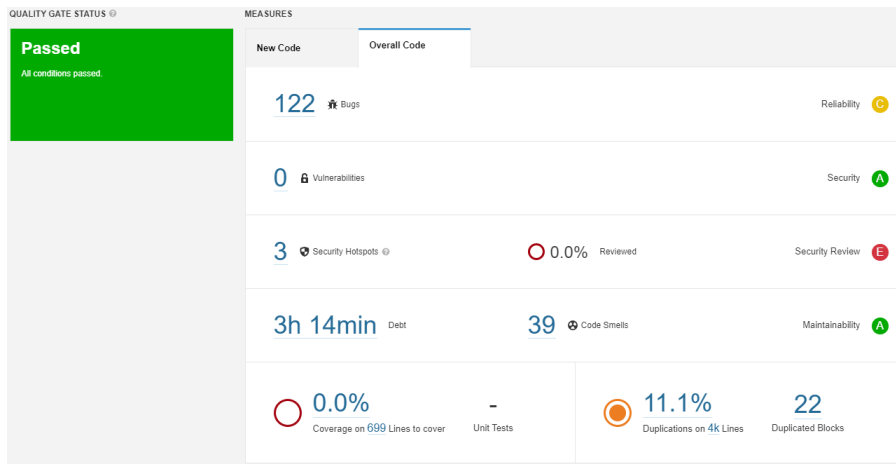
Nota: Esta tabla contiene los casos de pruebas de requerimientos funcionales por parte de los tres actores.

4.4.2 Pruebas de código

Para evaluar la calidad, claridad y limpieza del código escrito en la plataforma informática se utilizó la herramienta SonarQube. La cual permite detectar errores de código antes de que este sea implantado. Además, permite detectar errores, problemas de seguridad e incluso es capaz de sugerir mejoras de forma que este pueda ser limpio y fácil de mantener (SonarQube, 2021).

Antes de implementar la plataforma se validó que el mismo cumpla con las métricas que están establecidas en SonarQube. El escaneo de la plataforma tomó un tiempo total de 40.220 segundos. A continuación, se presenta el resumen:

Figura 38. Resumen métricas de SonarQube



Nota: Resumen del análisis realizado mediante SonarQube. Fuente: SonarQube

Como se pudo evidenciar en la figura anterior, la plataforma tiene diferentes observaciones en cuanto a seguridad, duplicidad del código y bugs, a pesar de esto, la plataforma aprueba las métricas de SonarQube. Sin embargo, se validó cada una de estas métricas para solucionar y aplicar las sugerencias.

En cuanto a la métrica de fiabilidad de la plataforma se observó que la mayoría de las alertas tuvieron como sugerencia cambiar las etiquetas HTML o agregar atributos a las mismas, tales como id o descripciones.

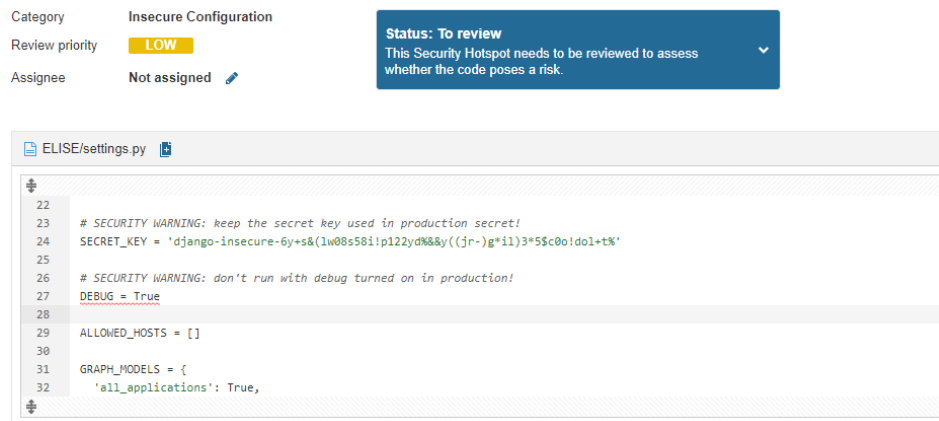
Figura 39. Métrica fiabilidad



Nota: Resumen de la métrica de fiabilidad. Fuente: SonarQube.

Para la métrica de seguridad se observó que el problema surge del archivo de configuraciones del proyecto, ya que la prueba se realizó antes de implementar la plataforma y por esta razón no se han modificado las configuraciones del ambiente de desarrollo.

Figura 40. Métrica seguridad



Nota: Resumen de la métrica de seguridad. Fuente: SonarQube.

En la métrica de duplicidad se observa que la mayoría de las alertas se encuentran en los diferentes formularios, equivalente al 11.1% de líneas de código duplicadas en la plataforma.

Figura 41. Métrica duplicidad

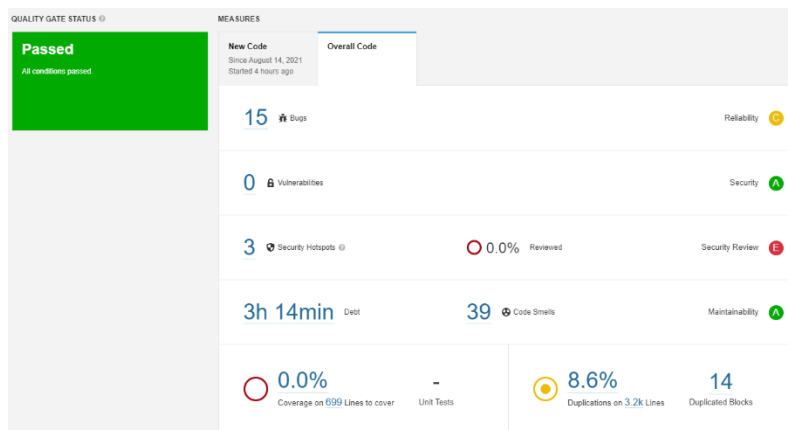


Nota: Resumen de la métrica de duplicidad. Fuente: SonarQube.

