



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

**PLATAFORMA DE EDUCACIÓN ONLINE UTILIZANDO REACTJS
PARA MEJORAR LA OFERTA DE CURSOS DE LA EMPRESA GAIA
DE LA CIUDAD DE AMBATO**

Trabajo de Integración Curricular Modalidad: Proyecto de Investigación, presentado
previo a la obtención del Título de Ingeniero en Tecnologías de la Información.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de software

AUTOR: Franklin Rafael Guillén Carrera

TUTOR: Ing. David Omar Guevara Aulestia, Mg.

Ambato – Ecuador

marzo - 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Integración Curricular con el tema: PLATAFORMA DE EDUCACIÓN ONLINE UTILIZANDO REACTJS PARA MEJORAR LA OFERTA DE CURSOS DE LA EMPRESA GAIA DE LA CIUDAD DE AMBATO, desarrollado bajo la modalidad Proyecto de Investigación por el señor Franklin Rafael Guillén Carrera estudiante de la Carrera de Ingeniería en Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 de las segundas reformas al Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y el numeral 7.4 del respectivo instructivo del reglamento.

Ambato, marzo 2023

Ing. David Omar Guevara Aulestia, Mg.

TUTOR

AUTORÍA

El presente trabajo de Integración Curricular titulado: PLATAFORMA DE EDUCACIÓN ONLINE UTILIZANDO REACTJS PARA MEJORAR LA OFERTA DE CURSOS DE LA EMPRESA GAIA DE LA CIUDAD DE AMBATO es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, marzo 2023



Franklin Rafael Guillén Carrera

C.C 1803553385

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Integración Curricular como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Integración Curricular en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, marzo 2023.



Franklin Rafael Guillén Carrera

C.C 1803553385

AUTOR

APROBACION TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Integración Curricular presentado por el señor Franklin Rafael Guillén Carrera, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado “PLATAFORMA DE EDUCACIÓN ONLINE UTILIZANDO REACTJS PARA MEJORAR LA OFERTA DE CURSOS DE LA EMPRESA GAIA DE LA CIUDAD DE AMBATO”, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 de las segundas reformas al Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y sus reformas y al numeral 7.6 del respectivo instructivo del reglamento. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, marzo 2023

Ing. Elsa Pilar Urrutia Urrutia, Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Edison Homero Álvarez Mayorga
PROFESOR CALIFICADOR

Ing. Julio Enrique Balarezo López
PROFESOR CALIFICADOR

DEDICATORIA

Dedico este proyecto de investigación al pilar fundamental de mi vida, a mi madre Maribel Carrera por ser esa luz que guía mi camino en los días de oscuridad, por brindarme su amor y su apoyo incondicional para formarme como persona y profesional.

A mi hermana, Mayra Guillén, por todo el tiempo brindado en aconsejarme y querer siempre lo mejor para mí y mis estudios.

Por último, dedico este trabajo a mi abuelita Mery Carrera por siempre estar ahí, en las buenas y en las malas. Por ser mi segunda madre y amarme como nadie en este mundo.

AGRADECIMIENTO

Agradezco de manera infinita a mi tía, Andrea López, por estar siempre presente cuando la necesitaba, tanto para mis interrogantes académicas como personales.

También, expreso mis agradecimientos a Andrea Guillén por ser un apoyo emocional en el desarrollo de este trabajo de investigación.

De igual forma, agradezco a la Empresa GAIA, y en especial al ingeniero Daniel Chimborazo, por permitirme desarrollar este trabajo dentro de su compañía y su predisposición para el mismo.

Índice general de contenidos

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
DERECHOS DE AUTOR	iv
APROBACION TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
RESUMEN EJECUTIVO	xii
ABSTRACT	xiii
CAPÍTULO I.- MARCO TEÓRICO	1
1.1 Tema de Investigación	1
1.1.1 Planteamiento del problema	1
1.2 Antecedentes Investigativos	2
1.3 Fundamentación Teórica	5
1.4 Objetivos	6
1.4.1 Objetivo general	6
1.4.2 Objetivos específicos	7
CAPÍTULO II.- METODOLOGÍA	8
2.1. Materiales	8
2.2. Métodos	9
2.2.1. Modalidad de Investigación	9
2.2.1.1. Investigación Campo	10
2.2.1.2. Investigación Bibliográfica-documental	10
2.2.2. Población y Muestra	10
2.2.3. Recolección de Información	10
2.2.4. Procesamiento y análisis de datos	21
CAPÍTULO III.- RESULTADOS Y DISCUSIÓN	22

3.1 Análisis y discusión de los resultados	22
3.1.1 Plataformas web	22
3.1.2 ReactJS	23
3.1.3 ReactJS en el desarrollo de una plataforma web	23
3.1.4 Nest.js	24
3.1.5 Axios	25
3.1.8 IDE de desarrollo Visual Studio	26
3.1.9 Determinación de la metodología de desarrollo	26
3.2 Desarrollo de la propuesta	26
3.2.1 Fase I: Planificación	26
3.2.2 Fase II: Diseño	33
3.2.3 Fase II: Codificación	41
3.2.4 Fase IV: Pruebas	76
3.2.5 Fase V: Implantación	84
CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES	86
4.1 Conclusiones	86
4.2 Recomendaciones	87
Bibliografía	88
Anexos	90

Índice de tablas

Tabla 1: Escala de Likert	8
Tabla 2: Preguntas aplicadas.....	9
Tabla 3: Población	10
Tabla 4: Entrevista a la gerente.....	18
Tabla 5: Entrevista a la persona responsable de la oferta de los cursos.....	20
Tabla 6: Análisis de una plataforma y ReactJS.....	24
Tabla 7: Ventajas y desventajas de Nest.js	25
Tabla 8: Asignación de roles.....	27
Tabla 9: Modelo historias de usuario.....	29
Tabla 10: Historia para la definición de la estructura de la plataforma web	29
Tabla 11: Historia para el diseño de la base de datos	30
Tabla 12: Historia para página informativa para la oferta de los cursos.....	30
Tabla 13: Historia para aparatado para la administración de la plataforma.....	31
Tabla 14: Historia para aparatado para los profesores	31
Tabla 15: Historia para aparatado para los estudiantes.....	32
Tabla 16: Estimación de historias de usuario	32
Tabla 17: Plan de iteraciones	33
Tabla 18: Tarjeta CRC 1	34
Tabla 19: Tarjeta CRC 2.....	34
Tabla 20: Tarjeta CRC 3	34
Tabla 21: Iteración 1	35
Tabla 22: Iteración 2	37
Tabla 23: Iteración 3	38
Tabla 24: Iteración 4	39
Tabla 25: Iteración 5	40
Tabla 26: Pruebas de aceptación 001	77
Tabla 27: Pruebas de aceptación 002.....	77
Tabla 28: Pruebas de aceptación 003.....	78
Tabla 29: Pruebas de aceptación 004.....	79

Índice de figuras

Figura 1: Pregunta 1.....	11
Figura 2: Pregunta 2.....	12
Figura 3: Pregunta 3.....	12
Figura 4: Pregunta 4.....	13
Figura 5: Pregunta 5.....	14
Figura 6: Pregunta 6.....	15
Figura 7: Pregunta 7.....	15
Figura 8: Pregunta 8.....	16
Figura 9: Pregunta 9.....	17
Figura 10: Pregunta 10.....	18
Figura 11: Arquitectura del sistema.....	28
Figura 12: Diseño de base de datos	36
Figura 13: Página informativa	37
Figura 14: Página administrativa	38
Figura 15: Página para profesores	39
Figura 16: Página para estudiantes	40

RESUMEN EJECUTIVO

Con el transcurso del tiempo la educación ha cambiado gracias a la tecnología, siendo así que tanto en las instituciones públicas como privadas es indispensable una plataforma complementaria a la enseñanza tradicional. Visto desde este punto, no cabe duda de que las plataformas de educación online permiten mejorar la comunicación a través de múltiples canales, las clases son más didácticas y existe un mejor control de los alumnos y profesores; en esta perspectiva las empresas de capacitación que ofertan cursos online también hacen uso de estas tecnologías, además incrementan la usabilidad y comodidad para que la educación sea completamente online.

En base a lo que antecede, el presente proyecto tiene la finalidad de desarrollar e implementar una plataforma de educación online en la empresa GAIA de la ciudad de Ambato. Cabe señalar que la empresa en mención no contaba con una plataforma de educación online y tampoco tenía un medio para ofertar los cursos online. Por lo tanto, la implementación del proyecto permite brindar un mejor servicio a sus estudiantes y aumentar la demanda de los cursos.

La plataforma de educación online fue desarrollada en ReactJS, para mejorar el rendimiento y la carga del sitio web, así mismo para gestionar los estados se utilizó Redux y para la interacción con la API se ocupó la herramienta Axios. La API se creó con el framework Nest.js. Se implementó el sistema de base de datos relacional llamado MySQL por su usabilidad y código abierto. Además, se realizó el deploy de los proyectos de React con la herramienta yarn. Finalmente, la metodología que se utilizó para el diseño y el desarrollo de la plataforma fue la metodología Extreme Programming (XP).

Palabras clave: Plataforma web, ReactJS, NextJS, MySQL, Vercel

ABSTRACT

Over time, education has changed thanks to technology, and in both public and private institutions a complementary platform to traditional teaching is essential. Seen from this point, there is no doubt that online education platforms allow communication to be improved through multiple channels, classes are more didactic and there is better control of students and teachers; In this perspective, training companies that offer online courses also make use of these technologies, in addition to increasing usability and comfort so that education is completely online.

Based on the foregoing, this project has the purpose of developing and implementing an online education platform in the GAIA company in the city of Ambato. It should be noted that the company in question did not have an online education platform and did not have a means to offer online courses. Therefore, the implementation of the project allows to provide a better service to its students and increase the demand for the courses.

The online education platform was developed in ReactJS, to improve the performance and loading of the website, likewise Redux was used to manage states and the Axios tool was used for interaction with the API. The API was built using the Nest.js framework. The relational database system called MySQL was implemented for its usability and open source. In addition, the React projects were deployed with the yarn tool. Finally, the methodology used for the design and development of the platform was the Extreme Programming (XP) methodology.

Keywords: Web platform, ReactJS, NextJS, MySQL, Vercel

CAPÍTULO I.- MARCO TEÓRICO

1.1 Tema de Investigación

PLATAFORMA DE EDUCACIÓN ONLINE UTILIZANDO REACTJS PARA MEJORAR LA OFERTA DE CURSOS DE LA EMPRESA GAIA DE LA CIUDAD DE AMBATO

1.1.1 Planteamiento del problema

A nivel mundial las transformaciones que diariamente surgen influyen significativamente en el manejo de cada empresa, la competencia desenfrenada, el acelerado desarrollo de las tecnologías, y la búsqueda por alcanzar resultados satisfactorios constituyen elementos importantes para la permanencia de una empresa en el entorno actual.

En base a lo que antecede, no cabe duda que las empresas constituyen el motor que impulsa una economía, generan fuentes de empleo y proporcionan múltiples beneficios a la sociedad, sin embargo, si se dirige una mirada hacia América Latina se puede observar que a nivel empresarial pocas apuestan por un cambio ya sea a nivel de innovación o de avances tecnológicos tal como lo señala Revista Lideres (2021) “ Los países de América Latina invierten en promedio un 0,8% de su PIB en I+D, en contraste con el del 2,8% en EE.UU., el 2,5% de la Unión Europea o del 2,2% en China, con estas cifras, se explica mejor la poca competitividad de las economías de la región, en su mayoría configuradas por mipymes que no logran crecer lo mismo que sus pares en las economías más avanzadas, ni tampoco consiguen generar empleo de calidad ni implementar tecnologías que les aporten valor agregado”[1].

Dicha situación esta enlazada a la resistencia de las empresas ante los nuevos cambios, limitándose a formar parte de las nuevas transformaciones tecnológicas sin darse cuenta de los múltiples beneficios que les proporcionarían.

En Ecuador el tema empresarial juega un rol de gran importancia, más si se refiere al sector de asesorías y capacitación, el cual ha tenido un gran crecimiento debido al interés que poseen las empresas y las personas por actualizar y fortalecer sus conocimientos.

No obstante, a raíz de su inclusión dicho sector ha tenido que adaptarse a los grandes cambios que surgen bajo este escenario ya que la formación y capacitación están ligadas a las nuevas tecnologías viéndose a estas como una herramienta para transmitir conocimientos efectivos y de calidad yéndose más allá de los modelos convencionales tal como lo explica Freire-Avilés “Muchos de los instrumentos de formación, en gran parte hoy están disponibles gracias a las nuevas Tecnologías de Información y Comunicación (TIC), las cuales posibilitan adquirir conocimientos y competencias, incluso, fuera de las estructuras educativas tradicionales”[2].

En la ciudad de Ambato, se encuentra la compañía GAIA constituida el 18 de febrero del 2016 la cual tiene como objetivo posesionarse en ECUADOR como un centro de capacitación integral, siendo su primer campo de trabajo la ciudad de Ambato.

Sin lugar a duda esta empresa ambateña busca ampliar y fortalecer los conocimientos de la población mediante una mejor oferta de sus cursos, además, de activar el gen emprendedor mediante su Escuela de Emprendedores, en tal sentido, tienen un gran interés de cambiar su metodología tradicional y darles paso a las nuevas plataformas digitales, las mismas que han ganado lugares significativos para la construcción de conocimientos.

1.2 Antecedentes Investigativos

Según MEDINA, Erazo Wilmer Fabricio [1] en su tesis “EVALUACIÓN DEL USO DE LAS PLATAFORMAS VIRTUALES EN LOS ESTUDIANTES DEL PROGRAMA DE MAESTRÍA EN DOCENCIA Y GERENCIA EDUCATIVA DE LA UNIDAD DE POSTGRADO INVESTIGACIÓN Y DESARROLLO DE LA UNIVERSIDAD DE GUAYAQUIL. PARA FORTALECER SUS CONOCIMIENTOS TECNOLÓGICOS. DISEÑO DE UN MANUAL” Trabajo realizado en la Universidad de Guayaquil en el año (2013) se pudo determinar que:

- Como podemos apreciar en las respuestas de nuestros encuestados, más del 65% considera necesario implementar nuevas formas de aprendizaje en que las Plataformas Virtuales se conviertan en herramientas didácticas de colaboración para los procesos de enseñanza-aprendizaje.
- De las opiniones de las personas entrevistadas, pudimos detectar que un gran porcentaje recomendaría a terceras personas estudiar en la modalidad educativa a distancia, y se resaltaron muchos de los principales atractivos de esta modalidad. Todos ellos vertieron criterios que nos hacen llegar a la conclusión de que la Unidad de Postgrado Investigación y Desarrollo de la Universidad de Guayaquil, no puede dejar de considerar esta problemática y debe incursionar en la modalidad educativa online en el país y por qué no a nivel internacional.

Según López Hinojosa José Daniel [2] en su tesis “ARQUITECTURA DE SOFTWARE BASADA EN MICROSERVICIOS PARA DESARROLLO DE APLICACIONES WEB DE LA ASAMBLEA NACIONAL” Trabajo realizado en la Universidad Técnica del norte en el año (2017) se pudo determinar que:

- El producto de software que se genera carece de principios de modularidad debido al uso de la arquitectura implícita de la plataforma de desarrollo utilizada. Como resultado las aplicaciones adquieren un carácter monolítico embebiendo en una sola unidad ejecutable todas sus funcionalidades lo que incide tanto en su desarrollo como mantenimiento.
- De forma unánime se concuerda que es de vital importancia la aplicación de una arquitectura de software porque permite aumentar la calidad del producto debido a la estandarización de construcción de aplicaciones.

Según María Salomé Dávila Silva [3] en su tesis “Análisis, diseño e implementación de un sistema de aula virtual para capacitación personal en la empresa UNDERMEDIA S.A” Trabajo realizado en la Universidad de las Fuerzas Armadas de Ecuador en el año (2013) se pudo determinar que:

- Los parámetros dictados por PACIE en las fases de Presencia y Alcance permitieron realizar un entorno virtual de aprendizaje que lleva la imagen corporativa de la empresa y se enfoca a los estándares, evaluación y destrezas que los empleados deben adquirir para realizar sus labores diarias.
- La utilización de SCRUM como metodología central en el proyecto permitió que el tiempo de desarrollo del mismo sea corto, produciendo un sistema de calidad en un tiempo reducido.

Según Chica González Jhon Denver y Rodriguez Mendoza Karla Lilibeth [4] en su tesis “DESARROLLO Y EJECUCIÓN DE UN AULA VIRTUAL PARA CURSOS DE FORMACIÓN CONTINUA PARA DOCENTES NO UNIVERSITARIOS EN PEQUEÑAS EMPRESAS” Trabajo realizado en la Universidad de Guayaquil en el año (2020) se pudo determinar que:

- Se identificó un listado de plataformas open source para realizar cursos en modalidades e-learning y/o blended learning resaltando sus características.
- Se desarrolló una comparación de las plataformas investigadas permitiendo evaluar las herramientas en base a una serie de parámetros específicos para una validación fundamentada.

Según Lincango Córdor Luis Isaías [5] en su tesis “Estudio de factibilidad para el diseño e implementación de la plataforma de educación on-line en la Facultad de Ciencias Económicas de la Universidad Central del Ecuador.” Trabajo realizado en la Universidad Central del Ecuador en el año (2018) se pudo determinar que:

- Entre los aspectos más seleccionados que tiene en cuenta los encuestados a la hora de seleccionar un curso en una plataforma on-line son: la claridad de la información seleccionados por el 61,4% de los encuestados, seguido por la calidad del servicio y la estructura de la información con el 53,0% y el 47,0%. Mientras que los aspectos asociados a la disponibilidad y al entorno amigable de la plataforma fueron seleccionados por el 33,2% y el 31,9%.

- En cuanto al interés de los estudiantes por seguir cursos virtuales en la Facultad de Ciencias Económicas de la Universidad Central del Ecuador, el 95,1% manifiesta que le gustaría dicha oferta académica.

1.3 Fundamentación Teórica

Tecnologías de la información

Las Tecnologías de la Información y la Comunicación (TIC) se desarrollan a partir de los avances científicos producidos en los ámbitos de la informática y las telecomunicaciones. Es el conjunto de tecnologías que permiten el acceso, producción, tratamiento y comunicación de información presentada en diferentes códigos (texto, imagen, sonido, video)[6].

Plataformas virtuales

Una plataforma e-learning, plataforma educativa web o Entorno Virtual de Enseñanza y Aprendizaje es una aplicación web que integra un conjunto de herramientas para la enseñanza-aprendizaje en línea, permitiendo una enseñanza no presencial (e-learning) y/o una enseñanza mixta (b-learning), donde se combina la enseñanza en Internet con experiencias en la clase presencial[7].

Plataforma de educación online utilizando ReactJS

El objetivo básico de una plataforma virtual, o también llamada, aula virtual, es crear un entorno donde se integren el uso de las distintas herramientas y aplicaciones que funcionen a través de Internet en el proceso de enseñanza-aprendizaje de nuestros alumnos[8].

ReactJS

React te ayuda a crear interfaces de usuario interactivas de forma sencilla. Diseña vistas simples para cada estado en tu aplicación, y React se encargará de actualizar y renderizar de manera eficiente los componentes correctos cuando los datos cambien[9].

Oferta de cursos

Las ofertas de cursos son variadas y responden a una necesidad de adquisición de conocimiento de forma ágil, concreta y aplicada para el desarrollo de una solución en un campo disciplinar, interdisciplinar o transversal en diferentes áreas del conocimiento[10].

Formación académica complementaria

Esta se refiere a toda la formación anexa a estudios formales, a los que se puede acceder mediante capacitaciones, cursos, talleres, jornadas o seminarios a los que atiende un individuo y de los que obtiene una formación integral en cierta área[11].

Son referidos con la palabra “complementaria” debido a que complementan a la educación formal pero nunca la sustituyen, siendo de igual manera una alternativa viable de formación para estudiantes y profesionales que deciden pulir sus habilidades[11].

Educación

Las formas más tradicionales que el uso común hace del significado de educación proceden de nuestra experiencia colectiva histórica y en muy diversos autores y pasajes históricos encontramos argumentaciones que se han transmitido como acervo cultural colectivo y forman parte de la experiencia y de la memoria colectiva que identifica la educación en los siguientes usos comunes: 1) la educación es cortesía, civismo y urbanidad; 2) la educación es crianza material y espiritual; 3) la educación es perfeccionamiento; 4) la educación es formación[12].

1.4 Objetivos

1.4.1 Objetivo general

Implantar una plataforma de educación online utilizando ReactJS para mejorar la oferta de cursos de la empresa GAIA.

1.4.2 Objetivos específicos

- Identificar los requerimientos de la empresa GAIA y de sus clientes.
- Investigar la aplicación de ReactJS para el desarrollo de una plataforma online.
- Implementar ReactJS en el desarrollo de la plataforma de educación online para mejorar la oferta de cursos de la empresa GAIA.
- Implantar una plataforma de educación online que permita mejorar la oferta de cursos de la empresa GAIA.

CAPÍTULO II.- METODOLOGÍA

2.1. Materiales

Para poder conocer la realidad de la empresa GAIA y la manera en la que realizan la oferta de cursos fue necesario obtener información mediante una entrevista a las personas que dirigen la empresa, el objetivo de la entrevista es conocer qué tipo de tecnologías utilizan en la oferta de cursos y saber si estarían dispuestos a actualizar la forma en la que publicitan sus cursos en internet. Además, se realizó una encuesta a los estudiantes con el fin de conocer sus opiniones acerca de la forma que GAIA tiene para ofertar sus cursos.

Encuesta aplicada a los estudiantes de los cursos de GAIA

El objetivo de esta encuesta es para determinar qué tipo de dificultades tuvieron que pasar los estudiantes para obtener información sobre los cursos de GAIA y para inscribirse en dichos cursos. Adicionalmente, se pregunta que tan de acuerdo estarían los estudiantes de implementar una plataforma web para mejorar la oferta de los cursos de GAIA.

La confiabilidad y la validez de esta encuesta fue validada mediante el coeficiente de alfa Cronbach, se obtuvo un coeficiente de 0.81 que refleja un nivel alto de confianza (Ver anexo A.1).

El tipo de escala que se utilizó en la encuesta fue Likert con 5 opciones de respuesta, en la siguiente tabla se puede observar la escala completa:

Tabla 1: Escala de Likert

Elaborado por: Investigador

Escala de Likert				
1	2	3	4	5
Totalmente desacuerdo	En desacuerdo	Indiferente	De acuerdo	Totalmente de acuerdo

Tabla 2: Preguntas aplicadas

Elaborado por: Investigador

N°	Preguntas aplicadas	1	2	3	4	5
1	¿Es difícil encontrar los cursos de GAIA en internet?					
2	¿La información que existe en las redes sociales sobre los cursos de GAIA es incompleta?					
3	¿Antes de adquirir el curso en GAIA, usted no conocía los temas que se iban a tratar dentro del curso?					
4	¿Fue difícil matricularse en el curso de su interés en GAIA?					
5	¿Antes de adquirir el curso, no tenía información sobre el profesor que iba a impartir el curso?					
6	¿Las personas cercanas a usted no conocen sobre la existencia de los cursos de la empresa GAIA?					
7	¿El pago del curso fue difícil e incómodo?					
8	¿Tuvo que cambiar su agenda diaria para poder asistir a los cursos presenciales de GAIA?					
9	¿Los cursos online son más accesibles que los cursos presenciales?					
10	¿Cree usted, que una plataforma web mejoraría la oferta de cursos en la empresa GAIA?					

2.2. Métodos

2.2.1. Modalidad de Investigación

Con la finalidad de recoger los datos y fundamentos necesarios para el desarrollo de este proyecto de investigación se utilizará una revisión bibliográfica y documental sobre estudios realizados referentes al tema planteado, de igual forma, la investigación de campo servirá como objeto de fuente para la investigación, finalmente se utilizará las modalidades especiales, ya que se refiere a la creación de proyectos creativos e innovadores con el objetivo de desarrollar tecnología y desarrollo de software.

2.2.1.1. Investigación Campo

Se realiza una investigación de campo ya que se necesita obtener información de los gerentes de la empresa GAIA para conocer la realidad de la empresa y además se debe obtener una información analítica de sus estudiantes.

2.2.1.2. Investigación Bibliográfica-documental

Se realiza un estudio bibliográfico para conocer las funcionalidades de ReactJS además de sus ventajas y desventajas para ser usado en la creación de una plataforma de educación online.

2.2.2. Población y Muestra

En este aspecto de la investigación la población que se considera oportuna está conformada por el personal administrativo, los docentes y los estudiantes de la empresa GAIA.

Tabla 3: Población

Elaborado por: Investigador

Población	Número	Porcentaje
Personal administrativo	5	17%
Docentes	4	14%
Estudiantes	20	69%
Total	29	100%

En base al cuadro que antecede contamos con una población de 29 personas, es decir una población finita ya que conocemos con exactitud cuántos elementos tiene dicha población.

2.2.3. Recolección de Información

A continuación, se muestran los datos obtenidos de la encuesta realizada a los estudiantes.

