



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA DE COMPUTACIÓN

**SISTEMA DE SOCIALIZACIÓN DE PROYECTOS Y DE CONOCIMIENTOS
EN ÁREAS ACADÉMICAS PARA ESTUDIANTES DE LA CARRERA DE
COMPUTACIÓN DE LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE
QUITO, CAMPUS SUR**

Trabajo de titulación previo a la obtención del
Título de Ingenieros en Ciencias de la Computación

AUTORES: PAUL ALEJANDRO MORALES ROSERO
LEYTON LUIS VALENCIA PORTILLA

TUTOR: DANIEL GIOVANNY DÍAZ ORTIZ

Quito-Ecuador
2024

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Paul Alejandro Morales Rosero, con documento de identificación N° 1719518027 y Leyton Luis Valencia Portilla con documento de identificación N° 1750290817 manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 04 de marzo de 2024

Atentamente,



Paul Alejandro Morales Rosero
1719518027



Leyton Luis Valencia Portilla
1750290817

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Nosotros, Paul Alejandro Morales Rosero con documento de identificación No. 1719518027 y Leyton Luis Valencia con documento de identificación No. 1750290817, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: “Sistema de socialización de proyectos y de conocimientos en áreas académicas para estudiantes de la carrera de computación de la Universidad Politécnica Salesiana sede Quito, campus Sur”, el cual ha sido desarrollado para optar por el título de: Ingenieros en las Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

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

Quito, 04 de marzo de 2024

Atentamente,



Paul Alejandro Morales Rosero
1719518027



Leyton Luis Valencia Portilla
1750290817

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Daniel Giovanni Díaz Ortiz con documento de identificación N° 1716975501 , docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: SISTEMA DE SOCIALIZACIÓN DE PROYECTOS Y DE CONOCIMIENTOS EN ÁREAS ACADÉMICAS PARA ESTUDIANTES DE LA CARRERA DE COMPUTACIÓN DE LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE QUITO, CAMPUS SUR, realizado por Paul Alejandro Morales Rosero con documento de identificación N° 1719518027 y por Leyton Luis Valencia Portilla con documento de identificación N° 1750290817, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 04 de marzo de 2024

Atentamente,



Ing. Daniel Giovanni Díaz Ortiz MSc.
1716975501

DEDICATORIA

Dedico este trabajo a la persona que confió en mí desde el principio e incluso cuando ni siquiera ella sabía cómo resultarían mis proyectos, me empuja a seguir adelante y nunca me quita su apoyo, la que me enseñó que podemos hacer un cambio solo si empezamos a avanzar, aunque se tenga todo en contra, y a la que le debo todo lo que soy y todo lo que tengo, a mi hermosa madre.

De la misma manera, quiero agradecer el amor y la guía de toda mi familia que incondicionalmente estuvieron presentes para mí, siguiéndome de cerca y dándome consejos y cariño que atesoraré toda mi vida, de mención especial a mi abuelita, que fue la persona que siempre cuida y cuidará de mí.

A una persona que fue una gran parte de mi inspiración en mi paso por la universidad, a mi novia que, a pesar de conocerme poco, me brindó un cariño que nunca pensé sentir de alguien que iba a conocer en esta universidad, la que me apoya cuando ni yo mismo comprendo mis sentimientos, gracias por ser mi novia, confidente y mi mejor amiga.

Y a mi compañero de tesis, otra alegría que me llevo es esta amistad que, a pesar de todo, sigue ahí para mí, gracias por todo y lo logramos hermano, luego de todas las madrugadas, malas noches y tiempo riéndonos en los pasillos y las aulas, terminamos la carrera, este agradecimiento es también para ti por tu apoyo incondicional.

Con gratitud, Paul Morales

DEDICATORIA

Queridos Padres, a ustedes les dedico este logro que representa no solo mi crecimiento académico, sino también es el vivo reflejo de los valores, enseñanzas y lecciones que han impartido en mi a lo largo de este viaje.

Este logro no solo es mío, sino nuestro. Es la culminación de noches de estudio, sacrificios y momentos de celebración compartidos en familia. Han sido testigos de mis esfuerzos, de mis caídas y de mis triunfos, y cada paso que he dado hacia la superación personal ha sido inspirado por su ejemplo de tenacidad y dedicación.

