



UNIVERSIDAD TECNOLÓGICA CENTROAMERICANA

FACULTAD DE INGENIERÍA

PROYECTO DE GRADUACIÓN

PLATAFORMA ESCOLAR Y PROMOCIONAL PARA PROYECTOS EL HOGAR

PREVIO A LA OBTENCIÓN DEL TÍTULO

INGENIERO EN SISTEMAS COMPUTACIONALES

PRESENTADO POR:

11441295 ALEXY OMAR CRUZ AGUILERA

ASESOR: LIC. TANIA MEZA

CAMPUS TEGUCIGALPA; ENERO, 2021

Para empezar, quiero dedicar este trabajo primeramente a Dios por permitirme culminar mi carrera universitaria y en segundo a mi familia, a mis padres Alexy Cruz y Karla Álvarez por quienes no tengo palabras suficientes para mostrarles toda mi gratitud, a mis hermanos Scarleth Cruz e Iván Sagastume y mi abuela Petronila Carias en quienes sé que siempre puedo encontrar apoyo.

Quisiera dedicárselo especialmente a Patricia Carpio quien además de ser una maestra fue una inspiración para mi vida, de igual forma quisiera agradecer a Nelly Chávez, Mirna Cerrato y Nelson Chávez quienes además de ser mis educadores fueron mi familia.

Para finalizar, quisiera dedicárselo a aquellos amigos que, aunque no compartieron mi vida universitaria, estuvieron conmigo desde que tengo memoria, ayudándome, apoyándome y acompañándome en cada etapa, es por eso que agradezco a Andrea Canaca y Eduardo Ramos quienes han sido amigos incondicionales. Así mismo quisiera dedicar mi tesis a aquellos amigos que caminaron a mi lado durante mi carrera universitaria, José Fiallos, Oscar Díaz y Dennis Cárcamo quienes estuvieron conmigo en incontables momentos y demostraron ser más fieles que un hermano. De igual manera quisiera agradecer a aquellos amigos que dejaron grandes enseñanzas en mi vida como lo son Asadia Rosado, David Molina, Juany Ramírez, Manuel Padilla, Félix Maldonado, Víctor Euceda, Christian Rodríguez, Herbert Paz, Didier Murillo y Nohelia Euceda.

AGRADECIMIENTO

Para empezar, quiero agradecer primeramente a Dios por permitirme culminar mi proyecto de graduación y en segundo a las Licenciadas Tania Meza y Jessika Claros quienes fueron mis asesoras durante el tiempo que requirió el desarrollo de este trabajo.

Para terminar, quisiera agradecer a Proyectos El Hogar por permitirme realizar un proyecto que ayude y fomente la educación en su institución.

RESUMEN EJECUTIVO

El presente documento se entrega como requisito para optar al título profesional de Ingeniero en Sistemas Computacionales mediante el desarrollo de una plataforma web para Proyectos El Hogar de Amor y Esperanza, el cual se desarrolló durante el periodo de Julio a Diciembre de 2020.

El proyecto consiste en una plataforma educativa y promocional las cuales tienen el objetivo de crear un espacio en el que el personal referente a la institución pueda interactuar entre sí y realizar las actividades que habitualmente concretan dentro de los espacios de la institución.

La plataforma educativa busca unir a maestros y estudiantes sustituyendo las aulas de clases por espacios virtuales de enseñanza, además de unir a personal administrativo con el objetivo de hacer eficiente la planeación y planteamiento de nuevo material y actividades.

La plataforma promocional busca dar a conocer todas las instituciones dentro de los Proyectos El Hogar de Amor y Esperanza y así facilitar la interacción entre el público general y entes administrativos, además busca facilitar las donaciones hacia la institución a través de campañas.

Para el desarrollo de ambas plataformas se utilizaron tecnologías de vanguardia como lo son React JS, Node JS y Heroku, además de buenas prácticas de programación que garantizan la escalabilidad y seguridad de los sitios web.

EXECUTIVE SUMMARY

This document is delivered as a requirement to qualify for the professional title of Computer Systems Engineer through the development of a web platform for El Hogar de Amor y Esperanza Projects, which was developed during the period from July to December 2020.

The project consists of an educational and promotional platform which have the objective of creating a space in which the personnel related to the institution can interact with each other and carry out the activities that they usually specify within the spaces of the institution.

The educational platform seeks to unite teachers and students by replacing classrooms with virtual teaching spaces, in addition to uniting administrative staff with the aim of making the planning and approach of new material and activities efficient.

The promotional platform seeks to publicize all the institutions within the El Hogar de Amor y Esperanza Projects and thus facilitate interaction between the general public and administrative entities, and also seeks to facilitate donations to the institution through campaigns.

For the development of both platforms, cutting-edge technologies such as React JS, Node JS and Heroku were used, as well as good programming practices that guarantee the scalability and security of the websites.

ÍNDICE DE CONTENIDO

I. Introducción	1
II. Planteamiento del Problema.....	3
2.1 Precedentes del Problema	3
2.2 Definición del Problema	3
2.3 Justificación	4
2.4 Preguntas de Investigación	4
2.5 Objetivos.....	4
2.5.1 Objetivo General.....	4
2.5.2 Objetivos Específicos	4
III. Marco Teórico.....	6
3.1 La nube.....	6
3.2 Heroku	8
3.3 React JS	8
3.4 Node JS	9
3.5 Metodologías Ágiles	12
IV. Metodología	18
4.1 Enfoque	18
4.2 Variables de Investigación	18
4.3 Técnicas e Instrumentos aplicados	18
4.3.1 Visual Studio Code.....	18
4.3.2 React JS	19
4.3.3 Express JS	19

4.3.4 MongoDB Atlas.....	19
4.3.5 Casos de Uso.....	19
4.4 Metodología de Estudio	22
4.4.1 Modelo Ágil.....	22
4.5 Cronograma de Actividades.....	24
V. Resultados y Análisis.....	25
5.1 Interfaz de Programación de Aplicaciones.....	25
5.2 Capturas de pantalla del sistema web informativo.....	28
5.3 Capturas de pantalla del sistema web educativo.....	33
VI. Conclusiones.....	45
VII. Recomendaciones	46
Bibliografía.....	47

ÍNDICE DE ILUSTRACIONES

Ilustración 1 - Modelos de Servicios de Computación en la Nube.....	7
Ilustración 2 - Desarrollo en Heroku.....	8
Ilustración 3 - Manejo de Peticiones de Node	10
Ilustración 4 - Comparativa de Peticiones por Segundo	11
Ilustración 5 - Estructura de Sprint.....	17
Ilustración 6 - Caso de uso de Administradores	20
Ilustración 7 - Caso de Uso de Maestros	21
Ilustración 8 - Caso de Uso de Estudiantes.....	22
Ilustración 9 - Ciclo de Trabajo de Scrum.....	23
Ilustración 10 - Sección Principal	28
Ilustración 11 - Sobre Nosotros.....	29
Ilustración 12 - Nuestros Valores.....	30
Ilustración 13 - Nuestro Equipo.....	31
Ilustración 14 - Contáctanos.....	32
Ilustración 15 - Formulario Inicio de Sesión	33
Ilustración 16 - Formulario de Creación de Cuenta	34
Ilustración 17 - Sección de Usuarios (Administrador).....	35
Ilustración 18 - Sección de Cursos (Administrador).....	36
Ilustración 19 - Sección de Cuenta (Administrador).....	36
Ilustración 20 - Sección de Cursos (Maestro).....	37
Ilustración 21 - Sección de Estudiantes (Maestro)	37
Ilustración 22 - Sección de Cuenta (Maestro)	38
Ilustración 23 - Sección de Archivos (Maestro)	38
Ilustración 24 - Sección de Cursos (Estudiante)	39
Ilustración 25 - Sección de Campañas	39
Ilustración 26 - Ingreso a Sala de Chat.....	40
Ilustración 27 - Sala de Chat.....	41
Ilustración 28 - Sección de Video Chat (Inicio).....	42

Ilustración 29 - Sala de Video Chat (Pre visualización).....	43
Ilustración 30 - Sección de Video Chat (Llamada).....	44

ÍNDICE DE TABLAS

Tabla 1 - Cronograma de Trabajo.....	24
Tabla 2 - API de Autenticación.....	25
Tabla 3 - API de Restablecimiento.....	25
Tabla 4 - API de Administradores.....	26
Tabla 5 - API Operaciones Básicas.....	27

LISTA DE SIGLAS Y GLOSARIO

AI	Artificial Intelligence
API	Application Programming Integration
CLI	Command Line Interface
CSS	Cascading Style Sheet
DB	Data base
ER	Entidad Relación
HRML	Human Resource Management Markup Language
HTML	HyperText Markup Language
HTTPS	HyperText Transfer Protocol Secure
IaaS	Infrastructure as a Service
JS	Javascript
JSON	Javascript Object Notation
NPM	Node Package Manager
PaaS	Product as a Service
SaaS	Software as a Service
SASS	Syntactically Awesome Style Sheets
UX	User Experience

UI User Interface

Apache: es un software de servidor web gratuito y de código abierto para plataformas Unix con el cual se ejecutan el 46% de los sitios web de todo el mundo. ("¿Qué es Apache? Descripción completa del servidor web Apache", 2020)

Backend: es la parte del desarrollo web que se encarga de que toda la lógica de una página web funcione. ("Qué es el Backend de una web y por qué es tan importante", 2020)

Browserify: es una herramienta que permite gestionar dependencias en forma de módulos del lado del cliente (en el navegador). (Browserify, 2020)

Cordova: es un framework de licencia libre que cuenta con muchas Apis de diversos dispositivos móviles para desarrollar aplicaciones nativas dentro de un Smartphone. ("¿Qué es Apache Cordova?", 2020)

Express JS: es el framework web más popular de Node, y es la librería subyacente para un gran número de otros frameworks web de Node populares (MDN Web Docs, 2020)

Flux: es una arquitectura para manejar el flujo de datos en el frontend de las aplicaciones. (Platzi, 2016)

Framework: es el esquema o estructura que se establece y que se aprovecha para desarrollar y organizar un software determinado. ("¿Qué es un Framework y para qué sirve? - Neo Wiki | NeoAttack", 2020)

Frontend: es la parte de una web que conecta e interactúa con los usuarios que la visitan. (Neo Wiki, 2020)

Gulp: es una herramienta, en forma de script en NodeJS, que ayuda a automatizar tareas comunes en el desarrollo de una aplicación (Retrungs, 2016)

Grunt: es una aplicación para gestionar tareas que permite automatizarlas. (Grunt, 2020)

Heroku: interfaz que permite hacer implementación, correr y manejar aplicaciones desarrolladas en distintos lenguajes de programación. (Heroku, 2020)

Javascript: es un lenguaje de programación que permite implementar características complejas en una página web. (MDN Web Docs, 2020)

JQuery: es una librería de JavaScript de código abierto que permite agregar interactividad y efectos visuales en un sitio web. ("Qué es JQuery y cómo implementarlo", 2020)

Node JS: es un entorno de tiempo de ejecución. (Apasionados del Marketing, 2015)

Redux: es un patrón de arquitectura de datos que permite manejar el estado de la aplicación de una manera predecible. Está pensado para reducir el número de relaciones entre componentes de la aplicación y mantener un flujo de datos sencillo. (Desarrolloweb6, 2018)

SASS: es un lenguaje de hoja de estilo que se compila en CSS. Ayuda a mantener bien organizadas las hojas de estilo grandes y facilita compartir diseños dentro y entre proyectos. (SASS Lang, 2020)

Tomcat: es un servidor web HTTPS tradicional. ("¿Qué es Apache Tomcat?", 2020)

I. INTRODUCCIÓN

A lo largo de la historia el sistema educativo no ha tenido ningún problema al adaptarse frente a los diversos obstáculos que se le han presentado, este año el sistema educativo se ve afectado por un problema sin precedentes en una época que da todas las herramientas para poder sobrellevar distintas dificultades. El distanciamiento social y la cuarentena son una realidad que afecta no solo la economía de un país sino también uno de sus pilares como lo es el sistema educativo, en una situación como esta, los diversos entes educativos apenas están implementando soluciones y cabe resaltar que algunos de ellos ni siquiera contemplan una solución estable al problema.

Honduras es un país en el que la digitalización apenas ha alcanzado a un pequeño porcentaje de los entes institucionales por lo que designar una solución en donde se puedan resumir las actividades que se realizan en un aula de clase es casi imposible. Escuelas públicas, privadas e incluso universidades carecen de ciertas aptitudes y tecnologías para poder enfrentar el problema.

Los humanos buscamos resolver problemas hasta que estos nos alcancen, en un país donde el método convencional de asistir e impartir clases es el único método, se hace difícil adaptarse a una solución completamente digital, "Compensa estar preparado contra el peligro." (Esopo, s.f.). Es por eso que desarrollar una plataforma centralizada de fácil uso para el usuario es algo primordial, una página web en la que no solo los estudiantes puedan interactuar con sus asignaciones, sino una relación bilateral más cercana entre docente y estudiante. Así mismo, espacios en donde los profesores no solo puedan calificar y asignar tareas sino discutir temas importantes entre el cuerpo administrativo el cual es el pilar de la institución. Adicionalmente se espera que la calidad de la institución no se ponga en riesgo por lo que la importancia de una sección que demuestre los resultados del uso de una solución tecnológica es no solo innovadora sino atractivo para nuevos estudiantes.

Una solución tecnológica no sería útil solamente para obstáculos como la cuarentena sino para abrir una puerta a una educación enfocada en el uso de herramientas digitales, en donde se pueda facilitar la comunicación sin dejar de lado la interacción a la que se está acostumbrado.

Lo que se quiere lograr con el proyecto no es solo buscar una solución rápida y temporal sino también innovar en las herramientas que se utilizan para transmitir conocimiento a los estudiantes y evaluarlo.

El siguiente informe presenta de forma detallada el proceso de realización de un sistema educativo y promocional, así como la información necesaria para su desarrollo. Dentro del planteamiento del problema se realiza una explicación de la problemática a resolver, por qué es necesario resolverla y los objetivos a alcanzar. En el marco teórico se presenta la información sobre herramientas y técnicas que se utilizaron durante el desarrollo de las distintas fases del proyecto. En la metodología se exponen brevemente las actividades que se realizaron y de qué forma se llevaron a cabo. Finalmente se presentan los resultados del proyecto obtenidos al finalizar el desarrollo.

II. PLANTEAMIENTO DEL PROBLEMA

2.1 PRECEDENTES DEL PROBLEMA

“Investigaciones han demostrado que la distribución de los recursos financieros que se dedican a la expansión del sistema escolar y al mejoramiento de la calidad de la educación impartida entre las escuelas y las entidades federativas ha sido bastante errática.” (Carlos Muñoz Izquierdo, 2012).

En el pasado se ha errado grandemente en dónde se invertirán los recursos financieros de un centro educativo, se piensa que buenas inversiones serán centros de investigación, mejoras de infraestructura, libros académicos, etc., pero pocas veces los centros educativos piensan invertir en educación fuera de lo tangible, fuera del aula de clases y pizarrones, cuando en la actualidad se hace difícil recurrir a ellos.

El problema en el desarrollo de la educación contemporánea no solo radica en la inversión que se realiza en las distintas áreas sino también en las oportunidades que se generan para cada tipo de estudiante, lo cual afecta grandemente la manera en la que este se desenvuelve y aprende dentro del centro educativo, al centrarse en la inversión de una herramienta que solventa los problemas de tener que ir a un centro educativo, las miradas pueden concentrarse en otro tipo de inversión como lo es equipo de cómputo para desarrollar capacidades que servirán en un mundo que se está digitalizando cada vez más.