Ítem 1: ¿Es difícil encontrar los cursos de GAIA en internet?

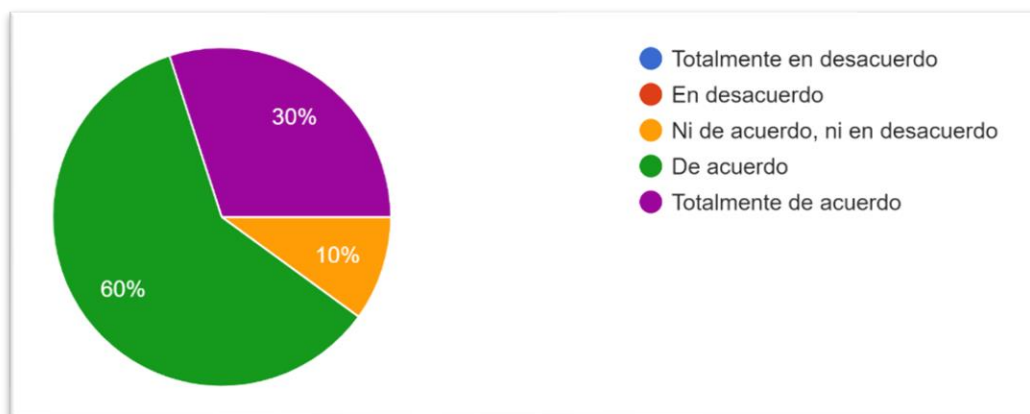


Figura 1: Pregunta 1

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

Como se puede observar en la Figura 1, de los 20 estudiantes encuestados el 60% está de acuerdo, seguidamente del 30% que opinan que están totalmente de acuerdo y, por último, el 10% de los encuestados están Ni de acuerdo, ni en desacuerdo. Estos datos indican que la mayoría de los estudiantes tuvieron problemas para encontrar los cursos de GAIA en internet.

Ítem 2: ¿La información que existe en las redes sociales sobre los cursos de GAIA es incompleta?

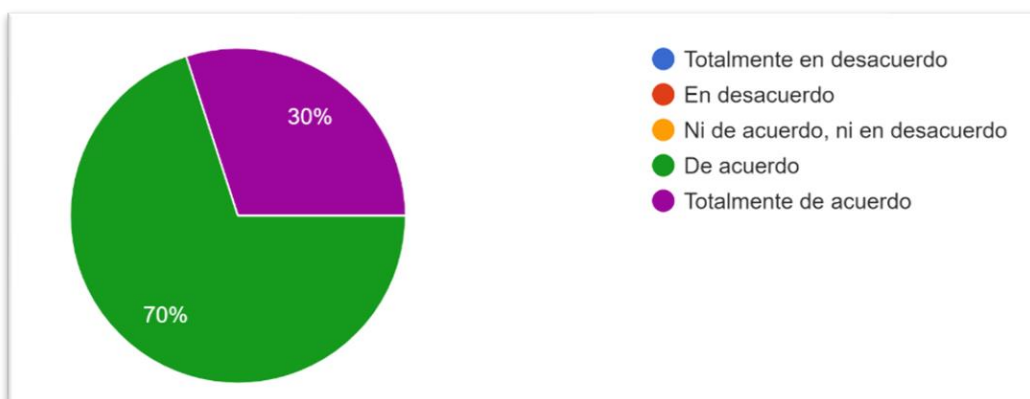


Figura 2: Pregunta 2

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

En la Figura 2 se puede observar que el 70% de los encuestados están de acuerdo, y el 30% de los estudiantes están totalmente de acuerdo con respecto a la poca información que existe de los cursos en las redes sociales de GAIA. Estos resultados indican que la mayoría de los estudiantes encontraron información incompleta sobre los cursos de GAIA en las redes sociales.

Ítem 3: ¿Antes de adquirir el curso en GAIA, usted no conocía los temas que se iban a tratar dentro del curso?

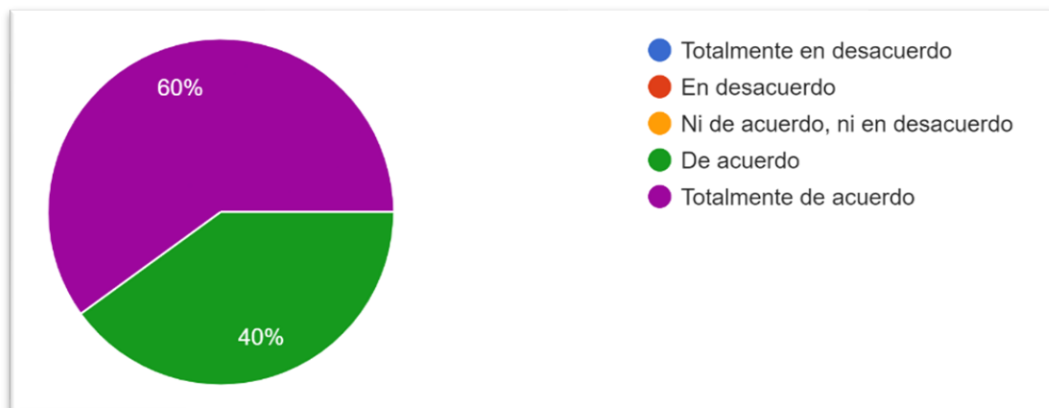


Figura 3: Pregunta 3

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

En la Figura 3 se puede observar que el 60% de los estudiantes encuestados responden que están totalmente de acuerdo y que el 40% están de acuerdo con la pregunta que corresponde al ítem número 3. Esto demuestra que los estudiantes no tenían una información completa de los contenidos de los cursos de la empresa GAIA.

Ítem 4: ¿Fue difícil matricularse en el curso de su interés en GAIA?

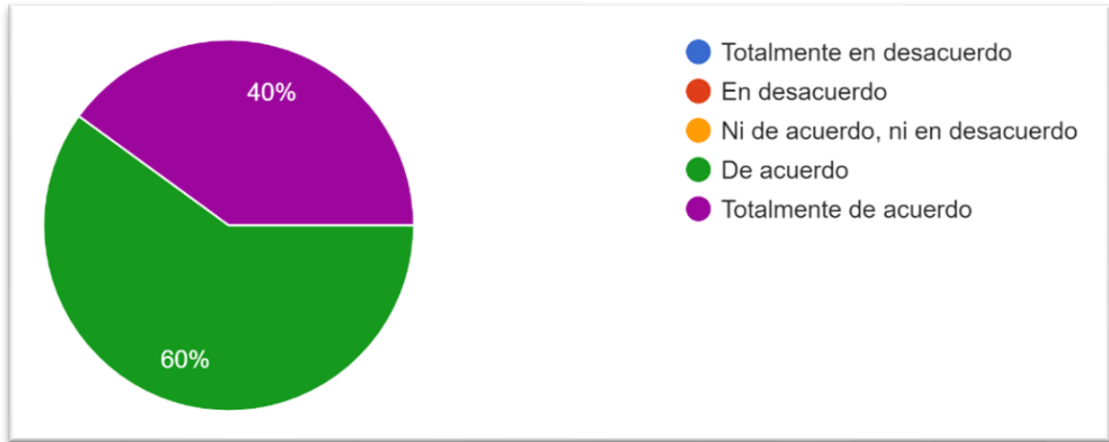


Figura 4: Pregunta 4

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados de la Figura 4, el 60% de los estudiantes encuestados está de acuerdo y el 40% restante está totalmente de acuerdo haciendo referencia a la dificultad para matricularse en los cursos de GAIA. Con estos resultados se puede concluir que la mayoría de los estudiantes no les resultó fácil el proceso de matriculación en los cursos de GAIA.

Ítem 5: ¿Antes de adquirir el curso, no tenía información sobre el profesor que iba a impartir el curso?

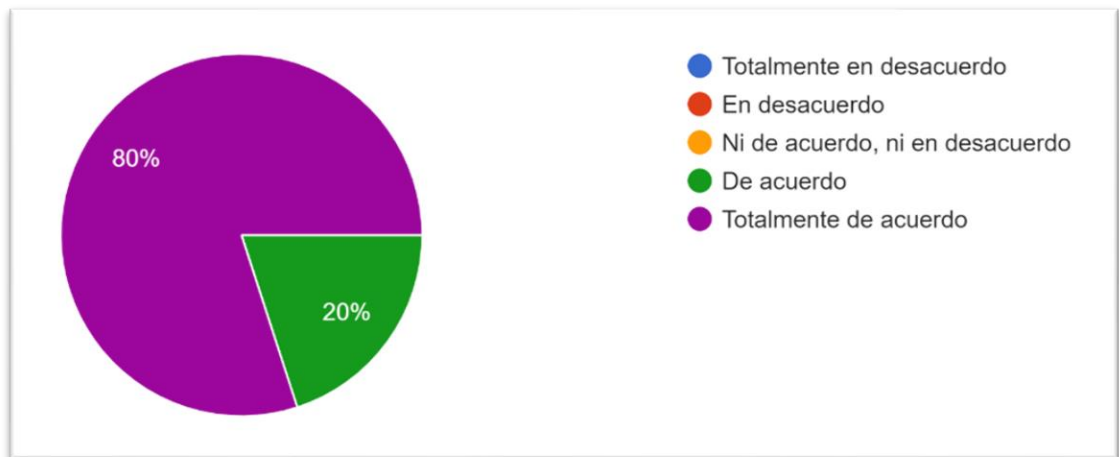


Figura 5: Pregunta 5

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

En la Figura 5 se puede observar que el 80% de los estudiantes encuestados están totalmente de acuerdo y que el 20% restante están de acuerdo con la pregunta de ítem número 5. Estos datos reflejan que los estudiantes de GAIA no tenían información del profesor que iba a impartir el curso.

Ítem 6: ¿Las personas cercanas a usted no conocen sobre la existencia de los cursos de la empresa GAIA?

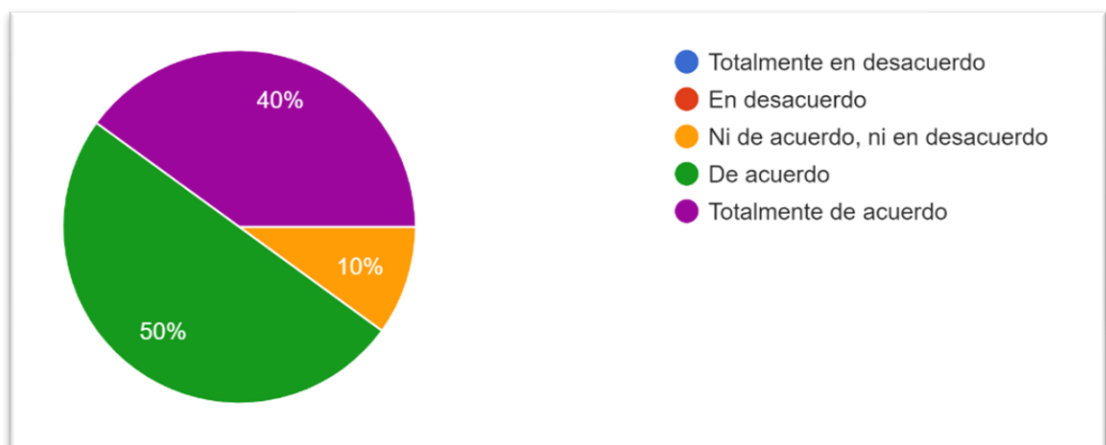


Figura 6: Pregunta 6

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

Según los resultados que refleja la Figura 6, el 50% de los estudiantes están de acuerdo, el 40% de los encuestados están totalmente de acuerdo y el 10% están ni de acuerdo, ni en desacuerdo. Con estos datos se puede concluir que la mayoría de los estudiantes no tienen personas cercanas que conozcan los cursos de GAIA.

Ítem 7: ¿El pago del curso fue difícil e incómodo?

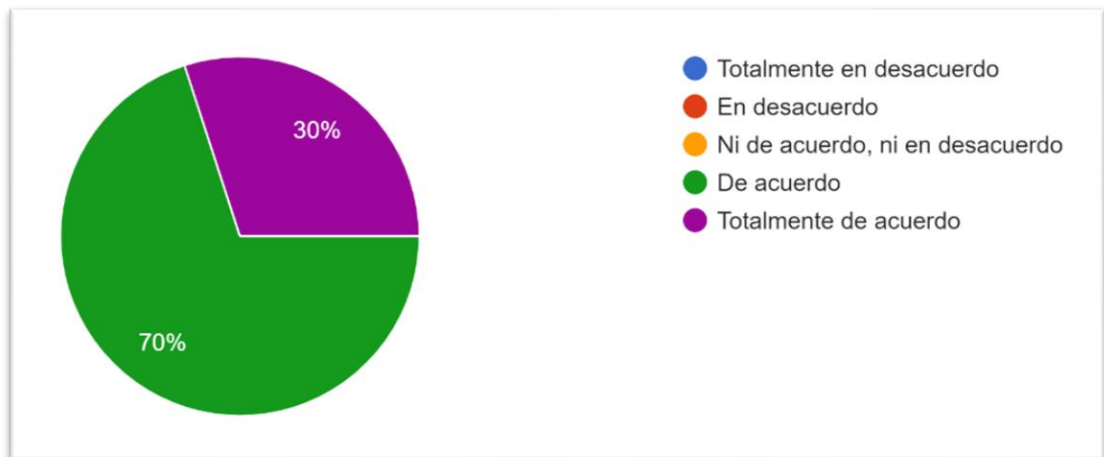


Figura 7: Pregunta 7

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con la Figura 7, el 70% de los estudiantes encuestados están de acuerdo y el 30% restante están totalmente de acuerdo con la pregunta del ítem número 7. Estos resultados obtenidos indican que el pago de los cursos de GAIA fue difícil e incómodo para los estudiantes de los cursos de GAIA.

Ítem 8: ¿Tuvo que cambiar su agenda diaria para poder asistir a los cursos presenciales de GAIA?

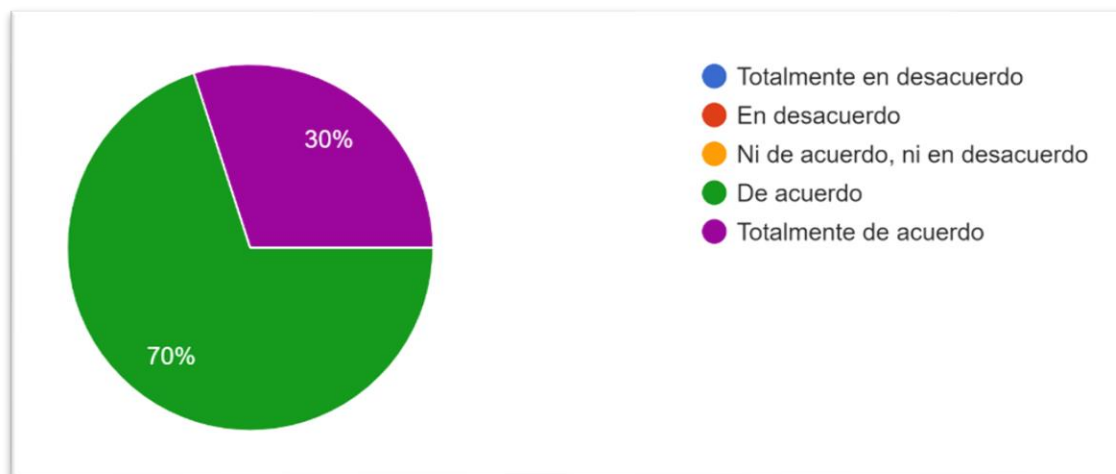


Figura 8: Pregunta 8

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

En la Figura 8 se puede observar que, el 70% de los estudiantes encuestados respondieron de acuerdo y el 30% restante respondieron que se encuentran totalmente de acuerdo. Con los datos mencionados anteriormente se puede determinar que los estudiantes tuvieron que cambiar su rutina diaria para poder asistir a los cursos presenciales de la empresa GAIA.

Ítem 9: ¿Los cursos online son más accesibles que los cursos presenciales?

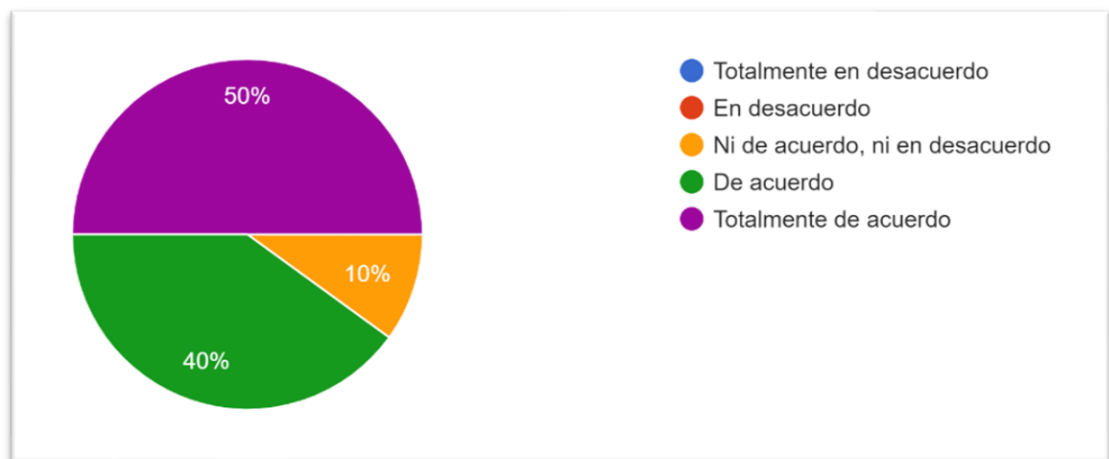


Figura 9: Pregunta 9

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados reflejados en la Figura 9, el 50% de los estudiantes encuestados están totalmente de acuerdo, el 40% de acuerdo y por último el 10% de los estudiantes están ni de acuerdo, ni en desacuerdo. Con estos datos se puede determinar que los estudiantes de GAIA opinan que los cursos online son más accesibles que los cursos tradicionales.

Ítem 10: ¿Cree usted, que una plataforma web mejoraría la oferta de cursos en la empresa GAIA?

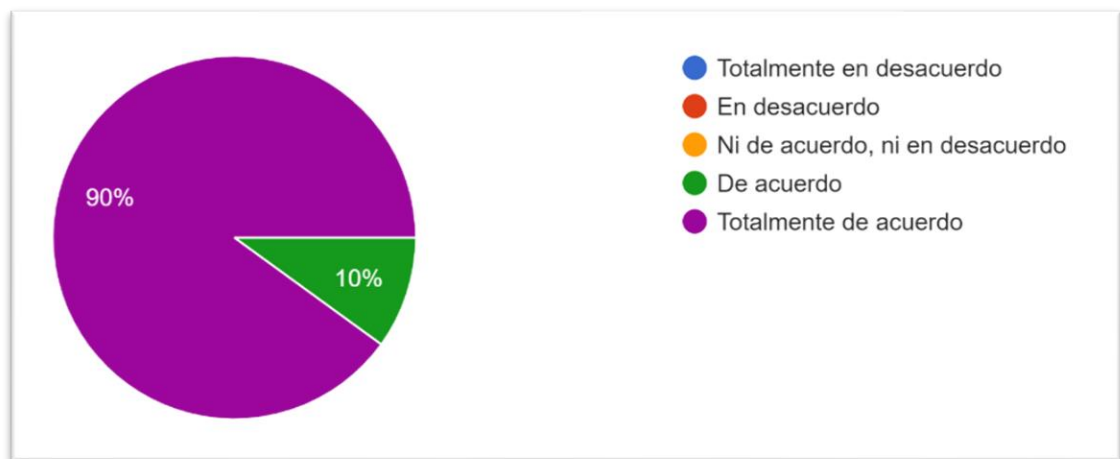


Figura 10: Pregunta 10

Elaborado por: Investigador

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

En la Figura 10 se puede observar que, el 90% de los estudiantes encuestados están totalmente de acuerdo y que el 10% de los estudiantes están de acuerdo. Con estos datos se puede evidenciar que los estudiantes de GAIA están a favor de la implementación de una plataforma web.

Resultados de la entrevista aplicada a la gerente de la empresa GAIA

Se realizó una entrevista a la ingeniera Pilar Rodríguez gerente de la empresa GAIA, para conocer la realidad y las necesidades de esta.

Tabla 4: Entrevista a la gerente

Elaborado por: Investigador

Preguntas	Respuestas
¿En promedio, cuantos estudiantes tienen matriculados al mes?	20 estudiantes.
¿Qué cantidad de cursos ofrecen actualmente y cuáles son?	Actualmente 3 cursos. Reparación y mantenimiento de computadoras, Excel intermedio y maquetación.

¿Qué herramienta utilizan para almacenar la información de los estudiantes?	La herramienta que utilizamos es Excel.
¿Qué métodos de pago usan para el cobro de los cursos?	Solamente efectivo.
¿Los profesores necesitan compartir recursos con los estudiantes?	Sí, por lo general se realizan por grupos de WhatsApp o Facebook.
¿Cuántos profesores trabajan en la empresa?	Cuatro profesores.
¿Existe un examen final que compruebe los conocimientos adquiridos en el curso dirigido a los estudiantes?	Sí, el examen es presencial.
¿Quién organiza el contenido de los cursos?	Siempre damos libertad a los profesores, son ellos los que organizan sus cursos y los temas.
¿Qué información se solicita a los estudiantes en la inscripción de un curso?	Nosotros pedimos información como el número de cédula, nombres completos, dirección y a veces si es que tienen algún tipo formación académica.
¿Qué información se solicita a los profesores contratados?	Casi la misma que los estudiantes, pero sobre todo la experiencia que tienen y su formación.
Bajo su criterio, ¿por qué la oferta de cursos no genera interés en las personas?	Las personas se ven interesadas en los cursos por nuestras redes sociales, pero a veces preguntan por más información y no siempre vienen a nuestras oficinas.
¿Cómo son las instalaciones que la empresa posee para impartir los cursos?	Tenemos solamente dos aulas y eso nos dificulta crear más cursos. Además, nos limita el número de estudiantes que podemos aceptar en cada curso.

<p>¿Estaría usted dispuesto a implementar nuevas tecnologías en los servicios que la empresa ofrece?</p>	<p>Claro que sí, nuestro problema es que no hemos encontrado a una persona que nos ayude con el tema de las nuevas tecnologías.</p>
<p>Conclusión:</p> <p>La empresa cuenta con pocos estudiantes y profesores, las herramientas que utilizan para almacenar los datos de estas personas no son los recomendables. Los cursos se ofertan por las redes sociales y esto conlleva a varios problemas con la comunicación de posibles estudiantes. El poco uso de las nuevas tecnologías está generando pérdidas económicas y disminuye la calidad del servicio.</p>	

Resultados de la entrevista aplicada al responsable de la oferta de cursos de la empresa GAIA

Se realizó una entrevista al ingeniero Daniel Chimborazo para conocer de qué forma se realiza la oferta de cursos y cuáles son sus falencias.

Tabla 5: Entrevista a la persona responsable de la oferta de los cursos

Elaborado por: Investigador

Preguntas	Respuestas
<p>¿A través de qué medios digitales se realiza la oferta de cursos?</p>	<p>Tenemos presencia online a través de las redes sociales como Facebook y WhatsApp Business.</p>
<p>¿Existe una persona encargada en responder las dudas de los sujetos interesados en los cursos?</p>	<p>No, muchas veces nos escriben para preguntarnos sobre más información, pero nos damos cuenta después de horas o días.</p>
<p>¿Cuáles son las interrogantes que normalmente tienen las personas interesadas en los cursos?</p>	<p>¿Cuáles son los temas del curso? ¿Cómo se llama el profesor del curso? ¿Cuál es la dificultad del curso? ¿Dónde se ubican las aulas? ¿Tienen página web?</p>

¿Se publica información sobre los profesores que imparten los cursos en las páginas voceras de la institución?	No, por lo general en nuestras publicaciones solo ponemos información básica del curso.
¿Considera que implementando nuevas tecnologías en los servicios que la empresa ofrece provocaría una mejora en la oferta de los cursos?	Sí, porque nos ayudaría a captar más estudiantes y en los procesos de inscripción de los alumnos.
<p>Conclusión:</p> <p>Las redes sociales no son una herramienta adecuada para la oferta de los cursos, existe poca comunicación con los posibles clientes. La información que se publica en las redes sociales es escasa.</p>	

2.2.4. Procesamiento y análisis de datos

Procesamiento y análisis de la encuesta

Una vez realizada la encuesta se puede concluir que la gran mayoría de los estudiantes encuentran ciertas dificultades para encontrar información de los cursos de GAIA en internet. Además, se muestra una gran necesidad de una plataforma de educación online ya que los estudiantes estarían dispuestos a recibir cursos online. En la encuesta se observó que los estudiantes modificaron sus horarios de que hacer para poder asistir a los cursos de forma presencial.