Al mirar atrás, veo el camino recorrido y sé que este no es el final, sino un nuevo comienzo. Gracias por enseñarme que el crecimiento no tiene límites y que la superación personal es un viaje constante. Este logro es solo un peldaño más en la escalera de nuestros sueños compartidos.

Quiero expresar mi sincero agradecimiento a la universidad, cada profesor, cada recurso y cada experiencia en la universidad ha sido un pilar fundamental en mi trayectoria educativa.

Gracias, querida universidad, por ser más que un lugar de estudio.

Con Gratitud,

Leyton Valencia.

INDICE DE CONTENIDO

RESUMEN	1
ABSTRACT	2
INTRODUCCION	3
Antecedente	4
Problemática	7
Justificación	8
Objetivos.....	9
Objetivo general	9
Objetivos específicos.....	9
CAPÍTULO I FUNDAMENTO TEÓRICO.....	10
Metodología.....	10
1.1 Marco Institucional.....	12
1.2. Marco Teórico o Referencial	12
1.3. Herramientas de desarrollo	13
1.3.1 MongoDB Cloud	13
1.3.2 Express	14
1.3.3 React.....	15
1.3.4 NodeJS	16
1.3.5 Clerk.....	16
1.3.6 Loader.io.....	17
1.4 Metodología Scrum	18
1.4.1 Scrum Team.....	18
1.4.2 Scrum Events.....	20
1.5 Sistemas de gestión para el repositorio de proyectos	21
1.5.1 On-Premise.....	21
1.5.2 Cloud	22
1.5.3 Híbrido	23
1.5.4 Mediante terceros	24
1.6 Arquitecturas para un sistema de gestión de repositorio de proyectos	25
1.7 Métricas de calidad de software para un Sistema de Repositorio.	27
1.8 Métodos de seguridad aplicados a sistemas de repositorios.....	28
1.8.1 Método 1: Protección contra NoSQL Injection.....	29
1.8.2 Método 2: Firewall perimetral.....	29

1.8.3 Módulos de seguridad avanzados.....	30
1.8.4 Protección antimalware en terminales y servidores	30
1.8.5 Procedimientos y prácticas.....	30
1.8.6 Integridad de los datos.....	31
1.8.7 Certificaciones.....	31
CAPÍTULO II ANÁLISIS Y DISEÑO	32
2.1 Encuestas al cuerpo estudiantil (Docentes y estudiantes)	32
2.1.1 Interpretación de respuestas	32
2.2 Requerimientos funcionales y no funcionales	42
2.2.1. Requerimientos funcionales	42
2.2.2. Requerimientos no funcionales	43
2.3 Historias de usuario	44
2.4 Caso de uso.....	47
2.5 Diagrama lógico de la base de datos	53
CAPÍTULO III CONSTRUCCION Y RESULTADOS	54
3.1 Diagrama de despliegue.....	54
3.1.1 Clerk – Fuente de datos de usuarios.....	55
3.1.2 S3 AWS – Fuente de datos, repositorio de archivos.	57
3.2 Pruebas y resultados	58
3.2.1 Pruebas de carga.....	58
3.2.2 Resultados: Pruebas de carga	59
3.2.3 Pruebas de caja negra	68
CONCLUSIONES	86
RECOMENDACIONES	87
BIBLIOGRAFÍA	88

ÍNDICE DE TABLAS

Tabla 1 Cuadro Comparativo de Las Metodologías Ágiles.....	10
Tabla 3: Requerimientos funcionales	42
Tabla 4: Requerimientos no funcionales	43
Tabla 5: Desglose de historia - Autenticación de Usuarios	44
Tabla 6: Desglose de historia - Gestión de Publicaciones	44
Tabla 7: Desglose de historia - Búsquedas Personalizadas	45
Tabla 8: Desglose de historia - Carga y descarga de Archivos ZIP/RAR	46
Tabla 9: Desglose de historia - Comentarios Interactivos en Publicaciones	46

