



UNIVERSIDAD ESTATAL DE MILAGRO
FACULTAD CIENCIAS E INGENIERÍA

TRABAJO DE PROPUESTA TECNÓLOGICA/PROYECTO TECNICO
PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
SISTEMAS COMPUTACIONALES

TEMA: “DESARROLLO DE UNA APLICACION WEB PARA
GESTIONAR LOS PROYECTOS DE EMPRENDIMIENTOS DE LA
FACULTAD CIENCIA DE LA INGENIERIA (FACI) DE LA
UNIVERSIDAD ESTATAL DE MILAGRO”

Autores:

Sr. LOPEZ ROBLES JHOYDER HAYAN

Sr. GUACHUN SALDAÑA JOSE LUIS

Tutor:

Mgr. VERA PAREDES DANIEL ALEXANDER

Milagro, Junio 2021

ECUADOR

DERECHOS DE AUTOR

Ingeniero.

Fabrizio Guevara Viejó, PhD.

RECTOR

Universidad Estatal de Milagro

Presente.

Yo, LOPEZ ROBLES JHOYDER HAYAN , en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de integración curricular, modalidad presencial, mediante el presente documento, libre y voluntariamente procedo a hacer entrega de la Cesión de Derecho del Autor, como requisito previo para la obtención de mi Título de Grado, como aporte a la Línea de Investigación 1.EDUCACIÓN, CULTURA, TECNOLOGÍA EN INNOVACIÓN PARA LA SOCIEDAD 2.SOCIEDAD DE LA INFORMACIÓN GESTIÓN, MEDIOS Y TECNOLOGÍA 3.DESARROLLO LOCAL Y EMPRESARIAL 2019 - FACI - PROYECTOS INTEGRADORES, de conformidad con el Art. 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación, concedo a favor de la Universidad Estatal de Milagro una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservo a mi favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo a la Universidad Estatal de Milagro para que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El autor declara que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Milagro, 4 de junio de 2021

LOPEZ ROBLES JHOYDER HAYAN

Autor 1

CI: 0604722090

DERECHOS DE AUTOR

Ingeniero.

Fabrizio Guevara Viejó, PhD.

RECTOR

Universidad Estatal de Milagro

Presente.

Yo, GUACHUN SALDAÑA JOSE LUIS, en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de integración curricular, modalidad *Elija un elemento.*, mediante el presente documento, libre y voluntariamente procedo a hacer entrega de la Cesión de Derecho del Autor, como requisito previo para la obtención de mi Título de Grado, como aporte a la Línea de Investigación 1.EDUCACIÓN, CULTURA, TECNOLOGÍA EN INNOVACIÓN PARA LA SOCIEDAD 2.SOCIEDAD DE LA INFORMACIÓN GESTIÓN, MEDIOS Y TECNOLOGÍA 3.DESARROLLO LOCAL Y EMPRESARIAL 2019 - FACI - PROYECTOS INTEGRADORES, de conformidad con el Art. 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación, concedo a favor de la Universidad Estatal de Milagro una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservo a mi favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo a la Universidad Estatal de Milagro para que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El autor declara que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Milagro, 4 de junio de 2021

GUACHUN SALDAÑA JOSE LUIS

Autor 2

CI: 0302244926

APROBACIÓN DEL TUTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Yo, VERA PAREDES DANIEL ALEXANDER en mi calidad de tutor del trabajo de integración curricular, elaborado por los estudiantes LOPEZ ROBLES JHOYDER HAYAN y GUACHUN SALDAÑA JOSE LUIS, cuyo título es DESARROLLO DE UNA APLICACION WEB PARA GESTIONAR LOS PROYECTOS DE EMPRENDIMIENTOS DE LA UNIVERSIDAD ESTATAL DE MILAGRO, que aporta a la Línea de Investigación 1. EDUCACIÓN, CULTURA, TECNOLOGÍA EN INNOVACIÓN PARA LA SOCIEDAD 2. SOCIEDAD DE LA INFORMACIÓN GESTIÓN, MEDIOS Y TECNOLOGÍA 3. DESARROLLO LOCAL Y EMPRESARIAL 2019 - FACI - PROYECTOS INTEGRADORES previo a la obtención del Título de Grado INGENIERIA EN SISTEMAS; considero que el mismo reúne los requisitos y méritos necesarios en el campo metodológico y epistemológico, para ser sometido a la evaluación por parte del tribunal calificador que se designe, por lo que lo APRUEBO, a fin de que el trabajo sea habilitado para continuar con el proceso previa culminación de Trabajo de Integración Curricular de la Universidad Estatal de Milagro.

Milagro, 4 de junio de 2021

VERA PAREDES DANIEL ALEXANDER

Tutor

C.I: Haga clic aquí para escribir cédula (Tutor).

APROBACIÓN DEL TRIBUNAL CALIFICADOR

El tribunal calificador constituido por:

Elija un elemento. Haga clic aquí para escribir apellidos y nombres (tutor).

Haga clic aquí para escribir apellidos y nombres (Secretario/a).

Elija un elemento. Haga clic aquí para escribir apellidos y nombres (integrante).

Luego de realizar la revisión del Trabajo de Integración Curricular, previo a la obtención del título (o grado académico) de ELIJA UN ELEMENTO. presentado por Elija un elemento. Haga clic aquí para escribir apellidos y nombres (estudiante1).

Con el tema de trabajo de Integración Curricular: Haga clic aquí para escribir el tema del Trabajo de Integración Curricular.

Otorga al presente Trabajo de Integración Curricular, las siguientes calificaciones:

Trabajo Curricular	Integración	[]
Defensa oral		[]
Total		[]

Emite el siguiente veredicto: (aprobado/reprobado) _____

Fecha: Haga clic aquí para escribir una fecha.

Para constancia de lo actuado firman:

	Nombres y Apellidos			Firma
Presidente	Apellidos Presidente.	y	nombres	de _____
Secretario /a	Apellidos Secretario	y	nombres	de _____
Integrante	Apellidos Integrante.	y	nombres	de _____

APROBACIÓN DEL TRIBUNAL CALIFICADOR

El tribunal calificador constituido por:

Elija un elemento. Haga clic aquí para escribir apellidos y nombres (tutor).

Elija un elemento. Haga clic aquí para escribir apellidos y nombres (Secretario/a).

Elija un elemento. Haga clic aquí para escribir apellidos y nombres (integrante).

Luego de realizar la revisión del Trabajo de Integración Curricular, previo a la obtención del título (o grado académico) de ELIJA UN ELEMENTO. presentado por Elija un elemento. Haga clic aquí para escribir apellidos y nombres (estudiante2).

Con el tema de trabajo de Integración Curricular: Haga clic aquí para escribir el tema del Trabajo de Integración Curricular.

Otorga al presente Proyecto Integrador, las siguientes calificaciones:

Trabajo de Integración Curricular	[]
Defensa oral	[]
Total	[]

Emite el siguiente veredicto: (aprobado/reprobado) _____

Fecha: Haga clic aquí para escribir una fecha.

Para constancia de lo actuado firman:

	Nombres y Apellidos				Firma
Presidente	Apellidos Presidente.	y	nombres	de	_____
Secretario /a	Apellidos Secretario	y	nombres	de	_____
Integrante	Apellidos Integrante.	y	nombres	de	_____

DEDICATORIA

Dedico este presente trabajo de investigación a Dios por darme todas las fuerzas necesarias para seguir cada día y a mis padres por su gran fortaleza y compromiso que fue el motor que me impulso y permitió avanzar incluso en momentos muy difíciles, su confianza y por todos los valores inculcados desde la niñez me han permitido llegar hasta aquí. Gracias desde el fondo de mi corazón.

López Robles Jhoyder Hayan

Quiero dedicar este proyecto en primer lugar a Dios que con su infinito amor me ha sabido guiar en los retos que me impone la vida día a día, brindándome sabiduría para poder tomar buenas decisiones respecto a mi vida.

A mi madre que siempre me supo apoyar y guiar de la forma adecuada para afrontar los conflictos que se me presentaban en el camino.

A mi tía que supo criarme, enseñarme valores y principios que perduran hasta la actualidad, brindándome su apoyo y confianza en que realizaría lo correcto en todos los aspectos de la vida.

Guachun Saldaña José Luis

AGRADECIMIENTO

Gracias Dios por darme la oportunidad de estudiar y poder culminar con éxito mi carrera universitaria, a mis padres por todo el sacrificio que hicieron por verme salir adelante, por todos sus consejos y perseverancia que me transmitían cada día, a mis compañeros con quien compartí todo este tiempo de estudio. Finalmente, a todos los que de manera directa o indirectamente ayudaron para la realización de este trabajo

López Robles Jhoyder Hayan

Agradezco infinitamente a Dios por brindarme su ayuda incondicional en todo momento, a mi familia que siempre estuvieron para guiarme en todos los ámbitos de la vida.

A mis compañeros y amigos que convivimos todo el periodo Universitario, por brindarme su ayuda y constancia al momento de estudiar.

Guachun Saldaña José Luis

ÍNDICE GENERAL

DERECHOS DE AUTOR.....	I
DERECHOS DE AUTOR.....	II
APROBACIÓN DEL TUTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR...	III
APROBACIÓN DEL TRIBUNAL CALIFICADOR	IV
APROBACIÓN DEL TRIBUNAL CALIFICADOR	V
DEDICATORIA.....	VI
AGRADECIMIENTO.....	VII
ÍNDICE GENERAL.....	VIII
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS.....	XI
RESUMEN.....	1
ABSTRACT	2
CAPÍTULO 1	3
1. INTRODUCCIÓN.....	3
1.1. Planteamiento del problema.....	4
1.2. Objetivos.....	4
1.2.1. Objetivo General.....	4
1.2.2. Objetivos Específicos	4
1.3. Alcance	5
1.4. Estado del arte.....	6
1.4.1. Base Legal	6
1.4.2. Marco Teórico	6
1.4.3. Marco Referencial	16
CAPÍTULO 2	18
2. METODOLOGÍA.....	18
CAPÍTULO 3	22
3. PROPUESTA DE SOLUCIÓN.....	22
CONCLUSIONES.....	52
RECOMENDACIONES	53
REFERENCIAS BIBLIOGRÁFICAS	54
ANEXOS.....	56

ÍNDICE DE FIGURAS

Figura 1. Esquema del proyecto	5
Figura 2. Arquitectura de la aplicación Web	8
Figura 3. Tipos de requerimientos	10
Figura 4. Modelo Vista Controlador.....	11
Figura 5. Modelo MTV	12
Figura 6. Etapas del modelo Cascada.....	19
Figura 7. Diagrama contextual aplicación web para gestionar los proyectos de emprendimientos	22
Figura 8. Diagrama modular.....	24
Figura 9. Caso de Uso - Módulo Usuario.....	27
Figura 10. Caso de Uso - Módulo Perfil.....	28
Figura 11. Caso de Uso - Módulo Proyecto	29
Figura 12. Caso de Uso - Módulo Interacción.....	29
Figura 13. Modelo Entidad Relación de la aplicación	32
Figura 14: Arquitectura de la Aplicación web.....	40
Figura 15. Interfaz de Inicio de Sesión.....	41
Figura 16. Interfaz de Registro de Usuario.....	41
Figura 17. Mensaje de correo electrónico con código de verificación	42
Figura 18. Interfaz de Validación de Cuenta.....	43
Figura 19. Interfaz al momento de iniciar sesión (Feed).....	43
Figura 20. Interfaz de Publicar Proyecto	44
Figura 21. Interfaz de Proyectos por filtrado.....	45
Figura 22. Interfaz de perfil de Usuario	45
Figura 23. Interfaz del detalle de un proyecto	46
Figura 24. Interfaz de interacciones con el proyecto visualizado.....	47
Figura 25. Interfaz de búsqueda de usuarios	47
Figura 26. Interfaz de perfil de visita de usuario	48
Figura 27. Interfaz de chat.....	48
Figura 28. Interfaz de notificaciones	49
Figura 29. Interfaz de Actualización de Perfil.....	49
Figura 30. Interfaz de Actualización de usuario.....	50

Figura 31. Interfaz de Cambio de contraseña	50
Figura 32. Interfaz de Redes sociales	51
Figura 33. Interfaz de eliminación de cuenta	51

ÍNDICE DE TABLAS

Tabla 1. Requerimientos para la construcción del sistema web	23
Tabla 2. Jerarquía funcional de la Aplicación web de gestor de proyectos de emprendimiento	24
Tabla 3. Caracterización del módulo Usuario	25
Tabla 4. Caracterización del módulo Perfil	26
Tabla 5. Caracterización del módulo Proyecto.....	26
Tabla 6. Caracterización del módulo Interacción	27
Tabla 7. Requerimientos de tiempo de respuesta	30
Tabla 8. Requerimientos de seguridad de acceso	30
Tabla 9. Requerimiento de seguridad de datos.....	31
Tabla 10. Requerimiento de diseño adaptable a diferentes dispositivos	31
Tabla 11. Requerimiento estructura de interfaz gráfica	32
Tabla 12. Tabla Usuario	33
Tabla 13. Tabla de Perfil	34
Tabla 14. Tabla de Red_Social.....	34
Tabla 15. Tabla de Notificación	35
Tabla 16. Tabla de Ciudad.....	35
Tabla 17. Tabla de Provincia.....	35
Tabla 18. Tabla de País.....	35
Tabla 19. Tabla de Proyecto	36
Tabla 20. Tabla Categoría	36
Tabla 21. Tabla Tipo de Proyecto	37
Tabla 22. Tabla Estado	37
Tabla 23. Tabla de Me Gusta.....	37
Tabla 24. Tabla de Conversación	38
Tabla 25. Tabla de Comentario	38
Tabla 26. Tabla de Seguir.....	38

Título de Trabajo Integración Curricular: Desarrollo de una aplicación web para gestionar los proyectos de emprendimientos en la Facultad Ciencia de la Ingeniería de la Universidad Estatal de Milagro.

RESUMEN

El presente proyecto se desarrolló con el fin de ayudar a los estudiantes de la Universidad Estatal de Milagro, dando a conocer sus ideas de emprendimiento a la sociedad, los mismos que permitirán el desarrollo económico de la comunidad. Entendiéndose que un emprendedor busca alguien que lo conecte con más personas y que le brinde oportunidades de negocio. El sistema informático se desarrolló aplicando la metodología cascada, la cual asegura un mayor control sobre el proyecto, y una implementación más efectiva y eficiente. Para el desarrollo de la aplicación, se utilizaron las tecnologías de desarrollo web, como lo es Python, con el framework django y almacenamiento de los datos en PostgreSQL. Además de aquello se utilizó bootstrap con el objetivo de que las interfaces se adapten automáticamente a cualquier dispositivo. Finalmente se obtuvo un sistema ligero, amigable, responsivo e intuitivo para el usuario. El sistema permite al usuario registrar sus ideas y estas sean publicados siempre y cuando cumplan con los requerimientos establecidos.

PALABRAS CLAVE: Emprendimiento, Proyecto, Ideas, Sistema informático.

Título de Trabajo Integración Curricular: Development of a web application to manage entrepreneurial projects at the Faculty of Engineering Science of the State University of Milagro.

ABSTRACT

This project was developed in order to help the students of the State University of Milagro, making their entrepreneurship ideas known to society, which will allow the economic development of the community. Understanding that an entrepreneur is looking for someone who connects him with more people and provides business opportunities. The computer system was developed applying the waterfall methodology, which ensures greater control over the project, and a more effective and efficient implementation. For the development of the application, it will be used in web development technologies, such as Python, with the django framework and data storage in PostgreSQL. In addition to that, bootstrap was used with the aim that the interfaces were automatically adapted to any device. Finally, a light, friendly, responsive and intuitive system for the user was obtained. The system allows the user to register their ideas and they are published as long as they meet the established requirements.

KEY WORDS: Entrepreneurship, Project, Computer system.

CAPÍTULO 1

1. INTRODUCCIÓN

En la actualidad el crecimiento de las aplicaciones web están ayudando activamente a muchos emprendedores a crecer de una manera más sencilla, rápida y eficiente, siendo una herramienta muy importante para emprender en una idea de negocio o proyectos tecnológicos que tengan como fin satisfacer una necesidad, demanda o servicio.

Los estudiantes de la facultad de ciencia de la ingeniería de la Universidad Estatal de Milagro, en muchas ocasiones tienen ideas brillantes tales como emprendimientos o proyectos tecnológicos de gran importancia tanto para la institución como para la sociedad en general, pero no se dan a conocer o no reciben el apoyo necesario para empezar a emprenderlos. Teniendo como necesidad una plataforma que permita la integración de inversionistas, colaboradores, emprendedores y personas de igual interés que se centren en los proyectos de emprendimiento de los estudiantes, pudiendo interactuar con los titulares de dichos proyectos para encontrar ya sea socios, colaboradores entre otros. Considerando lo expuesto, proponemos la implementación de una aplicación web que gestione los proyectos de emprendimientos de los estudiantes.

La gran mayoría de emprendimientos tienen como finalidad crecer, pero para poder lograr aquello las personas deben conocer dicha propuesta o emprendimiento, el desarrollo de este sistema web lograra solucionar dicho problema optando por aplicar metodologías ágiles para brindar un servicio de calidad.