Procesamiento y análisis de las entrevistas

La empresa GAIA cuenta con pocos estudiantes y cursos, las instalaciones que poseen son insuficientes para aumentar el número de estos. Existe una necesidad en actualizar las herramientas que la empresa utiliza para almacenar la información de los profesores y de los estudiantes. Las redes sociales no son los medios adecuados para la oferta de los cursos de GAIA, no tienen una correcta comunicación con los posibles estudiantes y la información de los cursos y de los profesores es muy escasa.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión de los resultados

Ahora que ya se han identificado los requerimientos y las necesidades de los estudiantes y de la empresa, es necesario analizar y establecer las tecnologías y metodología que se va a emplear para el desarrollo de la propuesta.

3.1.1 Plataformas web

Las plataformas web son un conjunto de programas cuyo objetivo principal es brindar una mejor ejecución de diferentes tareas en un solo sitio web. Existen varios tipos de plataformas web dependiendo de las tareas que se requieren ejecutar, como, por ejemplo:

- Plataformas sociales. Son plataformas en las cuales su principal objetivo es que diferentes usuarios puedan interactuar entre sí. Son sitios web en donde el usuario suele tener un perfil y puede chatear, subir fotos, escribir publicaciones. También se les conoce mejor como redes sociales.
- Plataformas de comercio electrónico. Estos sitios web permiten realizar ventas en internet con pedidos a domicilio, algunos ejemplos son: WooCommerce, Tiendanube, Shopify y Magento.
- Plataformas educativas. Son páginas web que intentan ofrecer una experiencia educativa igual o mejor que la enseñanza tradicional. Como ejemplos se puede mencionar Blackboard, e-College y Moodle.

Para que una plataforma web funcione correctamente debe estar compuesta por módulos que trabajen en conjunto para solucionar varias tareas y que a su vez se puedan desarrollar por separado. También, debe ser fácil de actualizar y debe ser capaz de manejar diferentes tipos de usuarios dentro de la plataforma, teniendo en cuenta la seguridad y la consistencia de los datos.

3.1.2 ReactJS

React es una biblioteca de interfaz de usuario desarrollada por Facebook para facilitar la creación de aplicaciones interactivas, con estado y componentes de interfaz de usuario reutilizables. Se utiliza en Facebook en su producción. ReactJS es mejor para renderizar interfaces de usuario complejas con alto rendimiento. El fundamento básico detrás de React es el concepto de DOM virtual. ReactJS usa DOM virtual de manera efectiva, que se puede representar en el lado del cliente o en el lado del servidor y comunicarse de un lado a otro.

React intenta ayudar a los desarrolladores a construir aplicaciones que usan datos que cambian todo el tiempo. Su objetivo es ser sencillo, declarativo y fácil de combinar. React sólo maneja la interfaz de usuario en una aplicación; React es la Vista en un contexto en el que se use el patrón MVC (ModeloVista-Controlador) o MVVM (Modelo-vista-modelo de vista).[13]

El Virtual DOM renderiza subárboles de nodos basados en cambios de estado. Hace la menor cantidad de manipulación DOM posible para mantener sus componentes actualizados. React es más ligero que Angular, se llena con el mínimo de condiciones y elimina la necesidad de utilizar elementos adicionales como complementos. React está en contra de dos vías vinculantes, intencionalmente se mantiene alejado de él y utiliza actualizaciones explícitas en su lugar.

3.1.3 ReactJS en el desarrollo de una plataforma web

Las plataformas web tienen ciertas características que son necesarias para el buen funcionamiento de estas. A continuación, se mencionan las características que una plataforma necesita junto con los atributos de ReactJS.

Tabla 6: Análisis de una plataforma y ReactJS

Elaborado por: Investigador

Características de una plataforma	Atributos de ReactJS
Interfaz fácil de renderizar	Usa un DOM virtual para un mejor desempeño en la renderización de la interfaz
Compuesta por varios programas	La forma en la que se desarrolla una aplicación con ReactJS es mediante componentes que facilita la ejecución de varias tareas
Debe fallar solamente en el proceso de compilación	A diferencia de otras tecnologías como Angular, ReactJS no falla en el tiempo de ejecución sino en el tiempo de compilación
Comunicación con el backend de forma segura	La comunicación con el backend se puede realizar mediante una API con los métodos propios de ReactJS
La interfaz debe actualizarse fácilmente	En cada componente la interfaz se diseña con JSX que es una extensión de JavaScript que permite mezclar JS y HTML. JSX permite utilizar variables y otros atributos dentro de HTML, esto facilita la creación de la interfaz
Sitio web seguro	ReactJS permite utilizar módulos para incrementar la seguridad de la aplicación

3.1.4 Nest.js

Nest (NestJS) es un marco para construir de manera eficiente y escalable Node.js aplicaciones del lado del servidor. Utiliza JavaScript progresivo, está construido con y es totalmente compatible TypeScript (todavía permite a los desarrolladores codificar en JavaScript puro) y combina elementos de OOP (Programación orientada a objetos), FP (Programación funcional) y FRP (Programación reactiva funcional).[14]

3.1.4.1 Ventajas y desventajas

Tabla 7: Ventajas y desventajas de Nest.js

Elaborado por: Investigador

Ventajas	Desventajas
Tiene un alto rendimiento	No ofrece una gestión gratuita
Utiliza tecnologías modernas	La depuración del código es ardua
Contiene componentes y bibliotecas ya integrados	Es un framework difícil para personas con poca experiencia
Se puede crear varios tipos de servicios	El estilo de la sintaxis es parecido a Angular
Código abierto	La compilación del código no es rápida

El principal objetivo de usar el framework Nest.js es para desarrollar una API moderna y fácil de usar para que la plataforma tenga un funcionamiento rápido y eficiente. Este framework permite crear los servicios que se requieren para ser consumidos en el front-end con React.JS.

3.1.5 Axios

Axios es un cliente HTTP basado en promesas que funciona en navegador o del lado del servidor con node.js. Es una de las librerías más populares y cuenta con algunas características que la API Fetch no proporciona por defecto como son los interceptors, convierte la respuesta del servidor a JSON automáticamente y también proporciona protección contra ataques cross-site request forgery (XSRF).[15]

3.1.6 Redux

Es una herramienta compatible con React que se encarga de la gestión del estado de la aplicación. A medida que la tecnología avanza las aplicaciones tienen más capacidades y podemos implementar más y mejores funcionalidades. Pero esto provoca que la complejidad de nuestros proyectos se dispare. Una de las razones por las que aumenta la dificultad del mantenimiento es el manejo del estado, tanto del modelo (información de terceros, datos creados localmente, etc.) como de la interfaz (paginación, listas, elementos seleccionados, etc.).[16]

3.1.7 Ant Design

Ant es un framework enfocado al diseño y ofrece una gran variedad de herramientas y componentes disponibles para su uso y reutilización. La primera versión de este framework fue la 0.9.0 y actualmente se encuentra en la versión 4.6.2.[17]

En la mayoría de los casos se usa React para encapsular una librería de componentes. Algunos websites desarrollados usando Ant Design son: Tencent, Alibaba, Didi...[17]

3.1.8 IDE de desarrollo Visual Studio

Visual Studio es un IDE desarrollado por Microsoft, permite codificar en varios lenguajes como PHP, JavaScript, .NET y C++. Es compatible con plugins y tiene varias terminales integradas en el software. De igual forma es uno de los IDE más utilizados sobretodo en el desarrollo de páginas web modernas que usan frameworks como Laravel, ReactJS o AngularJS. Por todo lo anterior este IDE tiene las características necesarias para codificar una plataforma web usando el framework NextJS con ReactJS.

3.1.9 Determinación de la metodología de desarrollo

La metodología que se va a usar en el presente proyecto de investigación será Extreme Programming, esta metodología está diseñada para realizar un proyecto con pocas personas a cargo del desarrollo de software a diferencia de las demás metodologías ágiles, que están diseñadas para trabajar en grupo. Esta metodología es idónea para proyectos que tienen un tiempo de desarrollo limitado y una de sus características que la diferencia de las demás metodologías es que acepta cambios en cada iteración.

3.2 Desarrollo de la propuesta

3.2.1 Fase I: Planificación

3.2.1.1 Levantamiento de la información

Para el levantamiento de la información, se realizó una entrevista a los directores de la empresa GAIA con el objetivo de conocer la realidad de la empresa y sus necesidades. Además, se realizó una encuesta a los estudiantes de los cursos de GAIA para obtener información de la experiencia de los estudiantes en el proceso de ingreso de los cursos y de las clases.

3.2.1.2 Descripción del cliente

Actualmente la empresa GAIA proporciona cursos de capacitación de forma presencial y el manejo de la información de los estudiantes y de los profesores se realiza de forma manual, la forma en la que GAIA oferta sus cursos es mediante redes sociales. Teniendo en cuenta lo anterior, se determinó qué procesos son necesarios para la creación de la plataforma de educación web.

La plataforma debe contener los siguientes módulos:

- **Apartado para la oferta de los cursos**

Este módulo se utilizará para ofrecer información y publicitar los cursos de GAIA.

- **Apartado administrativo de la plataforma**

En este módulo el personal administrativo llevará el control de la plataforma.

- **Apartado de los estudiantes**

En este módulo los estudiantes tendrán acceso a los cursos de GAIA.

- **Apartado de los profesores**

En este módulo los profesores tendrán acceso a los cursos que deben impartir.

3.2.1.3 Asignación de roles del proyecto

Tabla 8: Asignación de roles

Elaborado por: Investigador

Nombre	Descripción	Rol
Franklin Guillén	Persona a cargo de realizar la planificación, diseño, desarrollo, codificación y pruebas para la implantación de la plataforma de educación online.	Programador
Ing. David Guevara	Persona con el deber de revisar y controlar el trabajo realizado por el	Coach

	programador para que el proyecto se finalice según la planificación.	
Ing. Daniel Chimborazo	Persona que necesita la plataforma de educación online y conoce todos los requisitos de la plataforma.	Tester

3.2.1.4 Arquitectura del sistema

Para la arquitectura de la plataforma web se utilizó varias tecnologías modernas, una de ellas es Nest.js para trabajar con el back-end y desarrollar una API Rest. Para la base de datos se utilizó MySQL por ser una base de datos relacional y de código libre y para el desarrollo del front-end se ocupó React.js.

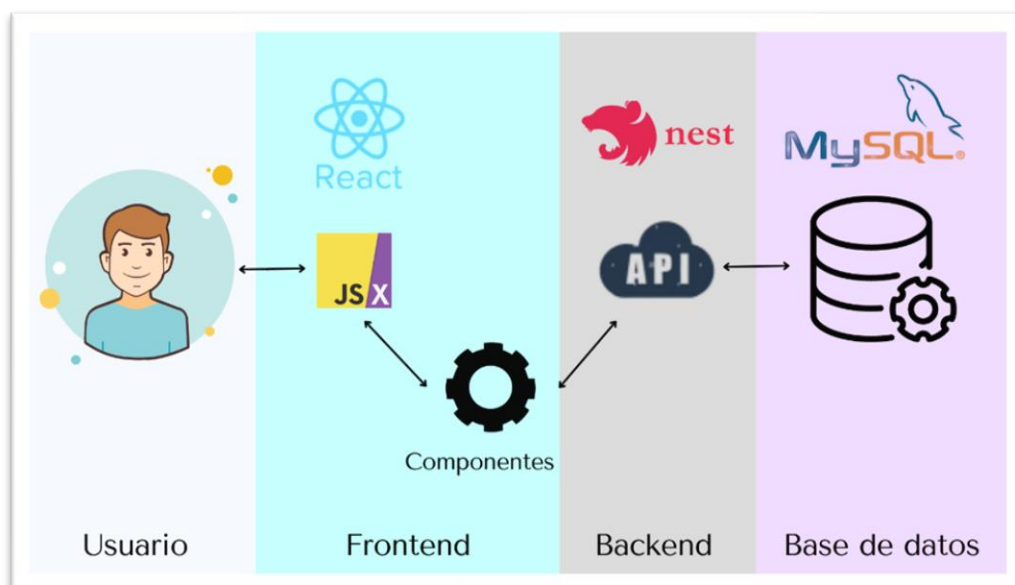


Figura 11: Arquitectura del sistema

Elaborado por: Investigador

3.2.1.5 Historias de usuario

Se elaboraron historias de usuario con la perspectiva del usuario y con el conocimiento del investigador para determinar las principales funcionalidades de la plataforma de educación web. El modelo que se utilizó para la creación de las historias de usuario fue el siguiente:

Tabla 9: Modelo historias de usuario

Elaborado por: Investigador

Historia de Usuario	
Número:	Usuario:
Nombre Historia:	
Prioridad en Negocio:	Riesgo en desarrollo:
Puntos Estimados:	Iteración Asignada:
Responsable:	
Descripción:	
Observaciones:	

A continuación, se muestran las siguientes historias de usuario elaboradas en base a la información obtenida por el cliente, el programador y los requerimientos de la plataforma.

Tabla 10: Historia para la definición de la estructura de la plataforma web

Elaborado por: Investigador

Historia de Usuario	
Número: 001	Usuario: programador
Nombre Historia: Definición de la estructura de la plataforma web	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Alta
Puntos Estimados: 5	Iteración Asignada: 1
Responsable: Franklin Guillén	
Descripción: Definir la estructura de la plataforma para establecer la unión de las tecnologías que se usarán en el desarrollo.	
Observaciones: Ninguna.	

Tabla 11: Historia para el diseño de la base de datos

Elaborado por: Investigador

Historia de Usuario	
Número: 002	Usuario: programador
Nombre Historia: Diseño de la base de datos	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Alta
Puntos Estimados: 5	Iteración Asignada: 1
Responsable: Franklin Guillén	
Descripción: Diseñar la base de datos para que la plataforma web tenga acceso a la información almacenada.	
Observaciones: Ninguna.	

Tabla 12: Historia para página informativa para la oferta de los cursos

Elaborado por: Investigador

Historia de Usuario	
Número: 003	Usuario: programador
Nombre Historia: Página informativa para la oferta de los cursos	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Media
Puntos Estimados: 2	Iteración Asignada: 2
Responsable: Franklin Guillén	
Descripción: Permite que los usuarios conozcan sobre la oferta de cursos que tiene GAIA.	
Observaciones: Los usuarios no deben estar registrados para ver esta página.	

Tabla 13: Historia para aparatado para la administración de la plataforma

Elaborado por: Investigador

Historia de Usuario	
Número: 004	Usuario: programador
Nombre Historia: Aparatado para la administración de la plataforma	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Alta
Puntos Estimados: 5	Iteración Asignada: 3
Responsable: Franklin Guillén	
Descripción: Tiene que permitir modificar ciertos aspectos de la plataforma como los cursos, profesores y estudiantes.	
Observaciones: Para poder ingresar a esta parte de la plataforma se debe ingresar como administrador.	

Tabla 14: Historia para aparatado para los profesores

Elaborado por: Investigador

Historia de Usuario	
Número: 005	Usuario: programador
Nombre Historia: Aparatado para los profesores	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Alta
Puntos Estimados: 5	Iteración Asignada: 4
Responsable: Franklin Guillén	
Descripción: Tiene que permitir modificar los cursos a los que tienen acceso los profesores.	
Observaciones: Para poder ingresar a esta parte de la plataforma se debe ingresar como profesor.	

Tabla 15: Historia para aparatado para los estudiantes

Elaborado por: Investigador

Historia de Usuario	
Número: 006	Usuario: programador
Nombre Historia: Aparatado para los estudiantes	
Prioridad en Negocio: Alta	Riesgo en desarrollo: Alta
Puntos Estimados: 5	Iteración Asignada: 5
Responsable: Franklin Guillén	
Descripción: Tiene que permitir acceder y visualizar las clases de los cursos en los que se encuentren matriculados.	
Observaciones: Para poder ingresar a esta parte de la plataforma se debe ingresar como estudiante.	

3.2.1.6 Estimación de historias de usuario

Después de crear las historias de usuario, el siguiente paso es para realizar una estimación de tiempo necesario para cumplir con todas las actividades establecidas. Por cada día se consideran 3 horas durante los días laborables de la semana para la elaboración de las historias de usuario.

Tabla 16: Estimación de historias de usuario

Elaborado por: Investigador

Número	Historia de usuario	Tiempo Estimado	
		Horas	Días
001	Definición de la estructura de la plataforma web.	9	3
002	Diseño de la base de datos.	15	5
003	Página informativa para la oferta de los cursos.	15	5
004	Aparatado para la administración de la plataforma	36	12
005	Aparatado para los profesores.	36	12
006	Aparatado para los estudiantes.	36	12
Total de tiempo estimado		147	49

3.2.1.7 Plan de iteraciones

Tabla 17: Plan de iteraciones

Elaborado por: Investigador

Iteración	Nº	Historia de usuario	Prioridad	Riesgo	Estado de desarrollo	Prueba
1	001	Definición de la estructura de la plataforma web.	Alta	Alta	Completo	Aceptada
	002	Diseño de la base de datos.	Alta	Alta	Completo	Aceptada
2	003	Página informativa para la oferta de los cursos.	Alta	Media	Completo	Aceptada
3	004	Aparatado para la administración de la plataforma.	Alta	Alta	Completo	Aceptada
4	005	Aparatado para los profesores.	Alta	Alta	Completo	Aceptada
5	006	Aparatado para los estudiantes.	Alta	Alta	Completo	Aceptada

3.2.2 Fase II: Diseño

3.2.2.1 Tarjetas CRC

Las tarjetas CRC (Clase, Responsabilidad y Colaboración) se utilizan para diseñar software y su interfaz. Para este proyecto de investigación se crea una tarjeta CRC para cada historia de usuario.

Tabla 18: Tarjeta CRC 1

Elaborado por: Investigador

Aparatado para la administración de la plataforma.	
Responsabilidades	Colaboradores
Tiene que permitir modificar ciertos aspectos de la plataforma como los cursos, profesores y estudiantes.	API ComponenteAdministrador ComponenteLogin
Observaciones: Ninguna.	

Tabla 19: Tarjeta CRC 2

Elaborado por: Investigador

Aparatado para los profesores.	
Responsabilidades	Colaboradores
Tiene que permitir modificar los cursos a los que tienen acceso los profesores.	API ComponenteProfesor ComponenteLogin
Observaciones: Ninguna.	

Tabla 20: Tarjeta CRC 3

Elaborado por: Investigador

Aparatado para los estudiantes.	
Responsabilidades	Colaboradores
Tiene que permitir acceder y visualizar las clases de los cursos en los que se encuentren matriculados.	API ComponenteEstudiante ComponenteLogin
Observaciones: Ninguna.	

3.2.2.2 Iteración 1

Según el plan de entrega establecido anteriormente, se ha establecido las siguientes historias de usuario.

Tabla 21: Iteración 1

Elaborado por: Investigador

Iteración	Nº	Historia de usuario	Prioridad	Riesgo	Estado de desarrollo	Prueba
1	001	Definición de la estructura de la plataforma web.	Alta	Alta	Completo	Aceptada
	002	Diseño de la base de datos.	Alta	Alta	Completo	Aceptada

Nº 001: Definición de la estructura de la plataforma web.

Para la correcta implementación de la plataforma de educación online se requiere del siguiente hardware y software:

Hardware

Un servidor contratado para el hosting y el dominio, el servidor cuenta con 6 GB de RAM, HTTP/2, certificado SSL, 10 GB de almacenamiento, cuentas FTP y acceso SSH.

Software

- MySQL
- ReactJS
- Nest.js
- Visual Studio Code

N° 002: Diseño de la base de datos.

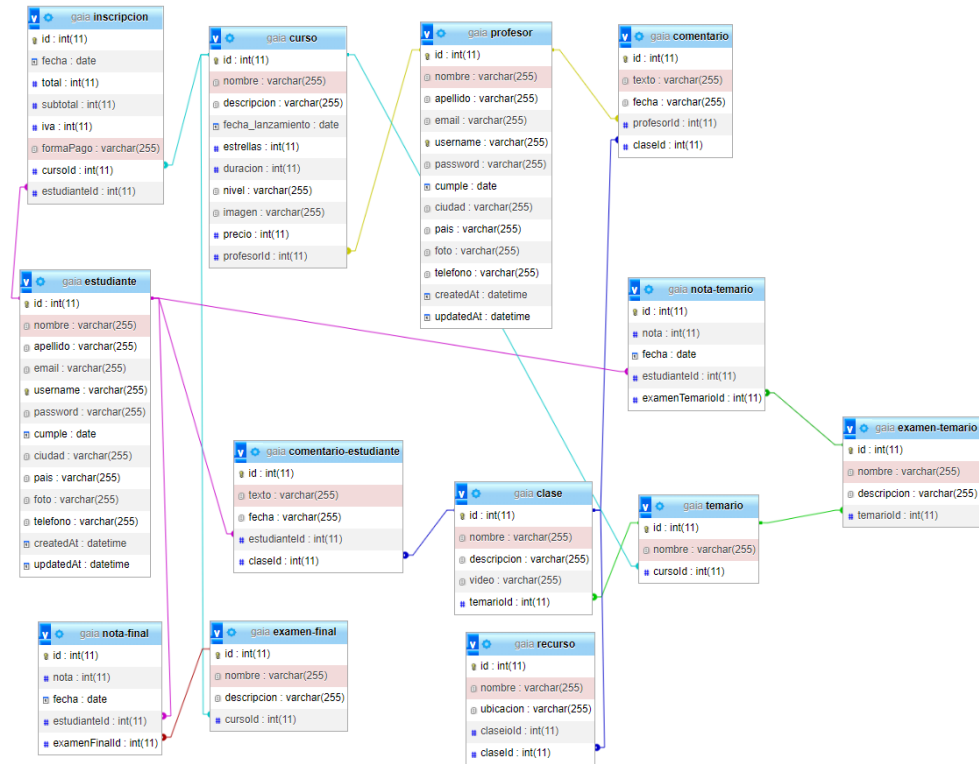


Figura 12: Diseño de base de datos

Elaborado por: Investigador

3.2.2.2 Iteración 2

Tabla 22: Iteración 2

Elaborado por: Investigador

Iteración	Nº	Historia de usuario	Prioridad	Riesgo	Estado de desarrollo	Prueba
2	003	Página informativa para la oferta de los cursos.	Alta	Media	Completo	Aceptada

Nº 003: Página informativa para la oferta de los cursos.



Figura 13: Página informativa

Elaborado por: Investigador

3.2.2.2 Iteración 3

Tabla 23: Iteración 3

Elaborado por: Investigador

Iteración	Nº	Historia de usuario	Prioridad	Riesgo	Estado de desarrollo	Prueba
3	004	Aparatado para la administración de la plataforma.	Alta	Alta	Completo	Aceptada

Nº 004: Aparatado para la administración de la plataforma.

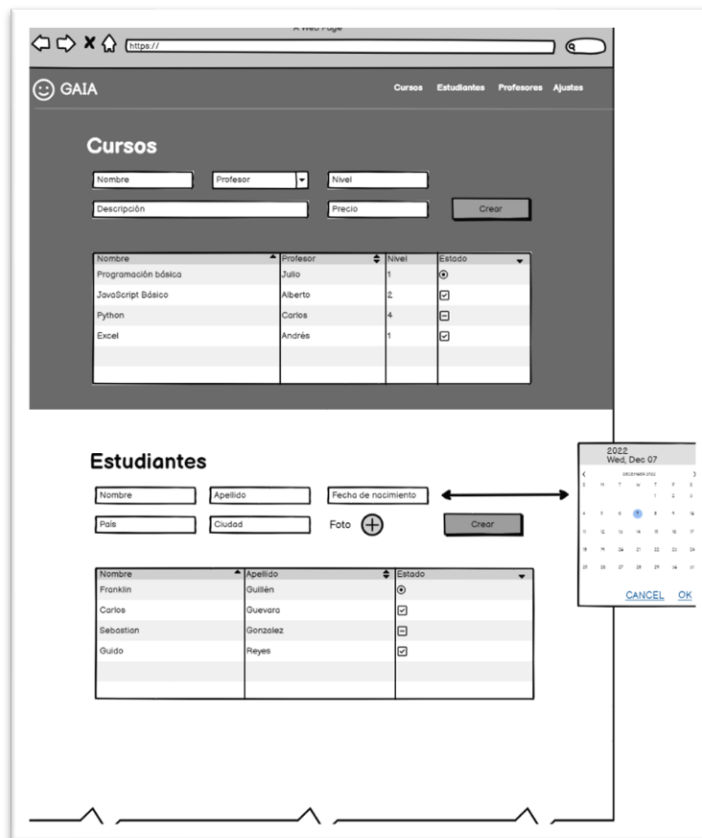


Figura 14: Página administrativa

Elaborado por: Investigador

3.2.2.2 Iteración 4

Tabla 24: Iteración 4

Elaborado por: Investigador

Iteración	Nº	Historia de usuario	Prioridad	Riesgo	Estado de desarrollo	Prueba
4	005	Aparatado para los profesores.	Alta	Alta	Completo	Aceptada

Nº 005: Aparatado para los profesores.