ÍNDICE DE FIGURAS

Figura 1 Arquitectura en premisa con almacenamiento en nube.....	21
Figura 2 Arquitectura en nube – ejemplo AWS.....	22
Figura 3 Arquitectura híbrida con alta disponibilidad.....	23
Figura 4 Arquitectura de alta disponibilidad por región - Hostinger Cloud.....	24
Figura 5 Gráfico de respuestas a pregunta #1.....	32
Figura 6 Gráfico de respuestas a pregunta #2.....	33
Figura 7 Gráfico de respuestas a pregunta #4.....	35
Figura 8 Gráfico de respuestas a pregunta #5.....	35
Figura 9 Gráfico de respuestas a pregunta #6.....	36
Figura 10 Gráfico de respuestas a pregunta #7.....	36
Figura 11 Gráfico de respuestas a pregunta #8.....	37
Figura 12 Gráfico de respuestas a pregunta #9.....	38
Figura 13 Gráfico de respuestas a pregunta #10.....	38
Figura 14 Gráfico de respuestas a pregunta #11.....	39
Figura 15 Gráfico de respuestas a pregunta #12.....	39
Figura 16 Gráfico de respuestas a pregunta #13.....	40
Figura 17 Gráfico de respuestas a pregunta #14.....	41
Figura 18 Diagrama de autenticación de usuario.....	47
Figura 19 Diagrama de gestión de Publicaciones.....	48
Figura 20 Diagrama de búsquedas personalizadas.....	50
Figura 21 Diagrama de carga y descarga de Archivos ZIP/RAR.....	51
Figura 22 Diagrama de comentarios interactivos en publicaciones.....	52
Figura 23 Diagrama lógico de la base de datos.....	53
Figura 24 Diagrama de despliegue.....	54
Figura 25 Imagen de permisos requeridos por Clerk sobre cuenta UPS.....	55
Ilustración 26 Código de llamado a librería de Clerk.....	55
Figura 27 Componente Clerk de usuario logeado.....	56
Figura 28 Función asíncrona de AWS S3 para subir archivos.....	57
Figura 29 Objeto con meta data de publicaciones en MongoDB Cloud.....	57
Figura 30 Tiempo de respuesta en milisegundos frente al número de usuarios y minutos de conexión.....	60
Figura 31 Gráfico de detalles de conexión por usuario.....	61
Figura 32 Gráfico de uso de ancho de banda contra tiempo en minutos.....	62
Figura 33 Gráfico de tiempos de respuesta en ms contra segundos, contra número de usuarios.....	64
Figura 34 Captura de reacción del WAF frente a los bots.....	65
Figura 35 Gráfico de solicitudes correctas contra tiempo en minutos.....	66
Figura 36 Gráfico de ancho de banda de usuarios contra tiempo en minutos.....	67
Figura 37 Botón inicio de sesión.....	68
Figura 38 Cuadro de ingreso de credenciales.....	68
Figura 39 Permisos requeridos de la aplicación Clerk.....	69
Figura 40 Inicio de la página web.....	69
Figura 41 Validación de ingreso de dominios no permitidos.....	70
Figura 42 Sección de publicaciones en la página web.....	70
Figura 43 Panel de administración de publicaciones.....	71

Figura 44 Opciones disponibles en la gestión de publicaciones	71
Figura 45 Sección de publicaciones luego de ejecutar una acción.....	72
Figura 46 Mensaje de validación cuando no se tienen publicaciones creadas	72
Figura 47 Filtros de sección de publicaciones.....	73
Figura 48 Opciones en filtros de publicaciones.....	73
Figura 49 Publicaciones en base a filtros usados	74
Figura 50 Mensaje de validación cuando no se encuentra una publicación	75
Figura 51 Ingreso a crear nueva publicación.....	75
Figura 52 Formulario de nueva publicación.....	76
Figura 53 Formulario de nueva publicación completo.....	77
Figura 54 Imagen del objeto creado en MongoDB luego de subir los datos del formulario	77
Figura 55 Imagen del archivo en AWS luego de subir el formulario	78
Figura 56 Publicación creada con el formulario completo.....	78
Figura 57 Sección de todas las publicaciones	79
Figura 58 Componente de publicación en sección de publicaciones	79
Figura 59 Vista principal cuando se ingresa a una publicación.....	80
Figura 60 Botón de descargar proyecto	80
Figura 61 Imagen referencia cuando se descarga él .zip	81
Figura 62 Validación con respecto al tamaño del archivo subido al formulario	81
Figura 63 Mensaje de control cuando no se logra cargar la información desde el servidor back-end.....	82
Figura 64 Validación para no cargar una imagen del usuario cuando no se ha elegido aun	82
Figura 65 Listado de publicaciones con sus tecnologías.....	83
Figura 66 Vista completa de página de publicación	83
Figura 67 Sección de comentarios en publicación	83
Figura 68 Campos de componentes para publicar un comentario.....	84
Figura 69 ComboBox de puntuación en el comentario	84
Figura 70 Visualización de comentarios general	84
Figura 71 Mensaje de validación cuando no se tienen comentarios en la publicación ..	85