4.5.2.1 Análisis de resultados finales

Con la finalidad de obtener un código limpio, mantenible y de calidad se realizaron cambios tomando en cuenta las sugerencias de SonarQube, en la Figura 26, se puede observar que se redujo notablemente la métrica de fiabilidad y duplicidad de código. En esta ocasión el escaneo de la plataforma redujo el tiempo de ejecución quedando en 18.955 segundos.

Figura 42. Resultados finales



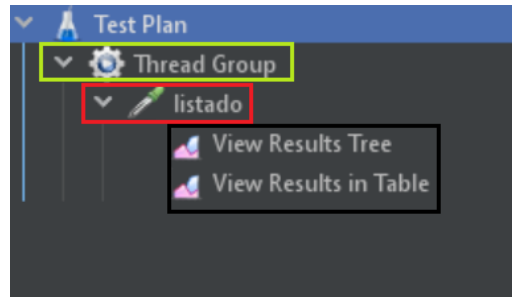
Nota: Resumen de la métrica de duplicidad. Fuente: SonarQube.

4.4.3 Pruebas de carga y rendimiento

Para estas pruebas se utilizó la herramienta de Apache JMeter la cual está completamente diseñada para cargar el comportamiento funcional de prueba y medir el rendimiento de las aplicaciones web (Foundation, 1999).

Esta aplicación además de ser gratuita cuenta con una interfaz muy amigable y de muy fácil uso en la cual debemos realizar algunos pasos para la correcta carga y visualización de resultados.

Figura 43. Pasos principales



Nota: Pasos realizar las pruebas de carga y rendimiento . Elaborado por: los autores.

Como se puede observar en la imagen existen 3 pasos:

1. El primer rectángulo de color verde hace referencia a un Thread group en él se colocará el número de usuarios, tiempo de cada petición (segundos) y el número de peticiones que se requiere hacer.

Figura 44. Thread Properties



Nota: Interfaz para el llenado de Thread group. Elaborado por: los autores.

2. El segundo rectángulo de color rojo hace referencia al tipo de actividad que se desea realizar, en nuestro caso elegimos HTTP Request para las pruebas de carga y rendimiento.

Esta opción cuenta con algunos campos muy importantes que deben ser llenados previamente como el Server Name, Path, protocolo, entre otros. Los cuales nos permitirán realizar diversas peticiones al sitio web.

Figura 45. HTTP Request Properties

Nota: Interfaz para el llenado de campos HTTP Request . Elaborado por: los autores.

3. El tercer rectángulo de color negro hace referencia a los “listener”, los cuales son las maneras de cómo se requiere visualizar los resultados de las diversas peticiones.

Figura 46. Listener view result in table

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
184	21:10:49.708	Thread Group 1-...	listado	135	✓	2863	117	135	0
185	21:10:49.713	Thread Group 1-...	listado	131	✓	2863	117	131	0
186	21:10:49.708	Thread Group 1-...	listado	139	✓	2863	117	139	0
187	21:10:49.718	Thread Group 1-...	listado	129	✓	2863	117	129	0
188	21:10:49.717	Thread Group 1-...	listado	130	✓	2863	117	130	0
189	21:10:44.020	Thread Group 1-...	listado	6057	✓	2863	117	3709	148
190	21:10:45.418	Thread Group 1-...	listado	4660	✓	2863	117	4660	0
191	21:10:44.471	Thread Group 1-...	listado	5608	✓	2863	117	2012	0
192	21:10:50.078	Thread Group 1-...	listado	114	✓	2863	117	114	0
193	21:10:44.456	Thread Group 1-...	listado	5872	✓	2863	117	2027	0
194	21:10:46.962	Thread Group 1-...	listado	3502	✓	2863	117	3502	0
195	21:10:47.568	Thread Group 1-...	listado	3059	✓	2863	117	3059	0
196	21:10:47.567	Thread Group 1-...	listado	3063	✓	2863	117	3063	0
197	21:10:47.567	Thread Group 1-...	listado	3064	✓	2863	117	3064	0
198	21:10:47.839	Thread Group 1-...	listado	2809	✓	2863	117	2809	0

Nota: Tabla de resultado de las peticiones. Elaborado por: los autores.

Para las pruebas de rendimiento se debe que observar en las columnas de Sample Time (El tiempo de muestreo) es el tiempo que transcurre desde que se hizo la petición hasta que se recibe toda la respuesta completa y Latency(Latencia) la cual es el tiempo en el cual se tardó en hacer la petición al servidor y en la que el servidor contesto.

Prueba login

Se realizo varias pruebas con diversos números de estudiantes con el fin de saber cuántos podrán acceder a dar el examen simultáneamente, en este caso el programa luego de varias

pruebas dio como resultado que un total de 400 estudiantes podrán acceder simultáneamente, cabe mencionar que este resultado fue obtenido de un servidor T2micro.

Figura 47. Interfaces prueba login

Thread Properties								
Number of Threads (users): 400								
Ramp up period (seconds): 1								
Loop Count: <input type="checkbox"/> Infinite 1								
Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency
381	23:30:34.176	Thread Group 1-...	listado	515	🟢	2863	117	515
382	23:30:34.179	Thread Group 1-...	listado	515	🟢	2863	117	515
383	23:30:34.183	Thread Group 1-...	listado	519	🟢	2863	117	519
384	23:30:34.189	Thread Group 1-...	listado	513	🟢	2863	117	513
385	23:30:34.192	Thread Group 1-...	listado	514	🟢	2863	117	514
386	23:30:34.194	Thread Group 1-...	listado	512	🟢	2863	117	512
387	23:30:34.197	Thread Group 1-...	listado	515	🟢	2863	117	515
388	23:30:34.203	Thread Group 1-...	listado	516	🟢	2863	117	516
389	23:30:34.200	Thread Group 1-...	listado	519	🟢	2863	117	519
390	23:30:34.207	Thread Group 1-...	listado	518	🟢	2863	117	518
391	23:30:34.186	Thread Group 1-...	listado	541	🟢	2863	117	541
392	23:30:34.213	Thread Group 1-...	listado	519	🟢	2863	117	519
393	23:30:34.209	Thread Group 1-...	listado	523	🟢	2863	117	523
394	23:30:34.218	Thread Group 1-...	listado	521	🟢	2863	117	521
395	23:30:34.221	Thread Group 1-...	listado	524	🟢	2863	117	524
396	23:30:34.224	Thread Group 1-...	listado	521	🟢	2863	117	521
397	23:30:34.173	Thread Group 1-...	listado	579	🟢	2863	117	579
398	23:30:34.114	Thread Group 1-...	listado	638	🟢	2863	117	638
399	23:30:34.214	Thread Group 1-...	listado	539	🟢	2863	117	539

Nota: Interfaces con resultados de las 400 peticiones. Elaborado por: los autores.