En la presente investigación se detallará el proceso de diseño e implementación de lo que hemos denominado “VentGram” “Sistema de gestión de proyectos de emprendimiento”. Describiendo el análisis aplicado, las tecnologías utilizadas y las metodologías que se aplicaron en el desarrollo de la aplicación web y destacando su alcance, optando por optimizar el proceso de producción utilizando menos recurso y tiempo.

1.1. Planteamiento del problema

Hoy en día los sistemas informáticos o páginas web por lo general son vistas como una manera de inversión, ya sea a corto o largo plazo. Estamos en una época donde los proyectos de emprendimiento tienen un alto grado de incertidumbre, pero tiene mucho peso en diferentes ámbitos como lo social y económico. La puesta en marcha de un emprendimiento tiene como finalidad mejorar la vida de las personas; embarcándolo con ideas innovadoras que satisfagan a un público objetivo.

Cuando se quiere llevar a cabo un emprendimiento necesitan de algo que les permita dar a conocer sus productos o ideas, para que a su vez pueda generar nuevas fuentes de empleo y ayuda al desarrollo socioeconómico. La Universidad Estatal de Milagro cuenta con estudiantes con ideas y/o proyectos emprendedores los mismos que necesitan concretarse y poco a poco ir superando obstáculos que en su mayoría es el miedo al fracaso o desmotivación.

La Universidad Estatal de Milagro no cuenta con un sistema informático, que permita a los estudiantes publicar sus proyectos de emprendimiento tanto para la institución como a la comunidad en general, para poder conectar con personas emprendedoras o de igual interés que estén dispuestos a realizar o invertir en dicho proyecto.

1.2. Objetivos

1.2.1. Objetivo General

Desarrollar una aplicación web para gestionar los proyectos de emprendimientos en la facultad de ciencias de la ingeniería de la Universidad Estatal de Milagro.

1.2.2. Objetivos Específicos

- Definir los distintos requerimientos que permitan el desarrollo de la aplicación web
- Estructurar un modelo de base de datos para la aplicación web que ayudara a gestionar los proyectos de emprendimiento en la facultad de ciencias de la ingeniería.
- Diseñar la interfaz gráfica para la aplicación web.

1.3. Alcance

Esta investigación va dirigida a la influencia de las aplicaciones web y sistemas informáticos en los proyectos de emprendimiento e ideas innovadoras, el proceso que se lleva a cabo para su realización, destinado por la Universidad Estatal de Milagro a la comunidad en general.

El proyecto en resumen es poder difundir y dar a conocer a la comunidad y a los estudiantes sobre ideas o proyectos emprendedores mediante una aplicación web para que personas interesadas o futuros inversionistas formen parte de dicho proyecto y así poder aportar a la generación de empleo y al desarrollo socioeconómico de la comunidad.

La aplicación web tiene como meta ayudar tanto a emprendedores como a inversionistas a poder entablar una comunicación directa para poder emprender un negocio y poder impulsar los proyectos de emprendimientos de los estudiantes y profesionales que buscan financiamiento.

La implementación de la aplicación web fue en sus principios incentivada por la gran cantidad de demanda que existe en la creación de proyectos por los estudiantes y docentes pertenecientes a la facultad ciencia de la ingeniería, debido a que estos proyectos que tienen potencial, al no tener un financiamiento por ningún tipo de entidad que les permita emprender sus proyectos para colaborar con el desarrollo de la sociedad, en la figura 1 se puede observar un esquema de manera general del proyecto.



Figura 1. Esquema del proyecto

Fuente: Elaboración propia

1.4. Estado del arte

1.4.1. Base Legal

La Asamblea Nacional durante los días 7 de agosto y 5 de septiembre del 2019 realizo el primer debate del “PROYECTO DE LEY ORGANICA DE EMPRENDIMIENTO E INNOVACION” y el segundo debate, siendo esta última fecha aprobado el día 7 de enero del 2019. Tal proyecto el 8 de febrero de 2020 fue revisado y presentado al Presidente Constitucional de la República del Ecuador, según lo dicho en el art. 64 de la Ley Orgánica de la Función Legislativa y en el art. 138 de La Constitución de la República del Ecuador, la LEY ORGANICA DE EMPRENDIMIENTO E INNOVACION finalmente fue aprobada el 18 de febrero del 2020

En el artículo 1 que corresponde al Objeto y ámbito de la LEY ORGANICA DE EMPRENDIMIENTO E INNOVACION dice lo siguiente:

Establecer normativas que fomenten e incentiven el emprendimiento, el desarrollo tecnológico y la innovación, logrando así una cultura emprendedora, enfocándose en nuevas modalidades de financiamiento y societarias para lograr fortalecer el ámbito emprendedor. (Hugo del Pozo, 2020)

1.4.2. Marco Teórico

La aplicación web propuesta a desarrollar permitirá solucionar una problemática que se percibe constantemente en los estudiantes universitarios que buscan una forma de llevar a producción sus proyectos de emprendimiento que piensan que tiene potencial de crecimiento, pudiendo con esta aplicación dar a conocer al mundo sus proyectos y posiblemente comunicarse con interesados a contribuir al crecimiento del mismo.

Proyecto

Surgen de una planificación definida realizando previamente un conjunto de actividades sincronizadas una con la otra con la ayuda de un equipo de personas preparadas idóneamente para alcanzar uno o varios objetivos estipulados siguiendo una metodología para que determine los resultados y la realización de buenas prácticas.

Para iniciar un proyecto se necesita una idea que se la debe poner en marcha y materializarla, esta acción se considera como un plan que requiere recursos que dispone el grupo de personas incluidas en el proyecto. Para que se lleve a cabo dicho proyecto se necesita asignar

recursos como: tiempo, materia prima, dinero, recursos humanos, etc., y como principal objetivo de todo proyecto es de obtener resultados en un tiempo que se haya determinado.

Tipos de proyectos

- **Proyectos de emprendimiento:** Son proyectos que surgen desde cero, partiendo solo de una idea determinada de una o varias personas, siguiendo un proceso que requiere una inversión ya sea a corto o largo plazo para obtener beneficios económicos.
- **Proyectos sociales:** La puesta en marcha de este proyecto surge de la necesidad de mejorar la calidad de vida de determinadas personas, busca eliminar desigualdades para encontrar el bien común, una estabilidad y sostenibilidad, pero necesita de una gran participación y compromiso de los involucrados.
- **Proyectos de investigación:** Para realizar este tipo de proyectos se necesita frecuentemente una inversión que por lo general esta no va a tener una recuperación, porque es una investigación que genera tecnología que puede ser utilizada en algún momento. (Beltrán, 2011)

La Web

Se denomina un conjunto de documentos que están interconectados. La World Wide Web, o www funciona por enlaces de hipertextos que estén disponibles en la internet, estos se comunican por medio de la tecnología digital. El término “hipertexto” surge por la mezcla de textos, archivos y gráficos en un solo documento.

Web no significa internet, pero forma parte de un subconjunto de internet con lo que se puede acceder a través de un navegador. Lo que forma parte de Internet son los correos electrónicos, las redes sociales, páginas de información, etc, pero la web no forma parte de ello. (Latorre, 2018)

Sistemas informáticos

Un sistema informático se define como un conjunto de partes interrelacionadas y procesos que se diseñan para desempeñarse y cumplir un objetivo en particular. Un sistema informático ayudara en la coordinación e integración de los procesos y actividades, este puede combinar varias tecnologías permitiendo el procesamiento y almacenamiento de información. (Matilla & Chalmeta, 2007)

Sistemas de Información

Se puede definir como un conjunto de componentes para tomar de decisiones que permita el procesamiento de información, el control de operaciones, analizar determinados problemas y la creación de servicios y nuevos productos en la empresa. El objetivo principal de los sistemas de información es el procesamiento y almacenamiento de información para tener un producto o una salida. (Barchini, Sosa, & Herrera, 2012)

Aplicación Web

A lo largo de los años las aplicaciones web se van convirtiendo en sistemas complejos con interfaces de usuarios personalizadas y muy parecidas a aplicaciones desktop, brindando un servicio muy rápido y dinámico a la accesibilidad y respuesta a procesos de negocios de gran importancia. Las aplicaciones web buscan que la información pueda ser accedida desde cualquier parte del mundo, con respecto a cómo sea estructurada y programada. (Castejón, 2004)

Arquitectura de la Aplicación Web

Desde la perspectiva de la arquitectura de la aplicación web se refiere a dos partes, la primera parte es el cliente, se refiere a el usuario final que interactúa con la aplicación a través de un navegador web y la otra es el servidor, aquí se alojan todos los datos, métodos, reglas y funciones de la aplicación. (Posada & Cardenas, 2015)

A continuación, se puede observar en la siguiente figura 2, la arquitectura de la aplicación con sus distintas capas, la capa de datos que utiliza, como también la capa de aplicación.

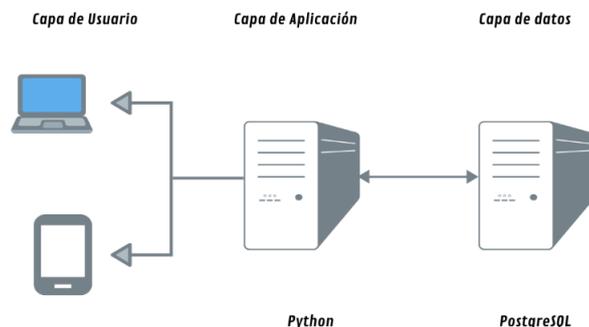


Figura 2. Arquitectura de la aplicación Web

Fuente: Elaboración propia

Metodologías

Cuando hablamos de estas metodologías se refiere a las metodologías para el desarrollo de proyectos informáticos.

Metodologías tradicionales

Esta metodología se la realiza por planeación, el desarrollo de un proyecto empieza con la investigación y comprensión de una manera rigurosa los requerimientos, mediante etapas como diseño y análisis de los mismos. La mala elicitación de los requerimientos puede ocasionar problemas y costos a lo largo de todo el proyecto y llevar al completo fracaso.

Cuando se emplea una metodología tradicional solo se piensa en un proyecto determinado, de gran tamaño y con una estructura bien definida. El proyecto se lo realiza siguiendo un proceso encaminado en una sola dirección y sin retroceso, este proceso no va a cambiar en el tiempo que dure la realización del proyecto; así mismo los requerimientos no pueden ser modificados y son acordados al inicio para todo el proyecto.

Metodologías Ágiles

Al momento de aplicar una metodología ágil a un proyecto es mucho más flexible su puesta en marcha, por lo que pueden ser modificados a la manera de cada equipo de trabajo. Esta metodología está diseñada para trabajar con varios proyectos al mismo tiempo, estos proyectos se los subdividen mediante características ordenadas, el trabajo de cada proyecto se lo trata de manera independiente.

Durante el desarrollo del proyecto es importante la constante comunicación con el cliente. La metodología ágil a comparación con la tradicional, los cambios son aceptados durante la realización del proyecto, es decir, los requerimientos pueden ser cambiados. Esta es una de las principales características de la metodología ágil, al igual que las entregas constantes de avances al cliente. (Navarro, Fernández, & Morales, 2013)

Importancia del análisis de requerimientos

A un requerimiento se lo puede describir como condición que debe cumplir un proyecto, ya sea porque el usuario así lo desee o porque se lo identifico al comienzo de un contrato o documento.

Tipos de requerimientos

Consisten en dos tipos de requerimientos, los cuales se pueden observar en la figura 3.

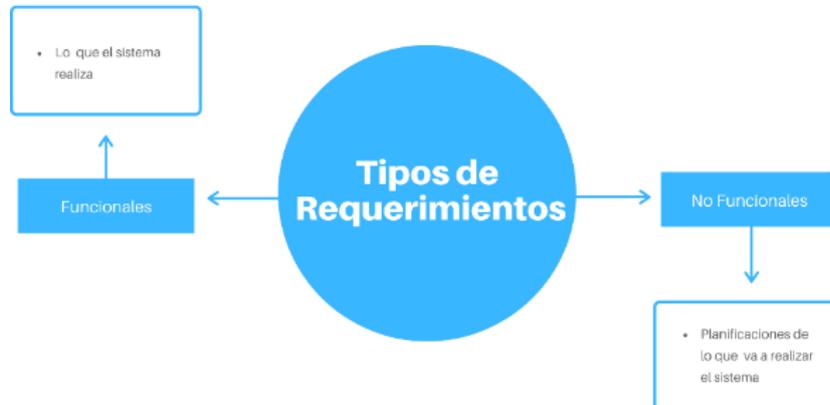


Figura 3. Tipos de requerimientos

Fuente: Elaboración propia

- **Requerimientos funcionales**

Estos requerimientos son los que delimitan las acciones que el sistema va a realizar, también especifican los cambios que realiza el sistema sobre las entradas para poder producir las salidas. Siempre es importante que primero se describa el “¿Qué?”. En el tiempo que avanza el proyecto de desarrollo de software estos requerimientos se convierten en algoritmos, lógica y una gran parte del código que tendrá el sistema.

- **Requerimientos no funcionales**

Los requerimientos no funcionales, por otra parte, son todos aquellos detalles que de alguna u otra manera puedan limitar o estancar el sistema, por ejemplo, el tiempo es un gran detalle que hay que planificarlo de una manera detallada, la seguridad, estándares, mantenimiento, etc. (Arias, 2005)

Framework

Un framework web se caracteriza por ser un tipo de estructura concreta de la cual se puede hacer uso de sus librerías, permitiendo la fácil creación de aplicaciones, siempre y cuando se respete la estructura que brinde esta tecnología (Jimmy, Mora, Mariuxi, & Erika, 2016) . En si es un conjunto de bibliotecas que permiten seguir un orden para crear una aplicación web funcional respetando todos los estándares de desarrollo.

Arquitectura de frameworks

La arquitectura que trabajan los frameworks se enlaza directamente con el Modelo Vista Controlador (MVC) ya que este modelo permite la división de trabajo y poder de esta forma trabajar de forma independiente en cada área y posteriormente lograr juntarlas sin ningún error de compatibilidad. (Jimmy, Mora, Mariuxi, & Erika, 2016)

El patrón de diseño suele ser el predominante en varios tipos de frameworks puesto que este contiene múltiples ventajas en sus patrones, debido a que este separa tanto el modelo que en si son los componentes que interactúan con el servidor y con los datos del usuario, la vista que se encarga de la interacción directa con el cliente siendo este el diseño que logra ver el usuario y posteriormente se comunican entre sí por medio del controlador el cual se encarga del procesamiento y comportamiento de acuerdo a las peticiones. (Nilsen, Oscar, & Jhonnathan, 2016)

Se puede programar en cada una de estos frameworks de forma independiente y luego el controlador que va ser el encargado de hacer la conexión de estos dos, brindándonos más seguridad a las aplicaciones que se estén desarrollando, como se puede observar a continuación en la figura 4.

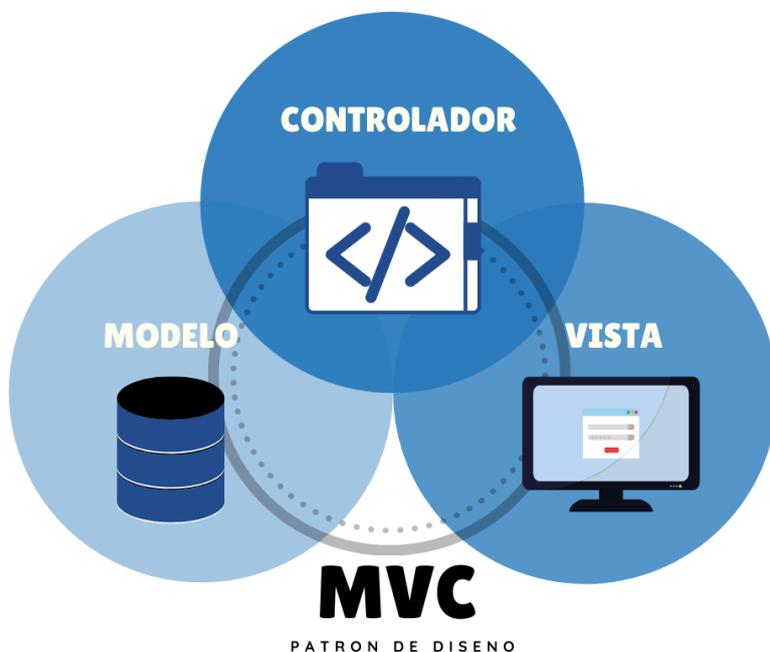


Figura 4. Modelo Vista Controlador

Fuente: Elaboración propia

Framework Django

Django es parte de los frameworks de desarrollo Web el cual está escrito puramente en el lenguaje de programación Python, su principal objetivo es el ahorro de recurso y tiempo, dando paso a la reutilización de sus apps y módulos que este contiene, permitiendo a los desarrolladores concentrarse en la lógica de negocio, respetando los parámetros de la modelo vista controlador (MVC). (Saul, 2015)

Los parámetros que utiliza Django son: Modelo Vista Template (MTV) parten del mismo principio de modelo vista controlador solo que las vistas vienen a ocupar el papel de controlador y el template ocupa el papel de vista, pero utilizan la misma arquitectura del framework siendo igual de eficientes sin mucha diferencia, como se puede observar en la figura 5.