RESUMEN

En la presente investigación se tiene como propósito diseñar y desarrollar un sitio web que cuente con un repositorio de proyectos y trabajos para los estudiantes de la carrera de Ingeniería en Ciencias de la Computación de la Universidad Politécnica Salesiana del Ecuador, consiguiendo así la preservación y difusión de conocimientos que se han ido trabajando a lo largo de la carrera para así rescatar los proyectos desarrollados por los estudiantes, debido a que muchos de estos trabajos son llamativos y útiles, se pueden llegar a aprovechar directamente por el docente como material para futuros ciclos, pero aun así gran parte de ellos se fueron olvidando y perdiendo a lo largo del tiempo.

En el marco teórico resaltamos la importancia de la información en la actualidad, tomando en cuenta que la mejor forma de llevar y mantener esta fuente de información es en un repositorio virtual, gracias a ellos se puede fortalecer el desarrollo y la competitividad en el área de recursos educativos que puedan aportar al estudiantado con conocimientos que se acoplen a los cambios y nuevas prácticas en el área.

El surgimiento de este proyecto viene de una falta de impacto de los proyectos realizados en la carrera de Ciencias de la Computación, en donde los estudiantes desarrollan una gran cantidad de trabajos y proyectos tecnológicos que han sido repetitivos o al mismo tiempo no han sido aprovechados pese al potencial que tienen, gracias a esto se ha realizado la evaluación de la implementación de un repositorio digital en donde se pueda almacenar e interactuar con las personas y trabajos anteriormente mencionados.

PALABRAS CLAVE: Repositorio Digital, Recursos Educativos, Software, TIC, Código Abierto, E-Learning.

ABSTRACT

The purpose of this research is to design and implement a website that has a repository of projects and works for students of Computer Science Engineering at the Salesian Polytechnic University of Ecuador, Thus achieving the preservation and dissemination of knowledge that have been working throughout the career to rescue the projects developed by students, because many of these works are striking and useful, you can get to take advantage directly by the teacher as material for future cycles, but still much of them were forgotten and lost over time.

In the theoretical framework we highlight the importance of information today, taking into account that the best way to carry and maintain this great source of information is in a virtual repository, thanks to them we can strengthen the development and competitiveness in the area of educational resources that can provide students with knowledge that can be coupled to the changes and new practices in the area.

The emergence of this project comes from a lack of project management in the career of Computer Science, where students have developed a lot of work and technology projects that have been repetitive or at the same time have not been exploited despite the potential they have, thanks to this has made the evaluation of the implementation of a digital repository where you can store and interact with people and work mentioned above.

KEYWORDS: Digital Repository, Educational Resources, Software, TIC, Open Source, E-Learning.

INTRODUCCION

En la actualidad, cada día se dan nuevas propuestas tecnológicas, surgen nuevos desafíos educativos que han sido planeados gracias a bases tecnológicas que se han ido trabajando con el tiempo.

Como menciona Gallegos y Polanco (2011): “La administración y gestión de los conocimientos representa para un país la memoria de su historia, cuya meta es transmitir y generar conocimientos”.

Al utilizar estos recursos educativos en un entorno virtual se logra maximizar la visibilidad, el uso y el impacto en la comunidad académica, tales recursos contribuyen al crecimiento del e-learning apoyándose en la tecnología para crear objetos virtuales de aprendizaje. En este aspecto, Roldan (2007) menciona: “Si queremos usar las nuevas tecnologías es necesario trabajar enseñando a utilizar las TIC, pero sin omitir que también es importante enseñar a los alumnos a buscar información, seleccionarla y evaluarla con criterio”.

En base a lo expuesto anteriormente, la elaboración de un repositorio virtual para la carrera servirá para la organización, preservación y difusión de las propuestas tecnológicas desarrolladas por el estudiantado, exhibiendo de forma sistemática, amigable y de fácil acceso, para lograr esto se desarrolló Code.Nest la cual es una página web que permitirá a la comunidad estudiantil guardar todo tipo de información que pueda ser de utilidad en un futuro.