Como resultado se puede observar que el tiempo de muestra y la latencia son bajas en las cuales muestran que para cada petición existe un promedio de 500 a 600 milisegundos.

Prueba para realizar encuesta

Se realizo varias pruebas con diversos números de estudiantes con el fin de saber cuántos podrán realizar el examen simultáneamente, en este caso el programa luego de varias pruebas dio como resultado que un total de 200 estudiantes podrán realizar el examen simultáneamente, cabe mencionar que este resultado fue obtenido de un servidor T2micro.

Figura 48. Interfaces prueba encuesta

Web Server

Protocol [http]: Server Name or IP: Port Number:

HTTP Request

GET Content encoding:

Redirect Automatically Follow Redirects Use KeepAlive Use multipart/form-data Browser-compatible headers

Number of Threads (users):

Ramp-up period (seconds):

Loop Count: Infinite

Nota: Interfaces con datos establecidos para las 200 peticiones. Elaborado por: los autores.

Figura 49. Resultados

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency
181	23:46:09.731	Thread Group 1-...	listado	568	✓	3149	261	311
182	23:46:09.815	Thread Group 1-...	listado	487	✓	3149	261	230
183	23:46:09.820	Thread Group 1-...	listado	490	✓	3149	261	238
184	23:46:09.835	Thread Group 1-...	listado	492	✓	3149	261	227
185	23:46:09.825	Thread Group 1-...	listado	503	✓	3149	261	234
186	23:46:09.838	Thread Group 1-...	listado	490	✓	3149	261	230
187	23:46:09.844	Thread Group 1-...	listado	489	✓	3149	261	226
188	23:46:09.854	Thread Group 1-...	listado	483	✓	3149	261	225
189	23:46:09.858	Thread Group 1-...	listado	481	✓	3149	261	226
190	23:46:09.849	Thread Group 1-...	listado	493	✓	3149	261	234
191	23:46:09.869	Thread Group 1-...	listado	477	✓	3149	261	232
192	23:46:09.864	Thread Group 1-...	listado	483	✓	3149	261	235
193	23:46:09.888	Thread Group 1-...	listado	466	✓	3149	261	240
194	23:46:09.883	Thread Group 1-...	listado	491	✓	3149	261	239
195	23:46:09.894	Thread Group 1-...	listado	485	✓	3149	261	249
196	23:46:09.830	Thread Group 1-...	listado	549	✓	3149	261	242
197	23:46:09.878	Thread Group 1-...	listado	504	✓	3149	261	235
198	23:46:09.873	Thread Group 1-...	listado	511	✓	3149	261	242
199	23:46:09.899	Thread Group 1-...	listado	508	✓	3149	261	234

Nota: Interfaces con resultados de las 200 peticiones para la realización de la encuesta. Elaborado por: los autores.

Como resultado se puede observar que el tiempo de muestra y la latencia son bajas en las cuales como resultado se muestra que para cada petición existe un promedio de 400 a 500 milisegundos.

4.4.4 Pruebas con usuarios

Para comprobar el correcto funcionamiento de la plataforma se configuró un estudio y se encuestó a 10 personas, entre estudiantes de la UPS y personas que querían aportar. La encuesta asignada fue el instrumento CIPRO. Las personas encuestadas nos pudieron colaborar enviando capturas de lo que estaba siendo la experiencia en la plataforma.

Figura 50. Presentación de la encuesta

**CIPRO
INSTRUMENTO**

1. Te gustaría solucionar ejercicios vectoriales

Totalmente desacuerdo
 Desacuerdo
 Ni de acuerdo, ni desacuerdo
 De acuerdo
 Totalmente de acuerdo

2. Servirías de voluntario en un hogar para las personas de la tercera edad

Totalmente desacuerdo
 Desacuerdo
 Ni de acuerdo, ni desacuerdo
 De acuerdo
 Totalmente de acuerdo

3. Te agrada leer libros a niños o personas de la tercera edad en sitios de acogida

Nota: Acceso a la interfaz de llenado de la encuesta desde el móvil. Elaborado por: los autores.

A continuación, se presenta otra captura de pantalla de los resultados para los usuarios que habían llenado la encuesta.

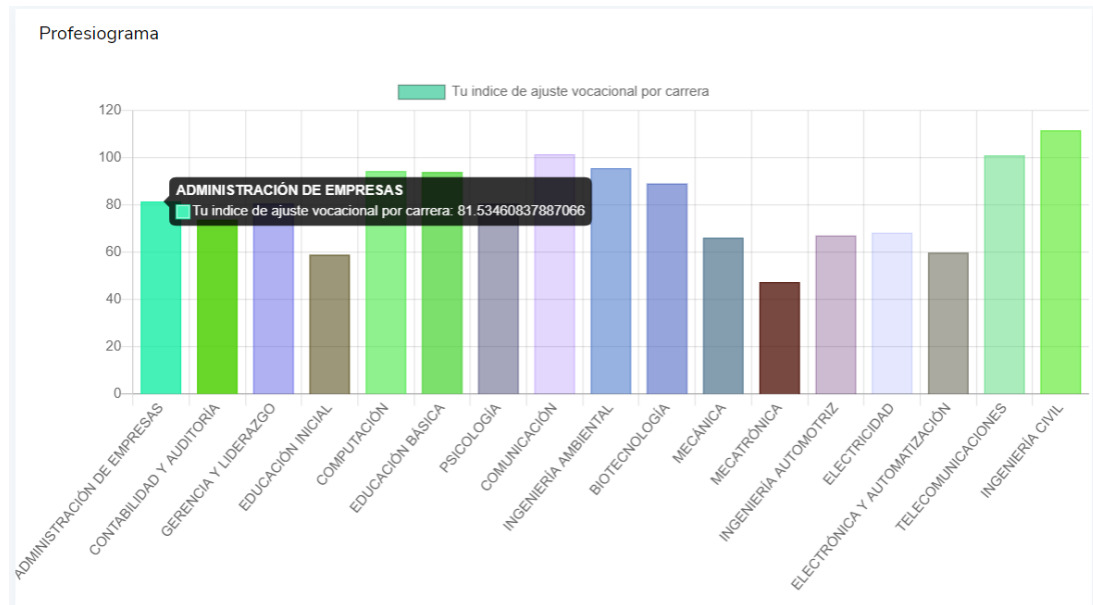
Figura 51. Porcentaje de rendimiento para cada carrera