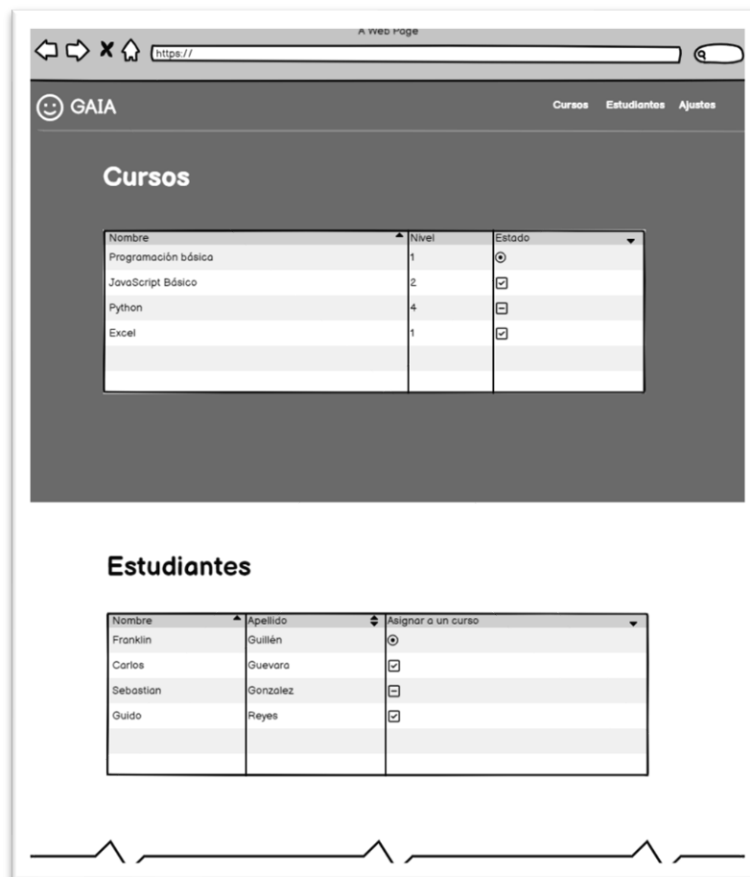


Figura 15: Página para profesores

Elaborado por: Investigador

3.2.2.2 Iteración 5

Tabla 25: Iteración 5

Elaborado por: Investigador

Iteración	Nº	Historia de usuario	Prioridad	Riesgo	Estado de desarrollo	Prueba
5	006	Aparatado para los estudiantes.	Alta	Alta	Completo	Aceptada

Nº 006: Aparatado para los estudiantes.

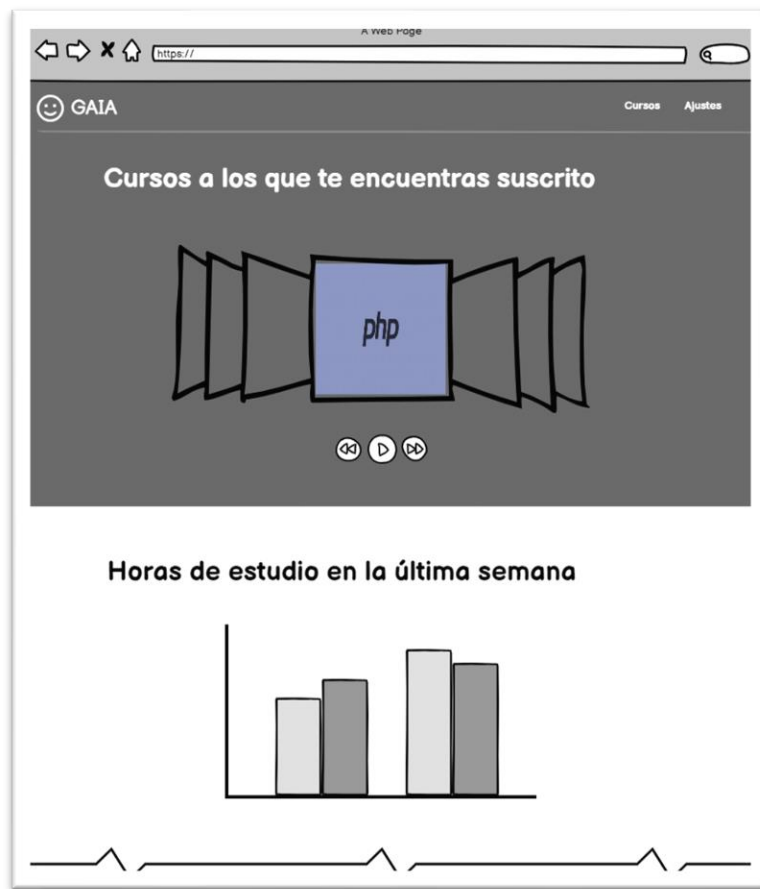


Figura 16: Página para estudiantes

Elaborado por: Investigador

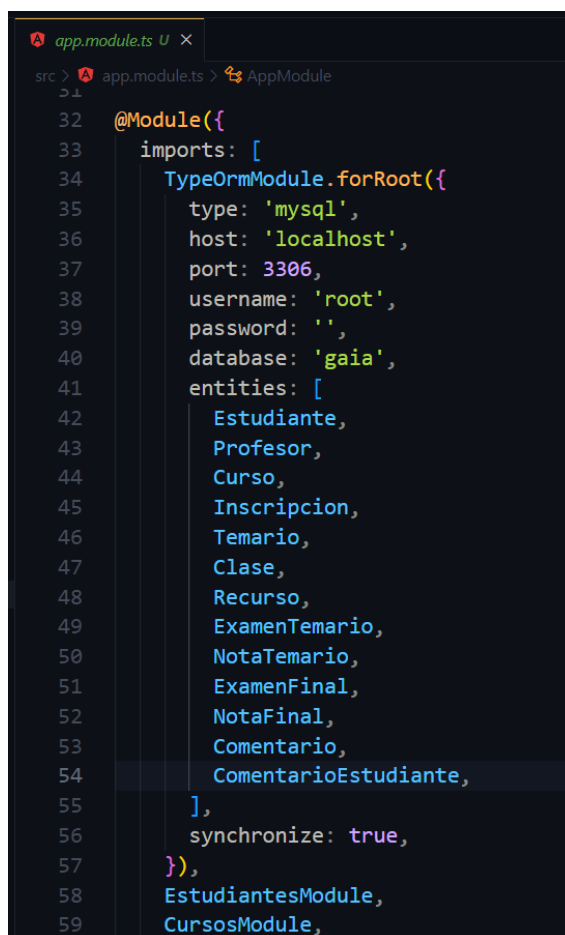
3.2.3 Fase II: Codificación

3.2.3.1 Creación de la API en Nest.js

Para que exista una conexión con la base de datos creada anteriormente se creó una API en Nest.js. Esta API además de estar conectada con la base de datos también contiene los métodos necesarios para acceder a la información y poder realizar modificaciones.

Conexión con la base de datos

En la figura 17 se puede observar la conexión con la base de datos MySQL, para lograr esta conexión se instaló los paquetes mysql2 y TypeORM con el comando npm. Además, se puede observar las entidades que obtiene de la base de datos y cada una de estas corresponde a una tabla.



```
app.module.ts U x
src > app.module.ts > AppModule
> 1
32 @Module({
33   imports: [
34     TypeOrmModule.forRoot({
35       type: 'mysql',
36       host: 'localhost',
37       port: 3306,
38       username: 'root',
39       password: '',
40       database: 'gaia',
41       entities: [
42         Estudiante,
43         Profesor,
44         Curso,
45         Inscripcion,
46         Temario,
47         Clase,
48         Recurso,
49         ExamenTemario,
50         NotaTemario,
51         ExamenFinal,
52         NotaFinal,
53         Comentario,
54         ComentarioEstudiante,
55       ],
56       synchronize: true,
57     }),
58     EstudiantesModule,
59     CursosModule,
```

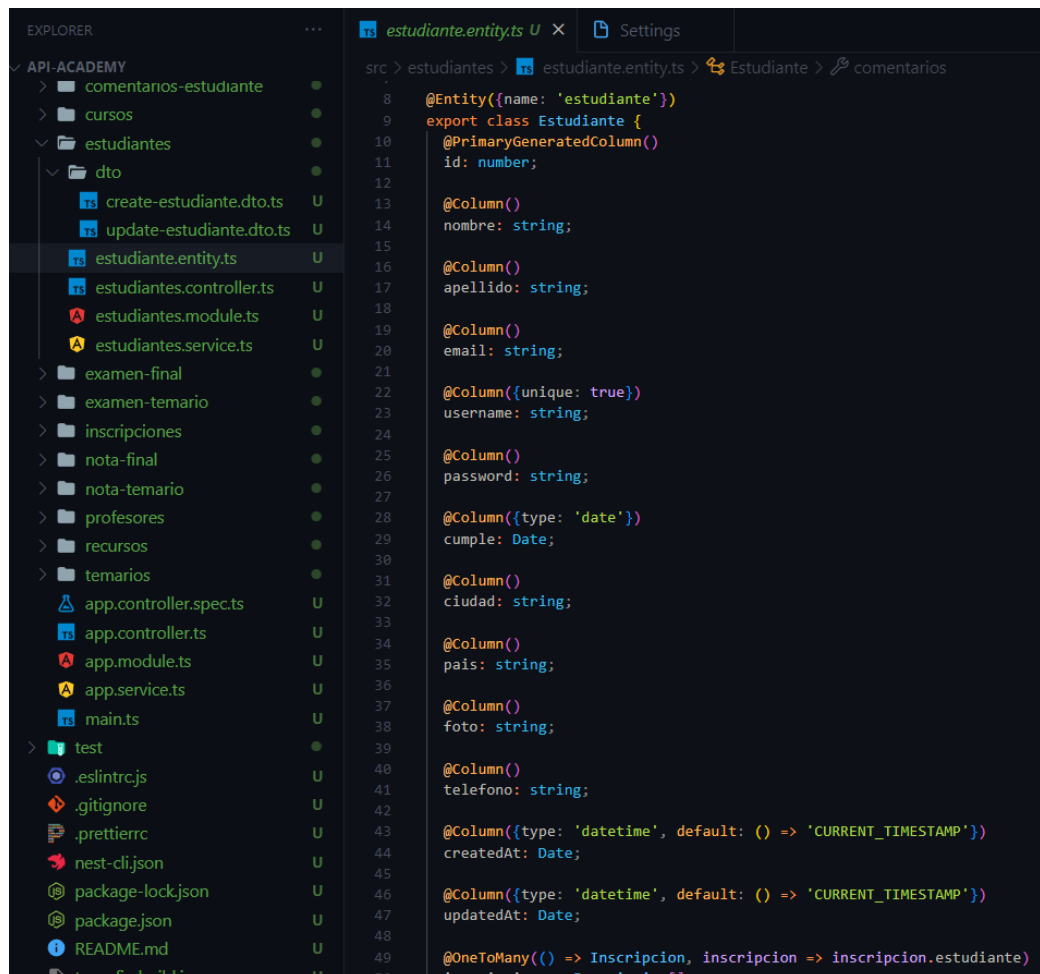
Figura 17: Conexión con la base de datos

Elaborado por: Investigador

A continuación, se procede a mostrar cada una de las entidades con sus módulos, controladores y servicios correspondientes.

Entidad estudiantes

Esta entidad hace referencia a la tabla estudiantes, en el archivo estudiante.entity.ts se detalla cada una de sus columnas con el tipo de valor y sus referencias.

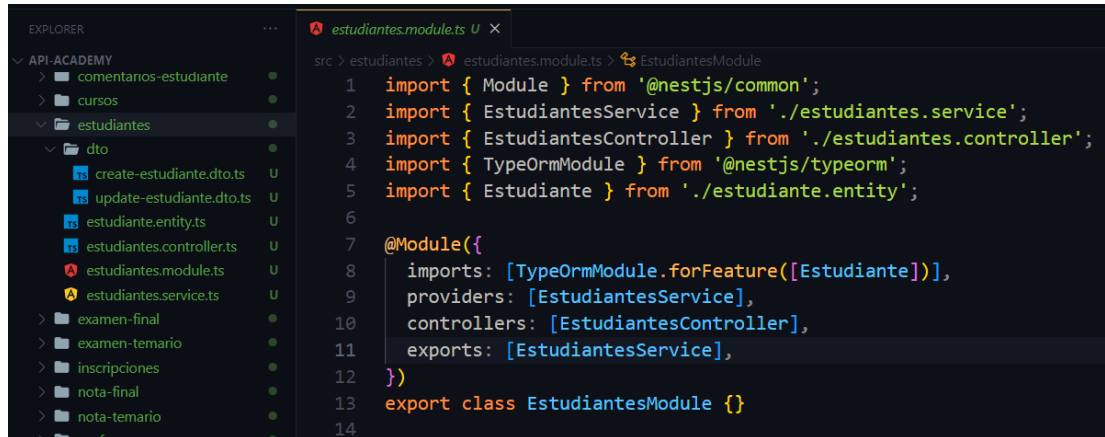


```
8 @Entity({name: 'estudiante'})
9 export class Estudiante {
10   @PrimaryGeneratedColumn()
11   id: number;
12
13   @Column()
14   nombre: string;
15
16   @Column()
17   apellido: string;
18
19   @Column()
20   email: string;
21
22   @Column({unique: true})
23   username: string;
24
25   @Column()
26   password: string;
27
28   @Column({type: 'date'})
29   cumple: Date;
30
31   @Column()
32   ciudad: string;
33
34   @Column()
35   pais: string;
36
37   @Column()
38   foto: string;
39
40   @Column()
41   telefono: string;
42
43   @Column({type: 'datetime', default: () => 'CURRENT_TIMESTAMP'})
44   createdAt: Date;
45
46   @Column({type: 'datetime', default: () => 'CURRENT_TIMESTAMP'})
47   updatedAt: Date;
48
49   @OneToMany(() => Inscpcion, inscripcion => inscripcion.estudiante)
```

Figura 18: Entidad estudiantes

Elaborado por: Investigador

En el módulo se encuentra la importación de la entidad mostrada anteriormente y se exporta el servicio de estudiantes para que pueda ser utilizado en otras entidades.

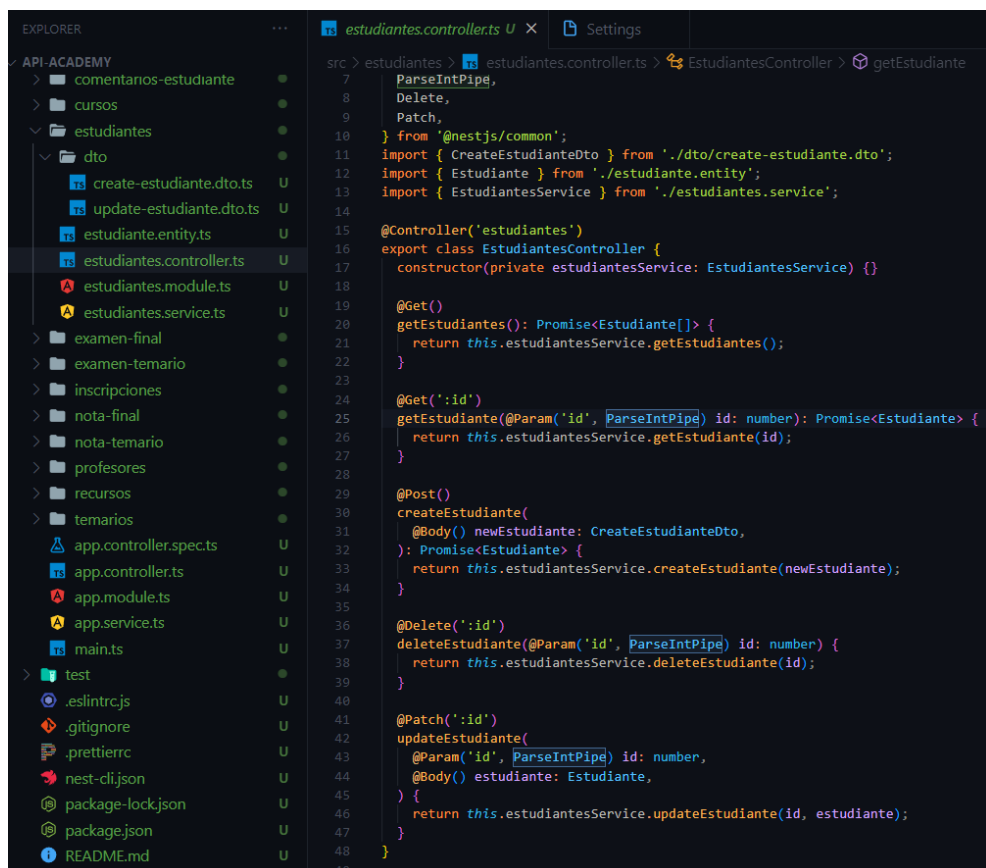


```
1 import { Module } from '@nestjs/common';
2 import { EstudiantesService } from './estudiantes.service';
3 import { EstudiantesController } from './estudiantes.controller';
4 import { TypeOrmModule } from '@nestjs/typeorm';
5 import { Estudiante } from './estudiante.entity';
6
7 @Module({
8   imports: [TypeOrmModule.forFeature([Estudiante])],
9   providers: [EstudiantesService],
10  controllers: [EstudiantesController],
11  exports: [EstudiantesService],
12 })
13 export class EstudiantesModule {}
14
```

Figura 19: Módulo estudiantes

Elaborado por: Investigador

En el controlador se detalla los métodos que se van a consumir desde el front-end, dependiendo de cada función se usará el método GET, POST, DELETE o PATCH.

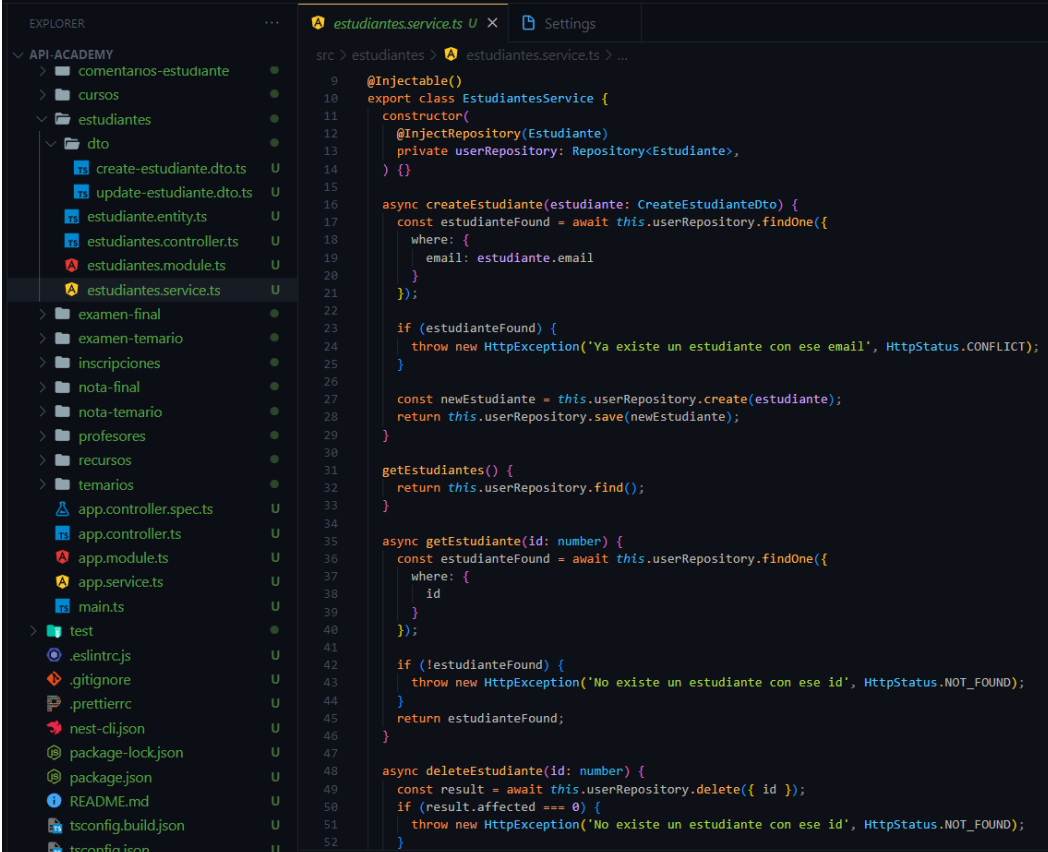


```
7   ParseIntPipe,
8   Delete,
9   Patch,
10  } from '@nestjs/common';
11 import { CreateEstudianteDto } from './dto/create-estudiante.dto';
12 import { Estudiante } from './estudiante.entity';
13 import { EstudiantesService } from './estudiantes.service';
14
15 @Controller('estudiantes')
16 export class EstudiantesController {
17   constructor(private estudiantesService: EstudiantesService) {}
18
19   @Get()
20   getEstudiantes(): Promise<Estudiante[]> {
21     return this.estudiantesService.getEstudiantes();
22   }
23
24   @Get('/:id')
25   getEstudiante(@Param('id', ParseIntPipe) id: number): Promise<Estudiante> {
26     return this.estudiantesService.getEstudiante(id);
27   }
28
29   @Post()
30   createEstudiante(
31     @Body() newEstudiante: CreateEstudianteDto,
32   ): Promise<Estudiante> {
33     return this.estudiantesService.createEstudiante(newEstudiante);
34   }
35
36   @Delete('/:id')
37   deleteEstudiante(@Param('id', ParseIntPipe) id: number) {
38     return this.estudiantesService.deleteEstudiante(id);
39   }
40
41   @Patch('/:id')
42   updateEstudiante(
43     @Param('id', ParseIntPipe) id: number,
44     @Body() estudiante: Estudiante,
45   ) {
46     return this.estudiantesService.updateEstudiante(id, estudiante);
47   }
48 }
49
```


Figura 20: Controlador estudiantes

Elaborado por: Investigador

En el servicio de estudiantes se realizan controles para no crear estudiantes duplicados, además, se configuran mensajes de error para tener una mejor interacción con la API y de esta forma mejorar el desarrollo en el front-end.



```
9 @Injectable()
10 export class EstudiantesService {
11   constructor(
12     @InjectRepository(Estudiante)
13     private userRepository: Repository<Estudiante>,
14   ) {}
15
16   async createEstudiante(estudiante: CreateEstudianteDto) {
17     const estudianteFound = await this.userRepository.findOne({
18       where: {
19         email: estudiante.email
20       }
21     });
22
23     if (estudianteFound) {
24       throw new HttpException('Ya existe un estudiante con ese email', HttpStatus.CONFLICT);
25     }
26
27     const newEstudiante = this.userRepository.create(estudiante);
28     return this.userRepository.save(newEstudiante);
29   }
30
31   getEstudiantes() {
32     return this.userRepository.find();
33   }
34
35   async getEstudiante(id: number) {
36     const estudianteFound = await this.userRepository.findOne({
37       where: {
38         id
39       }
40     });
41
42     if (!estudianteFound) {
43       throw new HttpException('No existe un estudiante con ese id', HttpStatus.NOT_FOUND);
44     }
45     return estudianteFound;
46   }
47
48   async deleteEstudiante(id: number) {
49     const result = await this.userRepository.delete({ id });
50     if (result.affected === 0) {
51       throw new HttpException('No existe un estudiante con ese id', HttpStatus.NOT_FOUND);
52     }
53   }
54 }
```

Figura 21: Servicio estudiantes

Elaborado por: Investigador

Entidad profesores

Esta entidad hace referencia a la tabla profesores, en el archivo profesor.entity.ts se crearon las columnas de acuerdo al diseño de la base de datos mostrado anteriormente. Esta entidad contiene referencias de tipo “OneToMany” (Uno a muchos) y “ManyToOne” (Muchos a uno).

```
src > profesores > profesor.entity.ts > Profesor > cursos
4 import { Curso } from '../cursos/curso.entity'
5
6 @Entity({name: 'profesor'})
7 export class Profesor {
8   @PrimaryGeneratedColumn()
9   id: number;
10
11   @Column()
12   nombre: string;
13
14   @Column()
15   apellido: string;
16
17   @Column()
18   email: string;
19
20   @Column({unique: true})
21   username: string;
22
23   @Column()
24   password: string;
25
26   @Column({type: 'date'})
27   cumple: Date;
28
29   @Column()
30   ciudad: string;
31
32   @Column()
33   pais: string;
34
35   @Column()
36   foto: string;
37
38   @Column()
39   telefono: string;
40
41   @Column({type: 'datetime', default: () => 'CURRENT_TIMESTAMP'})
42   createdAt: Date;
43
44   @Column({type: 'datetime', default: () => 'CURRENT_TIMESTAMP'})
45   updatedAt: Date;
46
47   @OneToMany(() => Curso, curso => curso.profesor)
```

Figura 22: Entidad profesores

Elaborado por: Investigador

En el módulo se importó la entidad profesor para que se conecte con la base de datos mediante la clase “TypeOrmModule”, adicionalmente se exportó los servicios de esta entidad para que sean usados en otras entidades que la necesiten.

```
src > profesores > profesores.module.ts > ProfesoresModule
1  import { Module } from '@nestjs/common';
2  import { ProfesoresController } from './profesores.controller';
3  import { ProfesoresService } from './profesores.service';
4  import { TypeOrmModule } from '@nestjs/typeorm';
5  import { Profesor } from './profesor.entity';
6
7  @Module({
8    imports: [TypeOrmModule.forFeature([Profesor])],
9    providers: [ProfesoresService],
10   controllers: [ProfesoresController],
11   exports: [ProfesoresService],
12 })
13 export class ProfesoresModule {}
14
```