Antecedente

Estudiantes de la UCE han abordado el mismo problema con un enfoque similar, en el año 2021 explican que existen proyectos buenos e innovadores que no tienen seguimiento, también comentando que toman como base la importancia que se le da a los repositorios universitarios para justificar por qué también se le debería dar importancia a los proyectos informáticos, así como a las tesis o artículos de otras carreras, posteriormente para explicar de qué manera iniciaron con el desarrollo de su propio repositorio siguiendo lineamientos y apegándose a los estándares (Arias Jaramillo Edison David, Quito 2021).

Buscar comunicarse es algo normal, y los desarrolladores y programadores constantemente están participando de manera activa y pasiva en redes como Reddit creando subredes que tienen toda la información referente al desarrollo de aplicaciones, noticias y actualizaciones del contenido (M. Aniche et al, 2018). Por lo que se busca que nuestra aplicación no solo sea un repositorio en el que los proyectos se almacenen, sino que también permita divulgación de los mismos y que los usuarios interesados tengan la oportunidad de interactuar y aclarar sus dudas entre estudiantes.

Según la World Engineering Education Forum en 2012, en los últimos años los repositorios institucionales han cobrado importancia en la sociedad estos son depósitos de archivos digitales a los cuales se puede acceder, difundirlos y preservarlos, es importante recalcar la importancia que tienen en el ámbito académico de la ingeniería como fuente de conocimiento por parte de los docentes, investigadores y alumnos para contribuir al desarrollo social y humano. Estos repositorios deben ser Open Access (OA) porque así es posible asegurar un acceso libre y abierto sin restricciones a los proyectos para preservarlos digitalmente como un bien común para la sociedad de hoy y del futuro.

Se deben seguir ciertas pautas que, si bien no son obligatorias, es importante tenerlas en cuenta como una base que ayude a marcar la línea lógica de implementación y así lograr potencialidades como mantener la información en el tiempo y garantizar su acceso a próximas generaciones (Texier, J., De Giusti, Oviedo, Villarreal, & Lira. Buenos Aires 2012).

El grupo de Investigación en Inteligencia Artificial del Programa de Ingeniería en Sistemas de la Universidad Francisco Paula Santander de Colombia comenta que el desarrollo ha sido caracterizado por ser un proceso complejo, que requiere mucha inversión de tiempo, conocimiento y herramientas para su correcto uso. Sin embargo, con el tiempo se ha buscado reducir esta complejidad con la reutilización de código para las aplicaciones y sistemas.

A estas unidades se les conoce como componentes reutilizables y con ayuda de un repositorio donde se pueda encontrar este tipo de componentes es posible permitir catalogar, organizar, descubrir y reutilizar código que agilice la creación de sistemas en base a la búsqueda de componentes en un repositorio, combinándolos y adaptándolos, agilizando de manera significativa todo el proceso de codificación.

La arquitectura de aplicaciones ha evolucionado bastante hasta llegar a reutilizar diferentes unidades como clases, paquetes, librerías, APIS, hasta llegar a microservicios actualmente es un área de mucho interés y se viene utilizando mucho en nuevas tecnologías. (Vargas-Fandiño, J. C., Sandoval-Ramirez, J. J., & Vera-Rivera, F. H.)

La tendencia actual es que, para mantenerse actualizado con las últimas tecnologías y desarrollos, muchos usuarios transitan sitios como Reddit, stackoverflow y otras redes con la finalidad de saber qué tipo de contenido se comparte y cómo la comunidad se va adaptando a estos cambios de conocimiento. Se ha logrado apreciar que

estas comunidades tienen como objetivo mejorar sus prácticas como desarrolladores, usar nuevas herramientas que permiten encontrar mejores soluciones a nuestros problemas para así poder seguir siendo relevantes, competitivos y lograr adaptarse al ecosistema actual del desarrollo. Para lograr esto, las comunidades deben sentirse en una plataforma que les permita conocimientos modernos, tener feedback sobre sus trabajos, poder analizar el contenido que se comparte para así lograr una motivación e incentivar a que participen en ella proponiéndose nuevos retos, aprender del contenido que leen y comparten (M. Aniche, Brasil 2018)

Problemática

Desde nuestra experiencia se sabe que existe una infravaloración de trabajos y proyectos académicos de estudiantes y docentes en cada semestre, pues han desarrollado un sin número de proyectos destacables e innovadores, que al no tener un espacio dedicado en el que puedan ser almacenados sus códigos, van quedando en el olvido con el paso del tiempo y plataformas como Github en un principio puede verse un poco intuitivas de utilizar pero en un principio su complejidad y robustez hace que no la terminen de utilizar correctamente, de manera que se pierden estructuras, líneas de código, grandes ideas y una fuente de conocimiento que puede ser aprovechada.