Carrera	Perfil		Satisfactorio	Riesg
	Vocacional	Satisfactorio		
ADMINISTRACIÓN DE EMPRESAS	AFIN	77.3 %	22.7	
CONTABILIDAD Y AUDITORÍA	AFIN	67.6 %	32.4	
GERENCIA Y LIDERAZGO	AFIN	54.4 %	45.6	
EDUCACIÓN INICIAL	NO AFIN	39.1 %	60.9	
COMPUTACIÓN	AFIN	81.6 %	18.4	
EDUCACIÓN BÁSICA	AFIN	67.3 %	32.7	
PSICOLOGÍA	AFIN	69.9 %	30.1	
COMUNICACIÓN	AFIN	68.6 %	30.4	
INGENIERÍA AMBIENTAL	AFIN	64.5 %	35.5	
BIOTECNOLOGÍA	AFIN	74.2 %	25.8	
MECÁNICA	AFIN	57.2 %	42.8	
MECATRÓNICA	NO AFIN	29.1 %	70.9	
INGENIERÍA AUTOMOTRIZ	AFIN	54.9 %	45.1	
ELECTRICIDAD	AFIN	63.0 %	37.0	
ELECTRÓNICA Y AUTOMATIZACIÓN	AFIN	67.2 %	32.8	
TELECOMUNICACIONES	AFIN	90.9 %	9.1	
INGENIERÍA CIVIL	AFIN	59.2 %	40.8	

Nota: Acceso a la interfaz de resultados para alumnos en móvil. Elaborado por: los autores.

Finalmente podemos observar el gráfico que presenta la plataforma a cada usuario, dándole el puntaje que obtuvo para cada carrera.

Figura 52. Puntajes para cada carrera



Nota: Acceso a la interfaz de resultado en web para alumno. Elaborado por: los autores.

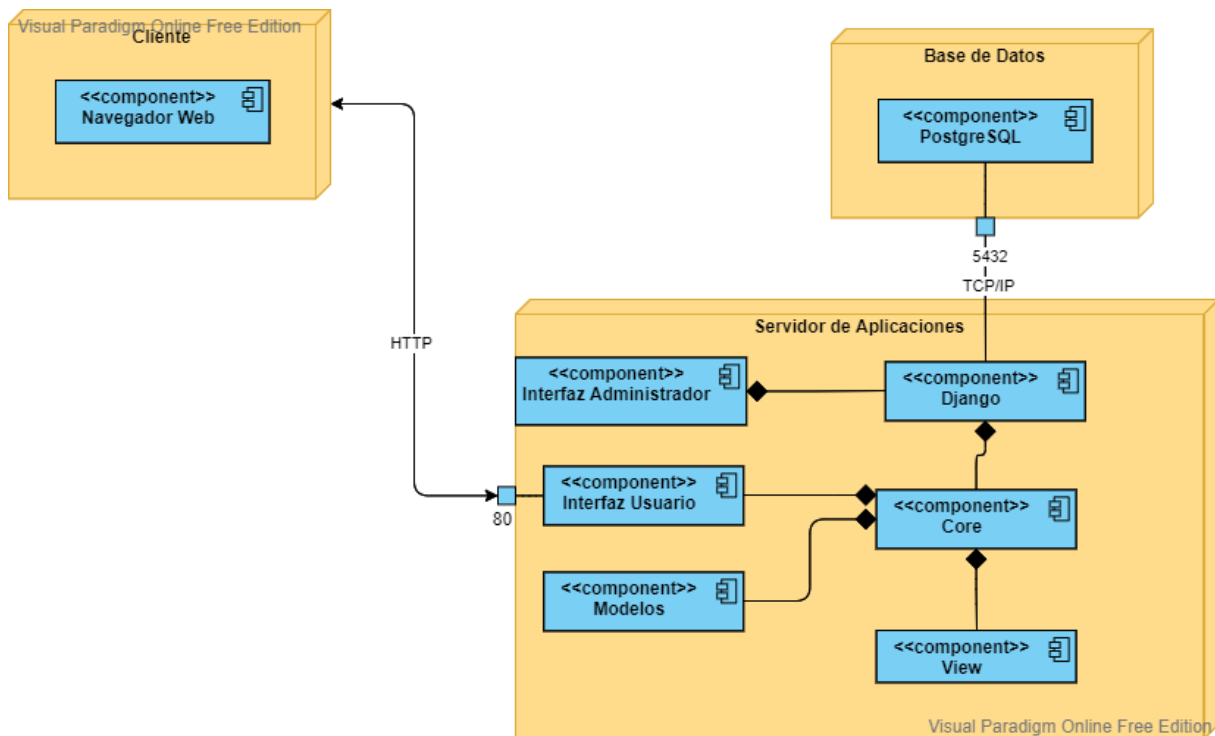
5. IMPLEMENTACIÓN

El presente capítulo presenta el diagrama de despliegue y configuraciones de los archivos de la plataforma, además, de la integración con la base de datos y la plataforma funcional en la instancia de AWS.

5.1 INFRAESTRUCTURA EN AWS

Para el despliegue de la plataforma se utilizó un servidor *t2.micro* con el sistema operativo *Ubuntu 18.04* alojado en AWS, además de una base de datos PostgreSQL alojada en el mismo servidor. A continuación, se presenta el diagrama de despliegue junto con la descripción de sus nodos y componentes:

Figura 53. Diagrama de despliegue



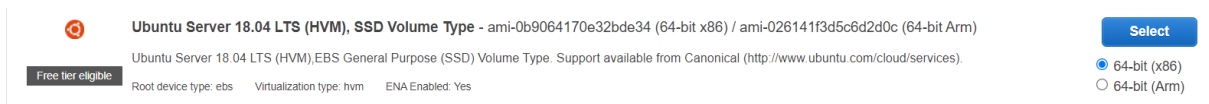
Nota: Nodos y componentes que hacen parte del despliegue de la plataforma. Fuente: Los autores.