Figura 23: Módulo profesores

Elaborado por: Investigador

En el controlador de los profesores se crearon los métodos para obtener los datos de todos los profesores, para obtener el dato de un solo profesor, para eliminar un profesor y para actualizar un profesor. Estas funciones se crearon los con los métodos POST, GET, DELETE y PATCH.

```
src > profesores > profesores.controller.ts > ProfesoresController > getProfesores
4
5
6
7
8
9
10 } from '@nestjs/common';
11 import { CreateProfesorDto } from './dto/create-profesor.dto';
12 import { Profesor } from './profesor.entity';
13 import { ProfesoresService } from './profesores.service';
14
15 @Controller('profesores')
16 export class ProfesoresController {
17   constructor(private profesoresService: ProfesoresService) {}
18
19   @Get()
20   getProfesores(): Promise<Profesor[]> {
21     return this.profesoresService.getProfesores();
22   }
23
24   @Get('/:id')
25   getProfesor(@Param('id', ParseIntPipe) id: number): Promise<Profesor> {
26     return this.profesoresService.getProfesor(id);
27   }
28
29   @Post()
30   createProfesor(@Body() newProfesor: CreateProfesorDto): Promise<Profesor> {
31     return this.profesoresService.createProfesor(newProfesor);
32   }
33
34   @Delete('/:id')
35   deleteProfesor(@Param('id', ParseIntPipe) id: number) {
36     return this.profesoresService.deleteProfesor(id);
37   }
38
39   @Patch('/:id')
40   updateProfesor(
41     @Param('id', ParseIntPipe) id: number,
42     @Body() profesor: Profesor,
43   ) {
44     return this.profesoresService.updateProfesor(id, profesor);
45   }
46 }
47
```

Figura 24: Controlador profesores

Elaborado por: Investigador

En los servicios de los profesores se crearon controles para que no existan correos o nombres de usuario duplicados, además, se creó controles para que no se pueda eliminar o actualizar un profesor que no existe. Los mensajes de error se configuraron con “HttpException”.

```
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
@Injectable()
export class ProfesoresService {
  constructor(
    @InjectRepository(Profesor)
    private userRepository: Repository<Profesor>,
  ) {}

  async createProfesor(profesor: CreateProfesorDto) {
    const profesorFound = await this.userRepository.findOne({
      where: {
        email: profesor.email
      }
    });

    if (profesorFound) {
      throw new HttpException('Ya existe un profesor con ese email', HttpStatus.CONFLICT);
    }

    const newProfesor = this.userRepository.create(profesor);
    return this.userRepository.save(newProfesor);
  }

  getProfesores() {
    return this.userRepository.find({
      relations: ['cursos']
    });
  }

  async getProfesor(id: number) {
    const profesorFound = await this.userRepository.findOne({
      where: {
        id
      }
    });

    if (!profesorFound) {
      throw new HttpException('No existe un profesor con ese id', HttpStatus.NOT_FOUND);
    }

    return profesorFound;
  }

  async deleteProfesor(id: number) {
    const result = await this.userRepository.delete({ id });
    if (result.affected === 0) {
```

Figura 25: Servicio profesores

Elaborado por: Investigador

Entidad curso

En esta entidad se encuentran las columnas correspondientes a la tabla curso, se configuro el tipo de valor de cada una de las columnas y se agregaron las referencias que existen con las tablas “profesor”, “inscripción” y “temario”.

```
8 @Entity({name: 'curso'})
9 export class Curso {
10   @PrimaryGeneratedColumn()
11   id: number;
12
13   @Column()
14   nombre: string;
15
16   @Column()
17   descripcion: string;
18
19   @Column({type: 'date'})
20   fecha_lanzamiento: Date;
21
22   @Column()
23   estrellas: number;
24
25   @Column()
26   duracion: number;
27
28   @Column()
29   nivel: string;
30
31   @Column()
32   imagen: string;
33
34   @Column()
35   precio: number;
36
37   @Column()
38   profesorId: number;
39
40   @ManyToOne(() => Profesor, profesor => profesor.cursos)
41   profesor: Profesor;
42
43   @OneToMany(() => Inscripcion, inscripcion => inscripcion.curso)
44   inscripciones: Inscripcion[];
45
46   @OneToMany(() => Temario, temario => temario.curso)
47   temarios: Temario[];
48
49   @OneToMany(() => ExamenFinal, examenFinal => examenFinal.curso)
50   examenesFinal: ExamenFinal[];
```

Figura 26: Entidad curso

Elaborado por: Investigador

En el módulo de la entidad cursos se importó el archivo curso.entity.ts para la conexión con la base de datos y adicionalmente se importó el módulo de la entidad “profesores” para hacer uso de sus servicios.

The screenshot shows the Visual Studio Code editor with the Explorer view on the left and the Editor view on the right. The Explorer view shows a project structure for 'API-ACADEMY' with a 'src' directory containing 'dto' and 'cursos' subdirectories. The 'cursos' directory contains 'create-curso.dto.ts', 'curso.entity.ts', 'cursos.controller.ts', 'cursos.module.ts', and 'cursos.service.ts'. The Editor view shows the code for 'cursos.module.ts' with the following content:

```
1 import { Module } from '@nestjs/common';
2 import { TypeOrmModule } from '@nestjs/typeorm';
3 import { CursosController } from './cursos.controller';
4 import { CursosService } from './cursos.service';
5 import { Curso } from './curso.entity';
6 import { ProfesoresModule } from 'src/profesores/profesores.module';
7
8 @Module({
9   imports: [TypeOrmModule.forFeature([Curso]), ProfesoresModule],
10  controllers: [CursosController],
11  providers: [CursosService],
12  exports: [CursosService],
13 })
14 export class CursosModule {}
15
```

Figura 27: Módulo curso

Elaborado por: Investigador

En el controlador de la entidad cursos se crearon los métodos POST y GET para crear nuevos cursos y para obtener todos los datos de los cursos o solamente uno de ellos.

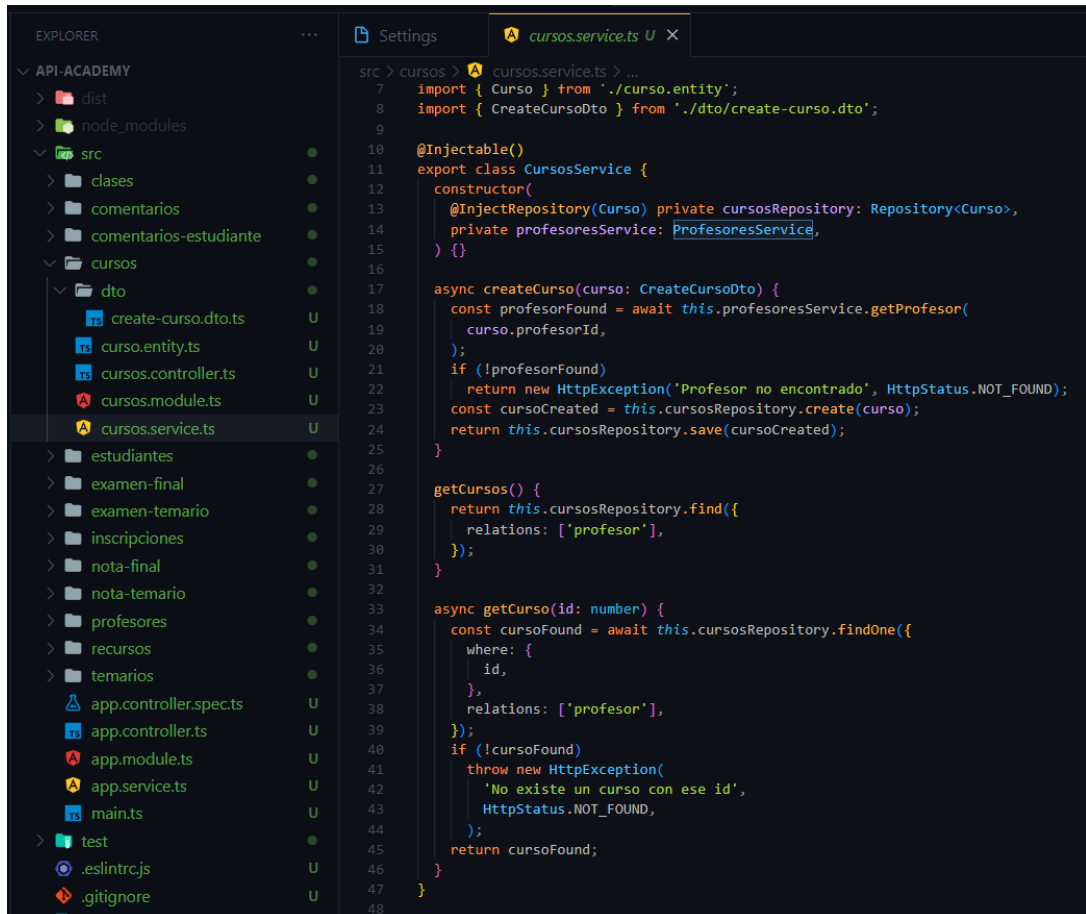
The screenshot shows the Visual Studio Code editor with the Explorer view on the left and the Editor view on the right. The Explorer view shows the same project structure as Figure 27, but with 'cursos.controller.ts' selected. The Editor view shows the code for 'cursos.controller.ts' with the following content:

```
1 /* eslint-disable @typescript-eslint/no-unused-vars */
2 import {
3   Body,
4   Controller,
5   Get,
6   Param,
7   ParseIntPipe,
8   Post,
9 } from '@nestjs/common';
10 import { CursosService } from './cursos.service';
11 import { CreateCursoDto } from './dto/create-curso.dto';
12
13 @Controller('cursos')
14 export class CursosController {
15   constructor(private cursosService: CursosService) {}
16
17   @Post()
18   createCurso(@Body() curso: CreateCursoDto) {
19     return this.cursosService.createCurso(curso);
20   }
21
22   @Get()
23   getCursos() {
24     return this.cursosService.getCursos();
25   }
26
27   @Get(':id')
28   getCurso(@Param('id', ParseIntPipe) id: number) {
29     return this.cursosService.getCurso(id);
30   }
31 }
32
```

Figura 28: Controlador curso

Elaborado por: Investigador

En los servicios de los cursos se detallaron los controles con el uso de los servicios de la entidad “profesores”. De esta forma se realiza un control para saber si existe el profesor al que se le va a asignar un nuevo curso.



```
7 import { Curso } from './curso.entity';
8 import { CreateCursoDto } from './dto/create-curso.dto';
9
10 @Injectable()
11 export class CursosService {
12   constructor(
13     @InjectRepository(Curso) private cursosRepository: Repository<Curso>,
14     private profesoresService: ProfesoresService,
15   ) {}
16
17   async createCurso(curso: CreateCursoDto) {
18     const profesorFound = await this.profesoresService.getProfesor(
19       curso.profesorId,
20     );
21     if (!profesorFound)
22       return new HttpException('Profesor no encontrado', HttpStatus.NOT_FOUND);
23     const cursoCreated = this.cursosRepository.create(curso);
24     return this.cursosRepository.save(cursoCreated);
25   }
26
27   getCursos() {
28     return this.cursosRepository.find({
29       relations: ['profesor'],
30     });
31   }
32
33   async getCurso(id: number) {
34     const cursoFound = await this.cursosRepository.findOne({
35       where: {
36         id,
37       },
38       relations: ['profesor'],
39     });
40     if (!cursoFound)
41       throw new HttpException(
42         'No existe un curso con ese id',
43         HttpStatus.NOT_FOUND,
44       );
45     return cursoFound;
46   }
47 }
48
```

Figura 29: Servicio curso

Elaborado por: Investigador

Entidad inscripciones

En esta entidad se detalla el tipo de columnas que tendrá cada inscripción para controlar el acceso de los estudiantes a cada curso. El valor pagado por el curso también se encuentra en esta entidad con el subtotal y el iva.


```

1  /* eslint-disable prettier/prettier */
2  import { Entity, Column, PrimaryGeneratedColumn, ManyToOne } from 'typeorm'
3  import { Estudiante } from '../estudiantes/estudiante.entity'
4  import { Curso } from '../cursos/curso.entity'
5
6  @Entity({name: 'inscripcion'})
7  export class Inscripcion {
8      @PrimaryGeneratedColumn()
9      id: number;
10
11     @Column({type: 'date'})
12     fecha: Date;
13
14     @Column()
15     total: number;
16
17     @Column()
18     subtotal: number;
19
20     @Column()
21     iva: number;
22
23     @Column()
24     formaPago: string;
25
26     @Column()
27     cursoId: number;
28
29     @Column()
30     estudianteId: number;
31
32     @ManyToOne(() => Estudiante, estudiante => estudiante.inscripciones)
33     estudiante: Estudiante;
34
35     @ManyToOne(() => Curso, curso => curso.inscripciones)
36     curso: Curso;
37
38 }

```

Figura 30: Entidad inscripción

Elaborado por: Investigador

En el módulo se importó la entidad creada anteriormente y en este caso no es necesario exportar los servicios de las inscripciones.

```

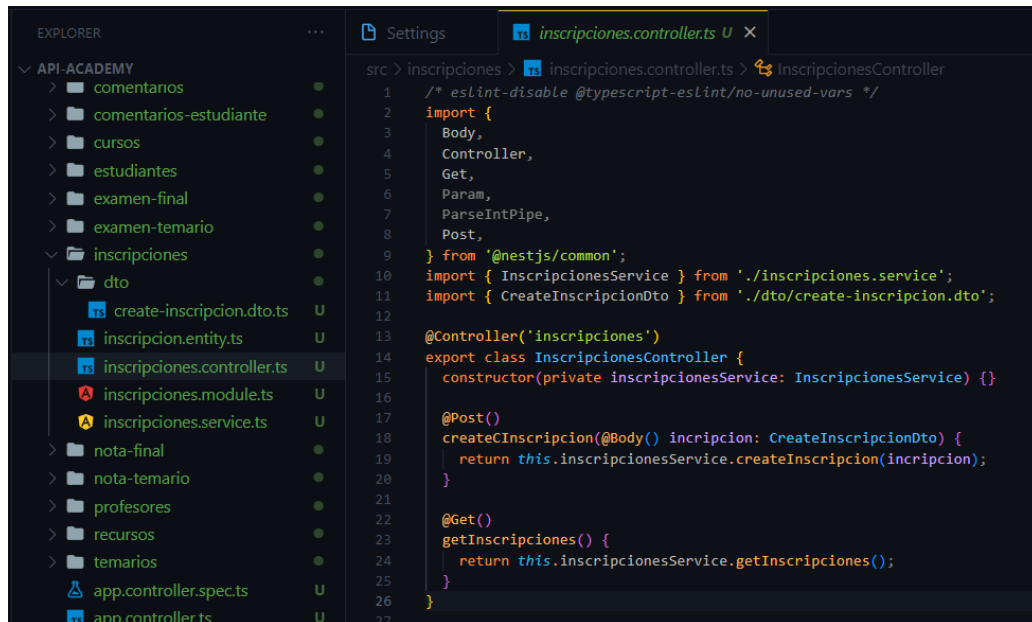
1  import { Module } from '@nestjs/common';
2  import { TypeOrmModule } from '@nestjs/typeorm';
3  import { IncripcionesController } from './inscripciones.controller';
4  import { IncripcionesService } from './inscripciones.service';
5  import { Inscripcion } from './inscripcion.entity';
6  import { EstudiantesModule } from 'src/estudiantes/estudiantes.module';
7  import { CursosModule } from 'src/cursos/cursos.module';
8
9
10 @Module({
11   imports: [
12     TypeOrmModule.forFeature([Inscripcion]),
13     EstudiantesModule,
14     CursosModule,
15   ],
16   controllers: [IncripcionesController],
17   providers: [IncripcionesService],
18 })
19 export class IncripcionesModule {}

```

Figura 31: Módulo inscripción

Elaborado por: Investigador

Este controlador está diseñado para crear y obtener nuevas inscripciones de los estudiantes en cada curso.



```
1  /* eslint-disable @typescript-eslint/no-unused-vars */
2  import {
3    Body,
4    Controller,
5    Get,
6    Param,
7    ParseIntPipe,
8    Post,
9  } from '@nestjs/common';
10 import { IncripcionesService } from './inscripciones.service';
11 import { CreateInscripcionDto } from './dto/create-inscripcion.dto';
12
13 @Controller('inscripciones')
14 export class IncripcionesController {
15   constructor(private inscripcionesService: IncripcionesService) {}
16
17   @Post()
18   createCInscripcion(@Body() inscripcion: CreateInscripcionDto) {
19     return this.inscripcionesService.createInscripcion(inscripcion);
20   }
21
22   @Get()
23   getInscripciones() {
24     return this.inscripcionesService.getInscripciones();
25   }
26 }
27
```

Figura 32: Controlador inscripción

Elaborado por: Investigador

En los servicios de las inscripciones se usó los servicios de los curso y de los estudiantes, estos últimos se utilizaron para saber si existen el curso y el estudiante que se asociará a la nueva inscripción. Además, se crearon códigos de error con la clase “HttpStatus”.

```

1  import { HttpException, HttpStatus, Injectable } from '@nestjs/common';
2  import { Inscricion } from './inscripcion.entity';
3  import { InjectRepository } from '@nestjs/typeorm';
4  import { Repository } from 'typeorm';
5  import { CreateInscricionDto } from './dto/create-inscripcion.dto';
6  import { EstudiantesService } from '../estudiantes/estudiantes.service';
7  import { CursosService } from '../cursos/cursos.service';
8
9  @Injectable()
10 export class InscricionesService {
11     constructor(
12         @InjectRepository(Inscricion)
13         private inscripcionRepository: Repository<Inscricion>,
14         private estudiantesService: EstudiantesService,
15         private cursosService: CursosService,
16     ) {}
17
18     async createInscricion(inscripcion: CreateInscricionDto) {
19         const estudianteFound = await this.estudiantesService.getEstudiante(
20             inscripcion.estudianteId,
21         );
22         const cursoFound = await this.cursosService.getCurso(inscripcion.cursoId);
23         if (!estudianteFound)
24             return new HttpException(
25                 'Estudiante no encontrado',
26                 HttpStatus.NOT_FOUND,
27             );
28         if (!cursoFound)
29             return new HttpException('Curso no encontrado', HttpStatus.NOT_FOUND);
30
31         const inscripcionCreated = this.inscripcionRepository.create(inscripcion);
32         return this.inscripcionRepository.save(inscripcionCreated);
33     }
34
35     getInscriciones() {
36         return this.inscripcionRepository.find({
37             relations: ['estudiante', 'curso'],
38         });
39     }
40 }
41

```

Figura 33: Servicio inscripción

Elaborado por: Investigador

Entidad temarios

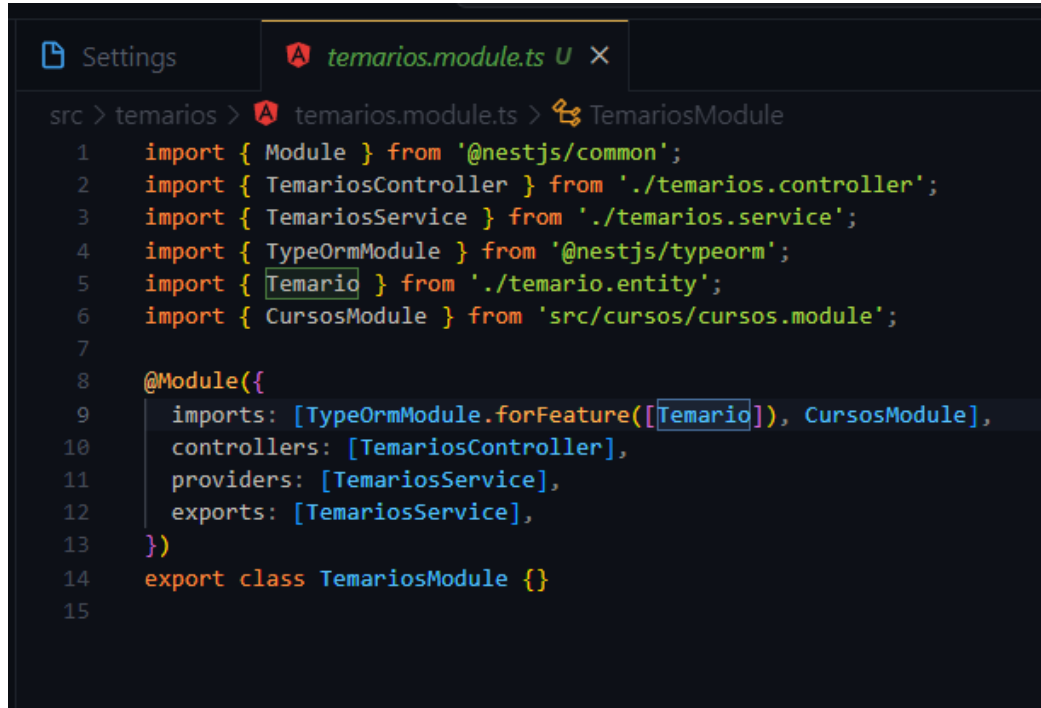
```

1  /* eslint-disable prettier/prettier */
2  import { Entity, Column, PrimaryGeneratedColumn, ManyToOne, OneToMany } from 'typeorm'
3  import { Curso } from '../cursos/curso.entity'
4  import { Clase } from '../clases/clase.entity'
5  import { ExamenTemario } from 'src/examen-temario/examen-temario.entity';
6
7  @Entity({name: 'temario'})
8  export class Temario {
9      @PrimaryGeneratedColumn()
10     id: number;
11
12     @Column()
13     nombre: string;
14
15     @Column()
16     cursoId: number;
17
18     @ManyToOne(() => Curso, curso => curso.temarios)
19     curso: Curso;
20
21     @OneToMany(() => Clase, clase => clase.temario)
22     clases: Clase[];
23
24     @OneToMany(() => ExamenTemario, examenTemario => examenTemario.temario)
25     examenesTemario: ExamenTemario[];
26 }

```

Figura 34: Entidad temario

Elaborado por: Investigador



```
src > temarios > temarios.module.ts > TemariosModule
1  import { Module } from '@nestjs/common';
2  import { TemariosController } from './temarios.controller';
3  import { TemariosService } from './temarios.service';
4  import { TypeOrmModule } from '@nestjs/typeorm';
5  import { Temario } from './temario.entity';
6  import { CursosModule } from 'src/cursos/cursos.module';
7
8  @Module({
9    imports: [TypeOrmModule.forFeature([Temario]), CursosModule],
10   controllers: [TemariosController],
11   providers: [TemariosService],
12   exports: [TemariosService],
13 })
14 export class TemariosModule {}
15
```

Figura 35: Módulo temario

Elaborado por: Investigador

```
1  /* eslint-disable @typescript-eslint/no-unused-vars */
2  import {
3    Body,
4    Controller,
5    Get,
6    Param,
7    ParseIntPipe,
8    Post,
9  } from '@nestjs/common';
10 import { TemariosService } from '../temarios.service';
11 import { CreateTemarioDto } from '../dto/create-temario.dto';
12
13 @Controller('temarios')
14 export class TemariosController {
15   constructor(private temariosService: TemariosService) {}
16
17   @Post()
18   createTemario(@Body() temario: CreateTemarioDto) {
19     return this.temariosService.createTemario(temario);
20   }
21
22   @Get()
23   getTemarios() {
24     return this.temariosService.getTemarios();
25   }
26
27   @Get('/:id')
28   getTemario(@Param('id', ParseIntPipe) id: number) {
29     return this.temariosService.getTemario(id);
30   }
31 }
32
```

Figura 36: Controlador temario

Elaborado por: Investigador

```
1 import { HttpException, HttpStatus, Injectable } from '@nestjs/common';
2 import { InjectRepository } from '@nestjs/typeorm';
3 import { CursosService } from '../cursos/cursos.service';
4 import { Repository } from 'typeorm';
5 import { CreateTemarioDto } from '../dto/create-temario.dto';
6 import { Temario } from '../temario.entity';
7
8 @Injectable()
9 export class TemariosService {
10   constructor(
11     @InjectRepository(Temario) private temariosRepository: Repository<Temario>,
12     private cursosService: CursosService,
13   ) {}
14
15   async createTemario(temario: CreateTemarioDto) {
16     const cursoFound = await this.cursosService.getCurso(temario.cursoId);
17     if (!cursoFound)
18       return new HttpException('Curso no encontrado', HttpStatus.NOT_FOUND);
19     const temarioCreated = this.temariosRepository.create(temario);
20     return this.temariosRepository.save(temarioCreated);
21   }
22
23   async getTemarios() {
24     return this.temariosRepository.find({
25       relations: ['curso'],
26     });
27   }
28
29   async getTemario(id: number) {
30     const temarioFound = await this.temariosRepository.findOne({
31       where: {
32         id,
33       },
34       relations: ['curso'],
35     });
36     if (!temarioFound)
37       throw new HttpException(
38         'No existe un temario con ese id',
39         HttpStatus.NOT_FOUND,
40       );
41     return temarioFound;
42   }
43 }
```

Figura 37: Servicio temario

Elaborado por: Investigador

Entidad clase

Esta entidad fue creada para almacenar el nombre, la descripción y el vídeo de cada clase, también se configuraron las relaciones con las tablas recurso, comentario y temario. En el caso de los recursos y comentarios se crearon un array para cada uno de ellos, de esta forma, la API podrá mostrar a qué recursos y comentario hace referencia la clase.