2.2 DEFINICIÓN DEL PROBLEMA

Al no contar con una plataforma centralizada entonces se generan diversos problemas, los maestros no cuentan con una vía de comunicación formal entre ellos y los alumnos, este problema hace que tareas simples como impartir una cátedra o asignar un trabajo sea una tarea que consuma no solo muchos recursos intangibles como el tiempo sino recursos de esfuerzo al pensar simplicidades como formatos de asignaciones, formas de envío de asignaciones y también la calificación y planeación de ciclos escolares.

Adicionalmente, un centro educativo no funciona a través de la comunicación Docente-Estudiante sino también en la comunicación docente-docente, es por ello que el trabajo como docente,

durante una época en la que asistir a centros educativos no es una opción, se duplica al tener diferentes canales para comunicarse con los estudiantes y otros para comunicarse con el personal administrativo.

2.3 JUSTIFICACIÓN

De no contar con un sistema que ayude a comunicar los diferentes tipos de usuarios dentro del centro educativo se hace casi imposible poder impartir clases, llevar a cabo eventos, calificar y asignar tareas y comunicarse entre docentes para la planeación de un ciclo escolar.

2.4 PREGUNTAS DE INVESTIGACIÓN

- ¿Qué alternativa para las aulas de clases utilizan los centros educativos para impartir cátedras cuando es imposible acceder a un centro educativo físico?
- ¿De qué manera impactaría el contar con una plataforma centralizada para poder impartir cátedra?
- ¿Cómo impactaría una plataforma que funcione como medio para llevar a cabo las reuniones entre personal administrativo?

2.5 OBJETIVOS

2.5.1 OBJETIVO GENERAL

Desarrollar una plataforma educativa en la que tanto docentes como estudiantes puedan interactuar entre sí, satisfaciendo todas sus necesidades y previniendo problemas futuros mediante el uso de tecnologías que permitan el uso de material didáctico e incentiven la colaboración.

2.5.2 OBJETIVOS ESPECÍFICOS

- Crear un espacio en el que los estudiantes puedan interactuar con los docentes y así visualizar asignaciones y otros temas de discusión.
- Crear un espacio de fácil acceso para que los docentes puedan interactuar entre sí y realizar la planeación de ciclos escolares.

- Realizar una página web en donde las personas puedan ver los resultados obtenidos del uso de la plataforma.
- Crear un espacio dentro de una página web en donde se pueda ayudar de manera monetaria a la continuidad de los Proyectos El Hogar.

III. MARCO TEÓRICO

Durante el desarrollo de la aplicación se contemplaron ciertas herramientas de desarrollo, para decidir dichas herramientas se llevó a cabo una investigación meticulosa de las necesidades del proyecto, algunos de los conceptos más relevantes de dicha investigación se encuentran a continuación.

3.1 LA NUBE

La nube es un método de gestión de recursos de TI que sustituye un ambiente físico con un centro de datos usando una infraestructura virtual, la nube cubre varias de las necesidades de los usuarios, dándoles acceso a recursos virtuales que podría tener una computadora física, red y almacenamiento, todo ello está disponible a través de un proveedor remoto.

Aunque la nube puede sonar como la solución más atractiva, puede ser complejo e incluso costoso migrar una organización entera de un sistema físico a una entidad virtual, de igual manera existen diversos beneficios por los cuales se debería considerar migrar a una entidad virtual:

- La facilidad de uso y mantenimiento de un sistema en la nube se vuelve más manejable, se pueden implementar recursos y actualizaciones de manera estándar, en lugar de actualizar equipo; las actualizaciones se harán de manera automática en la entidad virtual.
- Se reducen costos de infraestructura ya que al no tener computadoras físicas se elimina la necesidad de un cuarto refrigerado para almacenar el centro de procesamiento de datos.
- Resulta más fácil desarrollar un negocio si se mejora la tecnología implementada en este, al migrar de un sistema físico a uno virtual se acelera la digitalización del negocio y se facilita la actualización a tecnologías vanguardistas.

Existen diversas formas de computación en la nube:

- Infraestructura como servicio (IaaS): es la forma más básica, brinda a los usuarios el acceso a espacios de almacenamiento, acceso a servidores y redes.

- Plataforma como servicio (PaaS): ofrece un entorno de desarrollo completo, le facilita a los desarrolladores el trabajo de tener que ocuparse directamente de la capa de infraestructura al implementar y actualizar aplicaciones dentro del sistema.
- Software como servicio (SaaS): estas aplicaciones están diseñadas para usuarios normales, y mantienen toda la temática de desarrollo en un escenario al que el usuario no tiene conocimiento. Ejemplos de software SaaS pueden ser programas de procesamiento de texto, hojas de cálculo e incluso CRM. (Hewlett Packard Enterprise, 2020)

En la Ilustración 1 se puede ver una comparativa de las tres formas de computación en la nube, comparando su nivel de abstracción, control, flexibilidad y costo.

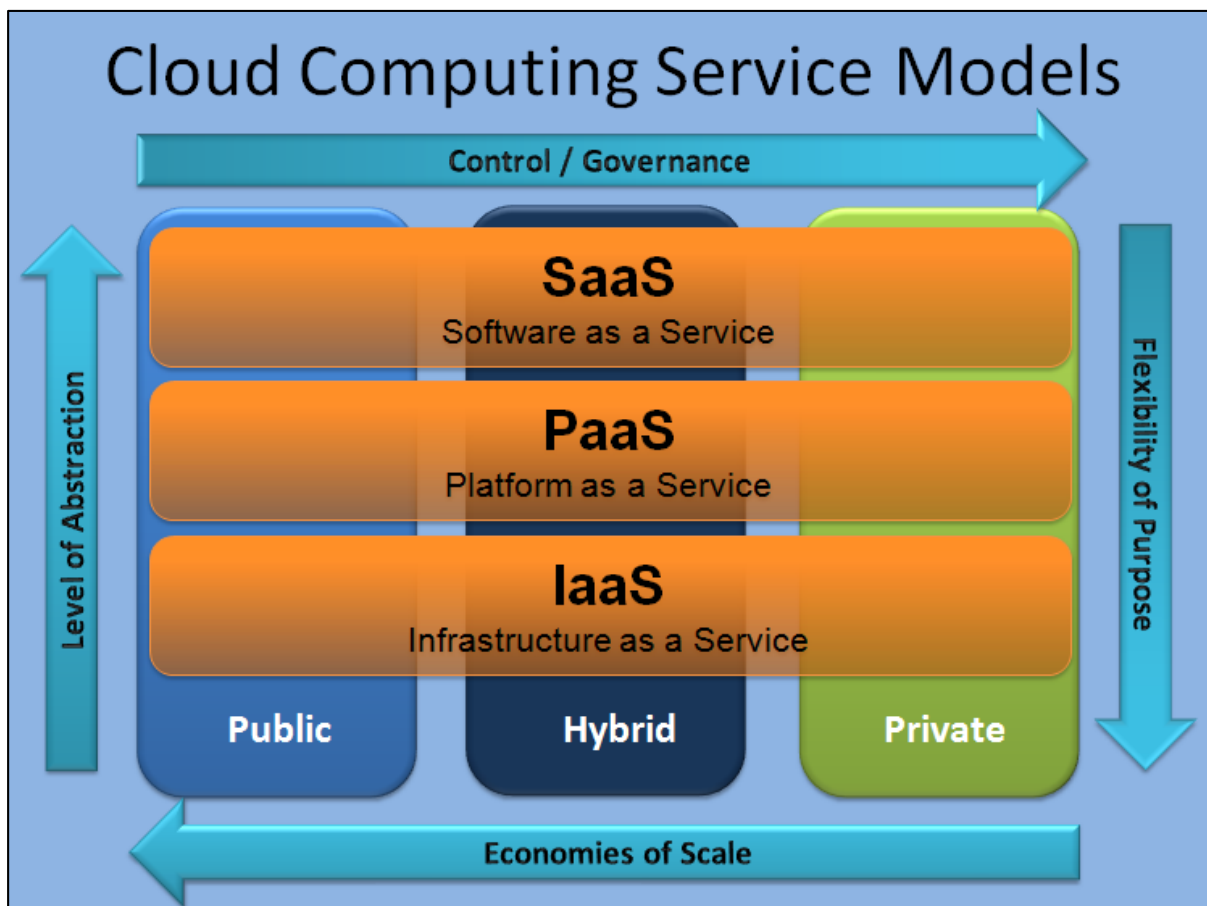


Ilustración 1 - Modelos de Servicios de Computación en la Nube

Fuente: (gangwar, 2020)

3.2 HEROKU

Heroku está catalogado como uno de los PaaS más utilizados en la actualidad para empresas grandes por su fuerte y rápida utilización al momento de desplegar una aplicación web, Heroku permite desplegar múltiples desarrollos web de manera fácil como se puede observar en la Ilustración 2 permite la implementación de una amplia lista de lenguajes de programación o frameworks como Javascript o React JS.

Heroku permite elegir la herramienta de Backend y Frontend que se utiliza en dicho desarrollo, dejando de lado la preocupación de mantenimiento por lo que al programador solo le concierne el desarrollo y la prueba del código. (Platzi, 2017)



Ilustración 2 - Desarrollo en Heroku

Fuente: (Platzi, 2017)

3.3 REACT JS

React es una librería de Javascript que se enfoca en el desarrollo rápido de interfaces de usuario. Su vasta versatilidad permite no solo crear páginas web sino también aplicaciones móviles, es por

eso que React cuenta con un completo ecosistema de módulos, herramientas y componentes que ayudan al desarrollador a cubrir objetivos avanzados con el mínimo de esfuerzo.

React representa la base sólida sobre la cual se puede construir un sin número de aplicaciones con Javascript. Es una librería desarrollada inicialmente por Facebook que desde su liberación al público cuenta con una comunidad de desarrolladores que sigue en incremento día a día. Se creó con el objetivo de suplir las necesidades del desarrollo ágil de componentes de interfaces de usuario, React sirve para desarrollar aplicaciones web de una manera más ordenada y con menos código sin usar un sin número de instrucciones de Javascript o librerías de JQuery, React permite asociar la interfaz de usuario con los datos de manera que si los datos cambian también lo hará la parte que el usuario está visualizando.

La característica más importante de React es el hecho de que es isomorfo, un concepto bastante importante en el desarrollo de aplicaciones actualmente, básicamente se trata de, con un mismo código cargar HTML tanto en el lado del servidor como en el lado del cliente, reduciendo así la carga de trabajo necesaria para llevar una aplicación web amigable a los buscadores. Al realizar una página web enteramente con Javascript esta deberá de recibir los datos del servidor mediante una API en un formato JSON lo cual luego produce HTML que debe presentar al navegador por lo que se vuelve una tarea difícil para el buscador.

Existe todo un ecosistema de herramientas, aplicaciones y librerías que al final equiparan React a un framework. Existen todo tipo de soluciones que se usan en múltiples proyectos como Redux o Flux, que aporta resolución a partes que React no controla. Estos dos mecanismos hacen que la información prevalezca dentro de la aplicación de manera optimizada, elegante y poniendo énfasis en la claridad e integridad de los datos. (Desarrolloweb6, 2019)

3.4 NODE JS

Es un entorno en tiempo de ejecución, un framework de Javascript para la creación de aplicaciones de red en tiempo real, ligero y escalable. Node JS es una forma de ejecutar Javascript desde un servidor, fue formulado para generar un sistema escalable que tuviese la consistencia suficiente para poder generar un elevado número de conexiones de forma simultánea con el servidor como se puede observar en la Ilustración 3.

Generalmente cuando se tiene varias páginas web que están realizando peticiones a una sola base de datos, esta responde a cada una de las páginas web ya que estas peticiones se trabajan como hilos, por lo que se ejecutan de manera independiente y simultánea, al generarse más peticiones, la carga del servidor es más grande y el tiempo de respuesta aumenta por lo que las respuestas al cliente pueden tardar más tiempo del esperado, Node JS ha sido concebido para aplicar una solución al problema y optimizarlo.

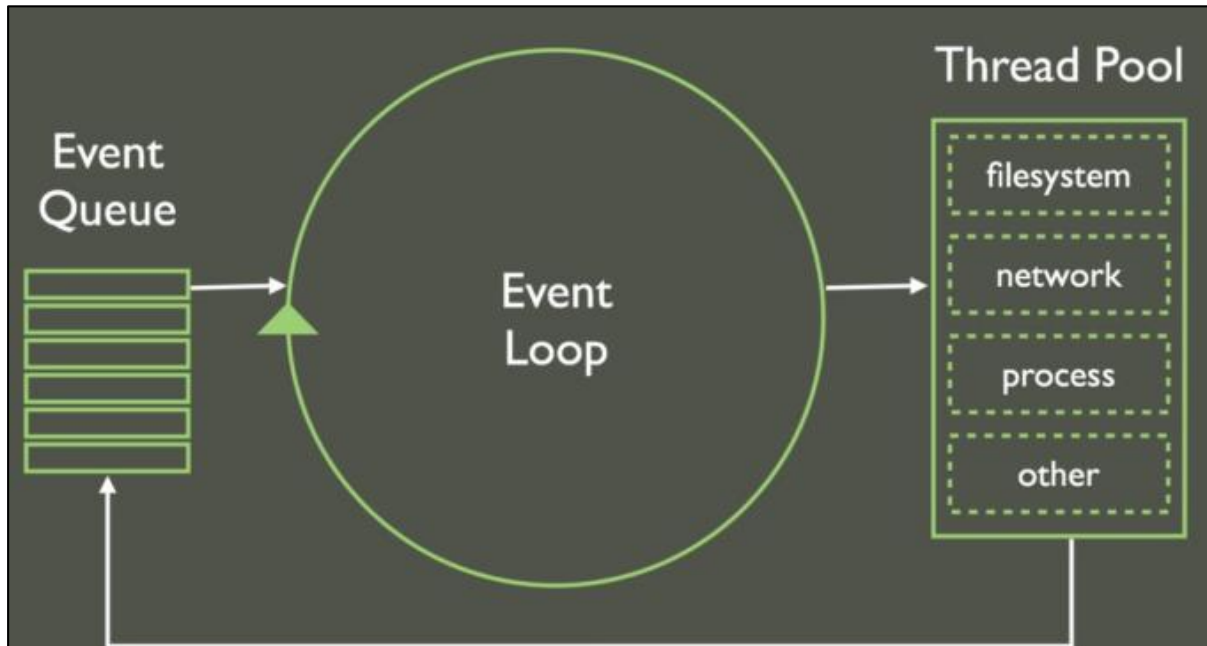


Ilustración 3 - Manejo de Peticiones de Node

Fuente: ("Node.js ▷ ¿Qué es y para qué sirve NodeJS? ¿Qué no es NodeJS?", 2020)

La propuesta a este problema de parte de Node JS es tratar las conexiones de manera unificada a partir de un único hilo que se complementa con una cola de eventos que luego entran en una categorización más detallada en un pool de hilos. Se eligió Javascript del lado del servidor para Node ya que posee un excelente modelo de eventos para programación asíncrona que es exactamente lo que buscaba Node, así mismo Javascript es un lenguaje omnipresente, conocido por millones de programadores por lo que esto reduce la curva de aprendizaje haciéndolo así un framework popular entre los desarrolladores por el fácil uso e implementación en sus aplicaciones.

En la Ilustración 4 se puede ver un análisis de las llamadas por segundo de Node JS, comparadas contra otros frameworks.