- **Nodo Servidor de Aplicaciones :** Este nodo contiene todos los servicios de la plataforma y sus componentes son:
 - Django:** El framework sobre el cual opera la plataforma.
 - Core:** El corazón de la plataforma, aquí está definida la lógica, cálculos, seguridad, importaciones creaciones y eliminaciones de registros.
 - Interfaz Usuario:** Es donde se cargan todas las preguntas.
- **Nodo Cliente:** Permite la interacción con la plataforma desde cualquier navegador web que permita la ejecución de código HTML, CSS y JavaScript, además debe tener acceso a internet.
- **Nodo Base de datos :** Permite la interacción de la base de datos con Django mediante el puerto 5432.

5.2 INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE

La instancia seleccionada fue una con Linux Ubuntu 18.04 de 64 bits y almacenamiento de tipo SSD. Además, se seleccionó el tipo *t2.micro* que tiene 1 CPU, 1 GB de RAM y pertenece a la capa gratuita de 12 meses que ofrece Amazon a sus nuevos usuarios.

Figura 54. Instancia AWS



Nota: Selección de sistema operativo para la instancia. Fuente: AWS Consolé.

Para acceder a la instancia, se utilizó *PuTTY* junto con la clave privada que se descarga al finalizar la creación de la instancia. La clave privada de la instancia es un archivo de tipo *pem*, por esta razón se necesita crear una clave de *PuTTY* con *PuTTYgen*. Además, se creó una IP elástica en la consola de Amazon para poder acceder siempre desde la misma IP.

Una vez finalizadas las anteriores configuraciones, se tiene acceso a la instancia y esto se puede evidenciar en la siguiente figura:

Figura 55. Acceso a la instancia mediante *PuTTY*

```
login as: ubuntu
* Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1045-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

System information as of Wed Sep  8 04:00:18 UTC 2021

System load:  0.0      Processes:    115
Usage of /:   31.3% of 7.69GB   Users logged in:  0
Memory usage: 47%      IP address for eth0: 172.31.28.59
Swap usage:  0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

8 updates can be applied immediately.
8 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

New release '20.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Tue Sep  7 23:24:36 2021 from 157.100.172.32
ubuntu@ip-172-31-28-59:~$
```

Nota: Acceso a la instancia mediante PuTTY. Fuente: Los autores.

5.2.1 Instalación de paquetes y servicios

Con el acceso a la instancia, se puede empezar a instalar tanto paquetes como servicios de los cuales depende la plataforma. Ya que Linux Ubuntu tiene preinstalado Python únicamente se realiza una actualización de los paquetes del sistema con los siguientes comandos:

```
$ sudo apt update
$ sudo apt upgrade
```

A continuación, se procede a instalar el administrador de paquetes de Python *pip*, también la base de datos *PostgreSQL*, el administrador de entornos virtuales *venv*, y algunas otras dependencias con el siguiente comando:

```
$ sudo apt install build-essential python3 python3-venv python3-pip
python3-dev nginx postgresql postgresql-contrib libpq-dev
```

Con esto se puede continuar, con la creación de la base de datos y usuarios en *PostgreSQL*, además de instalar *git* para acceder al proyecto y sus últimas actualizaciones desde la instancia. En la siguiente figura se presentan las dependencias del proyecto obtenidas del archivo *requirements.txt*:

Figura 56. Archivo *requirements.txt*

```
asgiref==3.3.4
defusedxml==0.7.1
diff-match-patch==20200713
Django==3.2
django-extensions==3.1.3
django-import-export==2.5.0
django-safedelete==1.0.0
django-widget-tweaks==1.4.8
et-xmlfile==1.1.0
MarkupPy==1.14
odfpy==1.4.1
openpyxl==3.0.7
pytz==2021.1
PyYAML==5.4.1
sqlparse==0.4.1
tablib==3.0.0
xlrd==2.0.1
xlwt==1.3.0
```

Nota: Dependencias del proyecto. Fuente: Los autores.

5.3 DESPLIEGUE

La plataforma funciona sobre el servidor de aplicaciones *uWSGI* y *nginx* para el contenido estático, por lo que es necesario realizar la instalación de *uwsgi* para Python con el siguiente comando:

```
$ sudo -H pip3 install uwsgi
```

A continuación, se procede con la configuración, descarga y levantamiento de los servicios y archivos del proyecto, se enumeran en la siguiente lista:

1. Clonar el proyecto en la instancia

```
$ git clone https://github.com/rtufino/elise.git
```

2. Acceder a la carpeta clonada y crear el entorno virtual del proyecto

```
$ python -m venv venv
```

3. Activar el entorno virtual

```
$ source venv/bin/activate
```


4. Actualizar el administrador de paquetes pip e instalar los requerimientos del proyecto.

```
$ pip install --upgrade pip
```

5. Agregar los hosts permitidos en el archivo settings.py y configurar la base de datos en el mismo archivo.

Figura 57. Hosts permitidos

```
ALLOWED_HOSTS = ['3.139.149.30']
```

Nota: Acceso a la instancia mediante PuTTY. Fuente: Los autores.

Figura 58. Configuración de la base de datos en la plataforma

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.postgresql_psycopg2',  
        'NAME': 'elise',  
        'USER': 'django_user',  
        'PASSWORD': ' ',  
        'HOST': 'localhost',  
        'PORT': '5432'  
    }  
}
```

Nota: Acceso a la instancia mediante PuTTY. Fuente: Los autores.

Finalmente Agregar el path de los recursos estáticos:

```
STATIC_ROOT = os.path.join(BASE_DIR, 'static')
```

6. Realizar las migraciones para crear las tablas en la base de datos.

```
$ python manage.py makemigrations
```

```
$ python manage.py migrate
```

7. Crear un super usuario

```
$ python manage.py createsuperuser
```

8. Crear los archivos estáticos del proyecto.