The screenshot shows the Visual Studio Code editor with the Explorer view on the left and the Code editor on the right. The Explorer view shows a project structure for 'API-ACADEMY' with folders like 'src', 'dto', 'comentarios', and 'temarios'. The Code editor displays the file 'class.entity.ts' with the following code:

```
1  /* eslint-disable prettier/prettier */
2  import { Entity, Column, PrimaryGeneratedColumn, ManyToOne, OneToMany } from 'typeorm'
3  import { Temario } from '../temarios/temario.entity'
4  import { Recurso } from '../recursos/recurso.entity'
5  import { Comentario } from 'src/comentarios/comentario.entity';
6
7
8  @Entity({name: 'class'})
9  export class Clase {
10     @PrimaryGeneratedColumn()
11     id: number;
12
13     @Column()
14     nombre: string;
15
16     @Column()
17     descripcion: string;
18
19     @Column()
20     video: string;
21
22     @Column()
23     temarioId: number;
24
25     @ManyToOne(() => Temario, temario => temario.clases)
26     temario: Temario;
27
28     @OneToMany(() => Recurso, recurso => recurso.clase)
29     recursos: Recurso[];
30
31     @OneToMany(() => Comentario, comentario => comentario.clase)
32     comentarios: Comentario[];
33 }
```

Figura 38: Entidad clase

Elaborado por: Investigador

En este módulo se importó la entidad creada anteriormente y el módulo de los temarios, este último se ocupará para realizar controles en los servicios. Adicionalmente, se realizó una exportación de los servicios de la clase para que sean utilizados en otras entidades.

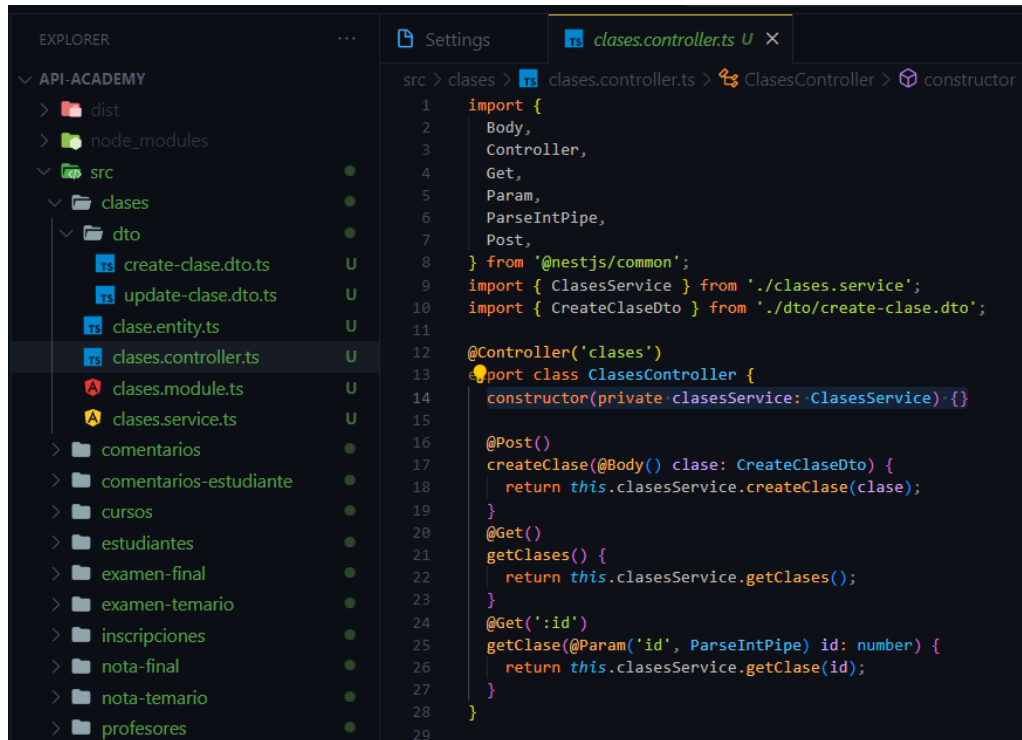
The screenshot shows the Visual Studio Code editor with the Explorer view on the left and the Code editor on the right. The Explorer view shows the project structure for 'API-ACADEMY' with folders like 'src', 'dto', 'comentarios', and 'temarios'. The Code editor displays the file 'classes.module.ts' with the following code:

```
1  import { Module } from '@nestjs/common';
2  import { ClasesController } from './clases.controller';
3  import { ClasesService } from './clases.service';
4  import { TypeOrmModule } from '@nestjs/typeorm';
5  import { Clase } from './clase.entity';
6  import { TemariosModule } from 'src/temarios/temarios.module';
7
8  @Module({
9     imports: [TypeOrmModule.forFeature([Clase]), TemariosModule],
10    controllers: [ClasesController],
11    providers: [ClasesService],
12    exports: [ClasesService],
13  })
14  export class ClasesModule {}
15
```

Figura 39: Módulo clase

Elaborado por: Investigador

El controlador se configuró para obtener los datos de todas las clases o de una en concreto, también cuenta con el método para crear una nueva clase con los datos configurados en la entidad.



```
1 import {
2   Body,
3   Controller,
4   Get,
5   Param,
6   ParseIntPipe,
7   Post,
8 } from '@nestjs/common';
9 import { ClasesService } from '../clases.service';
10 import { CreateClaseDto } from '../dto/create-clase.dto';
11
12 @Controller('clases')
13 export class ClasesController {
14   constructor(private clasesService: ClasesService) {}
15
16   @Post()
17   createClase(@Body() clase: CreateClaseDto) {
18     return this.clasesService.createClase(clase);
19   }
20
21   @Get()
22   getClases() {
23     return this.clasesService.getClases();
24   }
25
26   @Get('/:id')
27   getClase(@Param('id', ParseIntPipe) id: number) {
28     return this.clasesService.getClase(id);
29   }
30 }
```

Figura 40: Controlador clase

Elaborado por: Investigador

En los servicios de la clase se utilizaron los servicios de los temarios para controlar en la creación de una clase si previamente existe el temario al que se va a relacionar. Además, se configuraron los mensajes de error para cada uno de los controles.


```
7
8 @Injectable()
9 export class ClasesService {
10   constructor(
11     @InjectRepository(Clase) private claseRepository: Repository<Clase>,
12     private temariosService: TemariosService,
13   ): {}
14
15   async createClase(clase: CreateClaseDto) {
16     const temarioFound = await this.temariosService.getTemario(clase.temarioId);
17     if (!temarioFound)
18       return new HttpException('Tema no encontrado', HttpStatus.NOT_FOUND);
19     const claseCreated = this.claseRepository.create(clase);
20     return this.claseRepository.save(claseCreated);
21   }
22
23   async getClases() {
24     return this.claseRepository.find({
25       relations: ['temario'],
26     });
27   }
28
29   async getClase(id: number) {
30     const claseFound = await this.claseRepository.findOne({
31       where: {
32         id,
33       },
34       relations: ['temario'],
35     });
36     if (!claseFound)
37       throw new HttpException(
38         'No existe una clase con ese id',
39         HttpStatus.NOT_FOUND,
40       );
41     return claseFound;
42   }
43 }
44
```

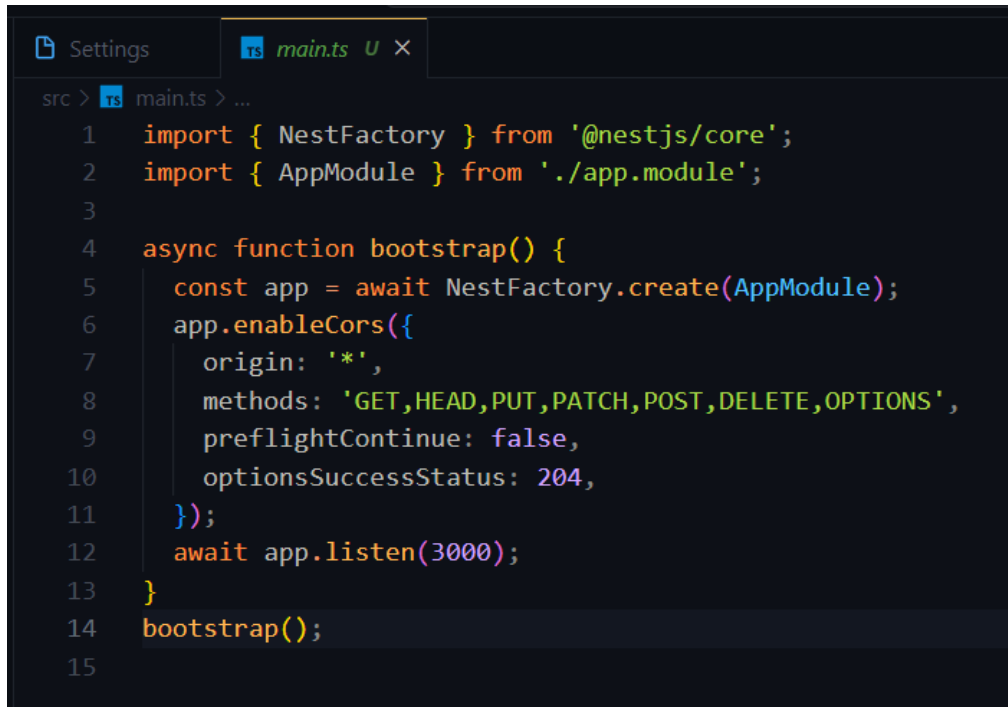
Figura 41: Servicio clase

Elaborado por: Investigador

Configuración del intercambio de recursos de origen cruzado

El intercambio de recursos de origen cruzado, también llamado CORS, es un mecanismo que permite una comunicación entre una API y varios sitios web. Los CORS deben estar habilitados para que la API pueda recibir y enviar datos mediante los métodos POST, GET, DELETE y PATCH, de lo contrario, este mecanismo no permitirá ningún tipo de interacción.

Por lo anterior mencionado, se realizó la configuración de los CORS en la API desarrollada en Nest.js. Mediante el módulo “AppModule” se configuró los orígenes y los métodos que estarán permitidos con el método “enableCors”.



```
src > main.ts > ...
1  import { NestFactory } from '@nestjs/core';
2  import { AppModule } from './app.module';
3
4  async function bootstrap() {
5    const app = await NestFactory.create(AppModule);
6    app.enableCors({
7      origin: '*',
8      methods: 'GET,HEAD,PUT,PATCH,POST,DELETE,OPTIONS',
9      preflightContinue: false,
10     optionsSuccessStatus: 204,
11   });
12   await app.listen(3000);
13 }
14 bootstrap();
15
```

Figura 42: Configuración de CORS

Elaborado por: Investigador

3.2.3.2 Configuración de los proyectos en React

Archivo package.json

En el archivo package.json se revisaron las dependencias que usará el proyecto con sus respectivas versiones, además, se configuró varios scripts que ayudan a ejecutar el proyecto como desarrollador o para crear una built con el proyecto comprimido.

```
package.json ×
package.json > ...
1  {
2    "name": "gaiacursos",
3    "private": true,
4    "version": "0.0.0",
5    "type": "module",
6    "proxy": "http://localhost:3000",
7    "scripts": {
8      "dev": "vite",
9      "build": "vite build",
10     "preview": "vite preview",
11     "re-dev": "rm -rf node_modules && yarn install && yarn dev"
12   },
13   "dependencies": {
14     "@ant-design/icons": "^5.0.1",
15     "antd": "^5.2.0",
16     "axios": "^1.3.3",
17     "cors": "^2.8.5",
18     "json-server": "^0.17.1",
19     "localforage": "^1.10.0",
20     "match-sorter": "^6.3.1",
21     "rc-footer": "^0.6.8",
22     "react": "^18.2.0",
23     "react-dom": "^18.2.0",
24     "react-query": "^3.39.3",
25     "react-router-dom": "^6.8.1",
26     "sort-by": "^1.2.0"
27   },
28   "devDependencies": {
29     "@types/react": "^18.0.27"
```

Figura 43: Archivo package.json

Elaborado por: Investigador

3.2.3.3 Página informativa

En este apartado se creó una página con componentes de React.js para ofrecer información sobre la empresa GAIA, además se agregó información de los cursos ofertados por la empresa y de los profesores que imparten estos cursos.



Figura 44: Interfaz de la página informativa

Elaborado por: Investigador

En la siguiente figura se puede observar parte del código referente al diseño que se utilizó para el desarrollo de la página informativa.

```
26 <nav class="navbar navbar-expand-lg navbar-dark ftco_navbar bg-dark ftco-navbar-light" id="ftco-navbar">
27 <div class="container">
28 
29 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#ftco-nav" aria-controls="ftco-na
30 <span class="oi oi-menu"></span> Menu
31 </button>
32
33 <div class="collapse navbar-collapse" id="ftco-nav">
34 <ul class="navbar-nav ml-auto">
35 <li class="nav-item active"><a href="index.html" class="nav-link" style="color: #005cb9;">Home</a></li>
36 <li class="nav-item"><a href="#nosotros" class="nav-link">Nosotros</a></li>
37 <li class="nav-item"><a href="#cursos" class="nav-link">Cursos</a></li>
38 <li class="nav-item"><a href="" class="nav-link">Profesores</a></li>
39 <li class="nav-item"><a href="#contacto" class="nav-link">Contacto</a></li>
40 </ul>
41 </div>
42 </div>
43 </nav>
44 <!-- END nav -->
```

Figura 45: Código de la página informativa

Elaborado por: Investigador

3.2.3.4 Login para administradores y profesores

Este componente se creó con la finalidad de ofrecer acceso a la página administrativa de GAIA, las variables utilizadas son emailRef y passwordRef. El diseño del componente se desarrolló en JSX con dos inputs, los cuales son el email y la contraseña.

```
src > components > LoginPage > LoginPage
1 import { useRef } from 'react';
2 import { LockClosedIcon } from '@heroicons/react/solid';
3
4 export default function LoginPage() {
5   const emailRef = useRef(null);
6   const passwordRef = useRef(null);
7
8   const submitHandler = (event) => {
9     event.preventDefault();
10    const email = emailRef.current.value;
11    const password = passwordRef.current.value;
12
13    console.log(email, password);
14  };
15
16  return (
17    <>
18      <div className="min-h-full flex items-center justify-center py-12 px-4 sm:px-6 lg:px-8">
19        <div className="max-w-md w-full space-y-8">
20          <div>
21            
22            <h2 className="mt-6 text-center text-3xl font-extrabold text-gray-900">Sign in to your account</h2>
23          </div>
24          <form className="mt-8 space-y-6" onSubmit={submitHandler}>
25            <input type="hidden" name="remember" defaultValue="true" />
26            <div className="rounded-md shadow-sm space-y-px">
27              <div>
28                <label htmlFor="email-address" className="sr-only">
29                  Email address
30                </label>
31                <input
32                  id="email-address"
```

Figura 46: Componente para el login

Elaborado por: Investigador

Para que se pueda realizar la autenticación satisfactoriamente se utilizó un “custom Hook” llamado useAuth. Esta función ayuda a controlar la sesión en todo momento y a mantener la seguridad de los datos dentro de la aplicación web.

3.2.3.5 Página administrativa

Para la elaboración de este apartado administrativo se creó un proyecto nuevo de React, con el objetivo de crear un acceso en un subdominio diferente al dominio utilizado para la página informativa.

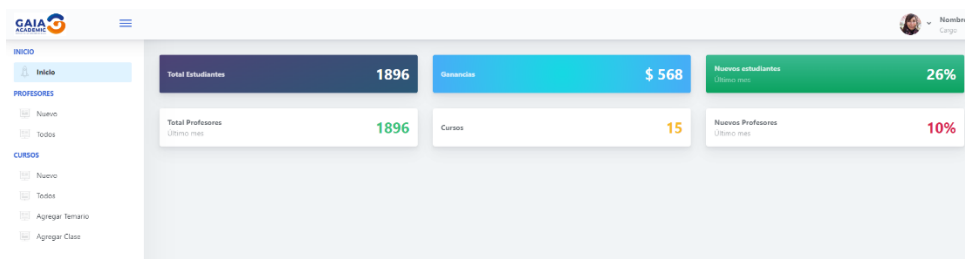


Figura 47: Interfaz de la página administrativa

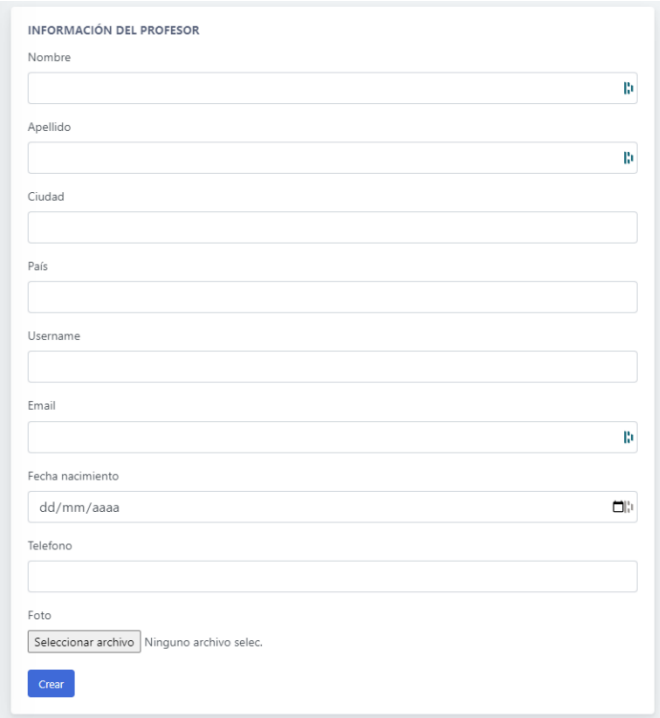
Elaborado por: Investigador

En este apartado de la plataforma de educación online se desarrolló diferentes componentes en React.js, los cuales son:

- Componente para profesores: la función de este componente es crear, actualizar, eliminar y leer toda la información de los profesores.
- Componente para cursos: la función de este componente es crear, actualizar, eliminar y leer toda la información de los cursos.

Gestión de profesores

Para crear un nuevo profesor se desarrolló el siguiente formulario:



INFORMACIÓN DEL PROFESOR

Nombre

Apellido

Ciudad

País

Username

Email

Fecha nacimiento

Telefono

Foto Ninguno archivo selec.

Figura 48: Gestión de profesores

Elaborado por: Investigador

Para observar, editar y eliminar los profesores que existen en la plataforma se creó la siguiente interfaz:

TODOS LOS PROFESORES					
#	Nombre	Apellido	Username	Editar	Eliminar
1	Mark	Otto	@mdo	Editar	Eliminar
2	Jacob	Thornton	@fat	Editar	Eliminar
3	Larry	the Bird	@twitter	Editar	Eliminar

Figura 49: Listado de profesores

Elaborado por: Investigador

Gestión de cursos

Para la creación de nuevos cursos se desarrolló el siguiente componente:

INFORMACIÓN DEL CURSO

Nombre

Precio

Profesor

Estrellas

Nivel

Descripción

Imagen
 Ninguno archivo selec.

Figura 50: Formulario del curso

Elaborado por: Investigador

El listado de todos los cursos que existen en la plataforma se puede observar en la Figura 51, este módulo se configuró para poder acceder al listado de estudiantes inscriptos en cada curso y sus notas, también permite actualizar o eliminar la información.

TODOS LOS CURSOS							
#	Nombre	Estrellas	Precio	Profesor	Listado de estudiantes	Editar	Eliminar
1	Excel	4	50	Pedro	Ver	Editar	Eliminar
2	Excel	4	50	Pedro	Ver	Editar	Eliminar
3	Excel	4	50	Pedro	Ver	Editar	Eliminar

Figura 51: Listado de los cursos

Elaborado por: Investigador

LISTADO DE ESTUDIANTES				
#	Nombre	Apellido	Username	Ver notas
1	Mark	Otto	@mdo	Ver
2	Jacob	Thornton	@fat	Ver
3	Larry	the Bird	@twitter	Ver

Figura 52: Listado de estudiantes

Elaborado por: Investigador

ESTUDIANTE JUAN		
#	Tema	Nota
1	Creación de componentes	8
2	Introduccion a React	7
3	Conexión con una API	9

Figura 53: Notas de un estudiante

Elaborado por: Investigador

En cada curso existen varios temas y para la creación de estos se desarrolló un componente que además permite crear exámenes para cada tema creado. En este módulo se contempla el curso al que va a estar asociado el tema.

CREAR TEMARIO

Nombre
Ejemplo: ReactJS

Curso
React

CREAR EXAMEN

Pregunta

Distractores

Ingresar pregunta

Crear Tema

Figura 54: Creación de un tema

Elaborado por: Investigador

Para la creación de cada clase se diseñó un componente que permite relacionar con un tema de un curso, además, permite subir los recursos y el vídeo correspondiente.

CREAR CLASE

Nombre
Ejemplo: ReactJS

Descripcion
Ejemplo: ReactJS

Curso
React

Tema
Introduccion

Video
Seleccionar archivo Ninguno archivo selec.

RECURSOS

Recurso
Seleccionar archivo Ninguno archivo selec.

Crear

Figura 55: Creación de una clase

Elaborado por: Investigador

3.2.3.5 Página para los estudiantes

Para los estudiantes se creó un nuevo proyecto en React con las mismas dependencias del proyecto de la parte administrativa. En este proyecto se crearon los componentes necesarios para que los estudiantes puedan acceder a la plataforma, ingresar en sus cursos, modificar datos de sus cuentas y visualizar los cursos online en un reproductor web.