Node proporciona una manera fácil de construir programas de red escalables. Existen diversas razones por las cuales un desarrollador debería elegir Node para el desarrollo de aplicaciones.

- Tiene una sintaxis idéntica a Javascript por lo que es fácil de aprender para los desarrolladores de Javascript.
- Se utiliza la misma sintaxis (Javascript) para desarrollar el cliente y el servidor por lo que se puede hacer reutilización de código.
- Su implementación es fácil y ligera con el cliente a diferencia de Apache o Tomcat.

Uno de los puntos fuertes de Node JS es su manejador de paquetes o más popularmente llamado Node Package Manager (NPM) el cual es un gestor que da acceso a un conjunto extenso de librerías gratuitas generadas por la comunidad de usuarios conformada por los desarrolladores que utilizan Node JS, algunos de estos paquetes (los más populares) son React, Gulp, Browserify, Grunt-cli, Bower, Grunt, Express, Cordova o Forever. ("Qué es NPM y para qué sirve", 2020)

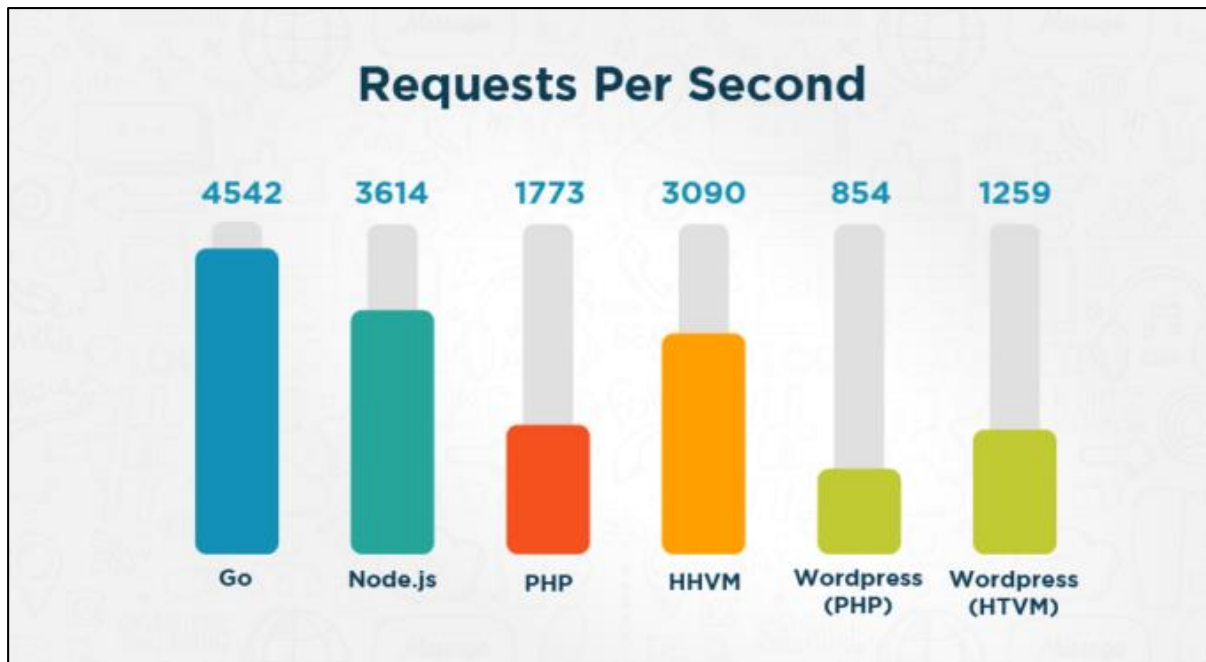


Ilustración 4 - Comparativa de Peticiones por Segundo

Fuente: ("Node.js ▷ ¿Qué es y para qué sirve NodeJS? ? ¿Qué no es NodeJS?", 2020)

3.5 METODOLOGÍAS ÁGILES

Las metodologías ágiles permiten adaptar la forma en que se trabaja a las condiciones del proyecto, por lo que son altamente adaptables. Este tipo de metodologías consiguen que se tenga flexibilidad en el desarrollo a las circunstancias específicas del entorno.

En la actualidad la mayoría de las empresas apuestan por este tipo de forma de trabajo y consiguen gestionar sus proyectos de manera flexible, autónoma y eficaz, lo que ayuda a reducir costos de implementación y también incrementa la productividad del equipo de trabajo.

Existen diversas ventajas por las que una empresa debería considerar la utilización de una metodología ágil, algunas de ellas son:

- Mejoran la satisfacción del cliente ya que lo involucran y comprometen a lo largo de todo el proyecto. En cada etapa o ciclo de trabajo se informa al cliente sobre los logros y progresos alcanzados hasta el momento. Se busca involucrar al cliente a fin de que este pueda aportar su visión, experiencia y conocimiento para que el equipo de trabajo pueda optimizar las características del producto final.
- Mejora la motivación e implicación del equipo de desarrollo, este tipo de metodologías permiten a todos los miembros del equipo conocer el estado del proyecto en cualquier momento.
- Ahorra tiempo y costo ya que trabaja de una manera eficiente y rápida, es por ello que se cumple de manera estricta el tiempo pactado en cada tarea asignada, llevando así al cumplimiento del presupuesto.
- Se trabaja con mayor velocidad ya que se implementa un sistema de trabajo con entregas parciales del producto, de este modo se puede entregar un producto mucho más funcional en un corto periodo de tiempo.
- Elimina cualquier característica innecesaria del producto gracias a las entregas parciales las cuales tienen como objetivo entregar las funcionalidades que aportan valor teniendo en cuenta la implicación del cliente en cada una de ellas.

- Mejoran la calidad del producto al tener una continua interacción entre el cliente y los desarrolladores. Con este enfoque es posible abrazar la excelencia tecnológica obteniendo así un producto tecnológicamente superior.
- Es posible alertar de forma rápida tanto errores o problemas que puedan sucederse a lo largo del proyecto. Es posible dar respuesta a todos aquellos problemas que puedan darse desde el inicio, con lo que mejoramos en costes y entrega.
- Permiten rentabilizar inversiones, gracias a la realización de entregas tempranas, el cliente tendrá rápido acceso a aquellas funcionalidades que aportan valor, acelerando el retorno de la inversión.

Existen diferentes opciones de metodologías ágiles, pero las más utilizadas son:

- Programación Extrema (XP) la cual es muy útil sobre todo para startups o empresas que están en proceso de consolidación.
- Scrum, se basa en un desarrollo incremental que le permite abordar proyectos complejos.
- Kanban conocida también como 'Tarjeta Visual', es muy útil para los responsables de proyectos ya que facilita la planificación de tareas.

Todas las metodologías antes descritas se guían a través de un patrón establecido por el manifiesto ágil realizado por varios autores que establecieron los 12 principios del software ágil. (Villán, 2019)

Actualmente existe una metodología de trabajo ágil que sobresale sobre las demás, la cual es Scrum, propone una serie de roles, artefactos y actividades que hay que asumir en el proyecto. (Monte Galiano, 2016)

Scrum es un proceso en el que se aplican buenas prácticas como las de trabajar de manera colaborativa para así obtener mejores resultados de un proyecto. Estas prácticas tienen como resultado el trabajo altamente productivo.

En Scrum se realizan entregas parciales, por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados rápidos y donde los requisitos del proyecto cambien frecuentemente o estén poco definidos. Así mismo, Scrum es ideal para manejar innovación, competitividad, flexibilidad y productividad de los proyectos.

Es utilizado para resolver situaciones en las que el equipo de trabajo no está entregando al cliente lo que en realidad necesita, cuando las entregas se alargan demasiado, los costos aumentan. (Monte Galiano, 2016)

Las fases que definen y en las que se divide un proceso de Scrum son las siguientes:

- El quién y el qué: identifica los roles de cada uno de los miembros del equipo y define su responsabilidad en el proyecto.
- El dónde y cuándo: representan el ciclo de trabajo o sprint.
- El por qué y el cómo: representan las herramientas que se utilizarán por los miembros de Scrum.

El equipo de Scrum consiste en tres diferentes roles:

- El product Owner / Dueño del product es la voz del cliente y también es el responsable de desarrollar, mantener y priorizar las tareas en el backlog de actividades.
- El Scrum master es el responsable de asegurarse que el equipo de trabajo este siguiendo las bases de Scrum. Además, se encarga de remover cualquier obstáculo que pueda encontrar el equipo de desarrollo
- Los Development Team Members / Miembros del Equipo de desarrollo son los encargados de desarrollar y probar el código.

El sprint es la unidad básica de trabajo para un equipo Scrum. Esta es la característica principal que diferencia Scrum de otros modelos de trabajo ágil. Es una iteración de trabajo llevada a cabo por los miembros del equipo, un equipo de trabajo puede completar varios sprint durante el desarrollo del proyecto.

Un Sprint inicia con un equipo que se compromete a realizar un trabajo y finaliza con la demostración de un entregable. El tiempo mínimo para realizar un sprint es de una semana y el máximo es de 4 semanas. Dentro del desarrollo de un sprint se llevan a cabo ciertos eventos, estos reciben el nombre de Scrum events o eventos Scrum. Estos son los eventos Scrum y su implicación en el ciclo de trabajo:

- Planeación del Sprint / Sprint Planning: todos los involucrados en el equipo se reúnen para planificar lo que será el sprint. Durante este evento se decide qué requerimientos o tareas

se le asignará a cada uno de los miembros del equipo. Cada uno de los integrantes deberá asignar el tiempo que crea prudente para llevar a cabo sus requerimientos, de esta manera se define el tiempo de duración del sprint.

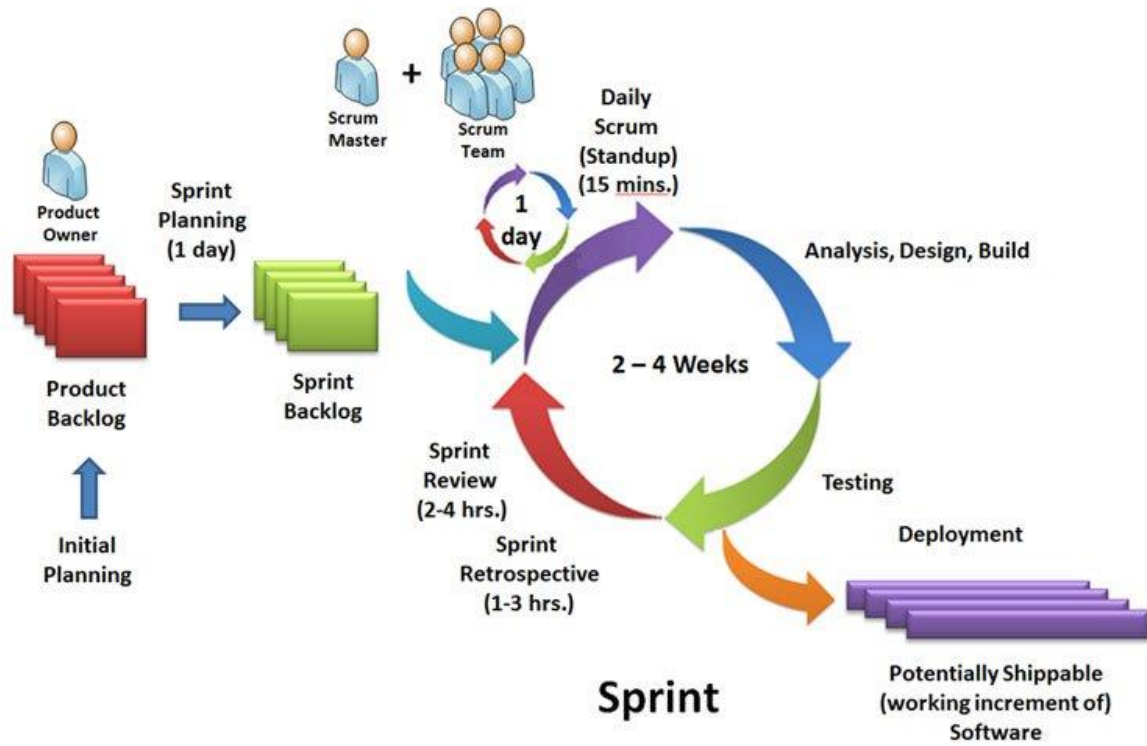
- Reunión de Equipo de Scrum / Scrum team meeting: se les dedica un máximo de 15 minutos diarios y deben ser siempre en el mismo horario y lugar. Estas reuniones sirven para que todos los miembros del equipo de trabajo se apoyen entre sí. Si alguno de ellos se ve de frente con algún inconveniente que obligue a extender el encuentro, este debe tratarse más a fondo en una reunión enfocada en buscar la mejor solución para ello. En esta reunión cada miembro del equipo responde 3 simples preguntas:
 - ¿Qué hiciste ayer?
 - ¿Qué tienes planeado hacer hoy?
 - ¿Qué obstáculos encontraste en el camino?
- Refinamiento del Backlog / Backlog Refinement: el product owner revisa cada uno de los elementos dentro del product backlog con el fin de esclarecer cualquier duda que pueda surgir por parte de los miembros del equipo de desarrollo. Así mismo sirve para volver a estimar el tiempo y esfuerzo dedicado a cada uno de los requerimientos.
- Revisión del Sprint / Sprint Review: los miembros del equipo scrum se reúnen para mostrar el trabajo de desarrollo de software que se ha completado. Se realiza una demostración de todos los requerimientos que se lograron completar dentro del sprint. En este punto no es necesario que todos los miembros del equipo hablen, pueden simplemente estar presentes a la demostración del Scrum master y product owner.
- Retrospectiva del Sprint / Retrospective: en este evento el product owner se reúne con todo el equipo de trabajo y su Scrum master para hablar sobre lo ocurrido durante el sprint. Los puntos principales a tratar en esta reunión son:
 - Qué se hizo mal durante el Sprint para poder mejorar el próximo.
 - Qué se hizo bien para seguir en la misma senda del éxito.
 - Qué inconvenientes se encontraron y no permitieron poder avanzar como se tenía planificado.

Para poder definir las respuestas a estas preguntas se vale de ciertas herramientas que Scrum provee. Estas son las siguientes:

- Backlog de producto / Product Backlog, se refiere a todo elemento que sea parte del proyecto.
- Historias de usuario / User Stories, proporcionan información sobre cómo debe ser el comportamiento del requerimiento que se está trabajando.
- Backlog del Sprint / Sprint Backlog, es el conjunto de elementos tomados del product backlog que fueron priorizados en la reunión de sprint planning.
- El panel de tareas / The Taskboard, muestra las tareas que tienen asignadas los miembros del equipo.
- Definición de "Listo" / Definition of Done, todo equipo eficaz y ágil tiene ciertos acuerdos que deben cumplirse antes de dar por finalizado un proyecto. Estos son:
 - Todas las tareas están completas.
 - Revisión de Código / Code Reviewed.
 - Pruebas realizadas a cada elemento desarrollado.
 - Revisión por parte de los clientes (que cumpla sus necesidades).
 - La revisión de las condiciones de Aceptación por parte del Product Owner.

Estas herramientas son útiles, no sólo durante un Sprint, sino también a lo largo del proyecto, pues ayudan al equipo a entender el porqué de cada actividad. Además, son visibles para el equipo y para los externos. (Cordero, 2015)

En la Ilustración 5 se muestra la estructura y proceso de un sprint dentro de Scrum.