```
$ python manage.py collectstatic
```

9. Crear el archivo de configuración para uWSGI

```
$ vi elise/elise.ini
```

Figura 59. Archivo *elise.ini*

```
[uwsgi]
home = /home/ubuntu/elise/venv/
chdir = /home/ubuntu/elise/
wsgi-file = /home/ubuntu/elise/ELISE/wsgi.py
http = 0.0.0.0:8000

master = true
processes = 5

socket = elise.sock
chmod-socket = 660
vacuum = true
die-on-term = true
```

Nota: Configuración para uWSGI. Fuente: Los autores.

10. Crear un servicio para la plataforma en la unidad *systemd*.

```
$ sudo vi /etc/systemd/system/elise.service
```

Figura 60. Archivo *elise.service*

```
[Unit]
Description = Instancia uWSGI para elise
After = network.target

[Service]
User = ubuntu
Group = www-data
WorkingDirectory = /home/ubuntu/elise
Environment = "PATH=/home/ubuntu/elise/venv/bin"
ExecStart = /usr/local/bin/uwsgi --ini elise.ini

[Install]
WantedBy=multi-user.target
```

Nota: Configuración del servicio para *elise*. Fuente: Los autores.

11. Habilitar e iniciar el servicio

```
$ sudo systemctl start elise.service
$ sudo systemctl enable elise.service
```

12. Configurar los sites de la plataforma en *nginx*

```
$ sudo vi /etc/nginx/sites-available/elise
```

Figura 61. Sites available en nginx

```
server{
    listen 80;
    server_name 3.139.149.30;

    location / {
        include uwsgi_params;
        uwsgi_pass unix:/home/ubuntu/elise/elise.sock;
    }
    location /static/ {
        alias /home/ubuntu/elise/core/static/;
        expires modified +1w;
    }
}
```

Nota: Configuración de los sitios disponibles para nginx. Fuente: Los autores.

13. Crear un enlace para los sitios habilitados en *nginx*

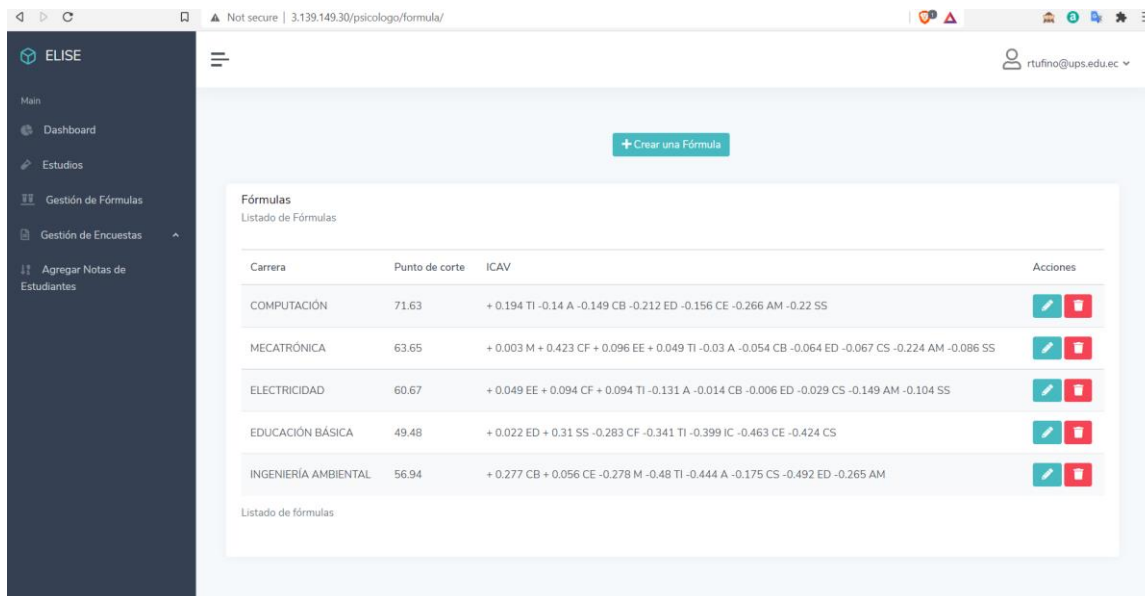
```
$ sudo ln -s /etc/nginx/sites-available/elise /etc/nginx/sites-enabled/
```

14. Verificar que la configuración de *nginx* este correcta y reiniciar el servicio.