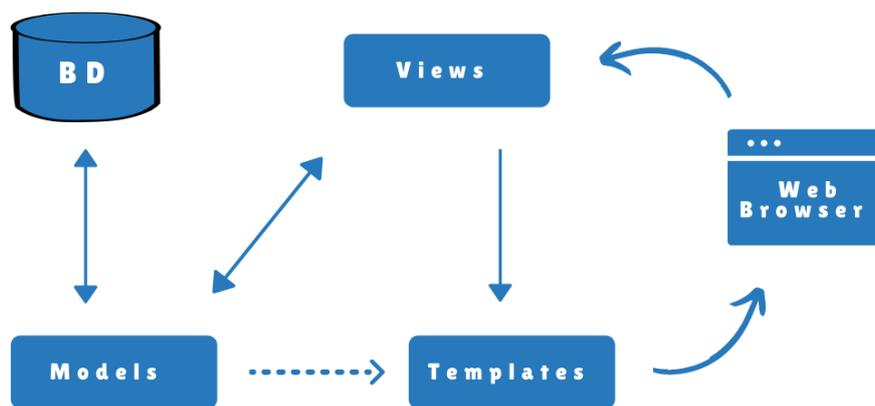


Figura 5. Modelo MTV

Fuente: Elaboración propia

Django busca optimizar o reutilizar al máximo los códigos generados por los programadores por lo que brinda muchas bibliotecas o apps genéricas que pueden ser editadas y adaptadas a la necesidad del desarrollador buscando incentivar la reutilización de código, siguiendo los estándares de calidad y la estructura básica de un desarrollo web.

Bootstrap

Bootstrap es un framework conformado por archivos CSS, archivos con Tipos de letra (Fonts) y archivos Javascript que permiten la creación de aplicaciones web responsivos e

interactivos que se adaptan a los distintos tamaños de dispositivos de una manera fácil. (Abigail & Miguel, 2018)

Nos brindan múltiples clases de diseño de interfaces que pueden ser utilizadas o editadas por medio de un archivo CSS permitiendo en si el uso de estas clases de forma ilimitada por su facilidad de edición, brindándonos un sistema de grilla constituida por 12 columnas que se adaptan automáticamente al tamaño del dispositivo, este sistema de grilla nos reduce el tiempo de desarrollo de frontend con estilos agradables y fáciles de implementar en nuestro proyecto.

Editor de código fuente

Los editores de código fuente son editores de texto plano dedicados al uso exclusivo de códigos de programación en su sintaxis adecuados de tal manera que facilitan al programador comodidad al momento de crear su desarrollo web.

Visual Studio Code

Visual Studio Code es un editor de código fuente que se encarga de mezclar lo simple de un editor y la experiencia que este ofrece, agregando nuevas funcionalidades que otros editores no cuentan como lo es la compilación-depuración. Puede trabajar tanto en OS X, Linux y Windows. (Tobias & Kay, 2016)

En si este editor de código fuente ofrece muchas funcionalidades respecto otro tipo de editores permitiendo integrar múltiples tecnologías en un solo tipo de editor ayudando al desarrollador de una forma acertada, brinda rapidez y precisión en sus múltiples plugin y extensiones que se adaptan a la necesidad del desarrollador.

Back-end

Back-end es parte del desarrollo web encargado del procesamiento de los datos de entrada desde el front-end, siendo este el controlador de la vista y realizando la conexión con la base de datos. (Nieto Rodrigo, 2016)

Está orientado en la parte del servidor, las aplicaciones y la forma en cómo se manejarán los datos del cliente, aquí predomina la base de datos y los SGBD (Sistema Gestores De Base de Datos) de los cuales pueden ser MySQL, Oracle, PostgreSQL, entre otros y por otro lado los lenguajes que se encargan de escribir la comunicación entre el front-end y el back-end

pueden ser Go, Python y Php, los lenguajes que se encargan de los datos pueden ser SQLServer, DDL entre otros.

Para la presente investigación se utilizará Python como principal lenguaje para la interacción con el servidor, SQL Server como lenguaje de tratamientos de datos y PostgreSQL como gestor de base de datos.

Base de datos relacional

Es aquella que está compuesta por un conjunto de datos almacenados y relacionados entre sí, permitiendo manipular los datos de una empresa, siendo su principal función el de mantener la seguridad e integridad de los datos ante cualquier incidente. (Jorge & Cyndi, 2015)

Una base de datos es un conjunto de datos que se encuentran almacenados de forma ordenada para luego poder ser utilizados por los mismos usuarios o por la entidad que resguarde estos datos.

PostgreSQL

PostgreSQL es un sistema gestor de base de datos el cual toma como referencia el modelo relacional de base de datos otorgando adicionalmente dos definiciones, tipos y funciones para que los desarrolladores puedan expandir fácilmente el sistema, brindando una escalabilidad más sencilla. (Thomas, 1996)

Permitiendo de esta manera la potenciación del modelado de base de datos posicionándolo en las categorías de base de datos conocidos como objeto-relacional.

Python

Python es un lenguaje de programación de alto nivel orientado a objetos, su sintaxis es considerada una sintaxis más legible que de otros lenguajes, siendo esta elegante y sencilla de comprender permitiendo a los programadores escribir código más entendible al lenguaje natural. (Andrés & Isabel, 2014)

Este lenguaje al ser orientados a objetos nos permite hacer uso de las ventajas que nos proporcionan herencia de tablas al momento de hacer nuestro modelado, también soluciona distintos tipos de errores que los compiladores pasan por alto, informado la detección de estos y dando solución a los mismos.

Front-end

El diseño de software front-end es la parte del software que interactúa con los usuarios. (Nieto Rodrigo, 2016) Proporcionándole la parte gráfica que el usuario interactúa para que posteriormente pueda realizar peticiones de información al back-end.

HTML

Html es considerado un lenguaje de etiquetas el cual otorga tres elementos al momento de diseñar un entorno web de los cuales son su estilo, funcionalidad y estructura estos tres elementos funcionaran en conjunto para el diseño de la aplicación web. (Gauchat, 2017)

Permite desarrollar la estructura del diseño que va a tener una aplicación web al momento de interactuar con el usuario dotando al desarrollador de múltiples funcionalidades de etiquetado para poder crear el entorno web.

CSS

CSS es la parte que trabaja conjuntamente con HTML el cual nos permite brindar estilos a la aplicación web, para que de esta forma no se vea rustico o anticuado nuestro sitio. brindándonos múltiples funcionalidades ya sea para crear nuestras propias clases de estilos o editar las que ya tiene HTML. (Gauchat, 2017)

Permitiendo la edición y diseño de nuestra aplicación web más dinámica y responsive, permitiendo crear estilos personalizados a modo de decoración del sitio e incluso validar campos que se necesiten de parte del back-end.

Java Script

JavaScript es un lenguaje de programación interpretado el cual no requiere de compilado por lo que trabaja directamente con los navegadores. (Gauchat, 2017)

Permiten interactuar con otros lenguajes, almacenar variables del lado del cliente y hacer cálculos sin necesidad de interactuar directamente con el back-end.

Control de versiones

Un sistema de control de versiones es aquel sistema que se encarga de gestionar los diversos cambios que se implementan dentro de un proyecto o producto.

Git bash

Git bash es un sistema de control de versiones orientada a los desarrolladores de software permitiendo a este navegar entre las distintas versiones que se hayan creado de este proyecto, dando a los desarrolladores una herramienta muy potente capaz de retroceder en el tiempo a su versión funcional o si desea ver algún cambio específico.

Github

GitHub es la nube y complemento de Git Bash permitiendo compartir, alojar, colaborar, administrar y revisar códigos de proyectos por la comunidad permitiendo un trabajo en equipo mucho más satisfactorio debido a que todos sus historiales de avances quedan registrados y estos posteriormente pueden ser revisados y analizados. (Leonardo, 2019)

Github se conecta con git bash y permite la interacción entre las versiones creadas en escritorio y las que se crearon en la plataforma de github, permitiendo juntar estas y crear grupos de trabajos más productivos, promocionando el trabajo en equipo y la participación de múltiples colaboradores que se interesen en el proyecto.

1.4.3. Marco Referencial

El emprendimiento ha sido y será crucial para el crecimiento económico de una nación y a su vez mejorando la calidad de vida, siendo una alternativa para generar empleo y combatir la desigualdad y la pobreza. Sin embargo, en los países latinoamericanos este concepto no se aplica debido a que los emprendimientos son tomados como algo esencial para crear empleos por la falta de puestos de trabajos y su consecuencia es que se crean negocios informales que no cuentan con un potencial de crecimiento y al pasar el tiempo estos desaparecen. (María Guillermina & Priscila Soledad, 2019)

En la Universidad Técnica de Ambato se realizó un sistema de informática para emprendimientos en la facultad de ciencia e ingeniería en alimentos, teniendo como autor a Daniel Isaías Paucar Quile, quien implemento su sistema en dicha facultad y teniendo resultados satisfactorios en la adaptabilidad e interacción del sistema, permitiendo a los estudiantes la publicación y gestión de proyectos de emprendimiento. (Daniel Isaias, 2019)

Existen casos de éxito de escuelas virtuales quienes permiten a sus estudiantes la publicación de sus proyectos desarrollados como es el caso de Platzi y Domestica quienes ofrecen cursos

virtuales y adicional a eso permiten que sus estudiantes publiquen sus aplicaciones, permitiendo de esta forma crear una comunidad en la cual puedan colaborar con los distintos proyectos que tengan potencial y posteriormente lanzarlos a producción.

En la actualidad el tener un proyecto con potencial es señal de que se puede fundar un negocio partiendo de sus bases. Ya sea este proyecto de índole social, investigativo o uno de emprendimiento, pudiendo generar empleos si este proyecto tiene éxito. La tecnología hoy en día es de mucha ayuda para enlazar a varios colaboradores con distintos tipos de visión para llegar a un acuerdo y montar una nueva empresa con potencial de crecimiento

CAPÍTULO 2

2. METODOLOGÍA

El sistema informático que se desarrollará permitirá gestionar proyectos de emprendimientos e informáticos de los estudiantes de la Facultad Ciencias de la Ingeniería, teniendo como principal objetivo la comunicación entre emprendedores e interesados al proyecto publicado en la plataforma, fomentando el desarrollo económico y social de los estudiantes.

El tipo de investigación que se utilizó para el desarrollo del proyecto es la documental, esta modalidad se la considero porque gran parte de la información obtenida se obtuvo de libros, tesis desarrolladas, artículos científicos para poder profundizar los temas acordes a nuestro proyecto de investigación. También se considera la investigación aplicada porque se realizó una investigación profunda para poder conocer los detalles del problema y así obtener resultados para el desarrollo del proyecto.

La metodología que se planteó a utilizar y la que mejor se adapta para el desarrollo del sistema web es la metodología “Cascada”.

La modelo cascada fue considerado el primer modelo notorio en la industria del software. Una de las principales características de este modelo metodológico es que permite la identificación de etapas muy específicas, manejándose de forma lineal. Estas etapas deben cumplirse de manera ordenada a medida que se vayan realizando respectivas modificaciones y pruebas para poder disminuir errores que se puedan presentar, este proceso es muy analítico, sistemático, técnico y disciplinado en el transcurso del desarrollo de la aplicación. Este proceso se lo representa de una manera gráfica en la siguiente figura 6. (González, Calero, & Loaiza, 2019)

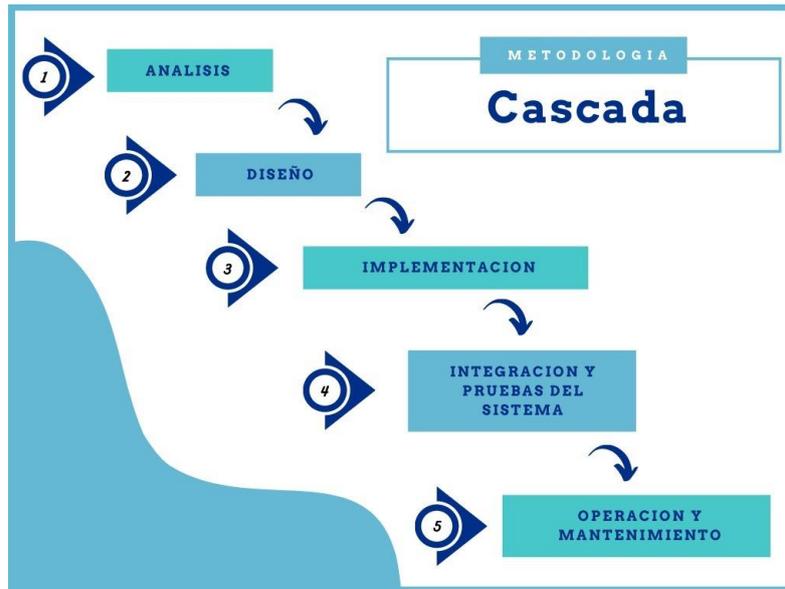


Figura 6. Etapas del modelo Cascada

Fuente: Elaboración propia

Etapas del modelo Cascada

2.1. Fase 1: Análisis y definición de requisitos

Es la primera fase de modelo cascada el cual consiste en dialogar con los usuarios pertinentes, con el objetivo de plantear restricciones, metas, y servicios de la aplicación a desarrollar. De los cuales estos elementos son detallados posteriormente para utilizarlo como especificaciones del software que se va implementar en el desarrollo. Como último punto se agrega el análisis de la definición de requisitos, permitiendo que los problemas más complicados sean divididos en varias secciones, creando las distintas estrategias para las respectivas soluciones.

En esta etapa se trabajó en conjunto con los estudiantes y docentes de la facultad ciencias de la ingeniería, analizando y especificando cada uno de los requisitos que se tendrán en cuenta en el desarrollo de la aplicación, llegando al planteamiento de los siguientes requisitos.

2.1.1. Requerimientos funcionales

Se detalla lo que va a realizar el sistema:

- RF1-La aplicación permitirá verificar el correo electrónico cuando se registre un nuevo usuario enviando un token único de validación al correo del nuevo usuario.

- RF2-Se permitirá el manejo de distintos tipos de perfil de usuario en el cual se podrá describir una autobiografía y el objetivo del usuario.
- RF3-La aplicación permitirá al usuario la publicación de proyectos de emprendimiento en la plataforma ligándolo a uno o varios usuarios, dependiendo de los autores.
- RF4-Se permitirá navegar en la aplicación con una cuenta de usuario o sin ella permitiendo revisar los distintos proyectos de emprendimientos publicados.
- RF5-La aplicación contara con una interfaz de usuario y administrador. Permitiendo al administrador la gestión de los usuarios activos, las publicaciones que se realicen, la organización de los proyectos y el registro de información requerida.

2.1.2 Requerimientos no funcionales

Se especifican las propiedades del sistema, es decir, como se las realizan:

- RNF1-Todo tipo de funcionalidad será capaz de responder en menos de 5 segundos
- RNF2-Los datos modificados en la base de datos tienen que actualizarse para todos los usuarios en menos de 5 segundos.
- RNF3-La aplicación debe de estar protegida por acceso no autorizado.
- RNF4-La comunicación entre servidor, aplicación y cliente tienen que estar encriptadas utilizando el algoritmo CSRF de Django.
- RNF5-La aplicación web debe de contar un diseño responsive con el fin de permitir la adecuada visualización en distintos tipos de computadoras personales, dispositivos móviles y tabletas.
- RNF6-La aplicación tiene que contener interfaces graficas bien estructuradas

2.2 Fase 2: Diseño

Esta es la segunda etapa de la metodología, aquí se trata de utilizar toda la información evaluada en la primera etapa, de acuerdo con los requerimientos considerados se definen claramente los puntos fuertes para la estructura del diseño de un modelo base para el desarrollo de la aplicación. Siguiendo todos los parámetros establecidos para una construcción optima de una base de datos relacional.

En esta etapa de diseño se comienza a detallar y se centran los componentes concretos, como lo es el entorno de trabajo a utilizar, las interfaces de usuario y la arquitectura del software. Esta fase da una idea inicial de cómo será la aplicación.

2.3 Fase 3: Implementación

El diseño de la arquitectura del software se pone en marcha en esta etapa de implementación, prácticamente se incluye la programación de la aplicación partiendo del modelo lógico establecido.

Se comenzó a desarrollar la aplicación web con el lenguaje de programación Python y su framework Django, se empezó a codificar el modelo, se creó sus distintos módulos que daban solución a problemas específicos del sistema permitiendo la reutilización de estos módulos en varias secciones del software, por último se realizó las distintas pruebas unitarias a cada módulo para detectar imperfecciones o errores de codificación permitiendo la reestructuración del mismo antes de ir a la siguiente etapa.

2.4 Fase 4: Integración y pruebas del sistema

Una vez que se concluye la etapa de implementación, se realizó una prueba beta de la aplicación a estudiantes de la Facultad Ciencias de la Ingeniería, esta prueba incluye el manejo de todos los módulos e interacciones, para identificar posibles errores y así realizar potenciales mejoras para que la aplicación cumpla con su correcto funcionamiento. Esta etapa permite determinar toda la información sobre la calidad del software.