Login

Este apartado permite dar acceso a los estudiantes a la plataforma, para su desarrollo se creó el siguiente componente:

```
LoginPage.jsx
src > screens > LoginPage.jsx > LoginPage
1 import React from 'react'
2 import { LockOutlined, UserOutlined } from '@ant-design/icons'
3 import { Button, Checkbox, Form, Image, Input } from 'antd'
4 import { useNavigate } from 'react-router-dom'
5 import logo from '../assets/logo.png'
6 // props react router dom redirect to Home
7 const LoginPage = () => {
8   const navigate = useNavigate()
9   const onFinish = values => {
10     navigate('/home')
11   }
12
13   return (
14     <Form
15       style={{
16         display: 'flex',
17         flexDirection: 'column',
18         justifyContent: 'center',
19         alignItems: 'center',
20         height: '100vh',
21
22         // background: 'rgb(12,22,51)',
23         // background:
24         // 'linear-gradient(180deg, rgba(12,22,51,1) 1%, rgba(0,212,255,1) 25%, rgba(0,212,255,1) 75%, rgba(12,22,51,1) 100%)',
25       }}
26     name='normal_login'
27     className='login-form'
28     initialValues={{
29       remember: true,
30     }}
31   )
32 }
```

Figura 56: Componente para el login

Elaborado por: Investigador



Figura 57: Interfaz del login de los estudiantes

Elaborado por: Investigador

Home

En esta página el estudiante puede observar los cursos que contiene la plataforma, el componente que se creó en React para este apartado se llama HomePage.

```
HomePage.jsx
src > screens > HomePage.jsx
245     price: '12000',
246     discount: '16',
247     learnMoreLink: '/course',
248   },
249 ]
250 return (
251   <>
252     <Navbar />
253     <div className='online-courses container'>
254       <Image
255         src={logo}
256         width={300}
257         style={{ marginLeft: '150%', marginTop: '20%', marginBottom: '20%' }}
258         preview={false}
259       />
260     <div className='banner-cards'></div>
261     <CourseContainer gradeRanges={['', '5-8', '9-12']} coursesData={COURSE_DATA} />
262   </div>
263   <FooterComponent />
264 </>
265 )
266 )
267 }
268
269 export default HomePage
270
```

Figura 58: Componente para la página de inicio

Elaborado por: Investigador

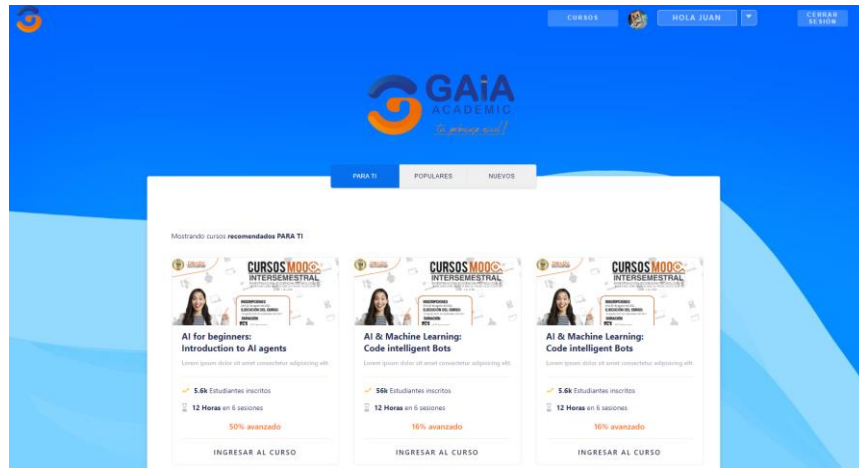


Figura 59: Diseño de estudiantes

Elaborado por: Investigador

Página del curso

En esta página se muestra la información del curso, sus temas y los vídeos. Para desarrollar este apartado se creó el siguiente componente que se comunica con la API para recibir los datos necesarios.

```
CoursePage.jsx ×
src > screens > CoursePage.jsx > ...
47   },
48   ];
49   const CoursePage = () => {
50     return (
51       <>
52         <Navbar />
53         <div className="online-courses container">
54           <div style={{ marginTop: "10%" }}>
55             {courses.map((course) => (
56               <CourseCard key={course.title} course={course} />
57             ))}
58           </div>
59           <div style={{ marginTop: "15%" }}>
60             {coursesInfo.map((courseInfo) => (
61               <CourseOutline
62                 key={courseInfo.title}
63                 courseInfo={courseInfo}
64                 videos={videos}
65               />
66             ))}
67           </div>
68         </div>
69         <FooterComponent />
70       </>
71     );
72   };
73
74   export default CoursePage;
75
```

Figura 60: Componente para la página del curso

Elaborado por: Investigador

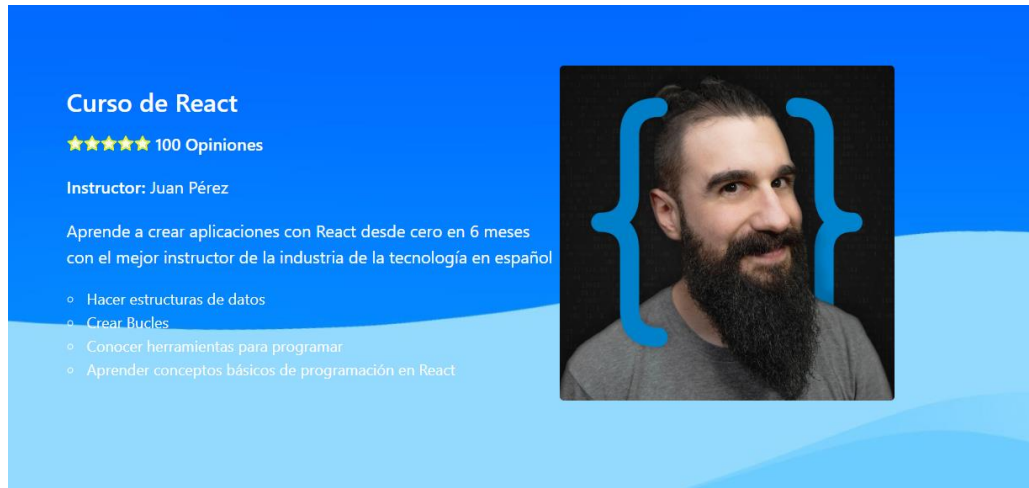


Figura 61: Diseño de la página del curso

Elaborado por: Investigador

Página para reproducir los vídeos de los cursos

Para el desarrollo de este apartado se implementó dos componentes los cuales son la lista de vídeos que contiene el curso y el reproductor del vídeo.

```
ListVideo.jsx x
src > components > CoursePage > ListVideo.jsx > ...
1 import { ClockCircleOutlined } from '@ant-design/icons'
2 import { List, Typography, Space } from 'antd'
3 import { useNavigate } from 'react-router-dom'
4 import './styles.scss'
5 function ListVideo({ videos, wv, hv }) {
6   const navigate = useNavigate()
7   return (
8     <List
9       itemLayout='horizontal'
10      dataSource={videos}
11      renderItem={video => (
12        <List.Item onClick={() => navigate('/video')} className='itemVideo'>
13          <Space direction='horizontal'>
14            <div style={{ width: wv, height: hv }}>
15              <iframe
16                title={video.title}
17                width='100%'
18                height='100%'
19                src={`https://www.youtube.com/embed/${video.id}`}
20                frameborder='0'
21                allow='accelerometer; autoplay; encrypted-media; gyroscope; picture-in-picture'
22                allowFullScreen
23              />
24            </div>
25            <div>
26              <Typography.Title style={{ color: 'white' }} level={4}>
27                {video.title}
28              </Typography.Title>
29              <Typography.Paragraph style={{ color: 'white' }}>
30                <ClockCircleOutlined /> {video.duration} minutos

```

Figura 62: Componente para reproducir los cursos

Elaborado por: Investigador

```
ListVideo.jsx X
src > components > CoursePage > ListVideo.jsx > ListVideo > <-function>
1 import { ClockCircleOutlined } from '@ant-design/icons'
2 import { List, Typography, Space } from 'antd'
3 import { useNavigate } from 'react-router-dom'
4 import './styles.scss'
5 function ListVideo({ videos, wv, hv }) {
6   const navigate = useNavigate()
7   return (
8     <List
9       itemLayout='horizontal'
10      dataSource={videos}
11      renderItem={video => (
12        <List.Item onClick={() => navigate('/video')} className='itemVideo'>
13          <Space direction='horizontal'>
14            <div style={{ width: wv, height: hv }}>
15              <iframe
16                title={video.title}
17                width='100%'
18                height='100%'
19                src={video.url}
20                frameborder='0'
21                allow='accelerometer; autoplay; encrypted-media; gyroscope; picture-in-picture'
22                allowFullScreen
23              />
24            </div>
25            <div>
26              <Typography.Title style={{ color: 'white' }} level={4}>
27                {video.title}
28              </Typography.Title>
29              <Typography.Paragraph style={{ color: 'white' }}>
30                <ClockCircleOutlined /> {video.duration} minutos
31              </Typography.Paragraph>
32            </div>
33          </Space>
34        </List.Item>
35      )
36    </List>
37  )
38}
```

Figura 63: Componente para listar los videos

Elaborado por: Investigador



Figura 64: Reproductor de vídeos

Elaborado por: Investigador

Actualizar datos del estudiante

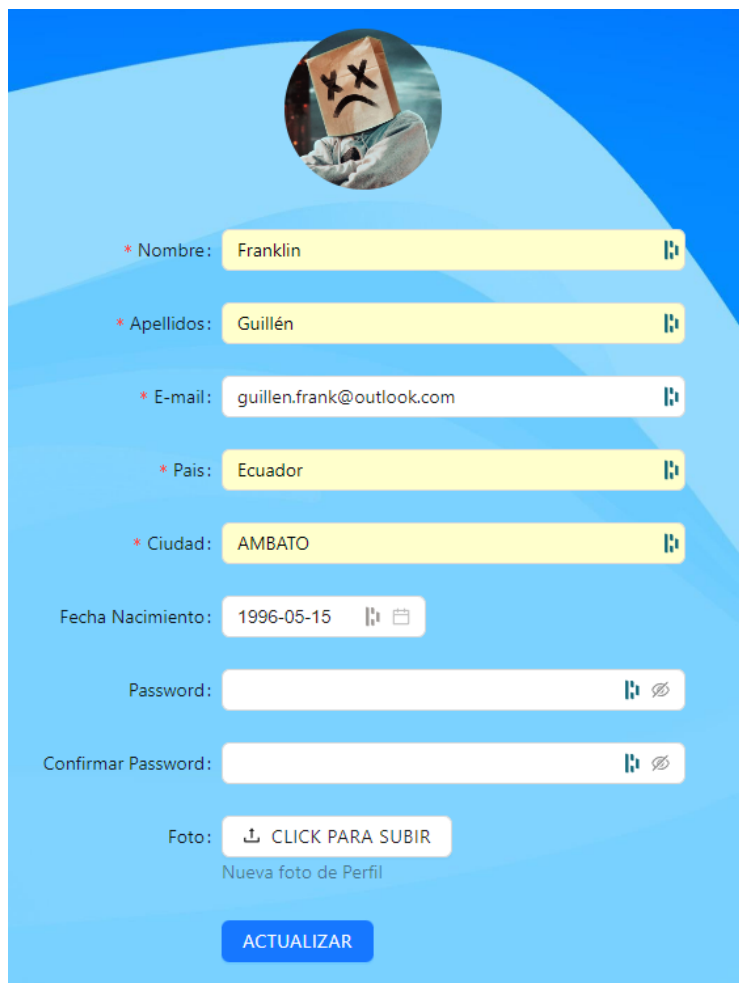
Se creó un componente para actualizar los datos de su usuario dentro de la plataforma, este componente se comunica con la API para hacer las peticiones y utiliza Redux para gestionar el estado.



```
70   },
71 };
72 const ProfilePage = () => {
73   const [form] = Form.useForm();
74   const onFinish = (values) => {
75     console.log("Received values of form: ", values);
76   };
77   const normFile = (e) => {
78     console.log("Upload event:", e);
79     if (Array.isArray(e)) {
80       return e;
81     }
82     return e?.fileList;
83   };
84   const prefixSelector = (
85     <Form.Item name="prefix" noStyle>
86       <Select
87         style={{
88           width: 100,
89         }}>
90         <Option value="593">+593</Option>
91       </Select>
92     </Form.Item>
93   );
94   const suffixSelector = (
95     <Form.Item name="suffix" noStyle>
96       <Select
97         style={{
98           width: 70,
```

Figura 65: Componente para actualizar los datos del estudiante

Elaborado por: Investigador



The image shows a user profile update form on a blue background. At the top is a circular profile picture of a person with a sad face drawn on their head. Below it are several input fields, each with a red asterisk indicating a required field. The fields are: 'Nombre' (Franklin), 'Apellidos' (Guillén), 'E-mail' (guillen.frank@outlook.com), 'Pais' (Ecuador), and 'Ciudad' (AMBATO). Below these are 'Fecha Nacimiento' (1996-05-15), 'Password', and 'Confirmar Password'. At the bottom is a 'Foto' section with a 'CLICK PARA SUBIR' button and the text 'Nueva foto de Perfil'. A blue 'ACTUALIZAR' button is at the very bottom.

* Nombre:	Franklin
* Apellidos:	Guillén
* E-mail:	guillen.frank@outlook.com
* Pais:	Ecuador
* Ciudad:	AMBATO
Fecha Nacimiento:	1996-05-15
Password:	
Confirmar Password:	
Foto:	CLICK PARA SUBIR
	Nueva foto de Perfil

ACTUALIZAR

Figura 66: Diseño para actualizar los datos del estudiante

Elaborado por: Investigador

3.2.4 Fase IV: Pruebas

3.2.4.1 Pruebas de aceptación

Se procede a realizar las pruebas de aceptación para evidenciar el cumplimiento de las historias de usuario mostradas anteriormente. Por consiguiente, de esta manera se demuestra el correcto funcionamiento de toda la plataforma de educación online para que los usuarios la utilicen con un rendimiento óptimo. Además, cada prueba de evaluación tendrá una evaluación que puede ser satisfactoria o insatisfactoria.

Tabla 26: Pruebas de aceptación 001

Elaborado por: Investigador

PRUEBA DE ACEPTACIÓN
Número: 001
Historia de usuario N°: 003
Nombre: Página informativa para la oferta de los cursos.
Descripción: Permite que los usuarios conozcan sobre la oferta de cursos que tiene GAIA.
Condiciones de ejecución: Cualquier usuario que ingrese a la página informativa puede observar la información de la empresa y de los cursos que se ofertan.
Interfaz: Se dispone de un menú para visualizar la información de la empresa, los cursos, los profesores y el contacto.
Resultado esperado: El menú debe dirigir al apartado correspondiente para que el usuario pueda leer la información.
Evaluación de la prueba: Satisfactoria

Tabla 27: Pruebas de aceptación 002

Elaborado por: Investigador

PRUEBA DE ACEPTACIÓN
Número: 002
Historia de usuario N°: 004
Nombre: Aparatado para la administración de la plataforma.
Descripción: Tiene que permitir modificar ciertos aspectos de la plataforma como los cursos, profesores y estudiantes.
Condiciones de ejecución: Si se ingresa como administrador debe permitir modificar los cursos y los profesores de la plataforma.

<p>Interfaz:</p> <p>Se dispone de un menú para abrir los componentes necesarios para editar los cursos y los profesores de la plataforma mediante formularios.</p>
<p>Resultado esperado:</p> <p>Las modificaciones o creaciones de los cursos y profesores se deben mostrar correctamente en los listados de cada apartado.</p>
<p>Evaluación de la prueba: Satisfactoria</p>

Tabla 28: Pruebas de aceptación 003

Elaborado por: Investigador

PRUEBA DE ACEPTACIÓN
<p>Número: 003</p>
<p>Historia de usuario N°: 005</p>
<p>Nombre: Aparatado para los profesores.</p>
<p>Descripción: Tiene que permitir modificar los cursos a los que tienen acceso los profesores.</p>
<p>Condiciones de ejecución:</p> <p>El usuario que ingresa a la parte administra tendrá el rol de profesor y solo debe acceder a los cursos que está asignado.</p>
<p>Interfaz:</p> <p>Existe un menú para que el usuario visualice o modifique los cursos. Para las actualizaciones de los datos de los cursos se encuentra un formulario.</p>
<p>Resultado esperado:</p> <p>La plataforma debe reflejar las modificaciones realizadas en el formulario de los cursos y listar correctamente los cursos a los que está asignado.</p>
<p>Evaluación de la prueba: Satisfactoria</p>

Tabla 29: Pruebas de aceptación 004

Elaborado por: Investigador

PRUEBA DE ACEPTACIÓN
Número: 004
Historia de usuario N°: 006
Nombre: Aparatado para los estudiantes.
Descripción: Tiene que permitir acceder y visualizar las clases de los cursos en los que se encuentren matriculados.
Condiciones de ejecución: Se procede a ingresar a este apartado con un usuario registrado en la plataforma para visualizar los cursos existentes.
Interfaz: Existe un menú para editar los datos del usuario con un formulario y para acceder a los cursos.
Resultado esperado: El usuario debe acceder a la plataforma, visualizar los cursos y modificar sus datos correctamente.
Evaluación de la prueba: Satisfactoria

3.2.4.2 Pruebas de rendimiento

Pruebas de rendimiento con Apache JMeter

Apache JMeter es un software que crea peticiones a un servicio web para realizar pruebas de estrés, con esta herramienta se hicieron 250 peticiones a varios servicios implementados en el proyecto Nest.js. Además, se configuraron 10 peticiones por segundo con 5 repeticiones cada prueba.

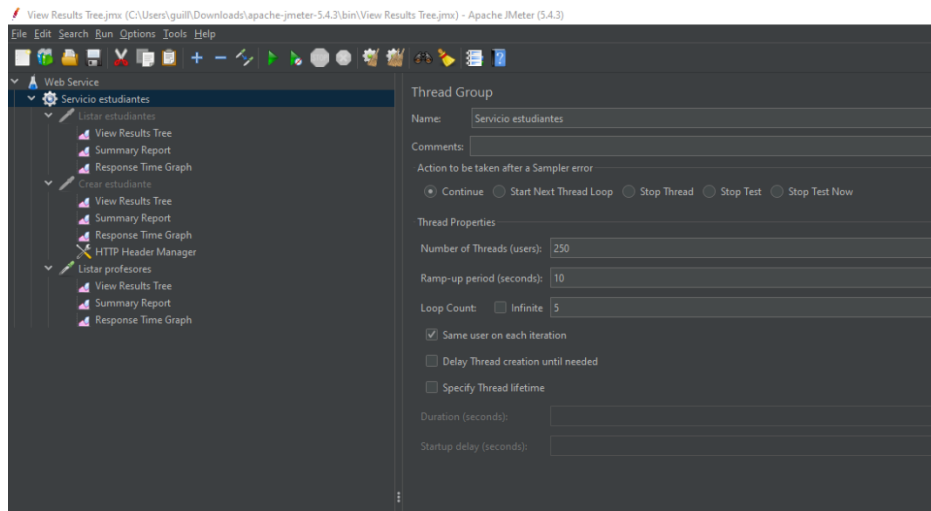


Figura 67: Configuración para las pruebas de estrés

Elaborado por: Investigador

Configuración para las pruebas de los estudiantes

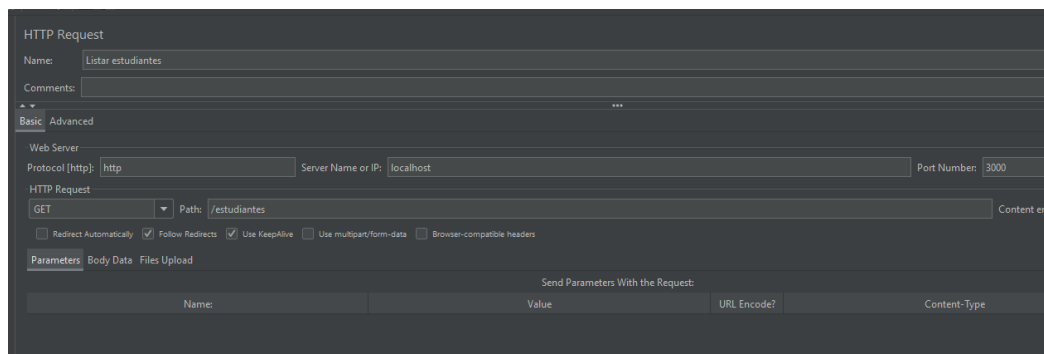


Figura 68: Configuración para las pruebas de los estudiantes

Elaborado por: Investigador

Resultados

Se realizaron 500 peticiones en un tiempo de 1.2/sec y con un porcentaje de error del 0% de todas las peticiones.

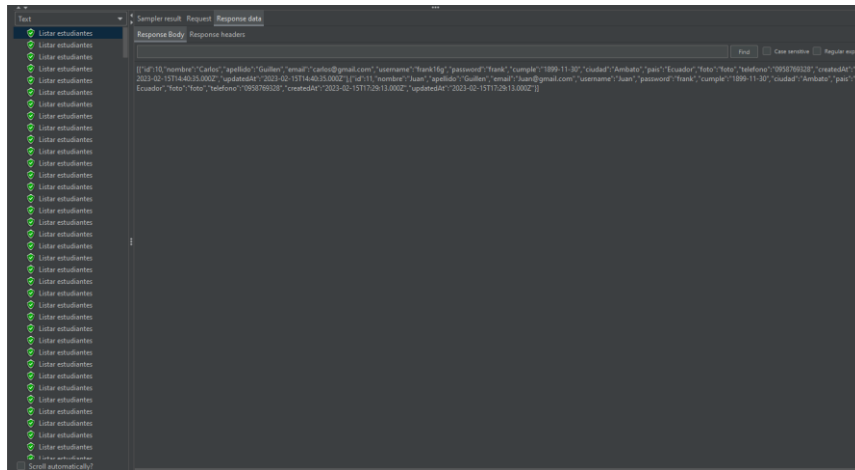


Figura 69: Datos de las pruebas de los estudiantes

Elaborado por: Investigador

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Listar estudiantes	500	2	1	42	3.15	0.00%	1.2/sec	1.00	0.15	842.0
TOTAL	500	2	1	42	3.15	0.00%	1.2/sec	1.00	0.15	842.0

Figura 69: Reporte de las pruebas de los estudiantes

Elaborado por: Investigador

Configuración para las pruebas de los profesores

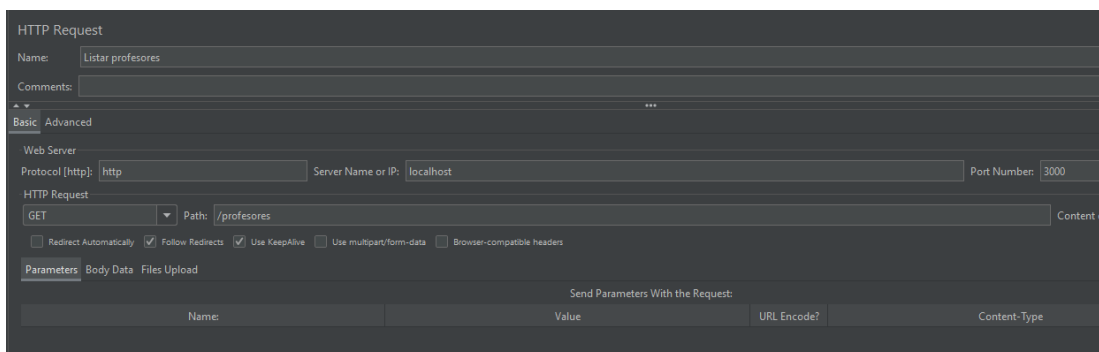


Figura 70: Configuración de las pruebas de los profesores

Elaborado por: Investigador

Resultados

Como resultado de las pruebas realizadas a los servicios de los profesores se obtuvo un 0% de error en las 250 peticiones realizadas en 25.1/sec.

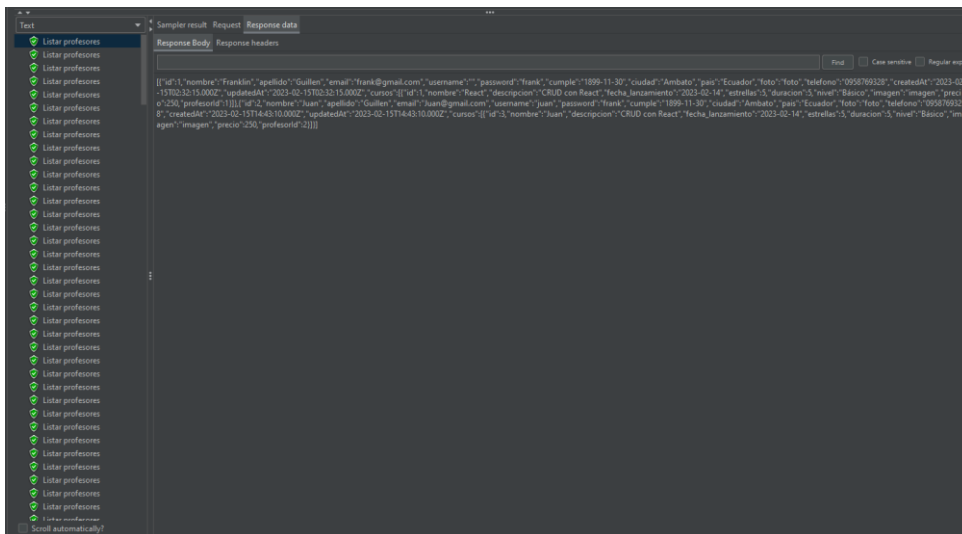


Figura 71: Datos de las pruebas de los profesores

Elaborado por: Investigador

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Listar profesores	250	2	1	5	0.55	0.00%	25.1/sec	29.81	3.16	1216.0
TOTAL	250	2	1	5	0.55	0.00%	25.1/sec	29.81	3.16	1216.0

Figura 72: Reporte de las pruebas de los profesores

Elaborado por: Investigador

Configuración para las pruebas de los cursos

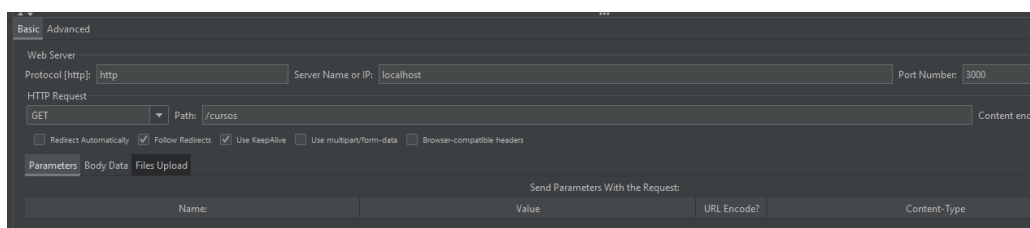


Figura 73: Configuración de las pruebas de los cursos

Elaborado por: Investigador

Resultados

Como resultados de las pruebas de los servicios de los profesores se obtuvo un 0% de error en las 1250 peticiones realizadas en 125.4/sec.