```
$ sudo nginx -t
$ sudo systemctl restart nginx
```

Finalmente, después de haber configurado todo, podemos acceder a la plataforma desde cualquier dispositivo con acceso a internet y un navegador web. A continuación, se muestra la plataforma funcionando desde el navegador en internet.

Figura 62. Plataforma desplegada en internet.



Nota: Plataforma elise funcionando en internet. Fuente: Los autores.

Se puede acceder a la plataforma mediante el siguiente enlace: <http://3.139.149.30/>

6. CONCLUSIONES

1. La automatización del proceso de importación de datos fue de gran utilidad debido a que se evitó el mal tipeo por parte del psicólogo y dando como resultado un ahorro de tiempo significativo.
2. La utilización de Django para el desarrollo del sistema ayudo en el desarrollo más rápido del sistema ya que se incorporó funciones nativas de Django tales como la parte administrativa, validaciones de login, funciones CRUD entre otras.
3. Para el manejo del sistema en la parte administrativa es necesario tener un conocimiento previo en el proceso del llenado de fórmulas.
4. La plataforma informática, permite el acceso seguro y oportuno a los datos sensibles que maneja el grupo GIE, además de unificar sus fuentes de datos.
5. Se puede decir que la gestión de fórmulas es el corazón del proyecto, ya que son las que permiten calcular y mostrar resultados, por lo que también fue una de las partes más complicadas de solventar. Finalmente se solucionó y se almacenan eficientemente.

7. RECOMENDACIONES

1. Se debe fomentar el uso de la aplicación entre las diferentes sedes de la Universidad Politécnica Salesiana, debido a que es una herramienta de gran utilidad en el área de psicología.
2. Se recomienda tener un mayor énfasis por parte de la Universidad Politécnica Salesiana en el aprendizaje del análisis y requerimientos para el desarrollo de proyectos.
3. Se debería contemplar la posibilidad de integrar esta aplicación para dispositivos móviles y en diferentes sistemas operativos de manera que se tenga un sistema completo de los exámenes psicométricos.
4. En una siguiente versión del sistema informático se recomienda incorporar la funcionalidad de enviar los resultados del examen al correo y al móvil del aspirante, con la finalidad de que le pueda servir para tomar una decisión correcta posteriormente.
5. El sistema está alojado en AWS con una instancia de una capacidad mínima, se recomienda el aumento de plan contratado para evitar algún colapso del sistema en alguna carga de datos masiva.

REFERENCIAS

- Zumárraga-Espinosa, M., Castro, M. I., Boada, P., Escobar, M. J., & Romero, J. C. (2019). Intereses profesionales y selección de carrera: criterios para la implementación de diagnósticos de compatibilidad automatizados en la Universidad Politécnica Salesiana. *Congresos CLABES*, 1303-1312.
- Carrillo Regalado, S., & Ríos Almodóvar, J. (2013). *Trabajo y rendimiento escolar de los estudiantes universitarios. El caso de la Universidad de Guadalajara, México*. Guadalajara: Revista de la educación superior.
- Flores Pelcastre, J. (2005). *Metodología para la automatización de procesos en la pequeña y mediana empresa*. Monterrey.
- Gomez Sánchez, D., Oviedo Marin, R., & Martínez López, E. (2011). *Factores que influyen en el rendimiento académico del estudiante universitario*. Chihuahua.
- Hernández Mendo, A., & Pastrana Bricones, J. (2015). *Evaluación y baremación de variables psicosociales a través de una plataforma online*. Servicio de Publicaciones y Divulgación Científica.
- Torres, B., & Vergara, T. (2018). *Factores que determinan la elección de carrera profesional: en estudiantes de undécimo grado de colegios públicos y privados de Barrancabermeja*. Psicoespacios.
- Vargas, G. M. (2007). *Factores asociados al rendimiento académico en estudiantes universitarios, una reflexión desde la calidad de la educación superior pública*.
- Elizabeth, C. Q. (2019). PROGRAMA DE ORIENTACIÓN VOCACIONAL- PROFESIONAL PARA. Ambato.
- Schwaber, K., & Sutherland, J. (2020). *La Guía Definitiva de Scrum: Las Reglas del Juego*.
- Tufiño, R. (Marzo de 2019). Ubuntu 18.04 Django + Nginx + PostgreSQL 10. Obtenido de Ubuntu 18.04 Django + Nginx + PostgreSQL 10.
- Lázaro, N., Callejas, Z., Griol, D., & Durán, M. (2017). La deserción estudiantil en

educación superior: s.o.s. en carreras de ingeniería informática. Obtenido de <https://revistas.utp.ac.pa/index.php/clabes/article/view/1674>

Zumárraga, M., Luzuriaga, J., Armas, R., González, Y., & Peñaherrera, L. (2020). “¿Los rasgos de personalidad importan? Medición y análisis de su correlación con el rendimiento académico como aporte a los procesos institucionales de orientación vocacional. 265-275.

Django Girls. (14 de 07 de 2021). Obtenido de [djangogirls.org: https://tutorial.djangogirls.org/es/django/#](https://tutorial.djangogirls.org/es/django/#)

Django Project. (14 de 07 de 2021). Obtenido de [djangoproject.com: https://www.djangoproject.com/start/overview/](https://www.djangoproject.com/start/overview/)

Foundation, A. S. (1999). *Apache JMeter* .

PostgreSQL. (15 de 07 de 2021). Obtenido de [postgresql.org: https://www.postgresql.org/about/](https://www.postgresql.org/about/)

Python Org. (14 de 07 de 2021). Obtenido de [python.org: https://www.python.org/about/](https://www.python.org/about/)