2.5 Fase 5: Operación y mantenimiento

En esta última etapa es necesario migrar todos los datos para poner en marcha la aplicación web. También se provee asistencia técnica para un manejo adecuado de la aplicación. Una vez puesto en producción, al momento de realizar sus respectivos manteamientos y mejoras al proyecto se lo realiza con procesos, métodos y herramientas adecuadas, según el desempeño del sistema y exigencias de los usuarios.

CAPÍTULO 3

3. PROPUESTA DE SOLUCIÓN

Se realizará la programación de la aplicación web para la gestión de proyectos informáticos o de emprendimientos. Partiendo de la necesidad de superación de muchos estudiantes, esta aplicación es la intermediaria para poder abrir puertas y dar a conocer a más personas sobre su visión de negocio o propuestas innovadoras de desarrollo.

Gracias a esta herramienta los usuarios podrán publicar sus ideas o proyectos teniendo el control de su información, en cualquier lugar donde se encuentren, no solo podrán contactar con personas del alrededor, sino también de otras partes, haciendo una aplicación muy útil para las personas y más que todo para estudiantes.

3.1. Descripción del producto

Desarrollo de una aplicación web para gestionar los proyectos de emprendimientos en la Facultad Ciencia de la Ingeniería de la Universidad Estatal de Milagro.

3.1.1. Perspectiva del producto

En la siguiente figura 7 se aprecia las funciones principales que realizara la aplicación, teniendo como entidades primordiales al módulo usuario, perfil, proyecto e interacciones. Siendo estas los pilares de la aplicación web.

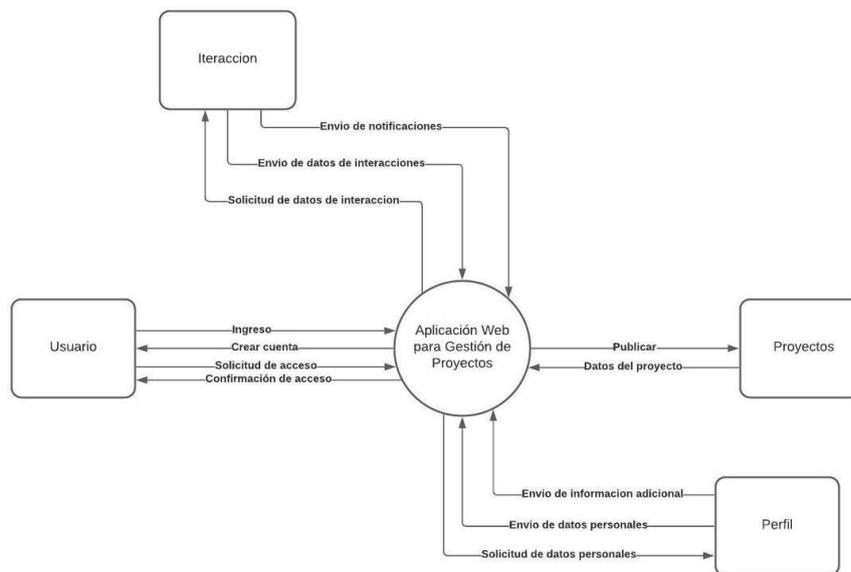


Figura 7. Diagrama contextual aplicación web para gestionar los proyectos de emprendimientos

Fuente: Elaboración propia

3.1.2. Funciones del producto

La aplicación web para gestionar los proyectos de emprendimientos en la Facultad Ciencia de la Ingeniería de la Universidad Estatal de Milagro permitirá realizar las siguientes funciones.

Id	Historias de Usuarios	Objetivos	Criterios de Aceptación	Tiempo	Rol
1	Modulo Usuario	Ingresar al sistema por medio de un correo y contraseña	<ul style="list-style-type: none"> • Crear usuarios • Ingresar con correo y contraseña 	15 días	<ul style="list-style-type: none"> • Usuario • Administrador
2	Modulo Perfil	Gestionar perfil de usuario	<ul style="list-style-type: none"> • Buscar usuarios • Ver mi perfil • Ver perfil de otros usuarios • Editar información personal 	15 días	<ul style="list-style-type: none"> • Usuario
3	Modulo Proyecto	Gestionar proyectos personales	<ul style="list-style-type: none"> • Publicar proyectos • Editar proyecto • Visualizar proyectos • Buscar proyectos 	20 días	<ul style="list-style-type: none"> • Usuario
4	Modulo Interacción	Interactuar con usuarios y proyectos	<ul style="list-style-type: none"> • Generar un me gusta en proyectos • Comentar Proyectos • Seguir Usuarios • Chat Privado • Notificaciones 	20 días	<ul style="list-style-type: none"> • Usuarios

Tabla 1. Requerimientos para la construcción del sistema web

Fuente: Elaboración Propia

Características del usuario

La aplicación web dispondrá de los perfiles de usuario: administrador, usuario.

3.1.3. Perfil de Usuario

- **Administrador de la aplicación:** En este perfil se tendrá acceso total a la aplicación, permitiendo gestionar los usuarios registrados, gestión de proyectos publicados, Gestión de comentarios, y mensajería.

- **Usuario:** En este perfil solo se tendrá acceso a los datos del usuario, permitiendo la publicación de proyectos, interacción con otros proyectos publicados y con otros usuarios en la aplicación.

3.1.4. Jerarquía de usuario



Tabla 2. Jerarquía funcional de la Aplicación web de gestor de proyectos de emprendimiento

Fuente: Elaboración Propia

3.2. Requerimientos funcionales del sistema



Figura 8. Diagrama modular

Fuente: Elaboración Propia

3.3. Proceso del método cascada

En el desarrollo de la aplicación web, se implementó la metodología cascada, debido a que esta permite tener resultados específicos, con un desarrollo ordenado y lineal, permitiendo realizar modificaciones y pruebas para disminuir errores que se puedan presentar en cada etapa, presentando resultados progresivos en el desarrollo.

3.3.1. Caracterización de casos de uso

3.3.1.1. Módulo Usuario

Módulo Usuario		
<i>Objetivos asociados</i>	Ingresar al sistema por medio de un correo y contraseña	
<i>Descripción</i>	La aplicación permitirá la creación de una cuenta con el rol de usuario automáticamente y su acceso.	
<i>Precondición</i>	Disponer de correo electrónico.	
<i>Secuencia Normal</i>	Paso	Acción
	1	El usuario creara una cuenta con sus datos personales.
	2	El sistema verificará sus datos y procederá a enviar un código único a su correo electrónico, para activar la cuenta.
	3	El usuario ingresara a su cuenta, con su correo y contraseña
<i>Postcondición</i>	ninguna	
<i>Excepciones</i>	Paso	Acción
	1	En caso de no ingresar el código de activación enviado al correo, la cuenta no será habilitada.
<i>Rendimiento</i>	Paso	Cota de tiempo
	2	5 segundos
<i>Frecuencia esperada</i>	70 veces por día.	
<i>Estabilidad</i>	Alta	
<i>Comentarios</i>	Sin comentarios	

Tabla 3. Caracterización del módulo Usuario

Fuente: Elaboración Propia

3.3.1.2. Módulo Perfil

Módulo de Perfil		
<i>Objetivos asociados</i>	Gestionar perfil de usuario	
<i>Descripción</i>	La aplicación permitirá la personalización de la cuenta de usuario.	
<i>Precondición</i>	ninguna	
<i>Secuencia Normal</i>	Paso	Acción
	1	El usuario iniciara sesión en su cuenta con su correo y contraseña
	2	El usuario puede ingresar información adicional a su perfil.
	3	El usuario podrá buscar otros usuarios registrados.
<i>Postcondición</i>	ninguna	
<i>Excepciones</i>	Paso	Acción
	1	Sin Excepciones
<i>Rendimiento</i>	Paso	Cota de tiempo

	1	5 segundos
<i>Frecuencia esperada</i>	50 veces por día	
<i>Estabilidad</i>	Alta	
<i>Comentarios</i>	Sin Comentarios	

Tabla 4. Caracterización del módulo Perfil

Fuente: Elaboración Propia

3.3.1.3. Módulo Proyecto

Módulo Proyecto		
<i>Objetivos asociados</i>	Gestionar proyectos personales	
<i>Descripción</i>	La aplicación permitirá la publicación de proyectos de emprendimiento.	
<i>Precondición</i>	ninguno	
<i>Secuencia Normal</i>	Paso	Acción
	1	El usuario iniciara sesión en su cuenta con su correo y contraseña
	2	Ingresar todos los datos requeridos por la aplicación del proyecto a publicar.
	3	Publicar proyecto.
	4	Visualizar proyecto publicado.
	5	Editar proyecto.
<i>Postcondición</i>	ninguna	
<i>Excepciones</i>	Paso	Acción
	1	Si la imagen subida en la cabecera no cumple con las dimensiones recomendadas, esta se distorsionará. Mostrando una cabecera poco aceptable.
	2	Si el archivo seleccionado no es de tipo PDF este no se guardará.
<i>Rendimiento</i>	Paso	Cota de tiempo
	3	5 segundos
<i>Frecuencia esperada</i>	35 veces por día	
<i>Estabilidad</i>	Alta	
<i>Comentarios</i>	Sin Comentarios	

Tabla 5. Caracterización del módulo Proyecto

Fuente: Elaboración Propia

3.3.1.4. Módulo Interacción

Módulo Interacción	
<i>Objetivos asociados</i>	Interactuar con usuarios y proyectos

<i>Descripción</i>	La aplicación permitirá la interacción de todos los usuarios registrados entre sí y con los proyectos publicados.	
<i>Precondición</i>	ninguna	
<i>Secuencia Normal</i>	Paso	Acción
	1	El usuario iniciara sesión en su cuenta con su correo y contraseña
	2	El usuario puede contactar con otros usuarios.
	3	El usuario puede colocar dar “likes” a proyectos.
	4	El usuario puede comentar los proyectos.
<i>Postcondición</i>	ninguna	
<i>Excepciones</i>	Paso	Acción
	1	ninguna
<i>Rendimiento</i>	Paso	Cota de tiempo
	4	5 segundos
<i>Frecuencia esperada</i>	100 veces por día	
<i>Estabilidad</i>	Alta	
<i>Comentarios</i>	Sin Comentarios	

Tabla 6. Caracterización del módulo Interacción

Fuente: Elaboración Propia

3.3.2. Diagramas de caso de uso

En la realización de los siguientes diagramas de casos de uso se tomó toda la información de los requerimientos anteriores para especificar toda la comunicación y comportamiento que tendrá el sistema, especificando sus relaciones y elementos.

3.3.2.1. Diagrama de caso de uso: Usuario

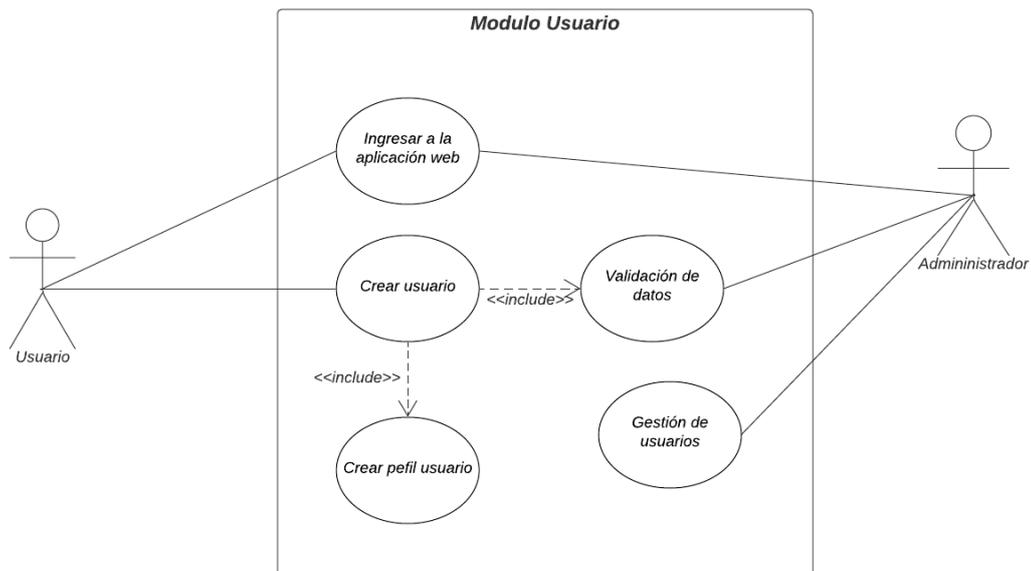


Figura 9. Caso de Uso - Módulo Usuario

Fuente: Elaboración Propia

En el módulo usuario tiene acceso tanto el administrador como lo es el usuario, permitiéndolo ingresar a la aplicación web o un crear un nuevo usuario. El administrador gestiona a los usuarios y valida los datos como se observa en la figura 9.

3.3.2.2. Diagrama de caso de uso: Perfil

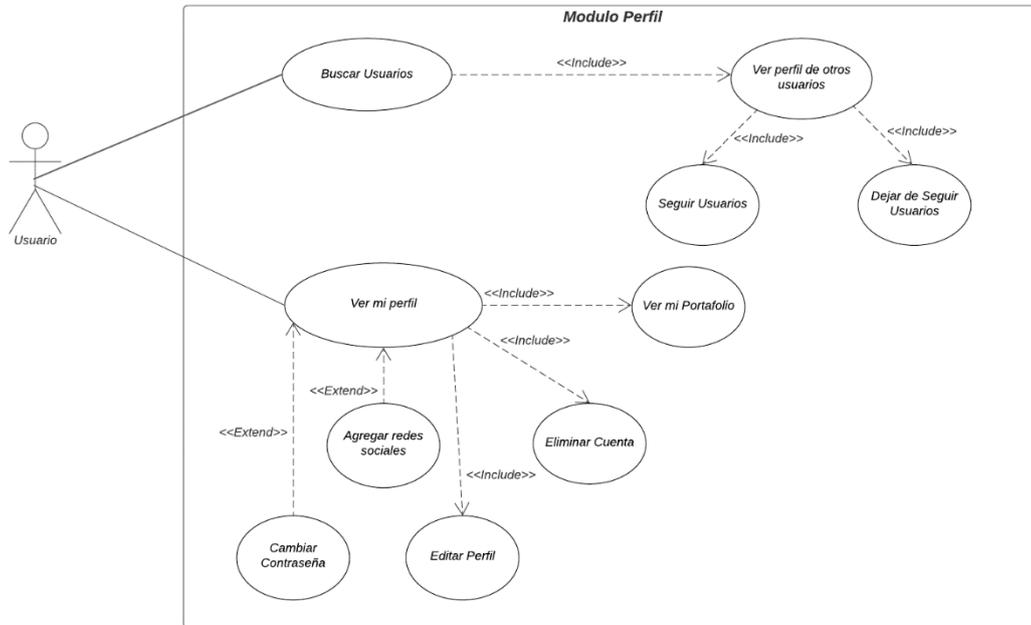


Figura 10. Caso de Uso - Módulo Perfil

Fuente: Elaboración Propia

En el diagrama que se observa en la figura 10 el usuario puede ver su perfil, buscar amigos, ver perfiles de otros usuarios, seguir y dejar de seguir usuarios, etc. Este módulo solo el usuario puede editar y manipular su información.

3.3.2.3. Diagrama de caso de uso: Proyecto

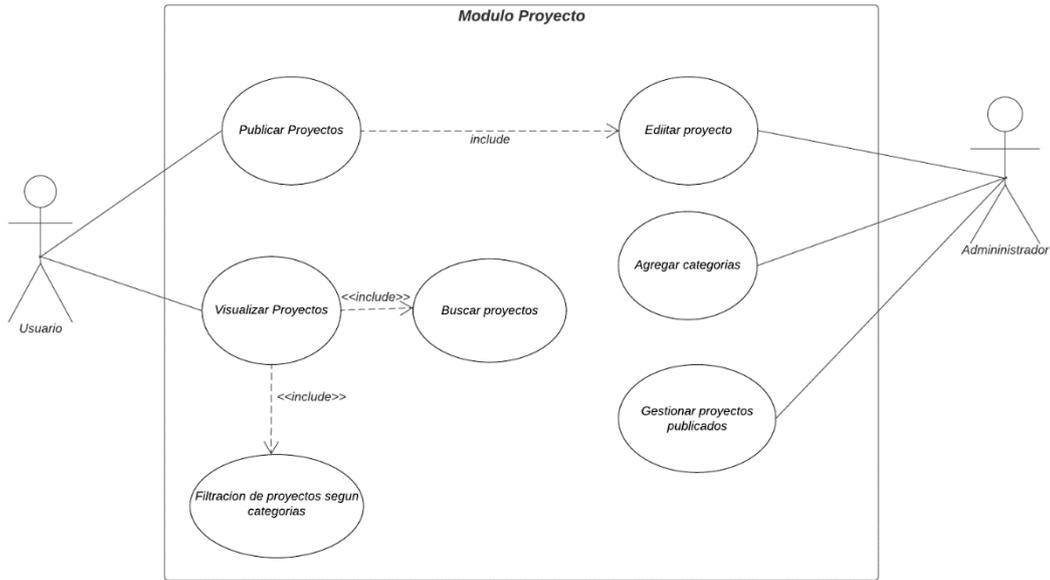


Figura 11. Caso de Uso - Módulo Proyecto

Fuente: Elaboración Propia

El usuario puede publicar proyectos, visualizarlos y buscar otros proyectos. En cambio, en administrador agrega las categorías para los proyectos, también los puede editar y se encarga de la gestión de los proyectos publicados tal como se observa en la figura 11.