La universidad dispone de diversas bibliotecas físicas y virtuales en las que es posible buscar información de varias fuentes que han sido publicadas por profesores y estudiantes. Estas fuentes resguardan datos y estudios relevantes para los integrantes de la institución, pero no se cuenta con un repositorio específico de código libre. En la actualidad, se vive en un mundo ampliamente influenciado por la información y la tecnología, por lo que es necesario comenzar a tener iniciativas de una biblioteca digital que los mismos estudiantes alimenten, que funcione como un repositorio donde se almacenen secciones de código o directorios completos con soluciones eficientes concebidas por los alumnos para el desarrollo de proyectos haciendo uso de una interfaz que permita filtrar y obtener la información de manera eficiente y relevante.

Justificación

El presente estudio creará una tendencia hacia el futuro, en la cual cada vez se buscará ser más competitivo y creativo entre los estudiantes, con la visión y alcance que se tengan en primera instancia de los trabajos que tengan más potencial durante la carrera.

Esto no solo permitiría entrar en un ciclo de mejora continua, sino que también brindaría mucho conocimiento, diversidad y valor a la carrera junto con un mejor nivel de abstracción.

Logrando establecer una base de conocimiento, la cual pueda ser usada para encontrar contenido, guiar a los mismos estudiantes, aclarar dudas y resolver problemas con ayuda de datos históricos que aporten valor a las diversas necesidades que pueden surgir, además de establecer un medio de comunicación entre estudiantes para conseguir una retroalimentación en cada proyecto y conectar ideas para mejorar constantemente.

Objetivos

Objetivo general

Crear un sistema de socialización de proyectos y trabajos del área de sistemas, mediante los conocimientos adquiridos para permitir que los estudiantes tengan un medio de comunicación y socialización de sus proyectos con el fin de mejorarlos y compartirlos con la sociedad.

Objetivos específicos

- Analizar los requerimientos necesarios para crear la plataforma de socialización de proyectos, tomando en cuenta las complicaciones y preparar respuestas para las posibles eventualidades. Además de realizar las validaciones pertinentes para la publicación de documentos y proyectos.
- Diseñar un espacio en el cual los alumnos puedan compartir contenido como trabajos o proyectos de forma sencilla, alojar proyectos y tener la posibilidad de comunicarse con personas que tengan interés en su trabajo.
- Construir un ambiente el cual permita dar continuidad y promoción a proyectos y trabajos para incentivar al cuerpo estudiantil a ofrecer al mundo su contenido de calidad.
- Realizar distintas pruebas para verificar la calidad del software mediante métricas y tratando de apegarnos a los estándares conocidos.

CAPÍTULO I FUNDAMENTO TEÓRICO

Metodología

Para el desarrollo de software se debe usar metodologías ágiles, debido a que permiten adaptar nuestras condiciones de trabajo a las necesidades del proyecto, brindando así flexibilidad y eficacia en productividad.

Tabla 1

Cuadro Comparativo de Las Metodologías Ágiles

Metodologías Ágiles	Scrum	Kanban	XP	FDD
Permite al equipo de desarrollo ser productivos y competitivos	Si	Si	Si	Si
El equipo de desarrollo se autogestiona y efectúa las actividades de manera autónoma	Si		Si	Si
Se realizan procesos iterativos que permitan la constante revisión, evaluación y actualización de funcionalidades	Si		Si	Si
Es posible monitorear y gestionar el progreso del proyecto mediante herramientas	Si	Si	Si	Si
Existe una gestión de posibles bloqueos puedan tener los integrantes del equipo	Si	Si		
El cliente tiene un papel activo en el proceso de desarrollo, y se involucra para evitar malentendidos	Si		Si	
Gestiona el proceso de modificaciones y cambios solicitados por el cliente en periodos	Si	Si	Si	

prudentes de tiempo				
Tiene la capacidad de ayudar con la creación de proyectos escalables de diferente alcance y tamaño	Si			

Nota: Esta tabla compara las diferentes metodologías ágiles en base a los requerimientos del proyecto para ver cual metodología es más factible.