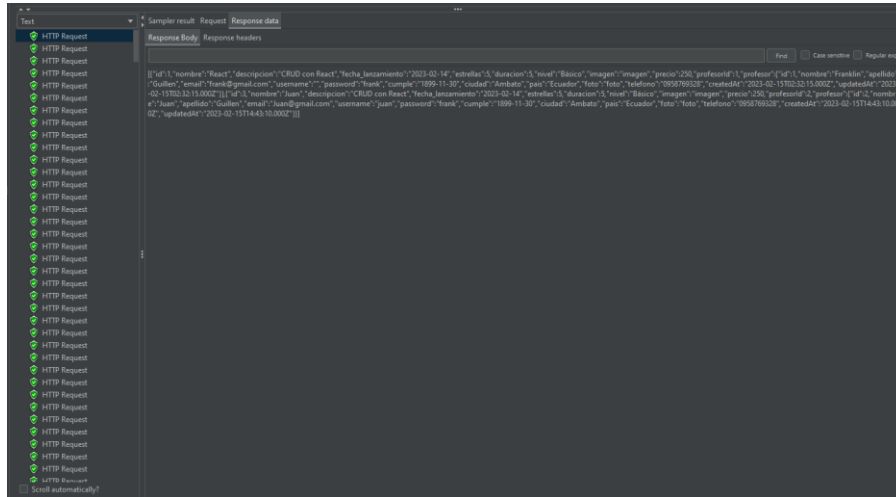


Figura 74: Datos de las pruebas de los cursos

Elaborado por: Investigador

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
HTTP Request	1250	1	0	4	0.60	0.00%	125.4/sec	148.93	15.31	1216.0
TOTAL	1250	1	0	4	0.60	0.00%	125.4/sec	148.93	15.31	1216.0

Figura 75: Reporte de las pruebas de los cursos

Elaborado por: Investigador

Conclusiones de las pruebas realizadas

La empresa Google tiene un conjunto de métricas enfocados a medir la experiencia del usuario llamados Core Web Vitals. En estas métricas se considera aceptable un tiempo de carga de hasta 300ms, pasado este tiempo se considera no aceptable.[18]

Según lo anterior mencionado, los resultados obtenidos son aceptables y satisfactorios para que los usuarios puedan hacer uso de la plataforma sin dificultades.

3.2.5 Fase V: Implantación

3.2.5.1 Deploy de los proyectos desarrollados en React

Para hacer el deploy se utilizó los comandos de yarn, estos comandos se ejecutan en la consola cmd para crear una carpeta llamada dist que contiene toda la página de forma comprimida y con los archivos encriptados para que desde el navegador no se pueda acceder al código de la plataforma.

```
EXPLORER
GAIACURSOS-MAIN
  dist
  node_modules
  public
  src
  .gitignore
  index.html
  package-lock.json
  package.json
  vite.config.js
  yarn.lock

TERMINAL
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\guill\Downloads\gaiacursos-main>npm build
Unknown command: "build"

Did you mean this?
  npm run build # run the "build" package script

To see a list of supported npm commands, run:
  npm help

C:\Users\guill\Downloads\gaiacursos-main>yarn build
yarn run v1.22.19
$ vite build
vite v4.1.1 building for production...
✓ 2649 modules transformed.
dist/assets/rupees-21c1aa26.svg      0.39 kB
dist/index.html                    0.45 kB
dist/assets/growth-12ddf8de.svg    0.78 kB
dist/assets/community-4c5bee12.svg 1.15 kB
dist/assets/timer-9fc50b66.svg     1.40 kB
dist/assets/offer-4098a1c6.svg     1.71 kB
dist/assets/favicon-27f9d585.png   150.66 kB
dist/assets/logo-1926f3fb.png     330.23 kB
dist/assets/profile-3e8e608e.jpg    362.30 kB
dist/assets/index-d43d7c45.css      10.55 kB | gzip: 2.62 kB
dist/assets/index-b0e7363b.js       932.72 kB | gzip: 291.57 kB

(!) Some chunks are larger than 500 kB after minification. Consider:
- Using dynamic import() to code-split the application
- Use build.rollupOptions.output.manualChunks to improve chunking: htt
- Adjust chunk size limit for this warning via build.chunkSizeWarningL
Done in 21.48s.

C:\Users\guill\Downloads\gaiacursos-main>
```

Figura 76: Deploy del proyecto en React

Elaborado por: Investigador

3.2.5.2 Subir la carpeta dist al hosting

La carpeta dist se subió hosting contratado y se modificó el nombre.

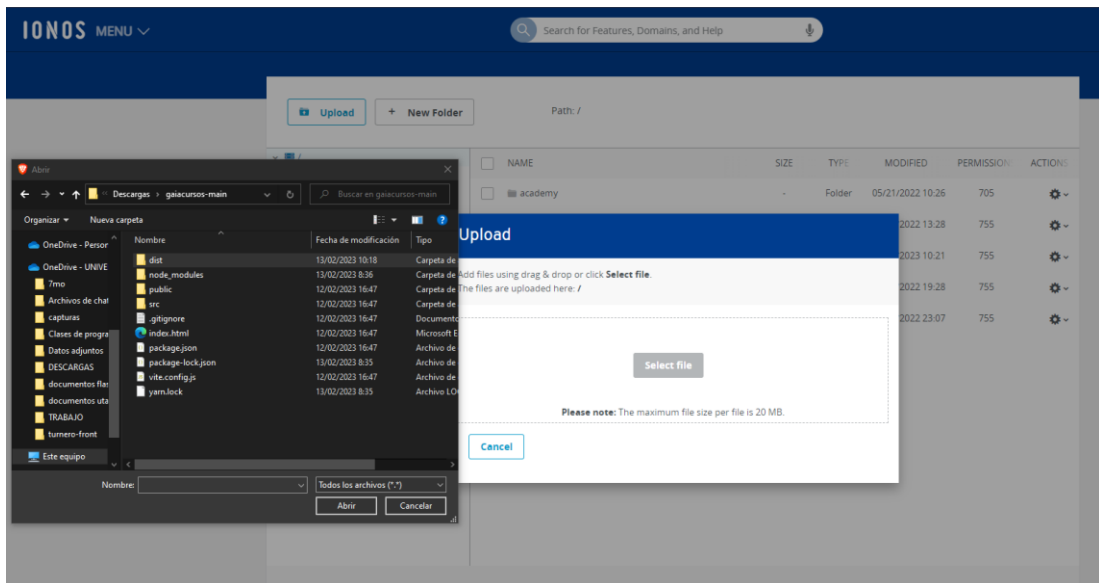


Figura 77: Subir carpeta al hosting

Elaborado por: Investigador

En la configuración de los dominios se asignó una carpeta a cada subdominio, los cuales son:

- Página informativa: gaiaecsa.com
- Página administrativa: ademic.gaiacsa.com
- Página para estudiantes: student.gaiacsa.com



	DOMAIN	STATUS
	ademic.gaiacsa.com Subdomain	Webspace (Contract 89830883) /ademic
	student.gaiacsa.com Subdomain	Webspace (Contract 89830883) /student

Figura 78: Subdominios

Elaborado por: Investigador

CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Mediante el análisis e interpretación de la información recolectada se logró identificar las necesidades de la empresa y de sus clientes, siguiendo la metodología de desarrollo se realizó pruebas para verificar el cumplimiento de los requerimientos.
- La información obtenida en la investigación de la aplicación de ReactJS para el desarrollo de una plataforma online fue útil para aprovechar las ventajas de ReactJS. Gracias a esta investigación se desarrolló una plataforma robusta, de carga rápida y fácil de actualizar en el futuro.
- La implantación se logró exitosamente utilizando un dominio y subdominios que ofrecen acceso a la página informativa, el acceso administrativo de la plataforma, el acceso a los profesores y por último el acceso de los estudiantes. El uso de la herramienta yarn facilitó el deploy y se implantó correctamente los proyectos de React al hosting.
- El uso de la plataforma desarrollada permite una mejor aceptación por parte de los estudiantes y un mejor seguimiento del trabajo de los profesores que elaboran los cursos; asimismo se puede llevar a cabo un adecuado control de las ofertas de los cursos. De igual forma su implementación permite un aumento de la productividad y mejorar la calidad de los cursos.

4.2 Recomendaciones

- Para crear módulos o controladores es importante utilizar la herramienta de interfaz de la línea de comandos de Nest.js que se llama NestCLI. Esta herramienta ayuda a generar código para facilitar la codificación y menorar los tiempos de desarrollo.
- Se recomienda monitorear con webpack el tamaño del bundle en el deploy del proyecto, ya que el uso de React puede provocar el crecimiento innecesario de esta carpeta y ralentizar la implantación.
- Se sugiere el uso del framework Nest.js para la creación de servicios para el desarrollo de una API Rest o GraphQL. Existen librerías para el uso de tipo de APIs que facilitan su codificación e implementación.
- En la implantación se aconseja modificar el nombre de la carpeta obtenida por el comando yarn build, de esta forma se garantiza una mayor seguridad dentro del hosting ya que los posibles atacantes no conocerán los nombres de las carpetas que están subidas al hosting.

Bibliografía

- [1] P. Villar, “Por qué América Latina invierte tan poco en innovación,” *El Comer.*, pp. 1–7, 2021, Accessed: May 30, 2022. [Online]. Available: <https://www.revistalideres.ec/lideres/innovacion-tecnologia-empresas-latinoamerica-caf.html>
- [2] R. Freire Avilés, J. Diaz Nava, W. Rodríguez Suárez, and N. Vera Lucio, “Modelo de educación virtual a distancia en el contexto universitario latinoamericano,” *Rev. Ibérica Sist. e Tecnol. Informação Iber.*, vol. 1, no. 01, pp. 1048–1062, 2018, Accessed: May 30, 2022. [Online]. Available: <http://www.risti.xyz/issues/ristie18.pdf>
- [3] E. W. F. MEDINA, “Evaluación del uso de las plataformas virtuales en los estudiantes del programa de maestría en docencia y gerencia educativa de la unidad de postgrado investigación y desarrollo de la universidad de Guayaquil. para fortalecer sus conocimientos tecnológicos. diseño de un manual,” Universidad de Guayaquil, Guayaquil, 2013. Accessed: May 30, 2022. [Online]. Available: [http://repositorio.ug.edu.ec/bitstream/redug/2032/1/Tesis final de Fabricio Medina.pdf](http://repositorio.ug.edu.ec/bitstream/redug/2032/1/Tesis%20final%20de%20Fabricio%20Medina.pdf)
- [4] López Hinojosa José Daniel, “Arquitectura de software basada en microservicios para desarrollo de aplicaciones web de la asamblea nacional,” Universidad Técnica del Norte, Ibarra, 2017. Accessed: May 30, 2022. [Online]. Available: [http://repositorio.utn.edu.ec/bitstream/123456789/7603/1/PG_569 TESIS.pdf](http://repositorio.utn.edu.ec/bitstream/123456789/7603/1/PG_569_TESIS.pdf)
- [5] M. Salomé, D. Silva, I. Carlos Caizaguano, and I. G. Gallegos, “Análisis, diseño e implementación de un sistema de aula virtual para capacitación de personal en la empresa Undermedia s.a”.
- [6] Chica González Jhon Denver and Rodríguez Mendoza Karla Lilibeth, “Desarrollo y ejecución de un aula virtual para cursos de formación continua para docentes no universitarios en pequeñas empresas,” Universidad de Guayaquil, Guayaquil, 2020. Accessed: May 30, 2022. [Online]. Available: [http://repositorio.ug.edu.ec/bitstream/redug/48815/1/B-CINT-PTG-N_509 Chica González Jhon Denver . Rodríguez Mendoza Karla Lilibeth.pdf](http://repositorio.ug.edu.ec/bitstream/redug/48815/1/B-CINT-PTG-N_509_Chica%20González%20Jhon%20Denver%20Rodríguez%20Mendoza%20Karla%20Lilibeth.pdf)
- [7] Lincango Córdor Luis Isaiás, “Estudio de factibilidad para el diseño e

- implementación de la plataforma de educación on-line en la Facultad de Ciencias Económicas de la Universidad Central del Ecuador.,” Universidad Central del Ecuador , Quito, 2018. Accessed: May 30, 2022. [Online]. Available: <http://www.dspace.uce.edu.ec/bitstream/25000/17278/1/T-UCE-0005-CEC-008-P.pdf>
- [8] Evelyn Elizabeth Ayala Ñiquen and Santiago Raúl Gonzales Sánchez, *Tecnologías de la información y la comunicación*, 1st ed. Lima: Fondo Editorial de la UIGV, 2015. Accessed: May 30, 2022. [Online]. Available: [http://repositorio.uigv.edu.pe/bitstream/handle/20.500.11818/1189/Libro TIC %282%29-1-76 %281%29.pdf?sequence=1&isAllowed=y](http://repositorio.uigv.edu.pe/bitstream/handle/20.500.11818/1189/Libro_TIC%282%29-1-76%281%29.pdf?sequence=1&isAllowed=y)
- [9] “elearningeuropa.info - contacto con el propietario del dominio | Epik.com.” https://elearningeuropa.info/extras/pdf/virtual_models.pdf (accessed May 30, 2022).
- [10] Á. M. Fernández Martínez, “Las plataformas virtuales en la enseñanza semipresencial de adultos,” 2012.
- [11] “React – Una biblioteca de JavaScript para construir interfaces de usuario.” <https://es.reactjs.org/> (accessed May 30, 2022).
- [12] D. Alberto *et al.*, “Relación entre oferta de cursos tipo MOOC e índices económicos por país en el contexto europeo y latinoamericano 1”.
- [13] “que es formacion complementaria | web oficial euroinnova.” <https://www.euroinnova.ec/blog/que-es-formacion-complementaria> (accessed Jun. 13, 2022).
- [14] J. Manuel and T. López, “El concepto de educación. carácter, sentido pedagógico, significado y orientación formativa temporal. hacia la construcción de ámbitos de educación”, Accessed: Jun. 13, 2022. [Online]. Available: <http://dondestalaeducacion.com/>

Anexos

A.1 Alfa de Cronbach

Se aplicó este método para medir la fiabilidad de la encuesta aplicada a los estudiantes y se obtuvieron estos resultados.

	Item	Item	Item	Item	Item	Item	Item	Item	Item	Item	Suma
	1	2	3	4	5	6	7	8	9	10	
Sujeto 1	4	4	5	4	5	4	4	4	5	5	44
Sujeto 2	4	4	4	5	5	5	5	5	5	5	47
Sujeto 3	5	5	5	5	5	5	5	5	5	5	50
Sujeto 4	5	4	4	4	4	4	4	4	5	4	42
Sujeto 5	4	4	4	4	5	4	4	4	4	5	42
Sujeto 6	4	4	4	4	5	4	4	4	4	5	42
Sujeto 7	5	5	5	5	5	4	4	5	5	5	48
Sujeto 8	3	4	5	4	4	3	4	4	4	5	40
Sujeto 9	4	4	5	4	5	5	4	4	3	5	43
Sujeto 10	4	5	5	5	5	5	5	4	4	5	47
Sujeto 11	4	4	5	4	5	4	4	4	5	5	44
Sujeto 12	4	4	4	5	5	5	5	5	5	5	47

Sujeto 13	5	5	5	5	5	5	5	5	5	5	50
Sujeto 14	5	4	4	4	4	4	4	4	5	4	42
Sujeto 15	4	4	4	4	5	4	4	4	4	5	42
Sujeto 16	4	4	4	4	5	4	4	4	4	5	42
Sujeto 17	5	5	5	5	5	4	4	5	5	5	48
Sujeto 18	3	4	5	4	4	3	4	4	4	5	40
Sujeto 19	4	4	5	4	5	5	4	4	3	5	43
Sujeto 20	4	5	5	5	5	5	5	4	4	5	47
Varianza	0,3789	0,2210	0,2526	0,2526	0,1684	0,4315	0,2210	0,2210	0,4631	0,0947	10,1578

Fórmula para el alfa de Cronbach

$$\alpha = \frac{K}{K - 1} \left(1 - \frac{\sum Vi}{Vt} \right)$$

Alfa	0,8102
k (número de ítems)	10
Vi (varianza de cada ítem)	2,7052
Vt (varianza total)	10,1578

A.2 Manual de usuario

Manual de usuario

“PLATAFORMA DE EDUCACIÓN ONLINE UTILIZANDO REACTJS PARA MEJORAR LA OFERTA DE CURSOS DE LA EMPRESA GAIA DE LA CIUDAD DE AMBATO”

El presente manual de usuario se realizó con el afán de que los administradores de la plataforma de educación online puedan comprender todas las funcionalidades y configuraciones, en cada interfaz de la plataforma se detalla su objetivo y su uso para facilitar la interacción con el usuario.

Página informativa

Esta página cuenta con un menú en la parte superior para interactuar con las diferentes secciones.

El objetivo de esta página informativa es para mostrar los cursos y dar a conocer la empresa GAIA.

Página administrativa

Esta página cuenta con un menú para acceder a las configuraciones de los profesores y de los estudiantes.

Para acceder al usuario se debe dar clic en esta sección

Crear un nuevo profesor

GAIA ACADEMIC

INICIO

Inicio

PROFESORES

Nuevo

Todos

CURSOS

Nuevo

Todos

Agregar Temario

Agregar Clase

INFORMACIÓN DEL PROFESOR

Nombre

Apellido

Ciudad

País

Username

Email

Fecha nacimiento

dd/mm/aaaa

Telefono

Foto

Seleccionar archivo Ninguno archivo selec.

Crear

Para crear un nuevo profesor se debe ingresar dando clic aquí.

El siguiente paso será llenar el formulario y dar clic en el botón Crear.

Ver todos los profesores

INICIO

Inicio

PROFESORES

Nuevo

Todos

CURSOS

Nuevo

Todos

Agregar Temario

Agregar Clase

TODOS LOS PROFESORES

#	Nombre	Apellido	Username	Editar	Eliminar
1	Mark	Otto	@mdo	Editar	Eliminar
2	Jacob	Thomton	@fat	Editar	Eliminar
3	Larry	the Bird	@twitter	Editar	Eliminar

Para ver el listado de los profesores se debe ingresar dando clic aquí.

Opción para editar el profesor.

Opción para eliminar el profesor.

Crear un nuevo curso

INICIO

- Inicio

PROFESORES

- Nuevo
- Todos

CURSOS

- Nuevo
- Todos
- Agregar Temario
- Agregar Clase

INFORMACIÓN DEL CURSO

Nombre
Ejemplo: ReactJS

Precio
Ejemplo: 250

Profesor
Juan

Estrellas
1

Nivel
Fácil

Descripción

Imagen
Seleccionar archivo Ninguno archivo selec.

Crear

Para crear un nuevo curso se debe ingresar dando clic aquí.

El siguiente paso será llenar el formulario y dar clic en el botón Crear.

Ver todos los cursos

GAIA ACADEMIC

INICIO

- Inicio

PROFESORES

- Nuevo
- Todos

CURSOS

- Nuevo
- Todos
- Agregar Temario
- Agregar Clase

TODOS LOS CURSOS

#	Nombre	Estrellas	Precio	Profesor	Listado de estudiantes	Editar	Eliminar
1	Excel	4	50	Pedro	Ver	Editar	Eliminar
2	Excel	4	50	Pedro	Ver	Editar	Eliminar
3	Excel	4	50	Pedro	Ver	Editar	Eliminar

Para ver el listado de los cursos se debe ingresar dando clic aquí.

Opción para ver el listado de los estudiantes.

Opción para editar el curso.

Opción para eliminar el curso.

Agregar nuevo tema

INICIO

- Inicio

PROFESORES

- Nuevo
- Todos

CURSOS

- Nuevo
- Todos
- Agregar Temario
- Agregar Clase

CREAR TEMARIO

Nombre
Ejemplo: ReactUS

Curso
React

CREAR EXAMEN

Pregunta

Distractores

Ingresar pregunta

Crear Tema

Para crear un nuevo tema se debe ingresar dando clic aquí.

El siguiente paso será llenar el formulario y dar clic en el botón Crear Tema.

Agregar nueva clase

GAIA ACADEMIC

INICIO

- Inicio

PROFESORES

- Nuevo
- Todos

CURSOS

- Nuevo
- Todos
- Agregar Temario
- Agregar Clase

CREAR CLASE

Nombre
Ejemplo: ReactUS

Descripcion
Ejemplo: ReactUS

Curso
React

Tema
Introduccion

Video
Seleccionar archivo Ninguno archivo selec.

RECURSOS

Recurso
Seleccionar archivo Ninguno archivo selec.

Crear

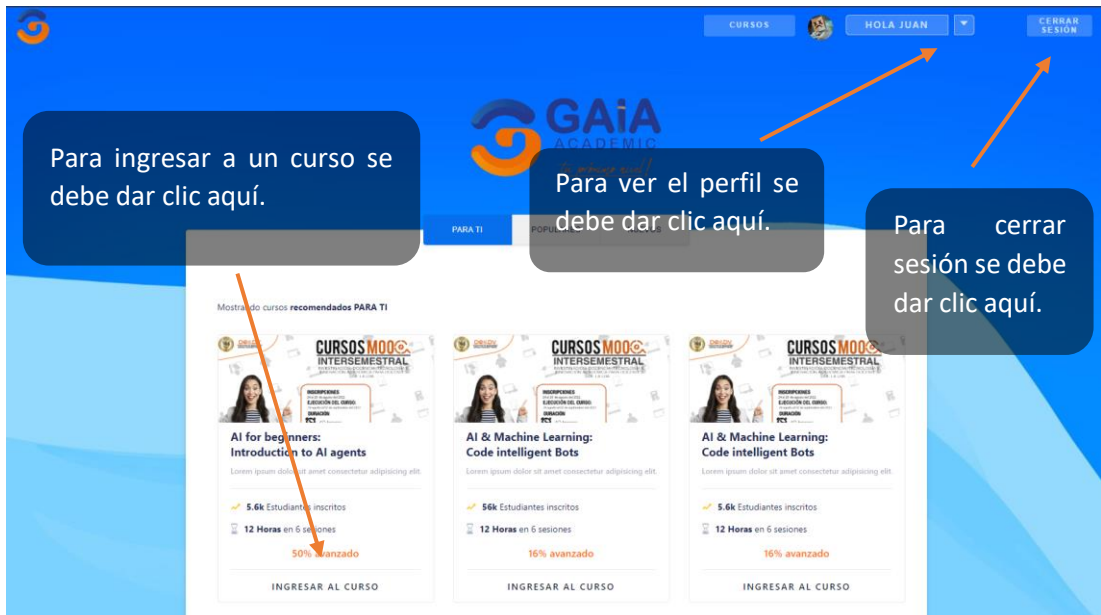
Para crear una nueva clase se debe ingresar dando clic aquí.

El siguiente paso será llenar el formulario y dar clic en el botón Crear.

Página estudiantes



Listado de cursos



Información del curso

Curso de React

★★★★★ 100 Opiniones

Instructor: Juan Pérez

Aprende a crear aplicaciones con React desde cero en 6 meses con el mejor instructor de la industria de la tecnología en español

- Hacer estructuras de datos
- Crear Bucles
- Conocer herramientas para programar
- Aprender conceptos básicos de programación en React



Temario y recursos del Curso de React

Para este curso vas a necesitar:

Para visualizar la clase se debe dar clic aquí.

Introducción a la programación:


- Introducción al curso
⌚ 5:00 minutos

Software para este Curso:

- Visual Studio Code
- Node.js
- Git

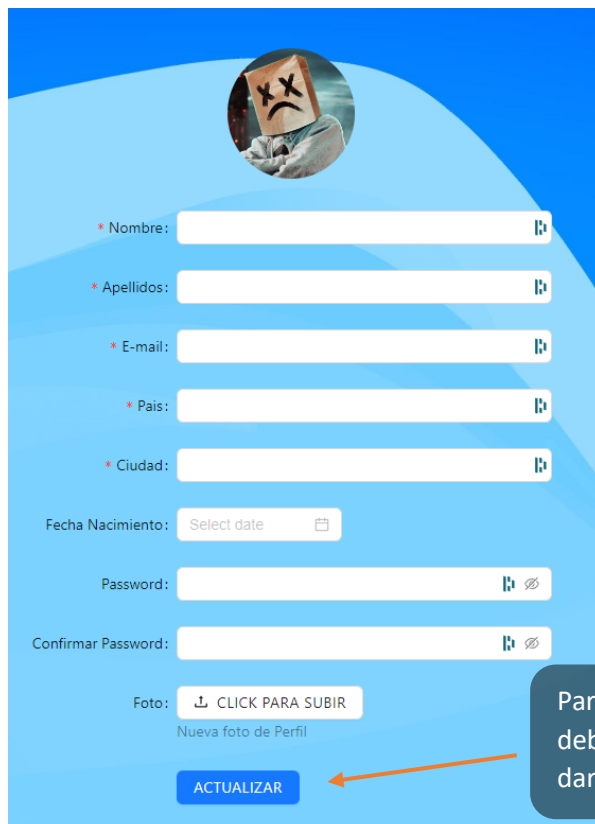
Configuración del entorno de desarrollo

Reproductor del curso



En el menú lateral se puede navegar por las clases que contiene el curso.

Actualizar datos de perfil



A user profile update form with a blue background. At the top left is a circular profile picture placeholder showing a person with a sad face. Below it are several input fields: Name, Surnames, E-mail, Country, and City, each with a red asterisk and a validation icon. The birth date field is labeled 'Fecha Nacimiento:' and has a 'Select date' button. The password fields are labeled 'Password:' and 'Confirmar Password:', both with a red asterisk and a validation icon. The photo field is labeled 'Foto:' and has a 'CLICK PARA SUBIR' button and the text 'Nueva foto de Perfil'. At the bottom is a blue 'ACTUALIZAR' button.

* Nombre:

* Apellidos:

* E-mail:

* Pais:

* Ciudad:

Fecha Nacimiento:

Password:

Confirmar Password:

Foto:
Nueva foto de Perfil

Para actualizar los datos se debe llenar el formulario y dar clic en actualizar