3.2.4.2 Diagrama de caso de uso: Interacción

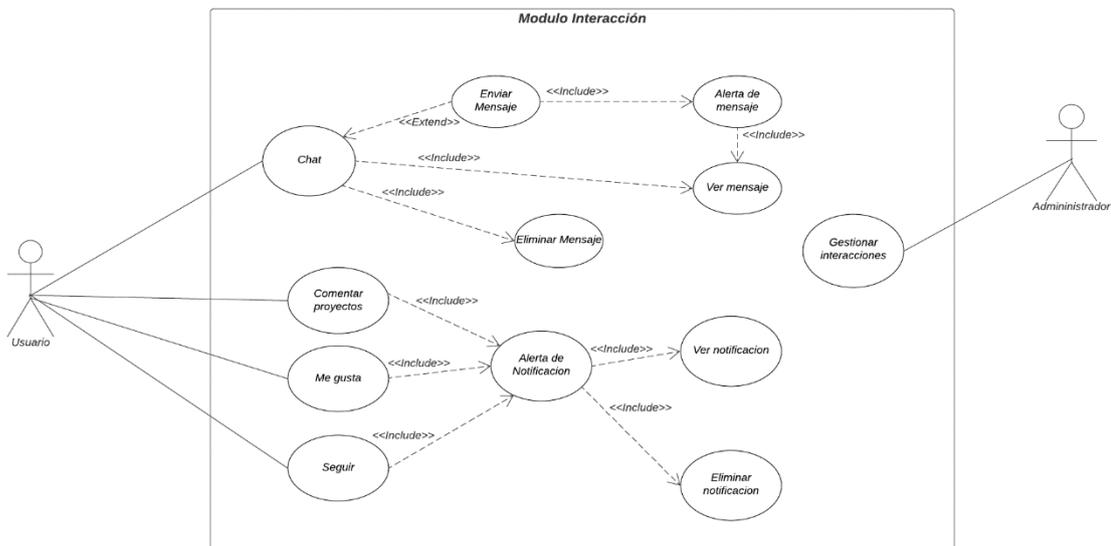


Figura 12. Caso de Uso - Módulo Interacción

Fuente: Elaboración Propia

En el módulo interacción el usuario es el único responsable de llevar a cabo los chats, comentar proyectos, dar likes y seguir a otros usuarios. También puede ver notificaciones en su cuenta y poder eliminarlas. El administrador solo puede gestionar las interacciones de los usuarios como se observa en la figura 12.

3.4. Requerimientos No Funcionales del sistema

3.4.1. Requerimientos de tiempo de respuesta de la aplicación

Tiempo de respuesta de todas las funcionalidades		
Versión	1.0	25-05-2021
Autores	Jose Guachun, Jhoyder López	
Descripción	Todo tipo de funcionalidad será capaz de responder en menos de 5 segundos	
Prioridad	Alta	
Estado	En Construcción	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 7. Requerimientos de tiempo de respuesta

Fuente: Elaboración Propia

3.4.2. Requerimiento de seguridad de acceso

Seguridad de acceso		
Versión	1.0	25-05-2021
Autores	Jose Guachun, Jhoyder López	
Descripción	La aplicación debe de estar protegida por acceso no autorizado.	
Prioridad	Alta	
Estado	En Construcción	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 8. Requerimientos de seguridad de acceso

Fuente: Elaboración Propia

3.4.3. Requerimiento de seguridad de datos

Seguridad de datos		
Versión	1.0	25-05-2021
Autores	Jose Guachun, Jhoyder López	
Descripción	La comunicación entre servidor, aplicación y cliente tienen que estar encriptadas utilizando el algoritmo CSRF de Django.	
Prioridad	Alta	
Estado	En Construcción	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 9. Requerimiento de seguridad de datos

Fuente: Elaboración Propia

3.4.4. Requerimiento de diseño adaptable a diferentes dispositivos

Diseño adaptable a diferentes dispositivos.		
Versión	1.0	25-05-2021
Autores	Jose Guachun, Jhoyder López	
Descripción	La aplicación web debe de contar un diseño responsive con el fin de permitir la adecuada visualización en distintos tipos de computadoras personales, dispositivos móviles y tabletas.	
Prioridad	Alta	
Estado	En Construcción	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 10. Requerimiento de diseño adaptable a diferentes dispositivos

Fuente: Elaboración Propia

3.4.5. Requerimiento estructura de interfaz gráfica

Estructura de interfaz gráfica.		
Versión	1.0	25-05-2021
Autores	Jose Guachun, Jhoyder López	
Descripción	La aplicación tiene que contener interfaces graficas bien estructuradas.	
Prioridad	Alta	
Estado	En Construcción	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 11. Requerimiento estructura de interfaz gráfica

Fuente: Elaboración Propia

3.5. Diagrama de dominio

3.5.1. Modelo Entidad Relación E- R

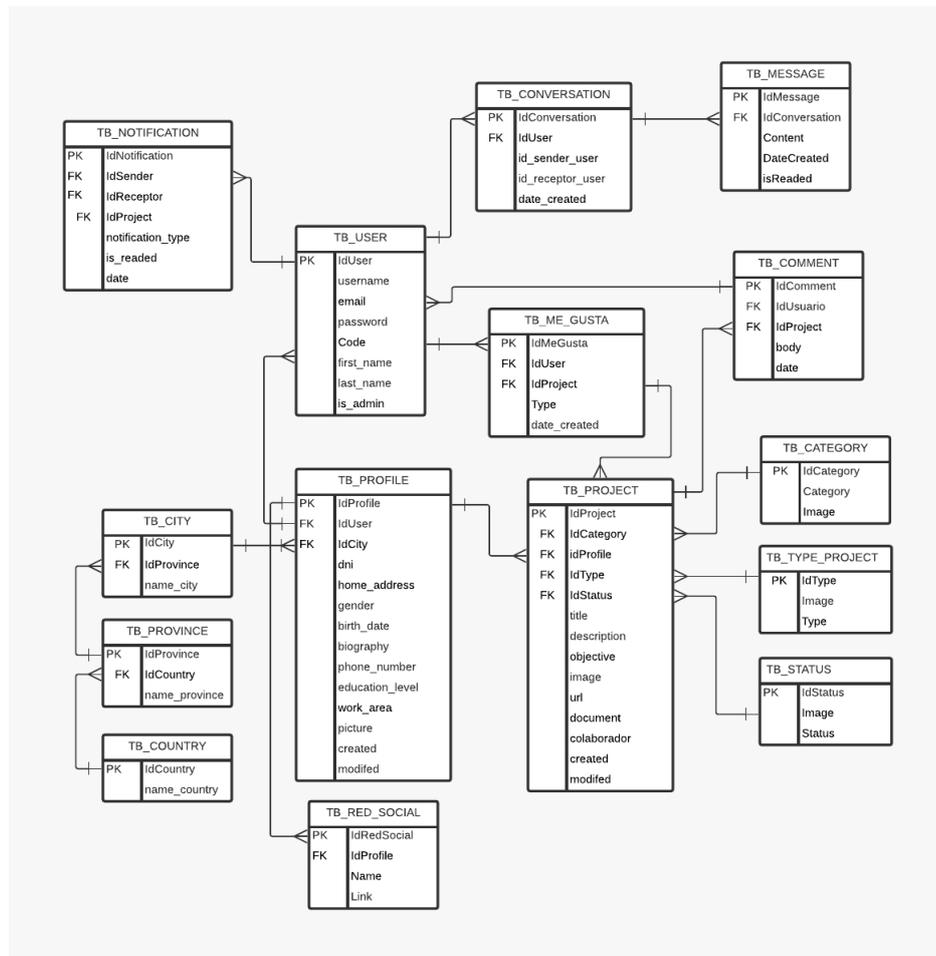


Figura 13. Modelo Entidad Relación de la aplicación

Fuente: Elaboración Propia

El diseño del modelo de base de datos consta de varias clases que permiten el crecimiento con el paso del tiempo de manera escalable, las mismas que se las detallan a continuación en las siguientes tablas:

Tabla de Usuario

Atributos	Tipo de dato	Tipo	Descripción
idUser	Integer	PK	Clave primaria que identifica al Usuario
username	Varchar	E	Usuario para registro
email	Varchar	E	Correo de usuario
password	Varchar	E	Contraseña de usuario
code	Varchar	E	Código generado para recuperar contraseña
first_name	Varchar	E	Nombre de usuario
last_name	Varchar	E	Apellido de usuario
is_admin	Boolean	E	Representa si es administrador o un usuario normal

Tabla 12. Tabla Usuario

Fuente: Elaboración Propia

Tabla de Perfil

Atributos	Tipo de dato	Tipo	Descripción
idProfile	Integer	PK	Clave primaria que identifica al Perfil
idUser	Integer	FK	Clave foránea que relaciona a Usuario
idCity	Integer	FK	Clave foránea que relaciona a Ciudad
dni	Varchar	E	Cedula de identificación del usuario
home_address	Varchar	E	Dirección del usuario
gender	Varchar	E	Género del usuario
birth_date	Date	E	Fecha de nacimiento del usuario

biography	Text	E	Biografía correspondiente al usuario
phone_number	Varchar	E	Número de teléfono del usuario
education_level	Varchar	E	Nivel de educación del usuario
work_area	Varchar	E	Trabajo en que se desempeña el usuario
picture	Varchar	E	Foto del usuario
created	Datetime	E	Fecha de creación del perfil
modified	Datetime	E	Fecha de modificación del perfil

Tabla 13. Tabla de Perfil

Fuente: Elaboración Propia

Tabla de Red_Social

Atributos	Tipo de dato	Tipo	Descripción
idRedSocial	Integer	PK	Clave primaria que identifica a la Red social
idProfile	Integer	FK	Clave foránea que relaciona a Perfil
name	Varchar	E	Nombre de la Red Social
link	Varchar	E	Link de la Red Social

Tabla 14. Tabla de Red_Social

Fuente: Elaboración Propia

Tabla de Notificación

Atributos	Tipo de dato	Tipo	Descripción
idNotification	Integer	PK	Clave primaria que identifica la notificación
idSender	Integer	FK	Clave foránea que relaciona al Usuario
idReceptor	Integer	FK	Clave foránea que relaciona al Usuario
idProject	Integer	E	Identificador del post
Notification_type	Integer	E	Tipo de notificación

Is_readed	Boolean	E	Si ya fue leída la notificación
date	datetime	E	Fecha de creación de la notificación

Tabla 15. Tabla de Notificación

Fuente: Elaboración Propia

Nota: El idSender va a almacenar el id del usuario que activa la notificación y el idReceptor va a obtener el id del usuario que va a recibir la notificación.

Tabla de Ciudad

Atributos	Tipo de dato	Tipo	Descripción
idCity	Integer	PK	Clave primaria que identifica a Ciudad
idProvince	Integer	FK	Clave foránea que relaciona a Provincia
name_city	Varchar	E	Nombre de la Ciudad

Tabla 16. Tabla de Ciudad

Fuente: Elaboración Propia

Tabla de Provincia

Atributos	Tipo de dato	Tipo	Descripción
idProvince	Integer	PK	Clave primaria que identifica a Provincia
idCountry	Integer	FK	Clave foránea que relaciona a País
name_province	Varchar	E	Nombre de la Provincia

Tabla 17. Tabla de Provincia

Fuente: Elaboración Propia

Tabla de País

Atributos	Tipo de dato	Tipo	Descripción
idCountry	Integer	PK	Clave primaria que identifica a País
name_country	Varchar	E	Nombre del País

Tabla 18. Tabla de País

Fuente: Elaboración Propia

Tabla de Proyecto

Atributos	Tipo de dato	Tipo	Descripción
idProject	Integer	PK	Clave primaria que identifica al proyecto
idCategory	Integer	FK	Clave foránea que relaciona a Categoría
idProfile	Integer	FK	Clave foránea que relaciona al Perfil
idType	Integer	FK	Clave foránea que relaciona Tipo_Proyecto
idStatus	Integer	FK	Clave foránea que relaciona Estado
title	Varchar	E	Título del Proyecto
description	Text	E	Descripción del Proyecto
objective	Varchar	E	Objetivo del Proyecto
image	Varchar	E	Imagen del Proyecto
url	Varchar	E	Dirección url del proyecto
colaborador	Varchar	E	Nombre del colaborador del proyecto
document	Varchar	E	Archivo del proyecto
created	datetime	E	Fecha de creación
modified	datetime	E	Fecha de modificación

*Tabla 19. Tabla de Proyecto**Fuente: Elaboración Propia*

Nota: La url es una opción si el proyecto está en la web.

Tabla de Categoría

Atributos	Tipo de dato	Tipo	Descripción
dCategory	Integer	PK	Clave primaria que identifica la Categoría
category	Varchar	E	Nombre de la categoría
image	Varchar	E	Imagen de la categoría

*Tabla 20. Tabla Categoría**Fuente: Elaboración Propia*

Nota: El atributo image sirve para poder guardar una imagen representativa de una categoría.

Tabla de Tipo de Proyecto

Atributos	Tipo de dato	Tipo	Descripción
idType	Integer	PK	Clave primaria que identifica Tipo_Pro
type	Varchar	E	Nombre del tipo de proyecto
image	Varchar	E	Imagen del tipo de proyecto

Tabla 21. Tabla Tipo de Proyecto

Fuente: Elaboración Propia

Tabla de Estado

Atributos	Tipo de dato	Tipo	Descripción
idStatus	Integer	PK	Clave primaria que identifica Estado
status	Varchar	E	Nombre del Estado del proyecto
image	Varchar	E	Imagen del Estado del proyecto

Tabla 22. Tabla Estado

Fuente: Elaboración Propia

Tabla de Me Gusta

Atributos	Tipo de dato	Tipo	Descripción
idMegusta	Integer	PK	Clave primaria que identifica a “Me gusta”
idUser	Integer	FK	Identificador del usuario que da Like
idProject	Integer	FK	Identificador del Proyecto

Tabla 23. Tabla de Me Gusta

Fuente: Elaboración Propia

Nota: El atributo idProject almacena el identificador del proyecto que se realiza el “Me gusta”

Tabla de Mensaje

Atributos	Tipo de dato	Tipo	Descripción
idMessage	Integer	PK	Clave primaria que identifica a “Mensaje”
idUser	Integer	FK	Identificador del user que inicia conversación
id_sender_user	Integer	FK	Identificador del user que envía el mensaje
id_receptor_user	Integer	FK	Identificador del user que recibe el mensaje
body	Varchar	E	Contenido del mensaje
date	Datetime	E	Fecha en que se envía el mensaje
is_read	Boolean	E	Si el mensaje fue leído

Tabla 24. Tabla de Conversación

Fuente: Elaboración Propia

Tabla de Comentario

Atributos	Tipo de dato	Tipo	Descripción
idComment	Integer	PK	Clave primaria que identifica a “Comentario”
idUser	Integer	FK	Identificador del user que realiza el comentario
idProject	Integer	FK	Identificador del proyecto que se comenta
body	Varchar	E	Contenido del comentario
date	Datetime	E	Fecha en que se el comentario

Tabla 25. Tabla de Comentario

Fuente: Elaboración Propia

Tabla de Seguir

Atributos	Tipo de dato	Tipo	Descripción
idFollow	Integer	PK	Clave primaria que identifica a “Follow”
from_user	Integer	FK	Identificador del user quien sigue
to_user	Integer	FK	Identificador del user a quien sigue

Tabla 26. Tabla de Seguir

3.6. Tecnologías aplicadas

Para el desarrollo de la aplicación de gestión de proyectos de emprendimientos de la Facultad Ciencias e Ingeniería se utilizó las tecnologías de backend como el lenguaje de programación de interpretación y desarrollo flexible Python, acompañado con su framework Django que se adapta de una manera ágil al desarrollo de aplicaciones web permitiendo implementar una estructura altamente escalable y teniendo como almacenamiento seguro en la base de datos PostgreSQL. En la parte del frontend se utilizó el framework Bootstrap para el diseño de las diferentes interfaces de usuario acompañado de JavaScript para una forma más dinámica de visualización entrelazándolo con el lenguaje de etiquetado HTML5.

3.7. Arquitectura de la aplicación

La Arquitectura de la aplicación web se basa de los objetivos y requerimientos tanto funcionales, como no funcionales que se describieron anteriormente, para su realización se usó la arquitectura de 3 capas.

3.7.1. Primera capa (Model)

Es la capa de acceso a los datos, es la que contiene la información sobre todos los datos. Permitiendo a Django aislar la base de datos relacional de todo el código. Se crea el modelo en la base de datos relacional PostgreSQL, cada uno de los modelos es una tabla en el database, la parte importante son los campos que se definen, los tipos de campos, como también los argumentos específicos del campo (ver Anexo 1).

3.7.2. Segunda Capa (View)

A esta capa Django la conoce como views. Aquí se procesa toda la información que el usuario consulta o solicita (request) a la aplicación, esta información es validada en el modelo de la base de datos para poder entregar u obtener datos. Esta vista genera una respuesta, en este caso el método render establece una plantilla de salida donde se van a entregar los datos solicitados. La vista es la lógica para poder entregar datos, y hacia donde mostrarlos (ver Anexo 2).