Copyright © 2011, William B. Heys

Ilustración 5 - Estructura de Sprint

Fuente: (Cordero, 2015)

IV. METODOLOGÍA

4.1 ENFOQUE

En este proyecto se utiliza un enfoque cualitativo ya que “asume una realidad subjetiva, dinámica y compuesta por multiplicidad de contextos.” (Mata Solís, 2019). Al finalizar el proyecto se busca realizar un análisis mediante encuestas para así poder observar el nivel de aceptación de la plataforma de parte de los alumnos y cuerpo administrativo.

Se utiliza un enfoque cualitativo ya que se desea analizar la manera en la que la plataforma soluciona el problema y cómo esta se adapta a las necesidades una vez se haya resuelto la incógnita.

4.2 VARIABLES DE INVESTIGACIÓN

- Nivel de satisfacción de estudiantes y cuerpo administrativo con la plataforma educativa.
- Nivel de respuesta ante los problemas cotidianos de parte de la plataforma educativa.
- Nivel de robustez del sistema ante múltiples transacciones.

4.3 TÉCNICAS E INSTRUMENTOS APLICADOS

Después de realizar un análisis minucioso sobre las posibles herramientas que se podrían utilizar para el desarrollo de una plataforma web educativa se concluyó que las mejores técnicas e instrumentos serían los siguientes:

4.3.1 VISUAL STUDIO CODE

Es un editor de código multiplataforma con el cual se puede desarrollar en una amplia gama de lenguajes de programación, ofrece extensiones no solo de desarrollo sino también de visualización de errores, control de versiones, etc.

Se eligió Visual Studio Code para el desarrollo del proyecto por su vasta librería, su popularidad en cuanto a uso y robustez al momento de gestionar el código desarrollado.

4.3.2 REACT JS

Es el framework de Javascript más popular del momento en cuanto a desarrollo de aplicaciones frontend, caracterizado por su fácil uso, hace que aplicaciones complejas y extensas sean de fácil desarrollo siempre y cuando se maneje un concepto claro y se realicen buenas prácticas de programación.

Se eligió React JS para desarrollo de frontend por ser un framework vanguardista y además el de más popularidad en la actualidad.

4.3.3 EXPRESS JS

Es un framework para desarrollo de aplicaciones backend, se basa en la creación de componentes para la elaboración de los distintos puntos de acceso que tendrá la base de datos.

En este proyecto se utiliza Express JS para elaborar el servidor de backend en donde estarán almacenadas las APIs a las que los usuarios harán solicitudes para poder acceder a la información.

4.3.4 MONGODB ATLAS

La base de datos utilizada en el desarrollo de la aplicación es MongoDB, la cual es una base de datos no relacional basada en documentos, se eligió una base de datos no relacional para así mejorar el entendimiento de las diversas entidades y tener un mejor control sobre ellas.

Al tener una gran cantidad de objetos dentro de la base de datos, se pueden visualizar de forma rápida los documentos con una base de datos no relacional, en comparación con una base de datos relacional.

MongoDB Atlas permite gestionar base de datos en la nube, dando fácil acceso a la manipulación de documentos y permisos de lectura y escritura para la base de datos.

4.3.5 CASOS DE USO

A continuación, se muestran los diversos casos de uso para los tipos de usuarios que interactúan dentro de la plataforma, detallando de esta forma los procesos que estos llevarán a cabo.

En la Ilustración 6 se muestra el caso de uso para el usuario administrador el cual tiene la capacidad de agregar campañas a la página principal de la institución y gestionar usuarios dentro de la plataforma.

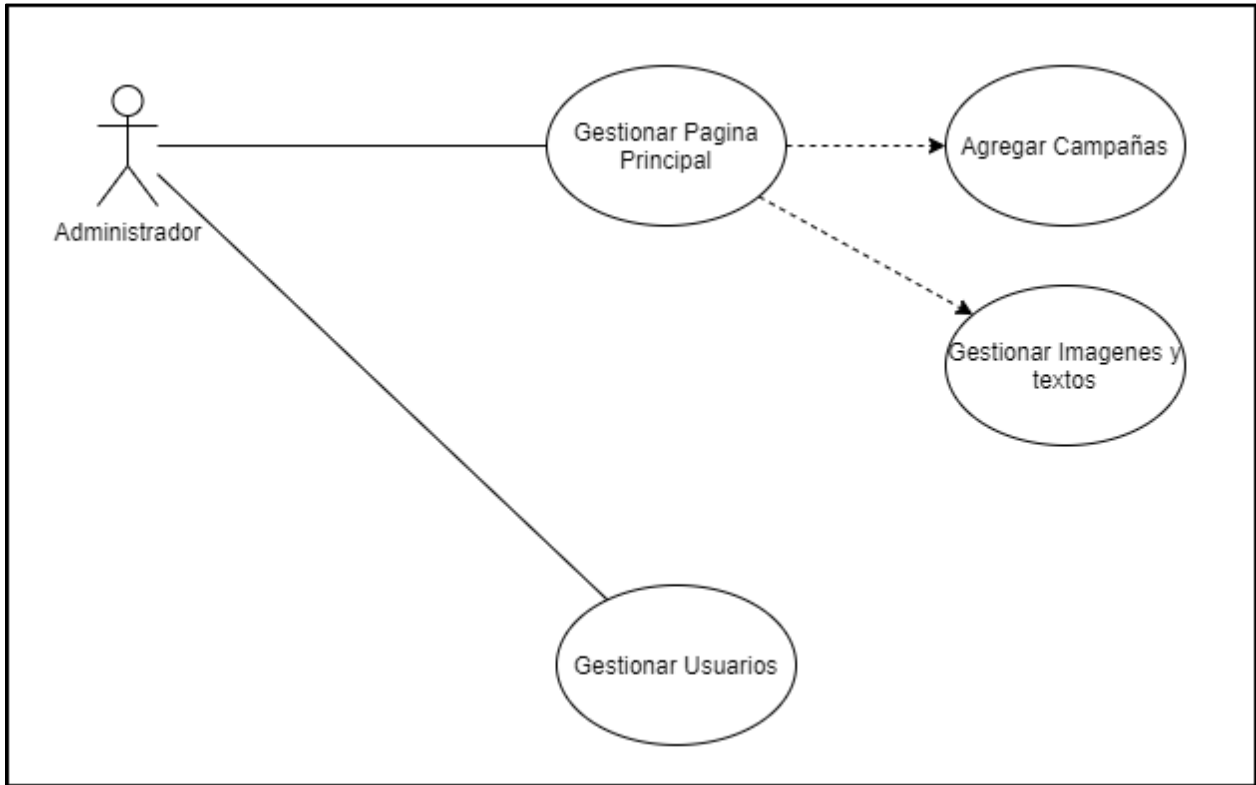


Ilustración 6 - Caso de uso de Administradores

Fuente: (Elaboración Propia)

En la Ilustración 7 se muestra el caso de uso de los usuarios que asumen el rol de maestros los cuales tienen la mayor parte de tareas dentro de los cursos, en los cuales pueden gestionar estudiantes, medios de comunicación y diversos tipos de actividades las cuales se convertirán en asignaciones para los estudiantes.

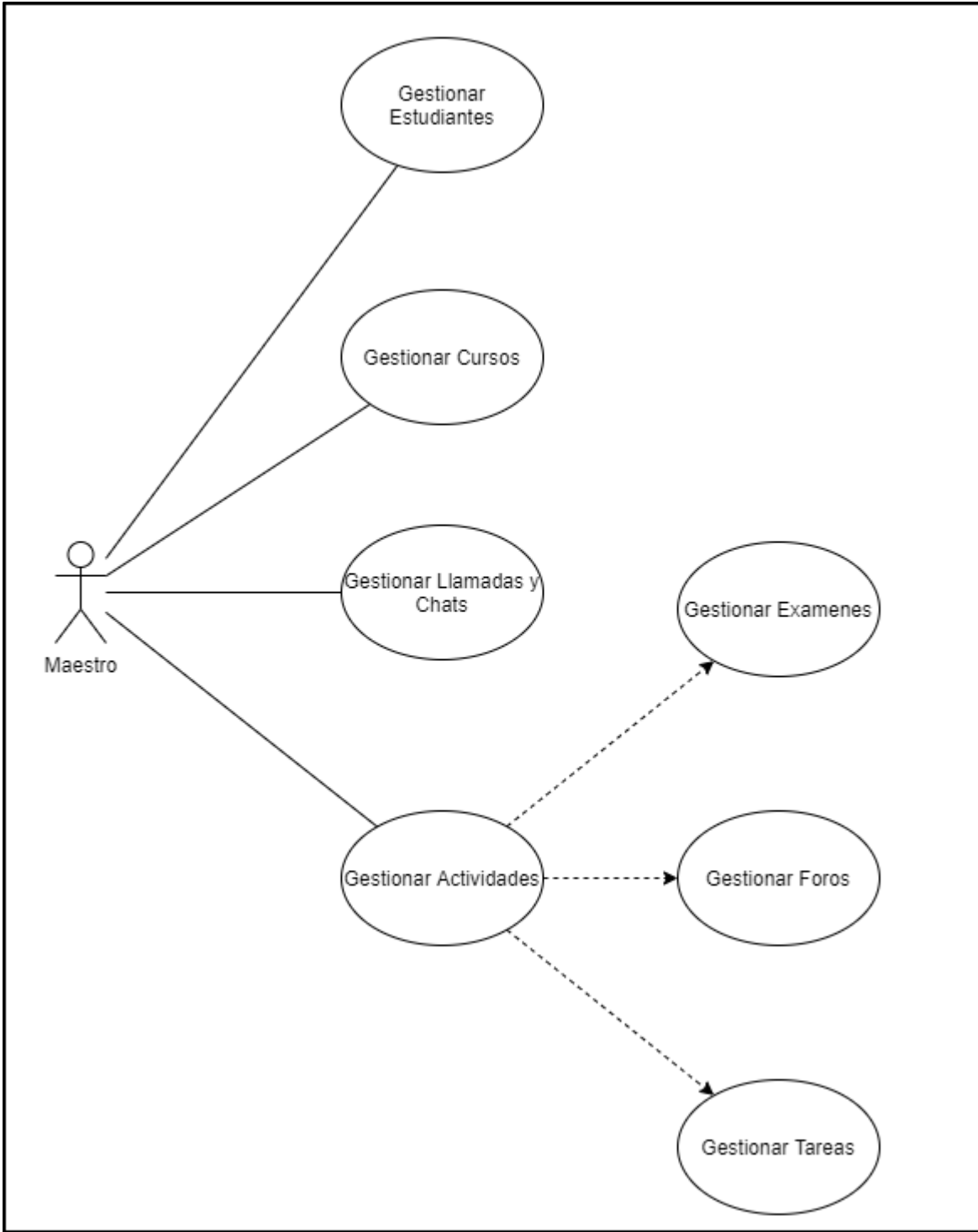


Ilustración 7 - Caso de Uso de Maestros

Fuente: (Elaboración Propia)

La Ilustración 8 se muestra el caso de uso para los usuarios con un rol de estudiante, los cuales tienen la capacidad de interactuar con sus maestros y también con los trabajos y asignaciones dentro de los cursos a los que pertenecen.

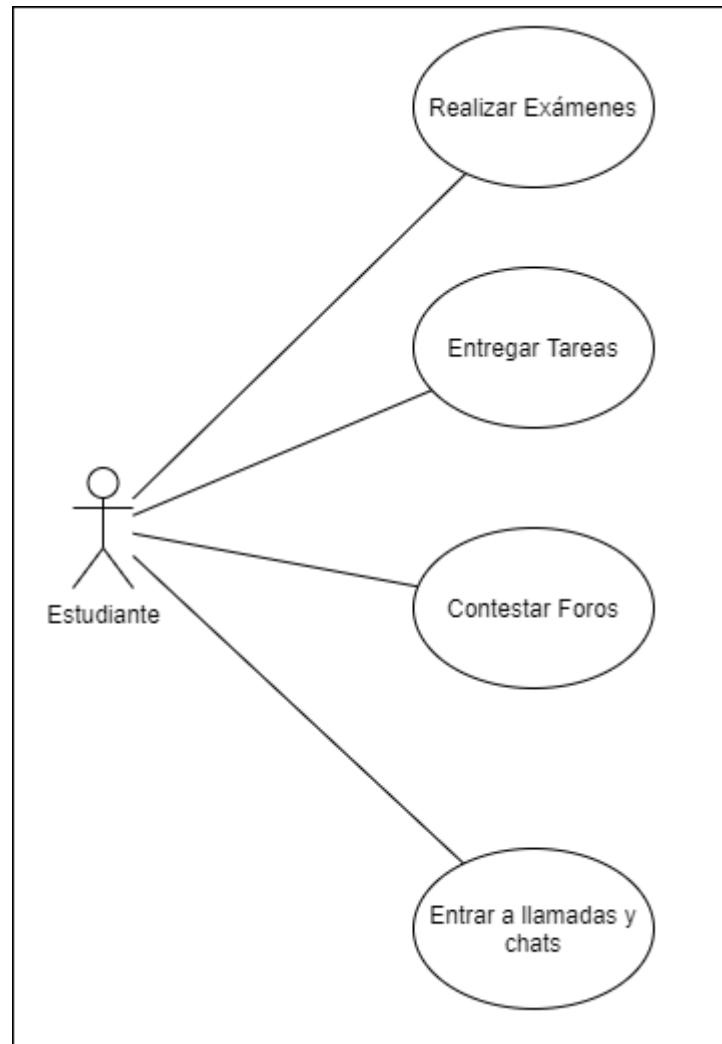


Ilustración 8 - Caso de Uso de Estudiantes

Fuente: (Elaboración Propia)

4.4 METODOLOGÍA DE ESTUDIO

4.4.1 MODELO ÁGIL

La metodología de desarrollo implementada en este proyecto es el de un modelo ágil como lo es Scrum, el cual es conveniente al momento de involucrar al cliente en ciclos de trabajos de una

longitud variable. Se utilizó este modelo de trabajo al considerar el alto alcance del proyecto y la importancia de generar entregables con bastante frecuencia. Así mismo se evaluó la implicación que debería tener el cliente dentro del proceso de desarrollo por lo que un modelo iterativo resultaba más conveniente. La Ilustración 9 muestra de manera condensada el ciclo de trabajo del modelo ágil utilizado dentro del proyecto.