Read the docs. (05 de 06 de 2020). Obtenido de [readthedocs.io: https://django-import-export.readthedocs.io/en/latest/](https://django-import-export.readthedocs.io/en/latest/)

SonarQube. (14 de 08 de 2021). Obtenido de https://www.sonarqube.org/?gads_campaign=South-America-DSA-SonarQube&gads_ad_group=DSA&gads_keyword=

(14 de 07 de 2021). Obtenido de [Pythones: https://pythones.net/que-es-python-y-sus-caracteristicas/](https://pythones.net/que-es-python-y-sus-caracteristicas/)

ANEXOS

Sprint 1

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
				1	2	3
					Sprint Planning Sprint Review Sprint Retrospective	
4	5	6	7	8	9	10
		Daily meet		Daily meet	Sprint Review Sprint Retrospective	
11	12	13	14	15	16	17
		Daily meet		Daily meet	Sprint Planning Sprint Review Sprint Retrospective	
18	19	20	21	22	23	24
		Daily meet		Daily meet	Sprint Review Sprint Retrospective	
25	26	27	28	29	30	
		Daily meet		Daily meet	Sprint Planning Sprint Review Sprint Retrospective	

Abril 2021

Nota: Planificación de SCRUM en el mes de abril. Elaborado por: Anthony Cabezas y Kevin Chacha

Tabla 53

Sprint 2

Mayo 2021

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
						1
2	3	4 Daily meet	5	6 Daily meet	7 Sprint Planning Sprint Review Sprint Retrospective	8
9	10	11 Daily meet	12	13 Daily meet	14 Sprint Review Sprint Retrospective	15
16	17	18 Daily meet	19	20 Daily meet	21 Sprint Planning Sprint Review Sprint Retrospective	22
23	24	25 Daily meet	26	27 Daily meet	28 Sprint Review Sprint Retrospective	29
30	31					

Nota: Planificación de SCRUM en el mes de mayo. Elaborado por: Anthony Cabezas y Kevin Chacha

Tabla 54 Sprint 3

Junio 2021						
Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
		1 Daily meet	2	3 Daily meet	4 Sprint Planning Sprint Review Sprint Retrospective	5
6	7	8 Daily meet	9	10 Daily meet	11 Sprint Review Sprint Retrospective	12
13	14	15 Daily meet	16	17 Daily meet	18 Sprint Planning Sprint Review Sprint Retrospective	19
20	21	22 Daily meet	23	24 Daily meet	25 Sprint Review Sprint Retrospective	26
27	28	29 Daily meet	30			

Nota: Planificación de SCRUM en el mes de mayo. Elaborado por: los autores.

Figura 63. Impresión resultados usuario

Datos Personales		Resultados por Carrera			
				Probabilidad de Rendimiento	
Nombres Completos	KATHERINE MARGOTH RAMIREZ VILAÑA	Carrera	Perfil Vocacional	Satisfactorio	Riesgoso
Cédula	1726280488	ADMINISTRACIÓN DE EMPRESAS	AFIN	77.3 %	22.7 %
Email Institucional	kramirezv@est.ups.edu.ec	CONTABILIDAD Y AUDITORÍA	AFIN	67.6 %	32.4 %
Carrera a la que postulas	COMPUTACIÓN	GERENCIA Y LIDERAZGO	AFIN	54.4 %	45.6 %
Edad	22	EDUCACIÓN INICIAL	NO AFIN	39.1 %	60.9 %
Sexo	Femenino	COMPUTACIÓN	AFIN	81.6 %	18.4 %
		EDUCACIÓN BÁSICA	AFIN	67.3 %	32.7 %
		PSICOLOGÍA	AFIN	69.9 %	30.1 %
		COMUNICACIÓN	AFIN	69.6 %	30.4 %
		INGENIERÍA AMBIENTAL	AFIN	64.5 %	35.5 %
		BIOTECNOLOGÍA	AFIN	74.2 %	25.8 %
		MECÁNICA	AFIN	57.2 %	42.8 %
		MECATRÓNICA	NO AFIN	29.1 %	70.9 %
		INGENIERÍA AUTOMOTRIZ	AFIN	54.9 %	45.1 %
		ELECTRICIDAD	AFIN	63.0 %	37.0 %
		ELECTRÓNICA Y AUTOMATIZACIÓN	AFIN	67.2 %	32.8 %
		TELECOMUNICACIONES	AFIN	90.9 %	9.1 %
		INGENIERÍA CIVIL	AFIN	59.2 %	40.8 %

Nota: Pantalla de resultados finales del estudiante Elaborado por: los autores.

Figura 64. Impresión resultados usuario

Datos Personales		Resultados por Carrera			
Nombres Completos	DIEGO JAVIER GONZALEZ QUINGA			Probabilidad de Rendimiento	
Cédula	1725670390				
Email Institucional	dgonzalezq2@est.ups.edu.ec				
Carrera a la que postulas	INGENIERÍA AMBIENTAL				
Edad	20				
Sexo	MASCULINO				
		Carrera	Perfil Vocacional	Satisfactorio	Riesgoso
		ADMINISTRACIÓN DE EMPRESAS	AFIN	77.3 %	22.7 %
		CONTABILIDAD Y AUDITORÍA	AFIN	67.6 %	32.4 %
		GERENCIA Y LIDERAZGO	AFIN	54.4 %	45.6 %
		EDUCACIÓN INICIAL	NO AFIN	39.1 %	60.9 %
		COMPUTACIÓN	AFIN	81.6 %	18.4 %
		EDUCACIÓN BÁSICA	AFIN	67.3 %	32.7 %
		PSICOLOGÍA	AFIN	69.9 %	30.1 %
		COMUNICACIÓN	AFIN	69.6 %	30.4 %
		INGENIERÍA AMBIENTAL	AFIN	64.5 %	35.5 %
		BIOTECNOLOGÍA	AFIN	74.2 %	25.8 %
		MECÁNICA	AFIN	57.2 %	42.8 %
		MECATRÓNICA	NO AFIN	29.1 %	70.9 %
		INGENIERÍA AUTOMOTRIZ	AFIN	54.9 %	45.1 %
		ELECTRICIDAD	AFIN	63.0 %	37.0 %
		ELECTRÓNICA Y AUTOMATIZACIÓN	AFIN	67.2 %	32.8 %
		TELECOMUNICACIONES	AFIN	90.9 %	9.1 %
		INGENIERÍA CIVIL	AFIN	59.2 %	40.8 %

Nota: Pantalla de resultados finales del estudiante Elaborado por: los autores.

Figura 65. Impresión resultados usuario

Datos Personales		Resultados por Carrera			
Nombres Completos	VERÓNICA ESTEFANÍA PILLAJO NOGALES			Probabilidad de Rendimiento	
Cédula	1725023327				
Email Institucional	vpillajon@est.ups.edu.ec				
Carrera a la que postulas	COMPUTACIÓN				
Edad	26				
Sexo	FEMENINO				
		Carrera	Perfil Vocacional	Satisfactorio	Riesgoso
		ADMINISTRACIÓN DE EMPRESAS	AFIN	77.3 %	22.7 %
		CONTABILIDAD Y AUDITORÍA	AFIN	67.6 %	32.4 %
		GERENCIA Y LIDERAZGO	AFIN	54.4 %	45.6 %
		EDUCACIÓN INICIAL	NO AFIN	39.1 %	60.9 %
		COMPUTACIÓN	AFIN	81.6 %	18.4 %
		EDUCACIÓN BÁSICA	AFIN	67.3 %	32.7 %
		PSICOLOGÍA	AFIN	69.9 %	30.1 %
		COMUNICACIÓN	AFIN	69.6 %	30.4 %
		INGENIERÍA AMBIENTAL	AFIN	64.5 %	35.5 %
		BIOTECNOLOGÍA	AFIN	74.2 %	25.8 %
		MECÁNICA	AFIN	57.2 %	42.8 %
		MECATRÓNICA	NO AFIN	29.1 %	70.9 %
		INGENIERÍA AUTOMOTRIZ	AFIN	54.9 %	45.1 %
		ELECTRICIDAD	AFIN	63.0 %	37.0 %
		ELECTRÓNICA Y AUTOMATIZACIÓN	AFIN	67.2 %	32.8 %
		TELECOMUNICACIONES	AFIN	90.9 %	9.1 %
		INGENIERÍA CIVIL	AFIN	59.2 %	40.8 %

Nota: Pantalla de resultados finales del estudiante Elaborado por: los autores