3.7.3. Tercera Capa (Template)

En esta última capa se renderiza toda la información que fue procesada en las views, esta información se la muestra en un template que funciona en formato de código HTML. Cada

una de estas plantillas tienen respuestas que proceden de las views con una estructura necesaria para su presentación (ver Anexo 3), utilizando estilo (CSS) y dinamismo gracias a JavaScripts. La arquitectura de la aplicación se la representa de la siguiente manera en la figura 14.

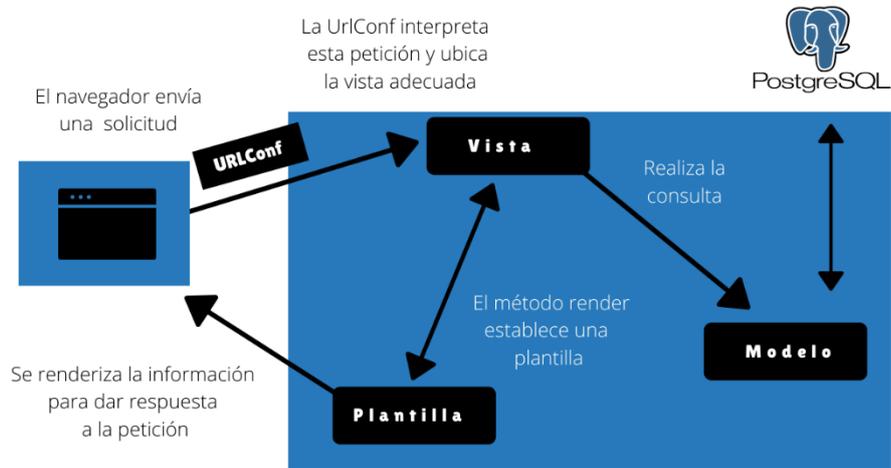


Figura 14: Arquitectura de la Aplicación web

Fuente: Elaboración Propia

3.8. Diseño de la interfaz

Inicio de Sesión

Después de culminar con el modelo se procedió al desarrollo de las interfaces y la programación correspondientes a cada una de los módulos. La primera pantalla que se observa en la figura 15 corresponde al LOGIN (ver Anexo 2), el cual permite el ingreso al sistema mediante un correo electrónico y contraseña.

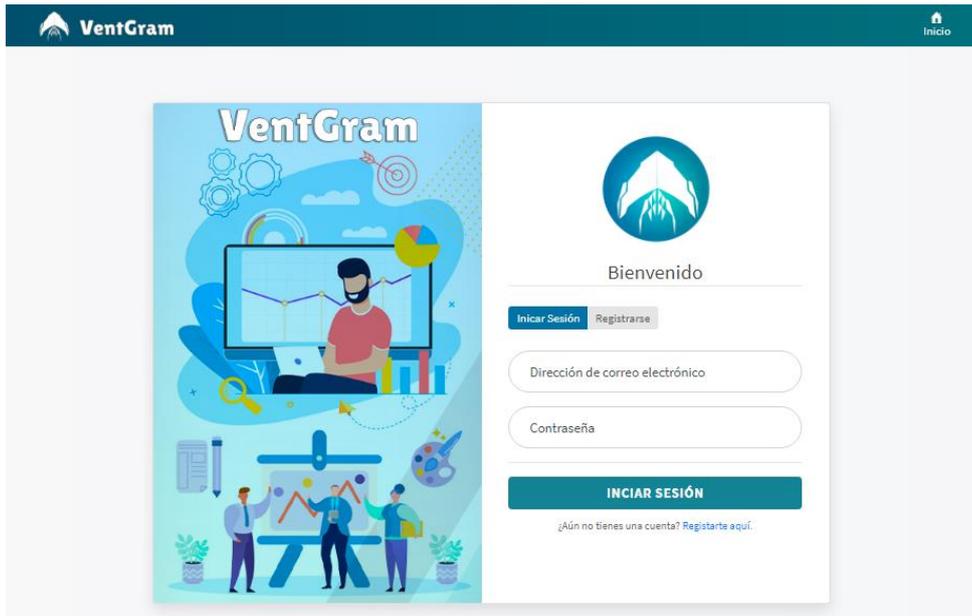


Figura 15. Interfaz de Inicio de Sesión

Fuente: Elaboración Propia

Registro de Usuario

Si no tiene una cuenta registrada, el usuario lo puede crear una dando click en la opción “Regístrate aquí” en inicio de sesión, se abrirá una interfaz de Registro de usuario como se muestra en la figura 16.

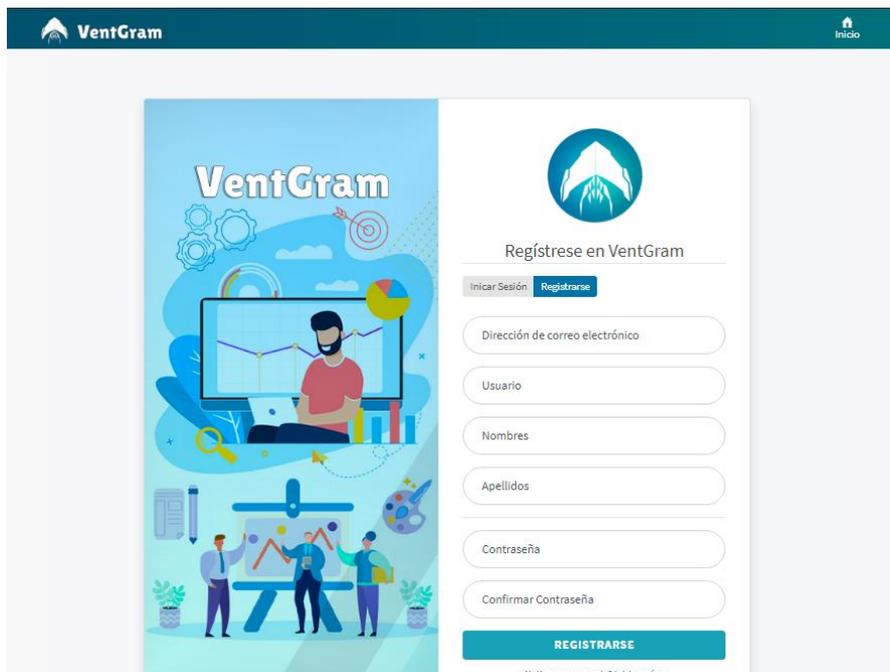


Figura 16. Interfaz de Registro de Usuario

Fuente: Elaboración Propia

Verificación de correo electrónico

Una vez enviado los datos para registrar, la aplicación enviará un correo electrónico al usuario que se está registrando, el cual tendrá que revisar su bandeja de correos y buscar “VentGram Activación de Cuenta” como se muestra en la figura 17.

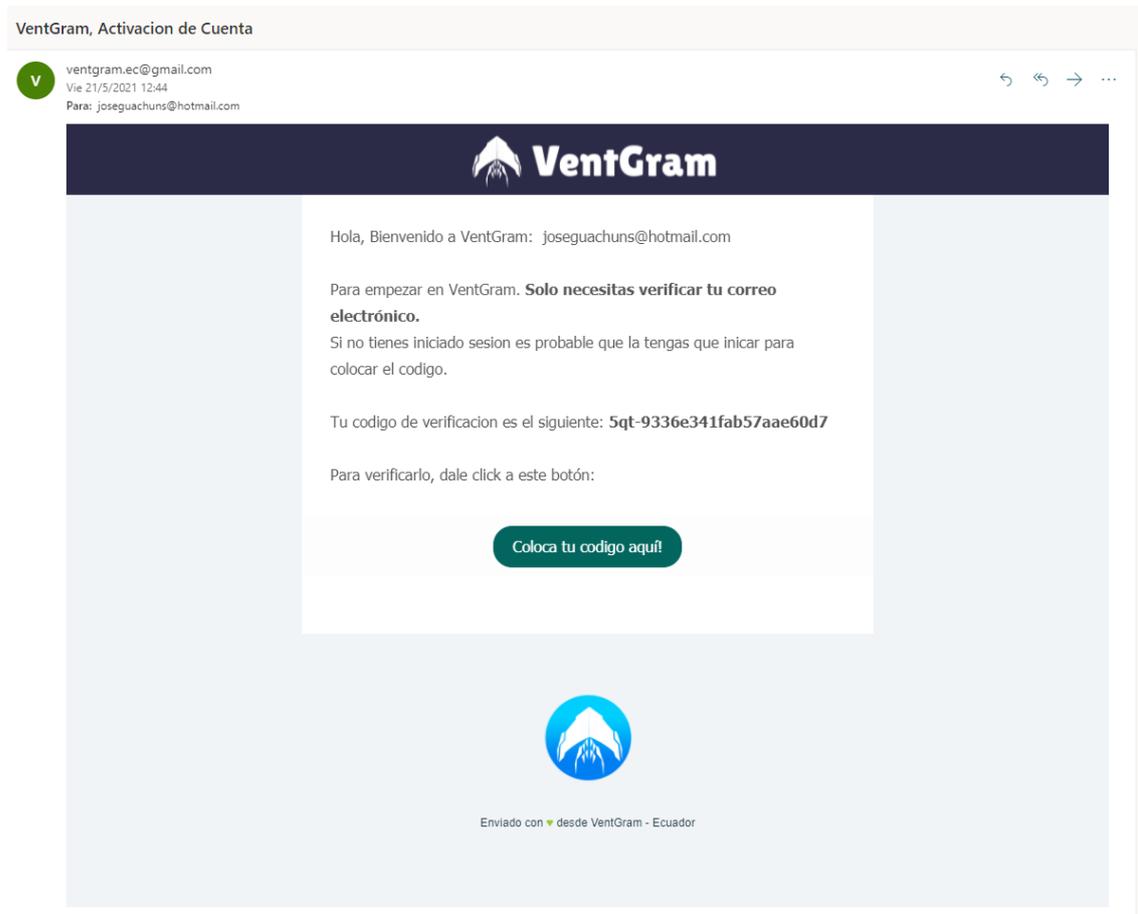


Figura 17. Mensaje de correo electrónico con código de verificación

Fuente: Elaboración Propia

El usuario posterior a recibir el código de verificación deberá copiar su código y pegarlo en la interfaz que se muestra en la figura 18 para poder activar su cuenta.

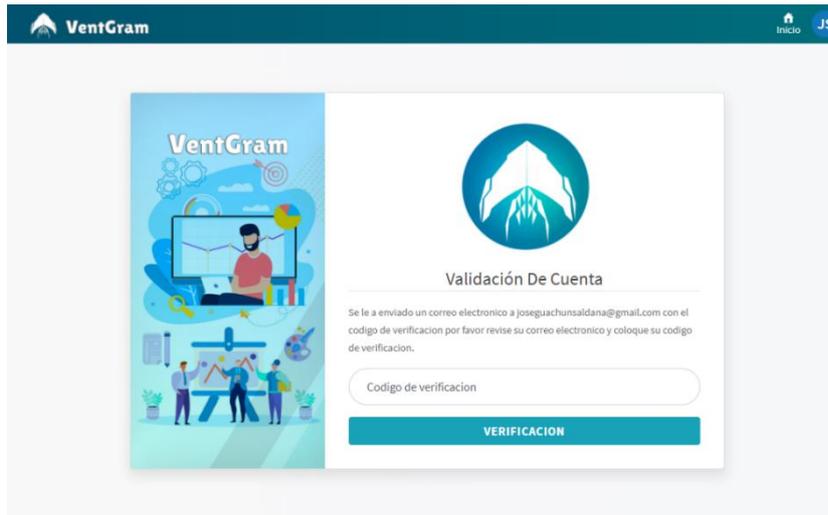


Figura 18. Interfaz de Validación de Cuenta

Fuente: Elaboración Propia

Feed

Al momento de ingresar a la aplicación lo primero que se mostrara es el feed de los proyectos publicados por otros usuarios, como se muestra en la figura 19. Los proyectos se mostrarán en forma de una tarjeta que muestra una reseña del proyecto, el usuario quien lo publicó, el título referente a la publicación, etc. También cuenta con un botón en la parte superior izquierda para publicar un proyecto nuevo.

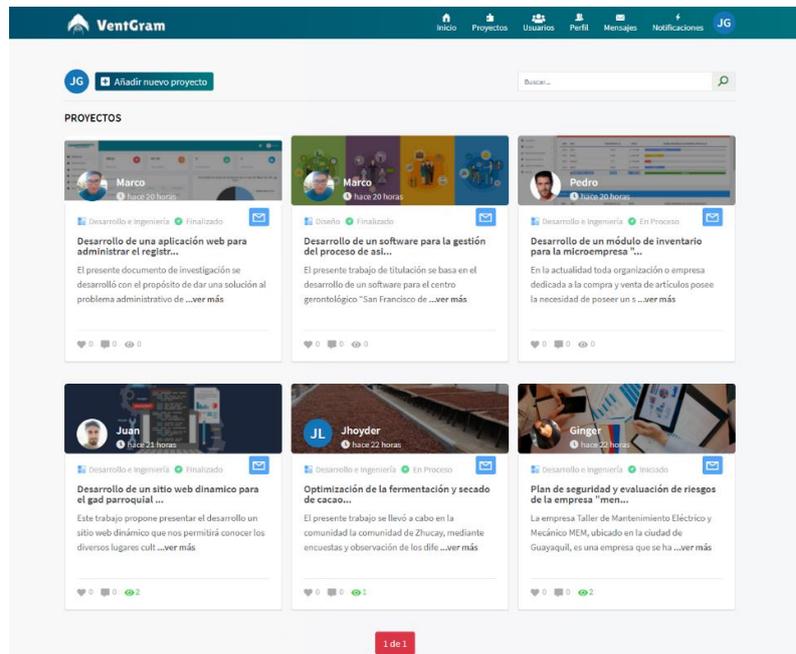


Figura 19. Interfaz al momento de iniciar sesión (Feed)

Fuente: Elaboración Propia

Publicar Proyectos

Cuando el usuario desea publicar un nuevo proyecto debe seleccionar el botón “añadir un nuevo proyecto” del feed y se habilitará una plantilla flotante como se observa en la figura 20, la cual permitirá poder ingresar todos los datos y archivos referentes a la nueva publicación que va a realizar.

The image shows a user profile for Jose Guachun on the left, with a blue circular avatar containing the letters 'JG'. Below the avatar, the name 'Jose Guachun' is displayed, along with 'Seguidos' and 'Seguidores' both at 0. There are buttons for 'Usuario' and 'Perfil', and a note 'Se unio hace una hora' with the email '@Jose43' and 'Joseguachuns@hotmail.com'. Social media icons for Facebook, Twitter, LinkedIn, and GitHub are also present. On the right, a 'Nuevo Proyecto' form is open. The form has a teal header and contains several input fields: 'Imagen de Cabecera (Obligatorio)' with a camera icon, 'Titulo' (Ingrese el titulo de su proyecto (Obligatorio)), 'Colaboradores' (Eje. Pedro Aguirre, John Cardenas (Opcional)), 'Tecnologías Utilizadas' (Eje. Python, HTML5, CSS3, Django (Obligatorio)), 'Estado' (dropdown), 'Objetivo' (Ingrese el objetivo del proyecto (Obligatorio)), 'Descripcion' (Ingrese la descripcion del proyecto (Obligatorio)), 'Url' (Ingrese la url del proyecto (Opcional)), 'Categoría' (dropdown), 'Tipo de Proyecto' (dropdown), and 'Documento' (Selecione un documento (Obligatorio)) with a 'SUBIR DOCUMENTO' button. At the bottom of the form are 'Publicar' and 'Cancelar' buttons.

Figura 20. Interfaz de Publicar Proyecto

Fuente: Elaboración Propia

Proyectos por filtrado

Cuando el usuario quiera ver proyectos específicos debe dar click en el botón “Proyectos” que se encuentra en la parte superior del nav como se muestra en la figura 21, se podrá hacer un filtrado de los proyectos por las categorías que han sido registradas.

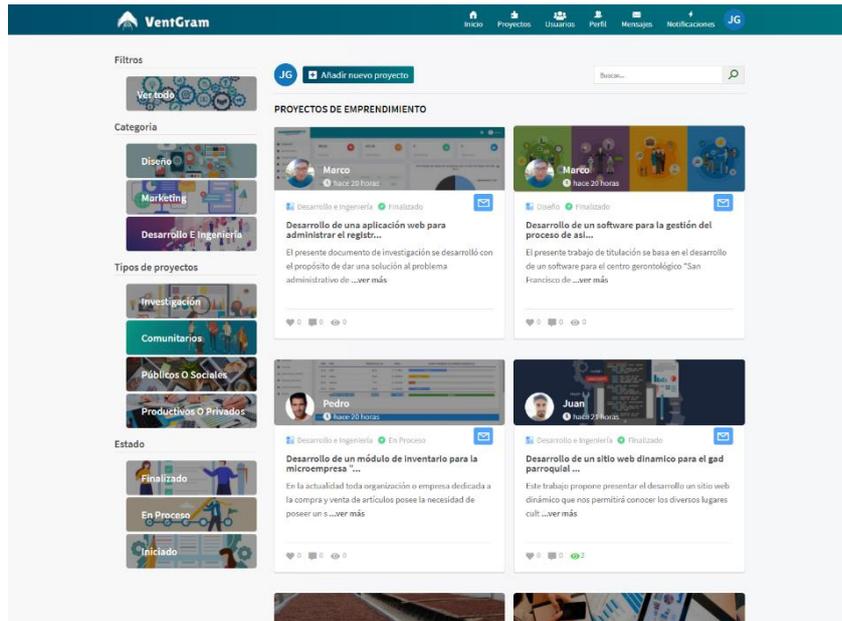


Figura 21. Interfaz de Proyectos por filtrado

Fuente: Elaboración Propia

Perfil de Usuario

Cuando el usuario ingresa a su perfil podrá visualizar su información en la parte izquierda, mientras que en la parte del centro se encuentran los proyectos publicados de manera descendente como se muestra en la figura 22.