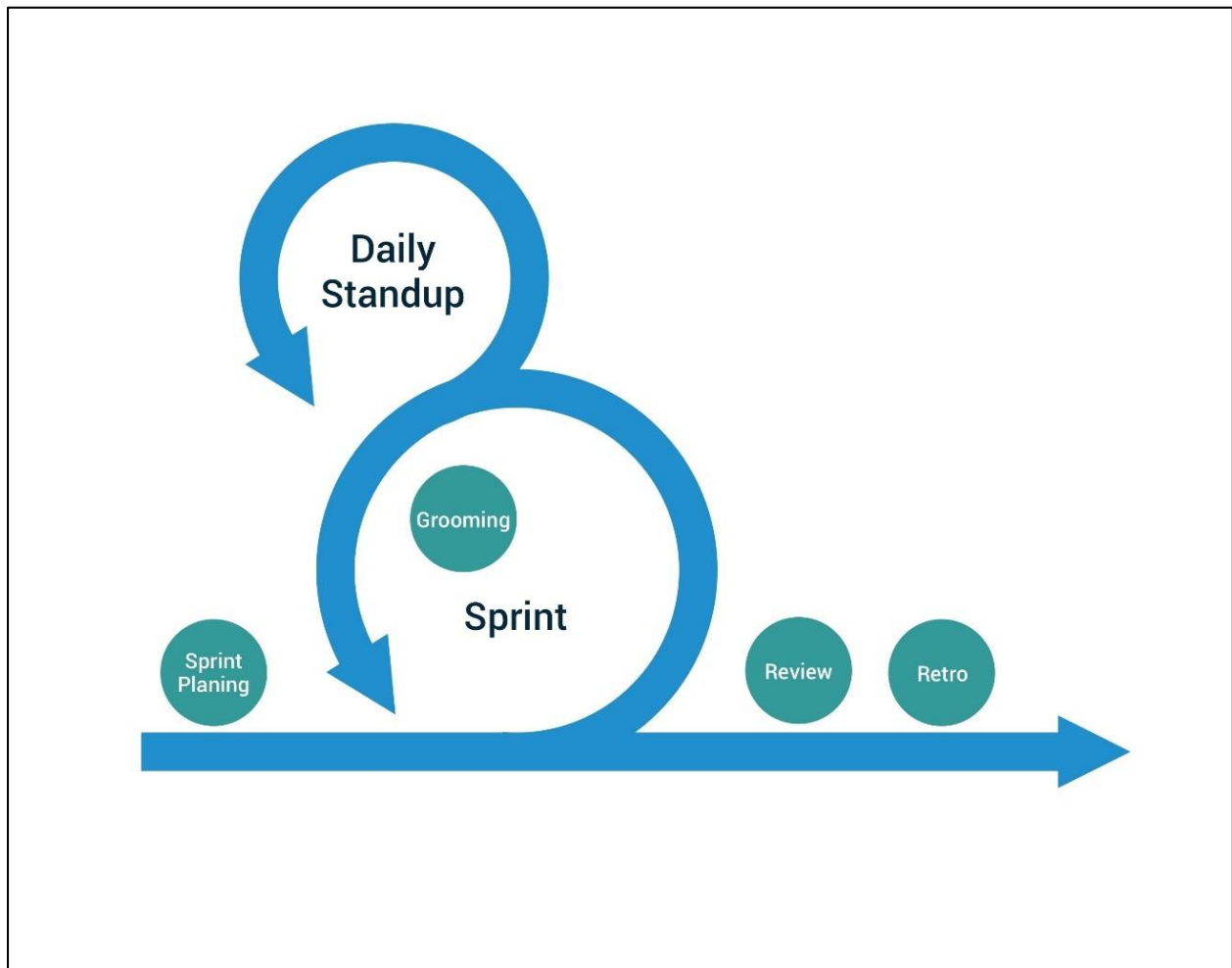


Ilustración 9 - Ciclo de Trabajo de Scrum

Fuente: (Blog & Scrum, 2020)

4.5 CRONOGRAMA DE ACTIVIDADES

La Tabla 1 muestra un cronograma detallado de las actividades y objetivos a completar para la correcta elaboración de la plataforma educativa dividiendo las actividades en tres categorías importantes como lo son el diseño, el desarrollo y la verificación del producto.

Tabla 1 - Cronograma de Trabajo

Actividades	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre
Requerimientos						
Definición de Metas y Procesos						
Definición Metodología de proyecto						
Análisis de tecnologías a utilizar						
Diseño						
Diagramas de caso de uso						
Diagramas Entidad-Relación						
Diseño de base de datos						
Desarrollo e Integraciones						
Creación de end points de bases de datos						
Hosting de base de datos						
Desarrollo de sección de docentes						
Desarrollo de sección de estudiantes						
Desarrollo de sección de administración						
Desarrollo de chatbot						
Integración de chatbot						
Desarrollo de chat y video llamada						
Integración de chat y video llamada						
Desarrollo de sección de donaciones						
Integración de método de donaciones						
Control de calidad de donaciones						
Verificación / Ejecución						
Control de calidad de las diversas secciones						

Fuente: (Elaboración propia)

V. RESULTADOS Y ANÁLISIS

5.1 INTERFAZ DE PROGRAMACIÓN DE APLICACIONES

La Tabla 2 se muestra las solicitudes que el usuario realiza al sistema en el momento de la autenticación y el estado de su sesión dentro de la plataforma. Así mismo se muestra la descripción de cada una de ellas.

Tabla 2 - API de Autenticación

Solicitud	Descripción	Tipo de Solicitud
Ingresar Sistema	Obtiene la información ingresada en el sistema y verifica las credenciales en la base de datos para que el usuario pueda ingresar al sistema	POST
Usuario Autenticado	Obtiene la información del usuario que está actualmente autenticado para así poder darle los permisos para poder realizar las transacciones de acuerdo a su rol.	POST

Fuente: (Elaboración propia)

En la Tabla 3 se muestran las solicitudes necesarias que se realizan para la creación y recuperación de contraseñas.

Tabla 3 - API de Restablecimiento

Solicitud	Descripción	Tipo de Solicitud
Restablecer Contraseña	Envía un correo electrónico al usuario que este solicitando el restablecimiento y procede a cambiar su contraseña	POST

Fuente: (Elaboración propia)

En la Tabla 4 se muestran las solicitudes que el administrador realiza durante los diversos procesos que este realiza, así mismo se presenta la descripción de los mismos.

Tabla 4 - API de Administradores

Solicitud	Descripción	Tipo de Solicitud
Crear Maestros	Obtiene la información mínima requerida dentro del formulario de creación de maestros para así poder crear un nuevo usuario maestro dentro del sistema.	POST
Crear Cursos	Obtiene la información mínima requerida dentro del formulario de creación de cursos para así poder crear un nuevo curso dentro del sistema.	POST
Asignar Cursos	Realiza una comparación entre cursos y maestros existentes dentro del sistema.	PUT
Eliminar Maestros	Muestra una lista de maestros y elije uno para eliminarlo del sistema.	DELETE
Eliminar Cursos	Muestra una lista de cursos y elije uno para eliminarlo del sistema.	DELETE
Crear Campañas	Obtiene la información mínima requerida dentro del formulario de creación de campañas para así poder crear una nueva campaña dentro del sistema.	POST
Editar Campañas	Obtiene la información mínima requerida dentro del formulario de creación de campañas para así poder editar una campaña existente dentro del sistema.	PUT
Eliminar Campañas	Muestra una lista de campañas y elije una para eliminarla del sistema.	DELETE

Fuente: (Elaboración propia)

En la Tabla 5 se muestran las solicitudes que el administrador concreta durante los diversos procesos que este realiza, así mismo se presenta la descripción de los mismos.

Tabla 5 - API Operaciones Básicas

Solicitud	Descripción	Tipo de Solicitud
Crear Asignaciones, Cursos, Entregas, Archivos, Estudiantes y Maestros	Obtiene la información mínima requerida dentro del formulario de creación de los registros para así poder crear un nuevo registro dentro del sistema.	POST
Editar Asignaciones, Cursos, Entregas, Archivos, Estudiantes y Maestros	Obtiene la información mínima requerida dentro del formulario de creación para así poder editar un registro existente dentro del sistema.	PUT
Eliminar Asignaciones, Cursos, Entregas, Archivos, Estudiantes y Maestros	Muestra una lista de registros y elige uno para eliminarlo del sistema.	DELETE
Obtener Asignaciones, Cursos, Entregas, Archivos, Estudiantes y Maestros	Obtiene la información del registro de manera específica o general.	DELETE

Fuente: (Elaboración propia)

5.2 CAPTURAS DE PANTALLA DEL SISTEMA WEB INFORMATIVO

El sistema web está formado por una página inicial y promocional y un portal educativo en donde estudiantes y docentes interactúan.

La Ilustración 10 muestra la sección principal junto con la barra de navegación de la misma.



Ilustración 10 - Sección Principal

Fuente: (Elaboración propia)

La Ilustración 11 muestra la sección “Sobre Nosotros” en la que se describe brevemente la misión y visión de la institución.



Ilustración 11 - Sobre Nosotros

Fuente: (Elaboración propia)

La Ilustración 12 muestra la sección “Nuestros Valores” en la que se muestran de manera breve la lista de valores con los cuales se rige la institución.

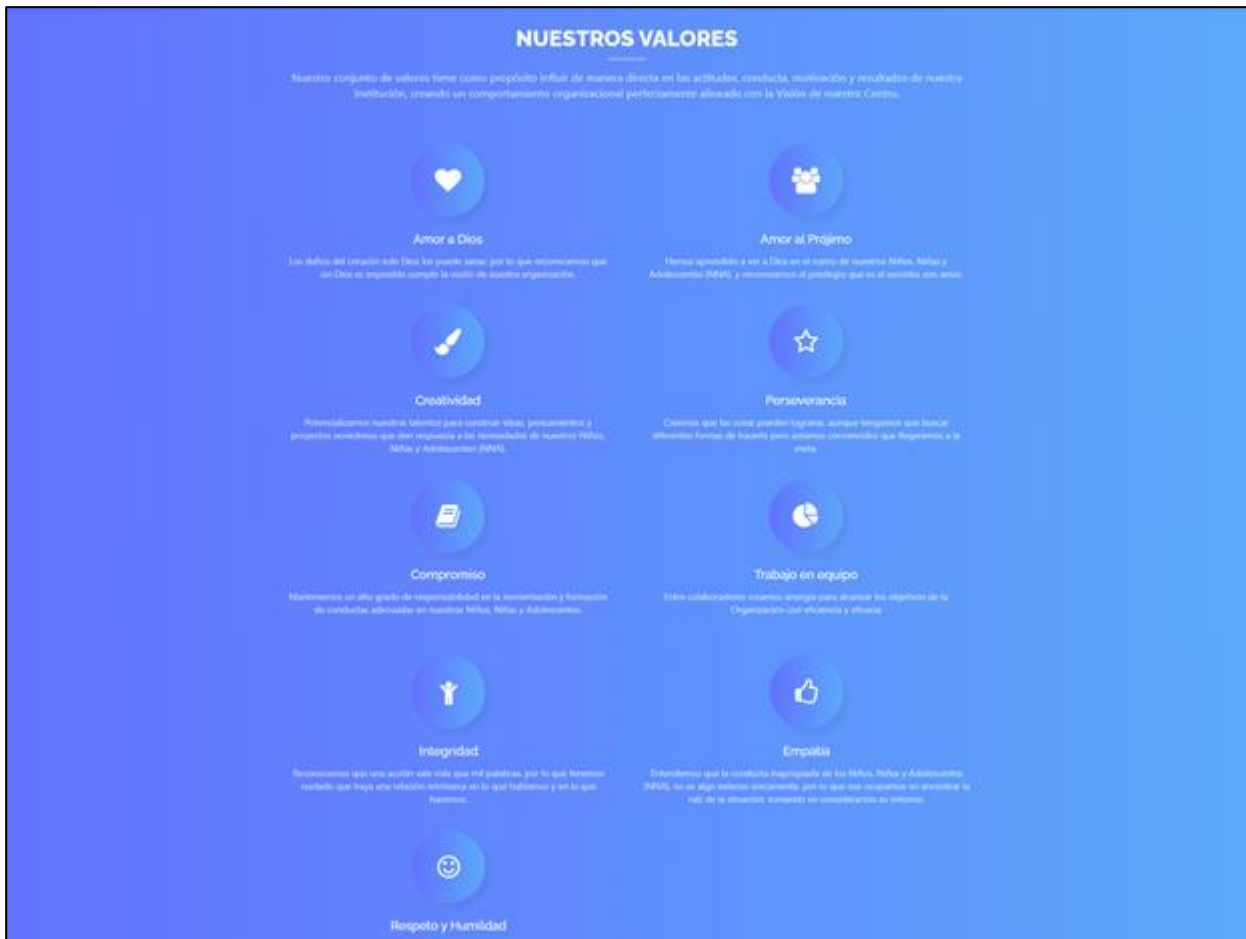


Ilustración 12 - Nuestros Valores

Fuente: (Elaboración propia)

La Ilustración 13 muestra la sección “Nuestro Equipo” en la que se muestran imágenes de la directiva de la institución junto con sus nombres y puestos en la misma.

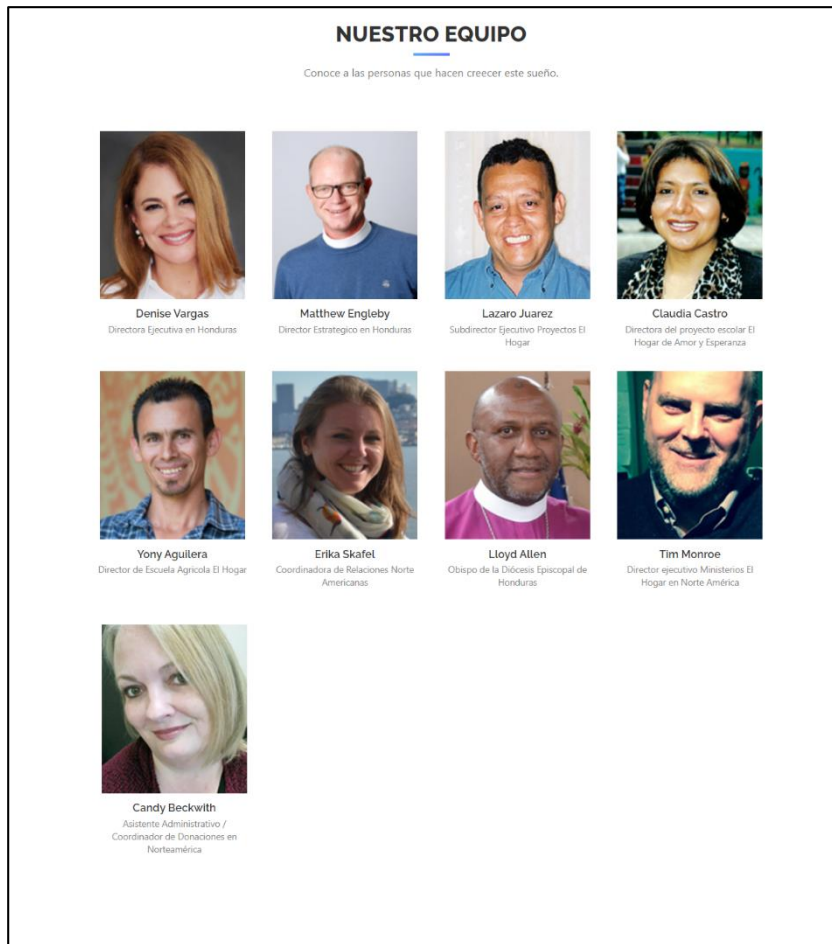


Ilustración 13 - Nuestro Equipo

Fuente: (Elaboración propia)

La Ilustración 14 muestra la sección “Contáctanos” en la que se muestra información de contacto de las diversas instituciones de Proyectos El Hogar junto con la opción de contactar a la institución.

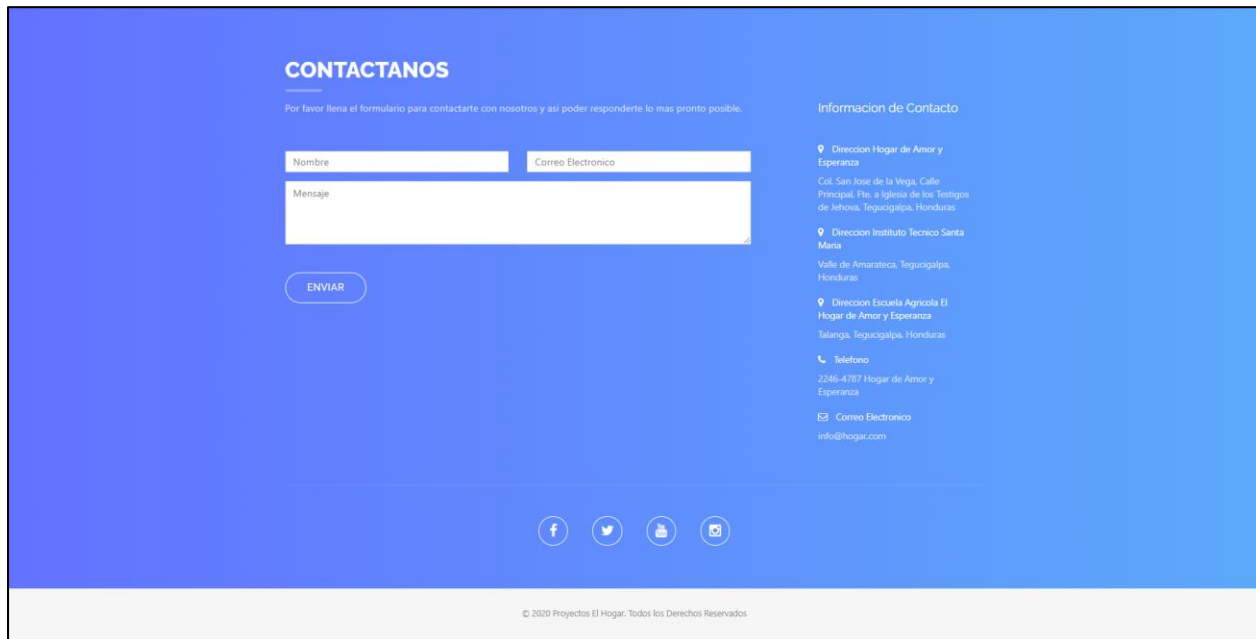
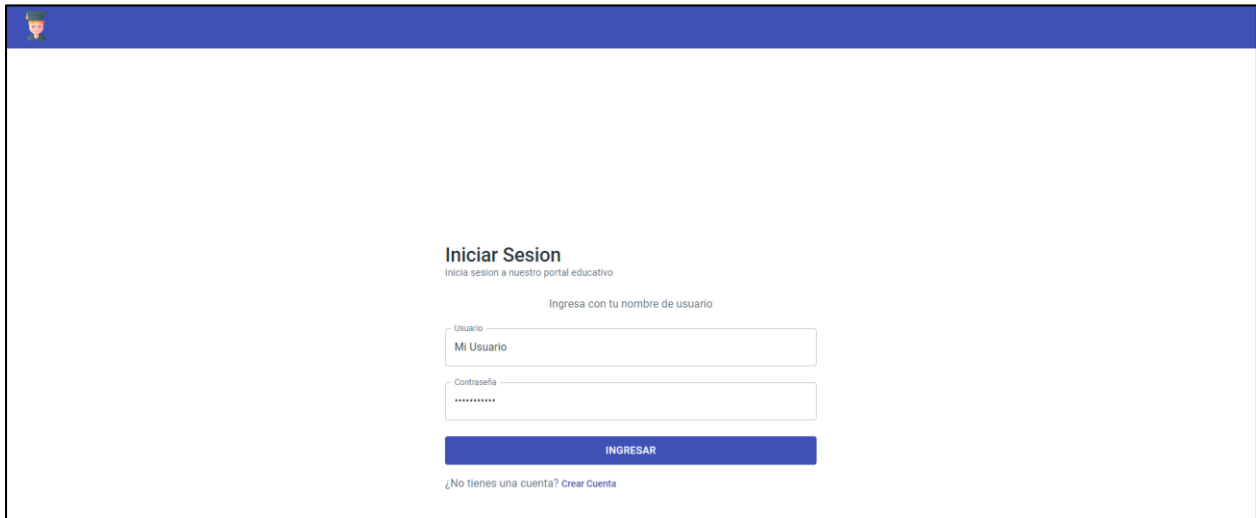


Ilustración 14 - Contáctanos

Fuente: (Elaboración propia)

5.3 CAPTURAS DE PANTALLA DEL SISTEMA WEB EDUCATIVO

La Ilustración 15 muestra el formulario para realizar el inicio de sesión al portal educativo, para realizar la autenticación se necesita el nombre de usuario y contraseña registradas al momento de completar el formulario de creación de nueva cuenta.



The screenshot shows a login interface with a blue header bar. The main content area is white and contains the following elements:

- Iniciar Sesión** (Login)
- Subtext: *Inicia sesión a nuestro portal educativo*
- Instruction: *Ingresar con tu nombre de usuario*
- Form fields:
 - Label: *Usuario*
 - Input: *Mi Usuario*
 - Label: *Contraseña*
 - Input: *******
- Submit button: **INGRESAR**
- Link: *¿No tienes una cuenta? [Crear Cuenta](#)*

Ilustración 15 - Formulario Inicio de Sesión

Fuente: (Elaboración propia)

La Ilustración 16 muestra el formulario para realizar la creación de una nueva cuenta tanto de estudiante como de maestro.

The image shows a web form for creating a new account. At the top left, there is a small logo. The main heading is "Crea una nueva cuenta" with a subtitle "Usa tu correo para crear una cuenta". The form consists of several input fields: "Nombre", "Apellido", "Usuario", a dropdown menu for "Selecciona un Rol*" with "Student" selected, "Correo Electronico", and "Contraseña". Below these fields is a checkbox labeled "Acepto los [Terminos y Condiciones](#)". A blue button labeled "CREAR CUENTA" is positioned below the checkbox. At the bottom, there is a link: "¿Ya tienes una cuenta? [iniciar Sesion](#)".

Ilustración 16 - Formulario de Creación de Cuenta

Fuente: (Elaboración propia)

La Ilustración 17 muestra la sección de usuarios para el administrador en la que puede buscar usuarios y autorizarlos para entrar a la plataforma escolar.

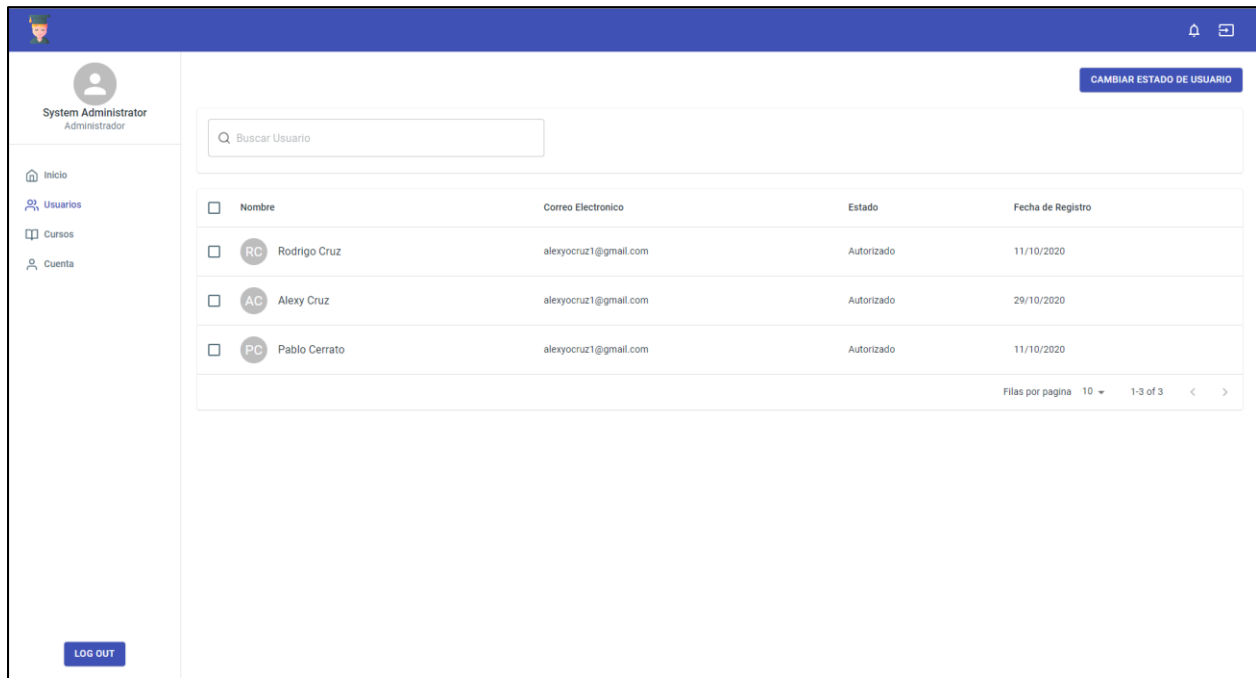


Ilustración 17 - Sección de Usuarios (Administrador)

Fuente: (Elaboración propia)

La Ilustración 18 muestra la sección de cursos para el administrador en la que puede ver una lista de ellos organizados por grado, así como también, crear, editar y eliminar los mismos. El administrador tiene la opción de crear grados en caso de que se quiera agregar una nueva práctica a grados extraordinarios.

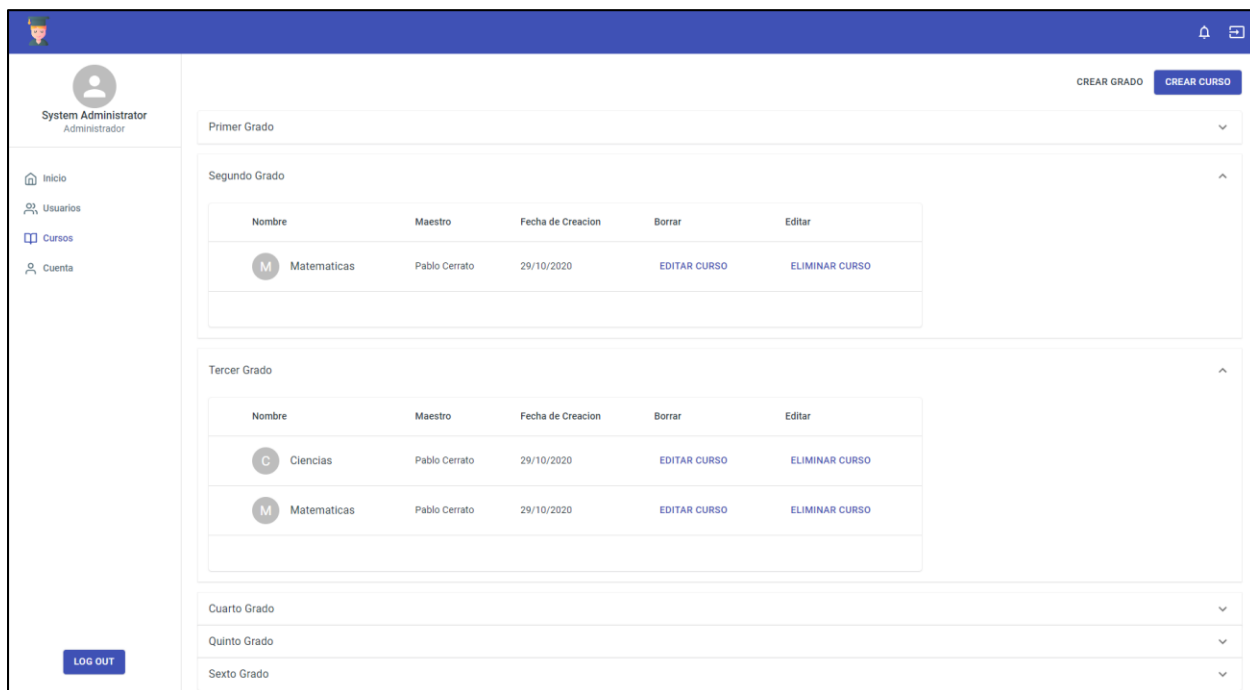


Ilustración 18 - Sección de Cursos (Administrador)

Fuente: (Elaboración propia)

La Ilustración 19 muestra la sección de la cuenta en la cual el administrador puede editar su información básica a excepción de su rol.

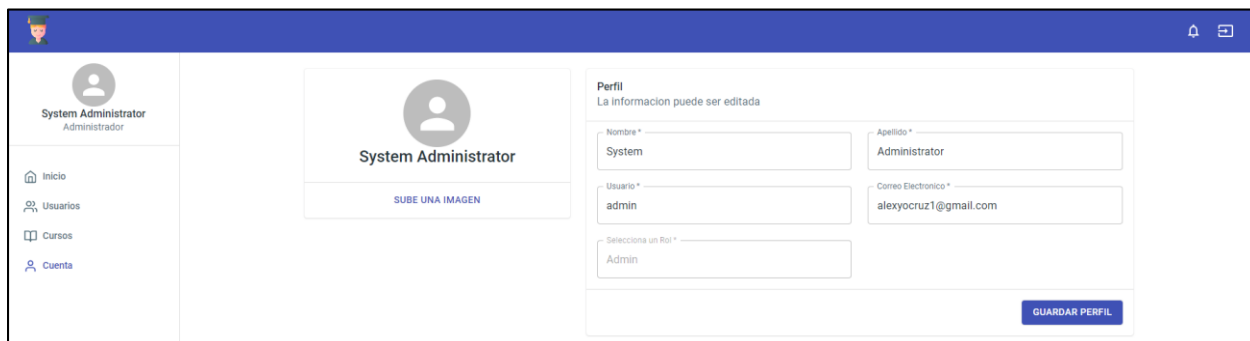


Ilustración 19 - Sección de Cuenta (Administrador)

Fuente: (Elaboración propia)

La Ilustración 20 muestra la sección de cursos en la que el maestro puede visualizar la lista de cursos asignados junto con su contenido.

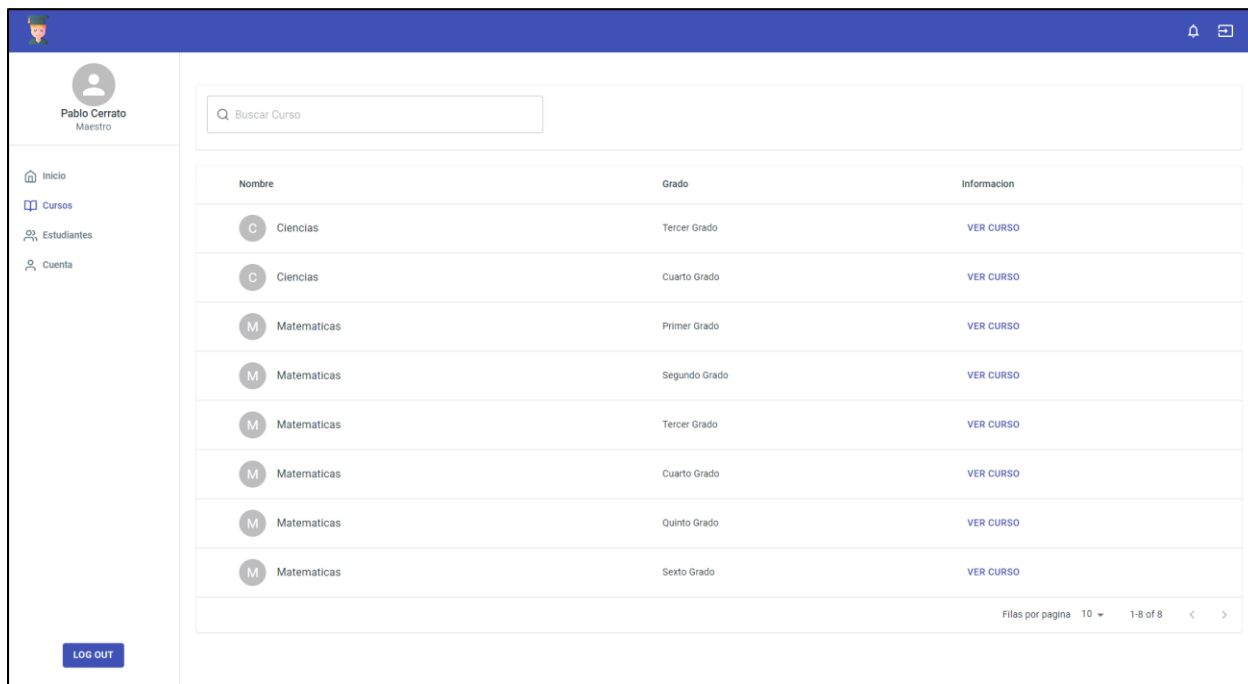


Ilustración 20 - Sección de Cursos (Maestro)

Fuente: (Elaboración propia)

La Ilustración 21 muestra la sección de estudiantes en la que el maestro puede visualizar el listado de alumnos que existen en la institución para visualizar en qué curso están asignados y así poder retirarlos de un curso o asignarlos a uno nuevo.