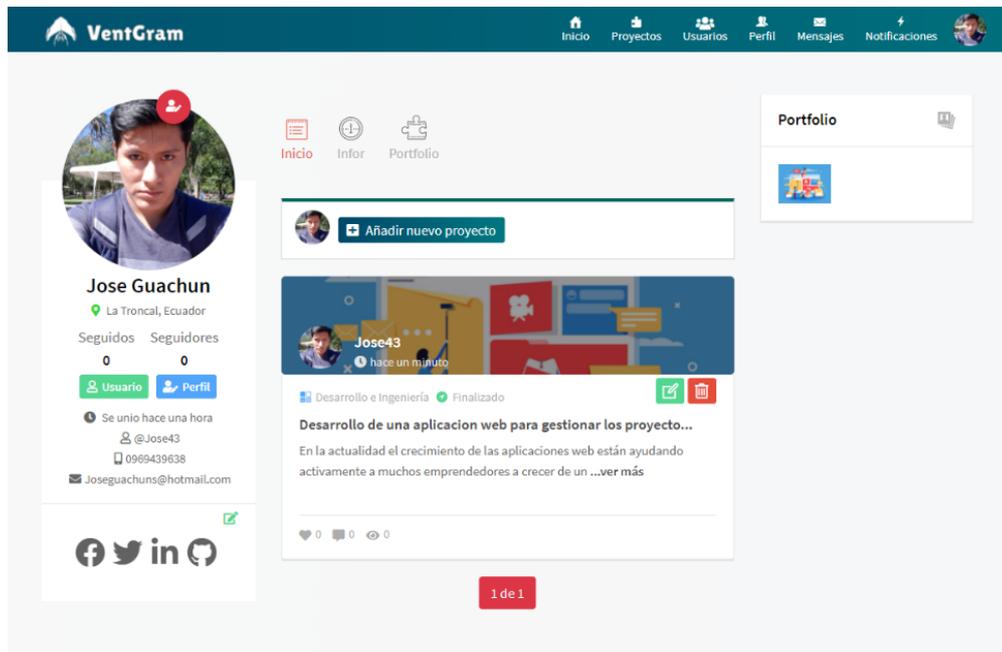


Figura 22. Interfaz de perfil de Usuario

Fuente: Elaboración Propia

Detalle de proyecto

Para poder visualizar una publicación completa se ingresa dando click en el título o imagen del proyecto publicado que se encuentran en las tarjetas de presentación, al momento de abrir una publicación se mostrara una interfaz como la siguiente figura 23.

The screenshot displays the 'VentGram' project detail interface. At the top, there is a navigation bar with icons for Inicio, Proyectos, Usuarios, Perfil, Mensajes, and Notificaciones. Below the navigation bar, there is a header with a profile picture and a button 'Añadir nuevo proyecto'. The main content area features a colorful banner with icons representing various project stages. The project title is 'Desarrollo de un software para la gestión del proceso de asistencia y actividades en los pacientes del centro gerontológico "san francisco de asís" de la ciudad de naranjal'. The author is 'Por: Marco Cardenas | @Marco' and the collaborator is 'Colaboradores: Jhoyder Aguirre'. The 'Objetivo' section lists three points: obtaining notifications from the parish of Lorezo de Garaicoa, presenting territorial conformations, and knowing cultural and tourist places. The 'Descripcion' section explains the software development process for the gerontological center, highlighting the use of Kanban methodology and the goal of improving information management efficiency. At the bottom, there is a 'Documento' section with a download button for a PDF file named '/DESARROLLO_DE_UN_MÓDULO_DE_INVENTARIO_PARA_LA_MICROEMPRESA.pdf'.

Figura 23. Interfaz del detalle de un proyecto

Fuente: Elaboración Propia

Interfaz de interacciones

En esa misma publicación se podrá guardar como favoritos dicho proyecto, generar un me gusta, comentar y descargar el documento completo del proyecto como se muestra en la figura 24.

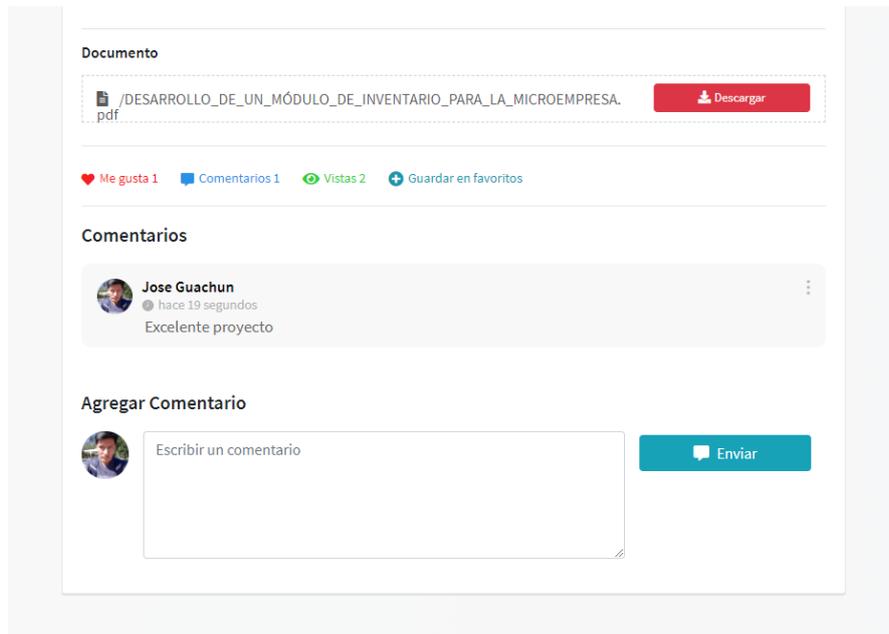


Figura 24. Interfaz de interacciones con el proyecto visualizado

Fuente: Elaboración Propia

Búsqueda de usuarios

Para visitar perfiles de otros usuarios, se tiene que seleccionar su nombre en proyecto que este público o a su vez dirigirse a la pestaña “Usuarios” y podrá buscar el usuario que desee como se muestra en la figura 25.

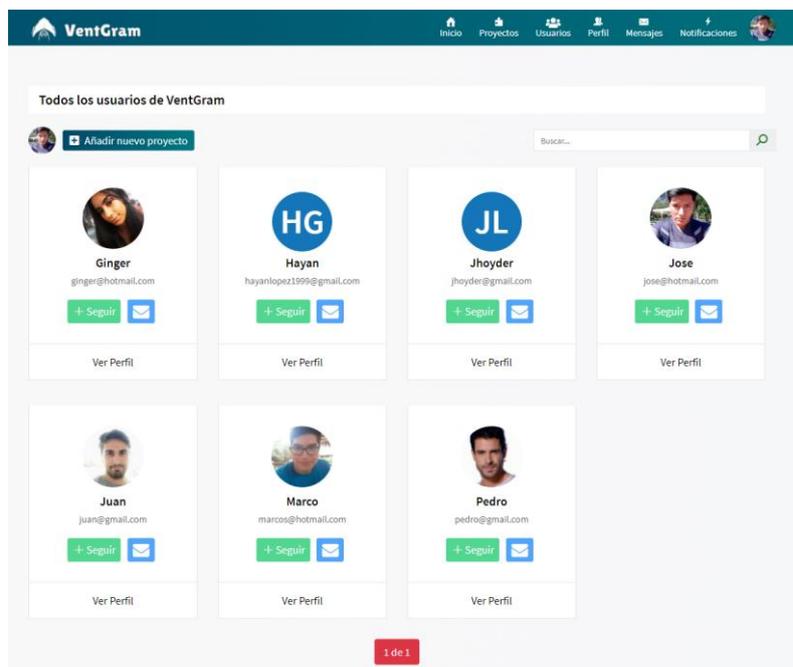


Figura 25. Interfaz de búsqueda de usuarios

Fuente: Elaboración Propia

Una vez seleccionado el usuario a visitar se visualizará el perfil del usuario, permitiendo interactuar de forma dinámica ya sea con él envió de mensajes, seguir a dicho usuario, ver su información de presentación, sus proyectos publicados o visitar sus redes sociales como se muestra en la figura 26.

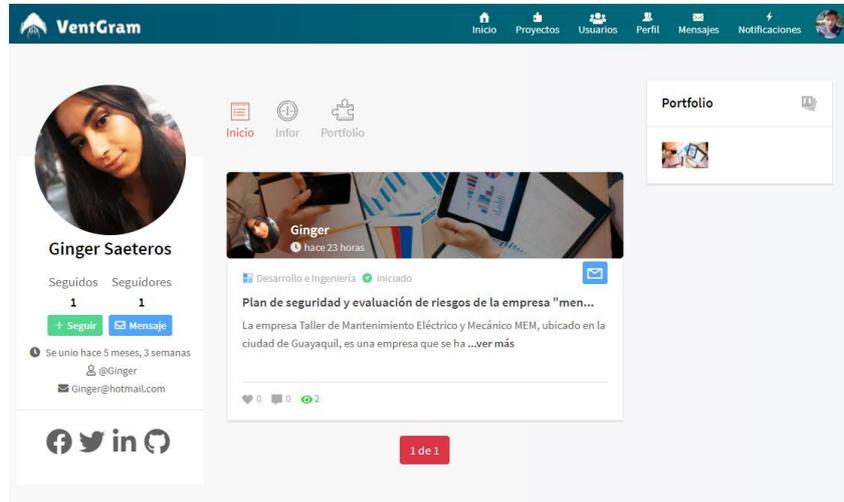


Figura 26. Interfaz de perfil de visita de usuario

Fuente: Elaboración Propia

Chat

Al seleccionar el botón mensaje este lo redirigirá a otra interfaz de chat con el usuario seleccionado, permitiendo enviar y recibir mensajes del usuario a contactar como se muestra en la figura 27.

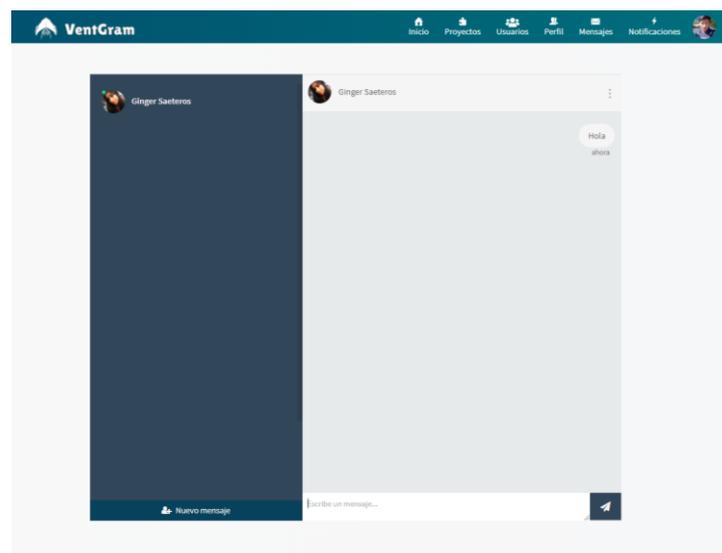


Figura 27. Interfaz de chat

Fuente: Elaboración Propia

Notificaciones

En la pestaña notificaciones se reflejará todas las interacciones que el usuario tenga de otros usuarios, ya sea comentarios, me gusta, o nuevos seguidores se reflejaran en esta sección como se muestra en la figura 28.

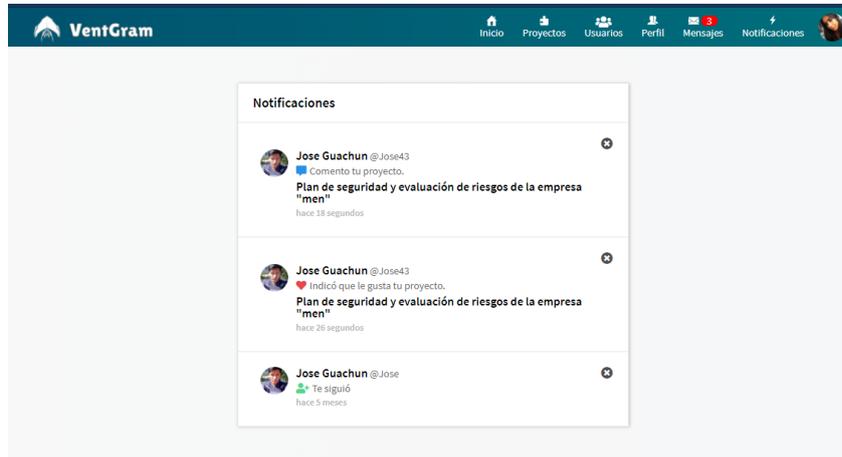


Figura 28. Interfaz de notificaciones

Fuente: Elaboración Propia

Actualización de perfil

Para realizar la actualización del perfil el usuario debe dar click en la imagen que le saldrá por defecto si aún no actualiza su imagen, se desplegará una opción de configuración de cuenta y mostrará la plantilla de configuración de perfil como se muestra en la figura 29.

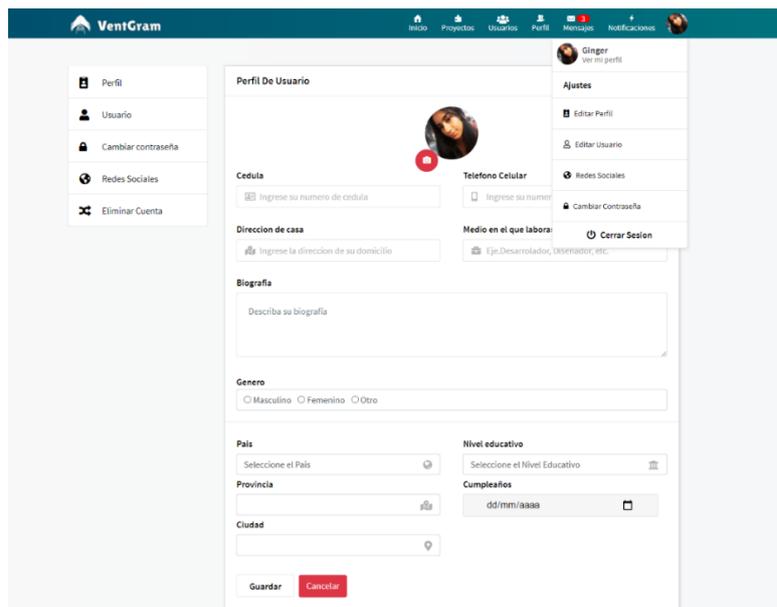


Figura 29. Interfaz de Actualización de Perfil

Fuente: Elaboración Propia

Actualización de usuario

En la figura 30 se muestra la interfaz para realizar la actualización de usuario.

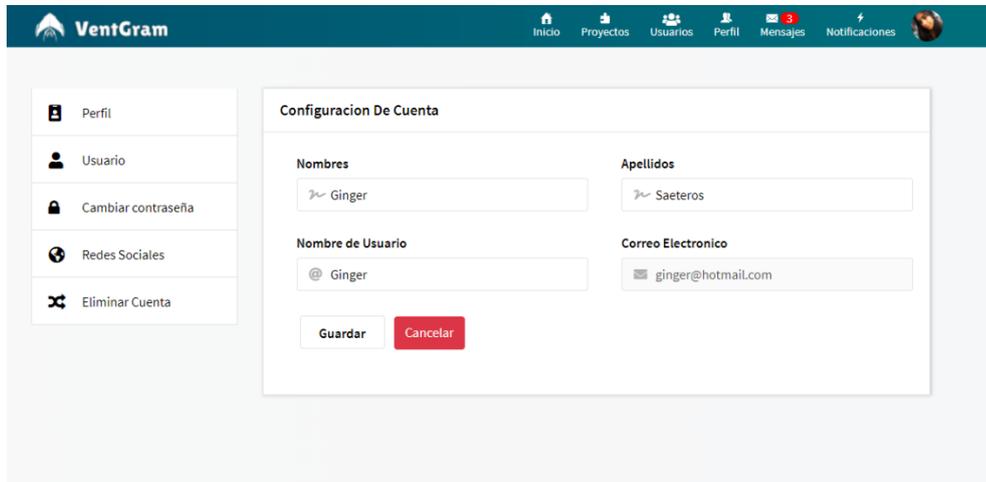


Figura 30. Interfaz de Actualización de usuario

Fuente: Elaboración Propia

Cambio de contraseña

Para realizar el cambio de contraseña el usuario debe dar click en la opción “Cambio de contraseña” que se encuentra en la parte izquierda, reflejara una plantilla como la que se muestra en la siguiente figura 31.