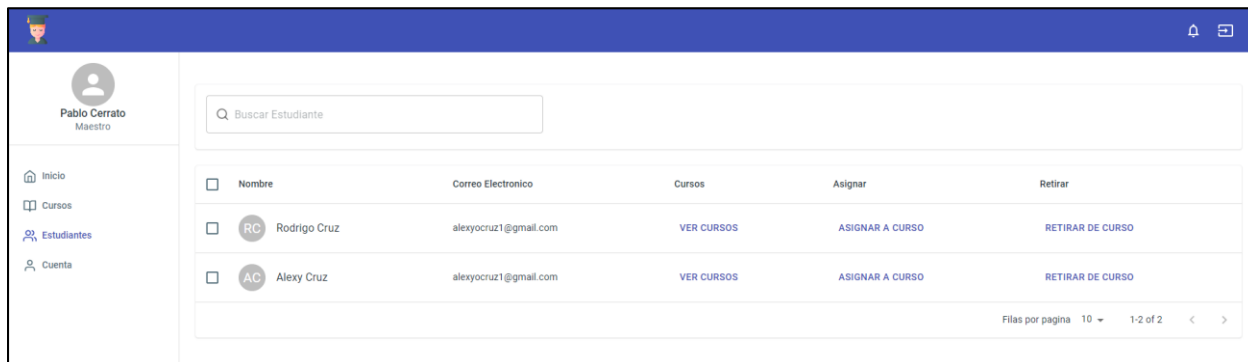
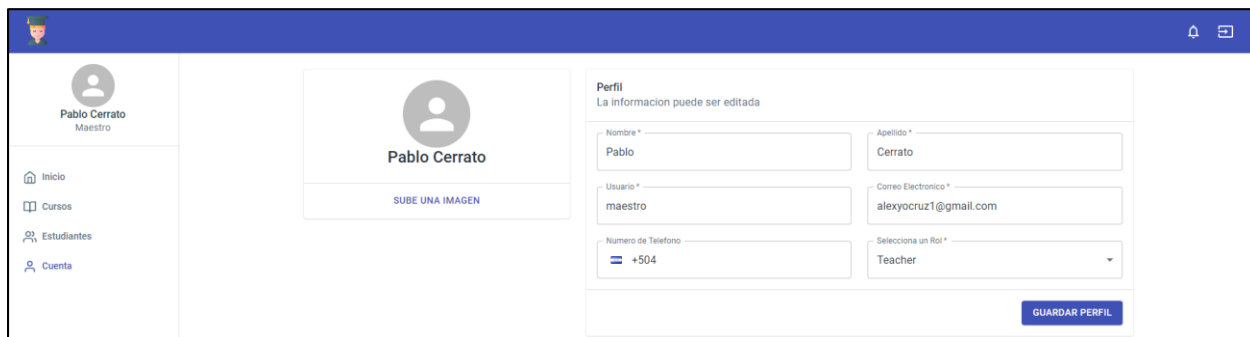


Ilustración 21 - Sección de Estudiantes (Maestro)

Fuente: (Elaboración propia)

La Ilustración 22 muestra la sección de la cuenta en la cual el Maestro puede editar su información básica, al igual que el estudiante.

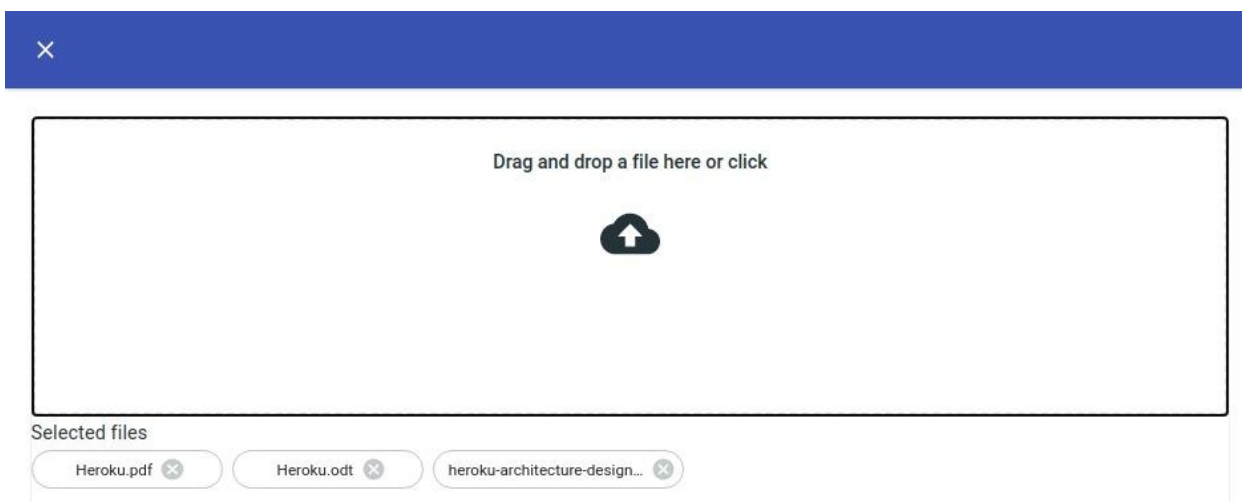


The screenshot shows a user profile editing interface. On the left is a navigation menu with 'Inicio', 'Cursos', 'Estudiantes', and 'Cuenta'. The main area features a profile card for 'Pablo Cerrato' with a 'SUBE UNA IMAGEN' button. To the right is a 'Perfil' form with the following fields: 'Nombre *' (Pablo), 'Apellido *' (Cerrato), 'Usuario *' (maestro), 'Correo Electronico *' (alexycruz1@gmail.com), 'Numero de Telefono' (+504), and 'Selecciona un Rol *' (Teacher). A 'GUARDAR PERFIL' button is at the bottom right.

Ilustración 22 - Sección de Cuenta (Maestro)

Fuente: (Elaboración propia)

La Ilustración 23 muestra la sección en la que los maestros pueden subir archivos o recursos, ya sean imágenes o documentos.



The screenshot shows a file upload interface. At the top is a blue bar with a close button. Below is a large white area with the text 'Drag and drop a file here or click' and a cloud upload icon. At the bottom, a 'Selected files' section shows three files: 'Heroku.pdf', 'Heroku.odt', and 'heroku-architecture-design...'. Each file has a close button.

Ilustración 23 - Sección de Archivos (Maestro)

Fuente: (Elaboración propia)

La Ilustración 24 muestra la sección de cursos en la que el estudiante puede ver los cursos en los que ha sido enlistado junto con su contenido.

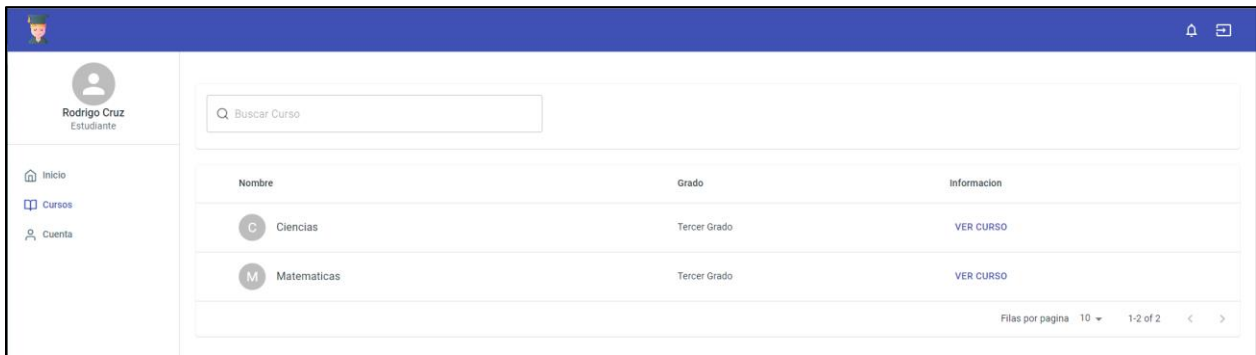


Ilustración 24 - Sección de Cursos (Estudiante)

Fuente: (Elaboración propia)

La Ilustración 25 muestra las campañas, con las cuales se pueden realizar donaciones y visualizar información referente a las mismas.

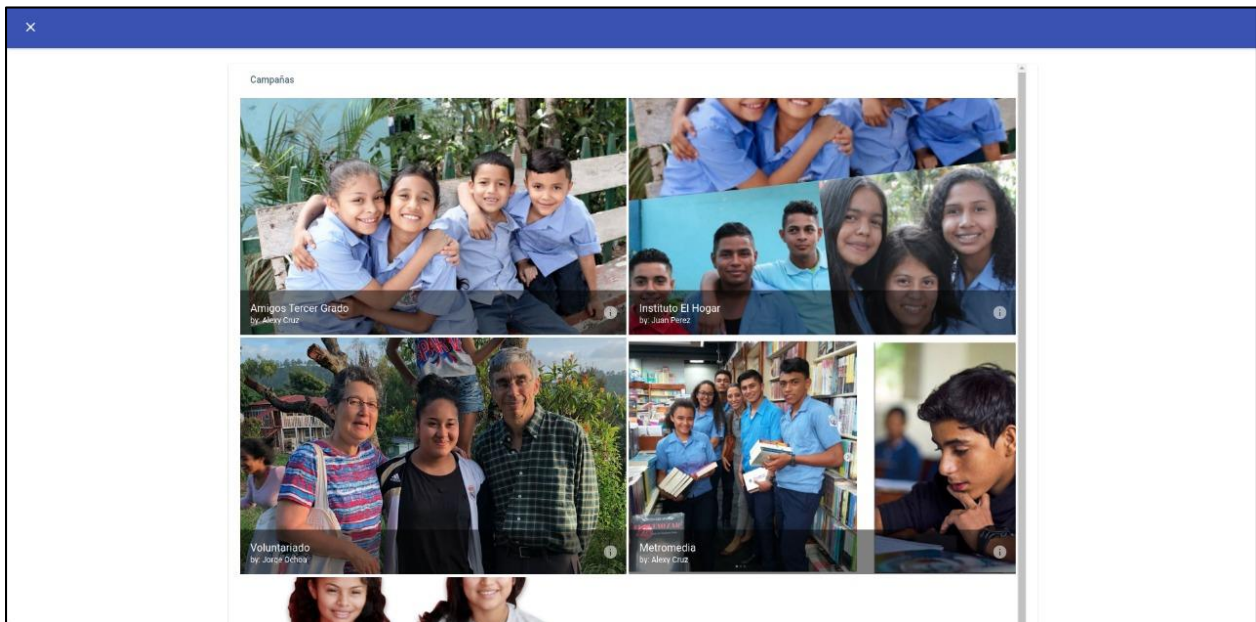


Ilustración 25 - Sección de Campañas

Fuente: (Elaboración propia)

La Ilustración 26 y 27 muestran el chat que se puede usar para iniciar conversaciones dentro de la aplicación, de igual manera se cuenta con un video chat para que maestros y alumnos puedan tener conversaciones de muchos a muchos.



The image shows a chat room entry interface. At the top, the word "CHAT" is displayed in a large, bold, black font. Below it is a horizontal line. Underneath the line are two input fields. The first input field contains the text "Alexy" followed by a vertical cursor. The second input field contains the text "sala1". Below these two input fields is a blue button with the text "UNIRSE" in white, uppercase letters.

Ilustración 26 - Ingreso a Sala de Chat

Fuente: (Elaboración propia)

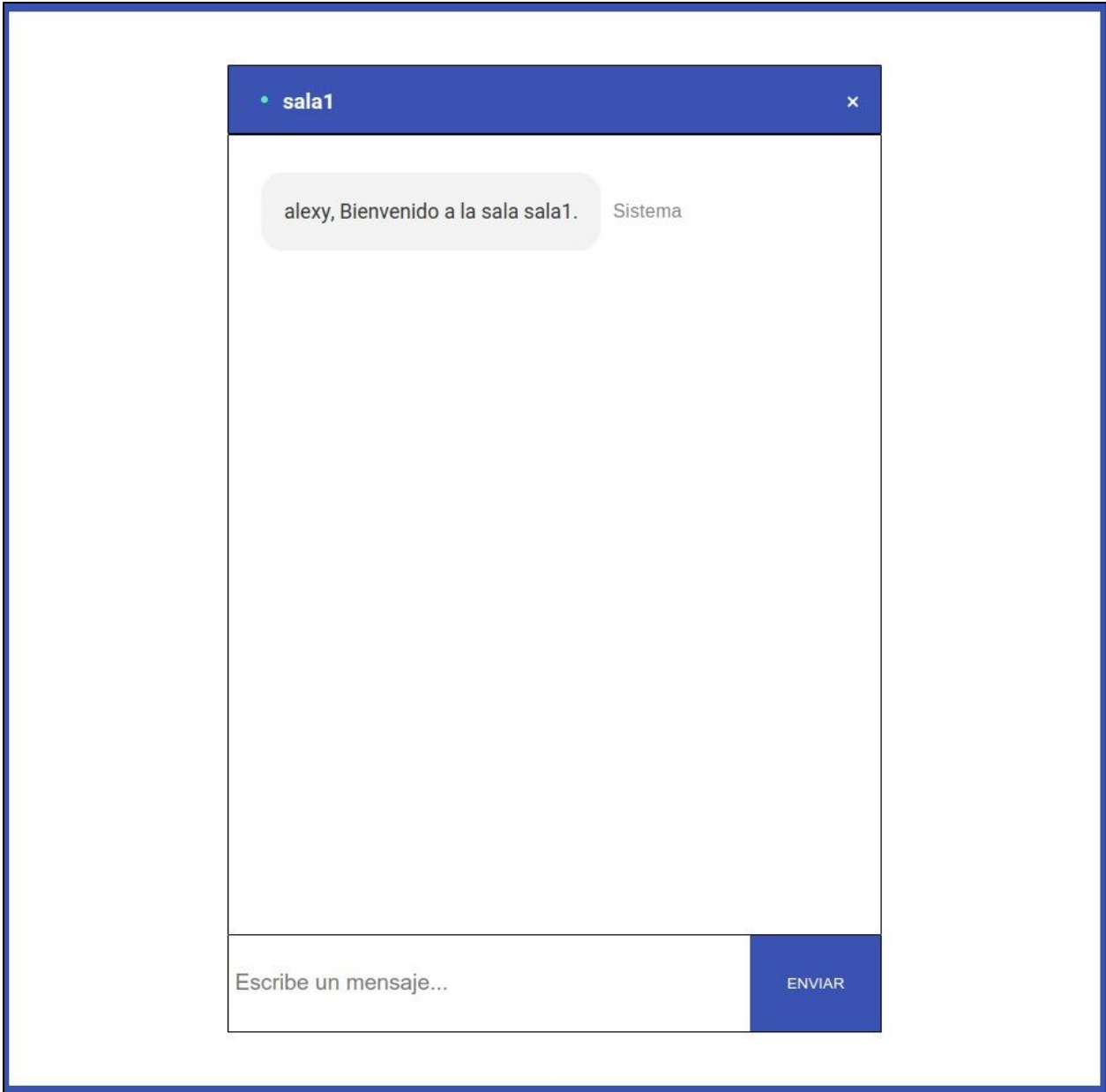
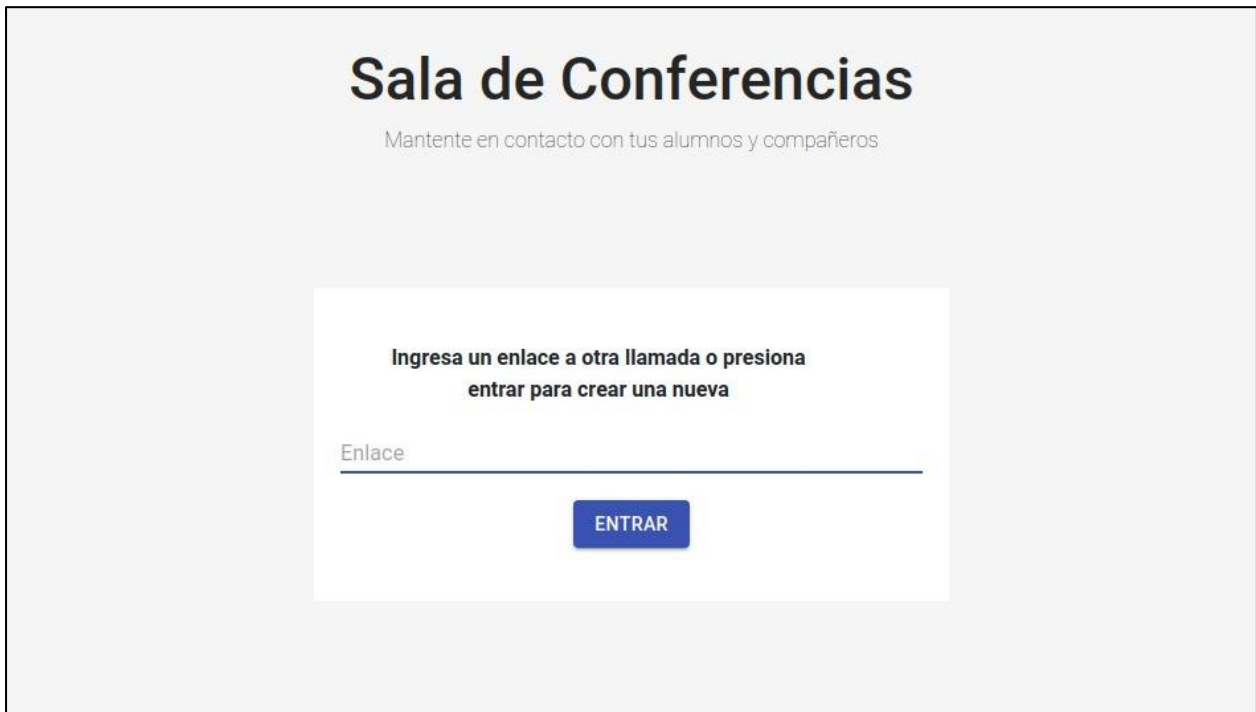


Ilustración 27 - Sala de Chat

Fuente: (Elaboración propia)

La Ilustración 28 muestra el inicio de la sala de conferencias en donde los maestros pueden elegir entrar a un enlace pre existente o crear una nueva sala de conferencias.



Sala de Conferencias

Mantente en contacto con tus alumnos y compañeros

Ingresa un enlace a otra llamada o presiona
entrar para crear una nueva

Enlace

ENTRAR

Ilustración 28 - Sección de Video Chat (Inicio)

Fuente: (Elaboración propia)

La Ilustración 29 muestra la sección de la sala de conferencias en donde el usuario ingresa su nombre de identificación y tiene una pre visualización de su cámara.

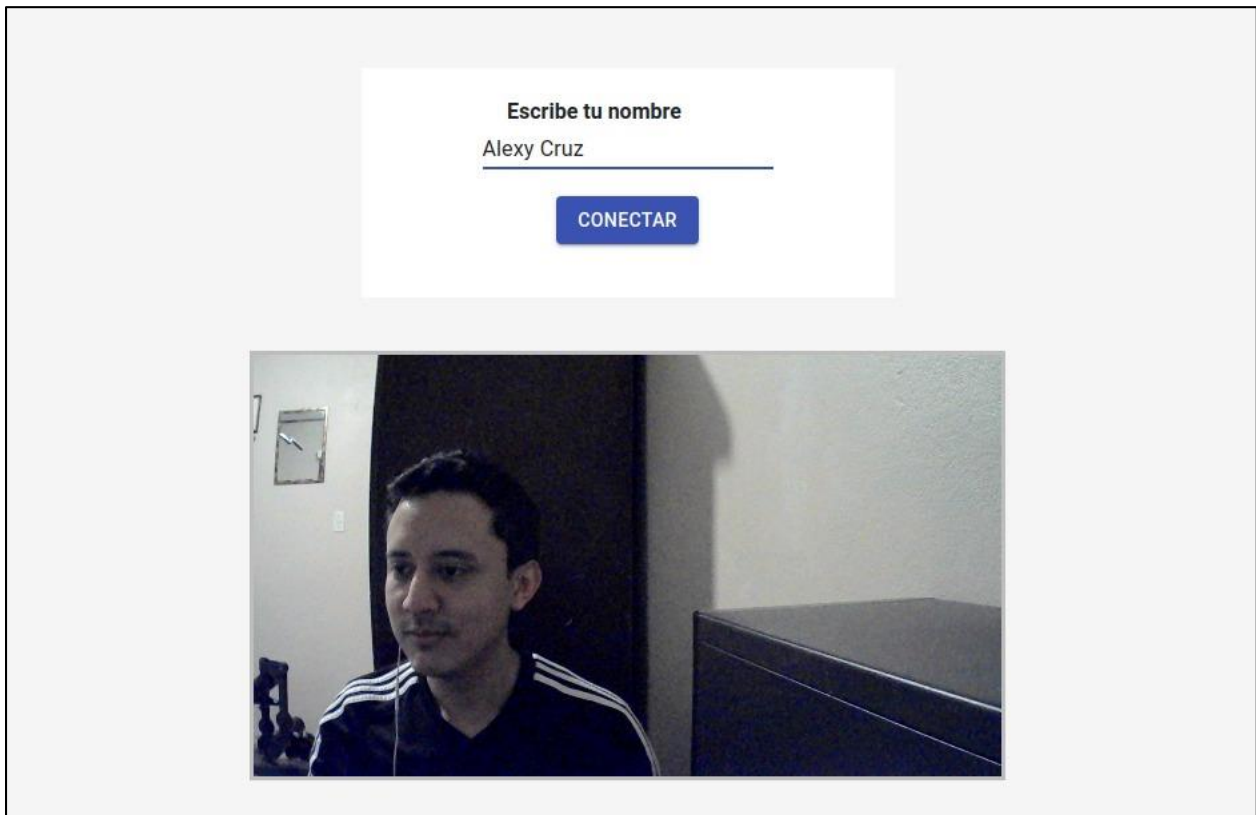


Ilustración 29 - Sala de Video Chat (Pre visualización)

Fuente: (Elaboración propia)

La Ilustración 30 muestra la sección de la sala de conferencias donde el usuario puede interactuar con las demás personas, colgar la llamada, compartir contenido y hacer uso del chat. Además, puede hacer uso del enlace de invitación para que otros miembros puedan unirse a la llamada.

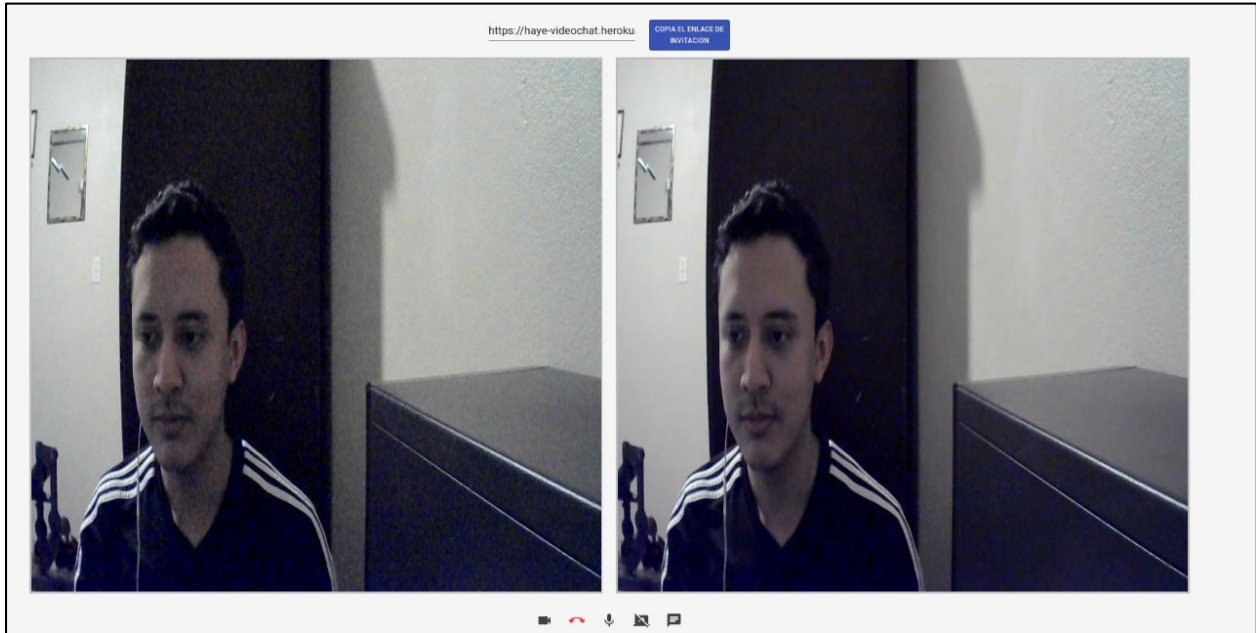


Ilustración 30 - Sección de Video Chat (Llamada)

Fuente: (Elaboración propia)

VI. CONCLUSIONES

1. Se desarrolló una aplicación educativa y promocional en la que tanto el docente, estudiantes y público en general pueden interactuar entre sí mediante chats, video llamadas y recursos de todo tipo.
2. Dentro de la plataforma escolar se desarrollaron espacios en donde el estudiante puede acceder a recursos creados por el docente como imágenes, documentos y videos.
3. La plataforma educativa cuenta con secciones en las que el docente puede realizar el planteamiento de cursos escolares al desarrollar contenido didáctico para cada ciclo escolar.
4. Mediante una plataforma de promoción se puede visualizar no solo la información vital de la institución sino también apoyar campañas de manera monetaria para asegurar el crecimiento y continuidad de las instituciones.
5. Mediante el uso de las plataformas se logró monitorear de mejor manera a la institución y sus miembros además de hacer algunos procesos mas eficientes.

VII. RECOMENDACIONES

1. Se recomienda ejecutar un análisis del equipo en donde se utilizará el sistema para asegurar cualquier problema de procesamiento, rendimiento, integridad y rapidez tanto de software como de hardware.
2. Fomentar el uso de recursos digitales mediante la plataforma escolar para así poder contribuir a la retención del contenido y facilitar la enseñanza.
3. Realizar análisis continuo acerca del número de usuarios (docentes, alumnos y personal administrativo) que accederá a la plataforma educativa y promocional diariamente y qué recursos se utilizarán para así poder monitorear el crecimiento de usuarios en la plataforma y verificar los límites de la misma para que esta pueda continuar de manera óptima.

BIBLIOGRAFÍA

1. Blog, T., & Scrum, T. (2020). What is Scrum and How Does It Work | Logic Solutions. Extraído el 31 de Agosto 2020, de <https://www.logicsolutions.com/what-is-scrum-how-it-works/>
2. Browserify. (2020). Extraído el 31 de Agosto 2020, de <http://browserify.org/>
3. Cordero, W. (2015). ¿Cómo funciona la metodología Scrum?. Extraído el 31 de Agosto 2020, de <https://platzi.com/blog/metodologia-scrum-fases/>
4. gangwar. (2020). Cloud models and platforms. Extraído el 31 de Agosto 2020, de <https://www.slideshare.net/Prabhatgangwar/cloud-models-and-platforms-62907419>
5. Grunt: Qué es y Cómo se Utiliza. (2020). Extraído el 31 de Agosto 2020, de <https://www.neoguias.com/grunt/#Que es Grunt>
6. Introducción a Express/Node. (2020). Extraído el 31 de Agosto 2020, de https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction
7. Monte Galiano, J. (2016). Implantar scrum con éxito. Barcelona: Editorial UOC.
8. Muñoz Izquierdo, C. (2012). Tres problemas fundamentales del sistema educativo. Extraído el 31 de Agosto 2020, de http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0185-26982012000500014
9. Node.js ▷ ¿Qué es y para qué sirve NodeJS? ? ¿Qué no es NodeJS?. (2020). Retrieved 31 August 2020, from <https://apasionados.es/blog/nodejs-4430/>
10. ¿Qué es Apache Cordova?. (2020). Extraído el 31 de Agosto 2020, de <https://www.vix.com/es/btg/tech/13241/que-es-apache-cordova>
11. ¿Qué es Apache Tomcat?. (2020). Extraído el 31 de Agosto 2020, de <https://www.hostdime.com.ar/blog/que-es-apache-tomcat/>
12. ¿Qué es Apache? Descripción completa del servidor web Apache. (2020). Extraído el 31 de Agosto 2020, de <https://www.hostinger.es/tutoriales/que-es-apache/>
13. Qué es el Backend de una web y por qué es tan importante. (2020). Extraído el 31 de Agosto 2020, de <https://rafarjonilla.com/que-es/backend/>
14. ¿Qué es Gulp y para que sirve? | return(GiS);. (2020). Extraído el 31 de Agosto 2020, de <https://www.returngis.net/2016/07/que-es-gulp-y-para-que-sirve/>

15. Qué es JQuery y cómo implementarlo. (2020). Extraído el 31 de Agosto 2020, de <https://www.laurachuburu.com.ar/tutoriales/que-es-jquery-y-como-implementarlo.php>
16. Qué es NPM y para qué sirve. (2020). Extraído el 31 de Agosto 2020, de <https://openwebinars.net/blog/que-es-node-package-manager/>
17. Qué es React. Por qué usar React. (2020). Extraído el 31 de Agosto 2020, de <https://desarrolloweb.com/articulos/que-es-react-motivos-uso.html>
18. Qué es Redux. (2020). Extraído el 31 de Agosto 2020, de <https://desarrolloweb.com/articulos/que-es-redux.html>
19. Qué es SCRUM. (2008). Extraído el 31 de Agosto 2020, de <https://proyectosagiles.org/que-es-scrum/>
20. Villán, V. (2019). Las metodologías ágiles más utilizadas y sus ventajas dentro de la empresa. Extraído el 31 de Agosto 2020, de <https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>