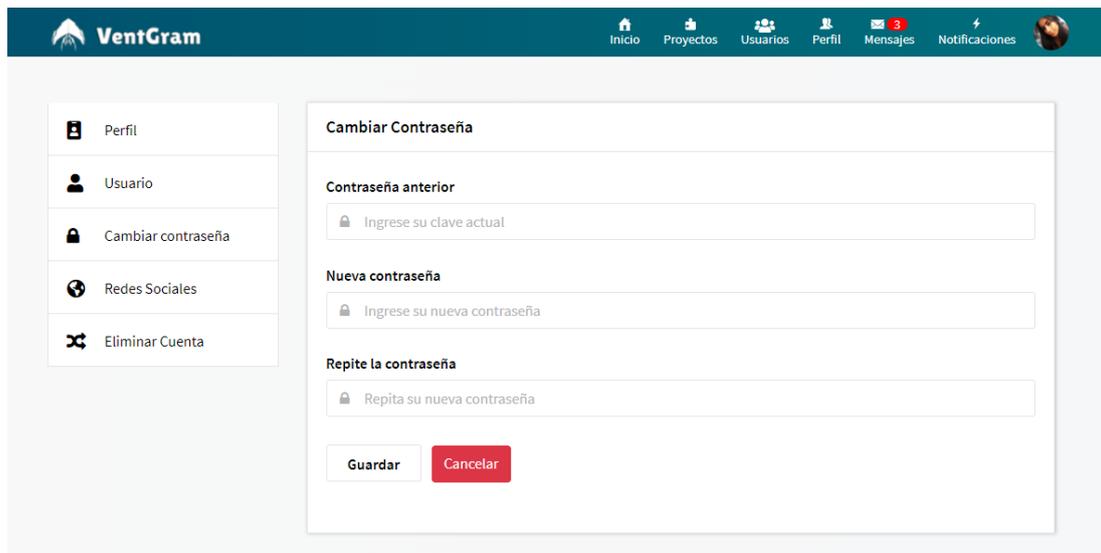


Figura 31. Interfaz de Cambio de contraseña

Fuente: Elaboración Propia

Redes Sociales

En configuración de la cuenta el usuario puede gestionar sus redes sociales como se puede observar en la figura 32.

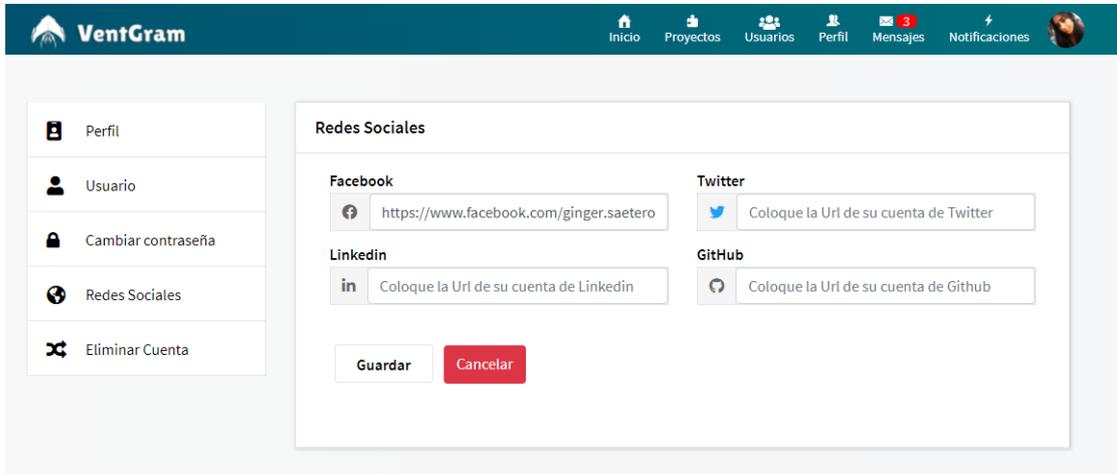


Figura 32. Interfaz de Redes sociales

Fuente: Elaboración Propia

Eliminación de cuenta

Por último, en la figura 33 se muestra la sección de Eliminar Cuenta, permitirá eliminar la cuenta del usuario si la desea, para ello debe proporcionar su clave de acceso para poder proceder con la eliminación permanente, también se borrarán todos sus proyectos publicados.

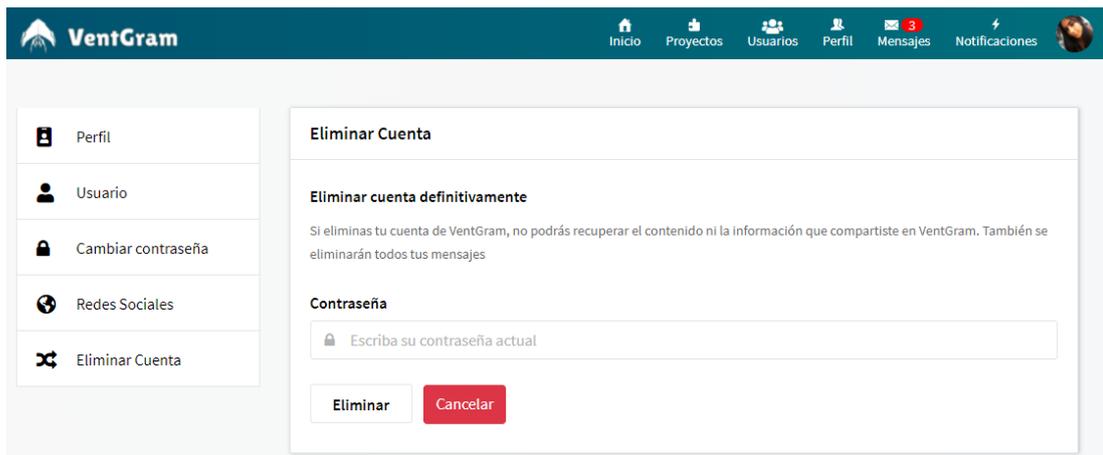


Figura 33. Interfaz de eliminación de cuenta

Fuente: Elaboración Propia

CONCLUSIONES

1. El desarrollo de esta aplicación web surge de la necesidad de superación de los estudiantes, mediante la cual los usuarios registrados podrán beneficiarse tanto al momento de publicar sus proyectos personales como también ser partícipe de otros proyectos de emprendimiento existentes. Permitiendo el desarrollo colaborativo y generando un entorno socioeconómico a los involucrados.
2. La estructuración de una base de datos relacional es muy importante debido a que se puede reutilizar conceptos o códigos de programación, permitiendo a la aplicación hacer consultas basándose puramente en claves foráneas ligadas a una tabla en específico, teniendo acceso a múltiples tablas utilizando la herencia aplicada, además permitiéndonos generar jerarquías de entidades es decir que nos da la posibilidad de guardar datos con una estructura de forma de un árbol.
3. La creación de las interfaces de usuario se centró en la comodidad del usuario, teniendo como prioridad el fácil manejo y el corto tiempo de respuesta de cada interacción. Para lograr estas prioridades se hizo uso del framework Bootstrap, el cual facilita el uso de clases de código abierto, reduciendo el tiempo de diseño de una manera organizada de forma visual y adaptable sin importar el dispositivo, la escala o resolución. Trabajando conjuntamente con CCS personalizados, JavaScript y la estructura HTML5

RECOMENDACIONES

Las recomendaciones después de la realización de la aplicación son las siguientes:

- Utilizar el Django REST framework para trabajar con un conjunto de herramientas potentes y flexibles que den la posibilidad de crear APIs web y a su vez reutilizarlas.
- Una alternativa a un lenguaje fácil de aprender cómo Node.js, para realizar el manejo de peticiones simultaneas impulsada por eventos.
- Completar todos los campos de perfil de usuario al momento de registrar un nuevo usuario para que las personas que visiten su perfil y visualicen un proyecto tengan una perspectiva clara de quien lo público.
- Agregar una funcionalidad de trabajo en equipo dentro de la plataforma con usuario propios registrados.

REFERENCIAS BIBLIOGRÁFICAS

- Abigail, H. D., & Miguel, M. S. (2018). *Bootstrap*. Mexico: Microsoft Mos Valuable Professional.
- Andrés, M., & Isabel, G. (2014). *Introducción a la programación con Python*. Castellón de la Plana: Universitat Jaume.
- Arias, M. (2005). La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. *InterSedes*, 1-13.
- Barchini, G., Sosa, M., & Herrera, S. (2012). La informática como disciplina científica. Ensayo de mapeo disciplinar. *Universidad Nacional de Santiago del Estero*.
- Beltrán, G. (2011). *PROYECTOS DE EMPRENDIMIENTO*. Quito: Editorial Universitaria Abya-Yala.
- Castejón, S. (2004). Arquitectura y diseño de sistemas web modernos. *InforMAS*, 1-6.
- Daniel Isaias, P. Q. (2019). SISTEMA INFORMÁTICO PARA EMPRENDIMIENTOS EN LA FACULTAD DE CIENCIA E INGENIERÍA EN ALIMENTOS DE LA UNIVERSIDAD TÉCNICA DE AMBATO Y COMUNIDAD. *Universidad Tecnica de Ambato*, 20-82.
- Gauchat, J. (2017). *El gran libro de HTML5, CSS3 y JavaScript*. Barcelona: MARCOMBO, S.A. .
- González, F., Calero, S., & Loaiza, D. (2019). Comparación de las metodologías cascada y ágil para el aumento de la productividad en el desarrollo de software. *Universidad Santiago de Cali*, 1-11.
- Hugo del Pozo, B. (2020). LEY ORGÁNICA DE EMPRENDIMIENTO E INNOVACIÓN. *Evolucion Juridica*, 1-49.
- Jimmy, R. M., Mora, N. M., Mariuxi, P. Z., & Erika, L. S. (2016). Evaluación de los Frameworks en el Desarrollo de Aplicaciones Web con Python. *Universidad Técnica de Machala*, 202.
- Jorge, M. C., & Cyndi, A. (2015). Fundamentos de base de datos. *UNIVERSIDAD TÉCNICA DE MACHALA*, 26.
- Latorre, M. (2018). HISTORIA DE LAS WEB, 1.0, 2.0, 3.0 y 4.0. *Universidad Marcelino Champagnat*, 1-8.
- Leonardo, P. O. (2019). MODELO PARA LA ORQUESTACIÓN DE MICROSERVICIOS CON KUBERNETES APLICADO AL SERVICIO DE CONTROL DE VERSIONES GIT. *UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS*, 30-68.
- María Guillermina, G. C., & Priscila Soledad, J. B. (2019). El emprendimiento y su evolución como una alternativa laboral en el. *UNIVERSIDAD DE CUENCA*, 8-9.
- Matilla, M., & Chalmeta, R. (2007). Metodología para la Implantación de un Sistema de Medición del Rendimiento Empresarial. *Información Tecnológica*, 119-126.

- Navarro, A., Fernández, J., & Morales, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Redalyc*, 30-39.
- Nieto Rodrigo, J. (2016). "Desarrollo de una aplicación web, con Front-end y Back-end, para compraventa de segunda mano. *UNIVERSIDAD POLITECNICA DE VALENCIA*, 13-16.
- Posada, M., & Cardenas, E. (2015). CONSTRUCCIÓN DE UNA APLICACIÓN WEB DE MAPAS, PARA VISUALIZACIÓN DE INFORMACIÓN RELACIONADA CON CALIDAD DEL AIRE EN LA CIUDAD DE BOGOTÁ, UTILIZANDO SOFTWARE LIBRE Y DE CÓDIGO ABIERTO. *UNIVERSIDAD MILITAR NUEVA GRANADA*, 1-10.
- Thomas, L. (1996). *Manual del usuario de PostgreSQL*. California: Postgres Global Development Group.
- Tobias, K., & Kay, G. (2016). *Visual Studio Code Tips & Tricks Vol. 1*. Alemania: Mathias Schiffer.

ANEXOS

Anexo 1

```
class User(AbstractBaseUser, PermissionsMixin):
    """User model
    extends from Django AbstractBaseUser, change the username field
    to email and some extra fields.
    """

    email=models.EmailField(
        'email address',
        unique=True,
        error_messages={
            'unique': 'Correo electrónico en uso.'
        }
    )

    username=models.CharField(
        'username',
        unique=True,
        max_length=30,
        error_messages={
            'unique': 'Nombre de Usuario en uso'
        }
    )

    first_name = models.CharField('first name', max_length=30, validators=[validators.MinLengthValidator(3)], blank=True, null=True)
    last_name = models.CharField('last name', max_length=30, validators=[validators.MinLengthValidator(3)], blank=True, null=True)
    code=models.CharField(
        'code',
        max_length=20,
        unique=True, blank=True, null=True,
        error_messages={
            'unique': 'Codigo en uso'
        }
    )

    is_admin=models.BooleanField(
        'client status',
        default=True,
        help_text=(
            'help easily distinguish users and perform queries. '
            'Clients are the main type of user. '
        )
    )

    is_verified=models.BooleanField(
        'verified',
        default=False,
        help_text=('se establece en verdadero cuando el usuario ha verificado su dirección de correo electrónico',
        )
    )

    date_joined = models.DateTimeField('date joined', default=timezone.now)
    is_active = models.BooleanField(default=True)
    is_staff = models.BooleanField(default=False)

    objects = UserManager()

    USERNAME_FIELD='email'
    REQUIRED_FIELDS=['first_name', 'last_name', ]

    def __str__(self):
        return self.username
```

Anexo 2

```
class ProfileView(LoginRequiredMixin, TemplateView):
    #muestra el perfil del usuario
    template_name='profile/me_profile.html'

class UpdateProfileView(FormView, LoginRequiredMixin, UpdateView):
    #update profile view
    template_name='profile/account_setting.html'
    form_class=ProfileForm
    def get_object(self):
        #return user profile
        return self.request.user.profile

    def get_context_data(self, **kwargs):
        context = super(UpdateProfileView, self).get_context_data(**kwargs)
        studies=('Primaria', 'Secundaria', 'Universidad', 'Maestria', 'Doctorado')
        work=('Desarrollo', 'Marketing', 'Diseño', 'Negocios', 'Electronica')
        gender=('Masculino', 'Femenino', 'Otro')
        context['studies']= studies
        context['work']= work
        context['gender']=gender
        return context

    def get_success_url(self):
        #Return to users profile.
        return reverse('users:setting')

class UpdateUserView(FormView, LoginRequiredMixin, UpdateView):
    #update User View
    template_name='profile/account_setting.html'
    form_class=UserForm

    def get_object(self):
        #return user profile
        return self.request.user

    def get_success_url(self):
        #Return to users profile.
        return reverse('users:setting')

class LoginView(FormView):
    #login view
    template_name='users/login.html'
    form_class=LoginForm
    success_url=reverse_lazy('posts:feed')

    @method_decorator(csrf_protect)
    @method_decorator(never_cache)
    def dispatch(self, request, *args, **kwargs):
        if request.user.is_authenticated:
            return HttpResponseRedirect(self.get_success_url())
        else:
            return super(LoginView, self).dispatch(request, *args, **kwargs)

    def form_valid(self, form):
        login(self.request, form.get_user())
        return super(LoginView, self).form_valid(form)

class SignupView(FormView):
    #Signup con classe base view
    template_name='users/signup.html'
```

Anexo 3

```
estGram > templates > users > login.html > hr:mt-1
1  {% extends 'users/base.html' %}
2  {% block head_content %}
3    login | VentGram
4  {% endblock %}
5  {% block title %}
6  <h5 class="card-title text-center mt-3 mb-0 pb-0" style="color: rgb(68, 68, 68);">Bienvenido</h5>
7  <hr class="mt-1" />
8  {% endblock %}
9
10 {% block container %}
11 {% if form.errors %}
12 <p class="alert alert-danger">El correo electrónico o contraseña están incorrectos.</p>
13 {% endif %}
14 <form method="POST" action="{% url 'users:login' %}">
15   {% csrf_token %}
16
17   <div class="form-label-group">
18     <input type="email" id="inputEmail" class="form-control" placeholder="Dirección de correo electrónico" name="username" required>
19     <label for="inputEmail">Dirección de correo electrónico</label>
20   </div>
21
22   <div class="form-label-group">
23     <input type="password" id="inputPassword" class="form-control" placeholder="Password" name="password" required>
24     <label for="inputPassword">Contraseña</label>
25   </div>
26
27   <div class="custom-control custom-checkbox mb-3">
28     <input type="checkbox" class="custom-control-input" id="customCheck1">
29     <label class="custom-control-label ml-3" style="color: rgb(92, 92, 92); font-weight:bold">Recordar Contraseña</label>
30   </div>
31
32   <hr class="my-4">
33   <button class="btn btn-lg btn-primary btn-block btn-login font-weight-bold text-uppercase" type="submit">Iniciar sesión</button>
34   <p class="d-block text-center mt-2 mb-1 small">¿Aún no tienes una cuenta? <a href="{% url 'users:signup' %}">Regístrate aquí.</a> </p>
35   <p class="d-block text-center small">¿Se te olvidó tu contraseña? <a href="{% url 'users:signup' %}">Recuperar contraseña.</a> </p>
36 </form>
37 {% endblock %}
38
```

Anexo 4