Sobra decir que todas las metodologías ágiles son útiles en escenarios de desarrollo de software, sin embargo, se diferencian en el nivel de adaptabilidad que brindan frente a los diferentes tamaños y arquitecturas de los proyectos actuales.

Las características presentadas en la tabla demuestran las necesidades generales que se encuentran en el trabajo de titulación, donde el cliente (docente tutor) está involucrado con un papel activo en el proceso de desarrollo, es un proyecto con varias funcionalidades, que necesitan una revisión y evaluación pertinente antes de continuar con un próximo punto.

El proyecto debe ser escalable e ir juntando funcionalidades de manera periódica y controlada, para evitar errores de compatibilidad o remplazos de componentes de código clave para funcionalidades implementadas en iteraciones pasadas, sin embargo, el código debe ser reusable y mantenerse estandarizado.

Por las necesidades y características expuestas se decide tomar como metodología escogida a la metodología ágil Scrum que nos facilitará la gestión de iteraciones, modificaciones y nos permitirá crear un proyecto escalable mucho más robusto y estructurado.

1.1 Marco Institucional

La Universidad Politécnica Salesiana, actuando como una institución de educación superior la cual está promovida por la Sociedad Salesiana del Ecuador, es una comunidad académica que promueve el desarrollo de las personas y el patrimonio cultural de la sociedad en base a la investigación, la docencia y vinculación social (IUS, 2016).

Desde la perspectiva del desarrollo humano y de la formación de un potencial perfil de estudiante de la UPS, el desarrollo de las capacidades es libremente elegidas para ser y formar un proyecto de vida con libertad. Esto con la carrera de Ciencias de la Computación permite relacionar muy bien los sistemas computacionales y la sociedad para conseguir una funcionalidad optima en desempeño y utilidades que se le puede dar a las soluciones informáticas.

1.2. Marco Teórico o Referencial

Evalúa el impacto social que puede llegar a tener el auto aprendizaje o la investigación para facilitar el diseño y construcción de sistemas computacionales.

En cuanto a los diversos tipos de proyectos, refiriéndose a proyectos académicos, integradores o de titulación, se crea la posibilidad de organizar mejoras a proyectos existentes, promoviendo la interpretación de código, una habilidad con la que deben contar los desarrolladores en el campo laboral, puesto que el uso de documentos almacenados en repositorios o en tiendas de templates o plantillas, también cuando tomamos proyectos en pleno desarrollo o que fueron pausados y debemos darle continuación, una vez desarrollada esta idea clave.

De la misma manera, no solo se incentiva prácticas que los estudiantes necesitan desarrollar, también les da la posibilidad a los docentes en pensar en ideas nuevas, no solo

reusar proyectos todos los años, por el contrario, van a idear funcionalidades que sean un reto posible para los nuevos estudiantes mientras se va actualizando, refactorizando y alimentando proyectos destacables, así los estudiantes realizaran entregas de calidad que también podrían evolucionar y ser de alta importancia para la universidad.

1.3. Herramientas de desarrollo

1.3.1 MongoDB Cloud

Es un sistema de gestión y administración de bases de datos de código abierto, software el cual es perfecto para almacenar gran volumen de datos y los interpreta como colecciones y documentos. Esto permite tener gran escalabilidad, disponibilidad y flexibilidad debido a que es una base de datos que no sigue ningún esquema. (Gómez Pedro, 2020).

Mongo DB está basado en bases de datos no relacionales NoSQL, las cuales cuentan con características específicas que presentan este tipo de bases como una mejor alternativa para conformar el almacén de datos del presente proyecto, entre los primeros puntos se encuentra la agilidad o velocidad de desarrollo, ya que permite conexiones y segmentación de manera sencilla al momento de desarrollar, la base adapta a los proyectos y a las actividades DevOps y microservicios, es sumamente compatible con las metodologías ágiles, sobre todo a Scrum, se considera que el modelo de datos tabular no se adapta correctamente a las necesidades del proyecto.

Como desarrolladores se trabaja con aplicaciones que rápida y constantemente van a estar cambiando, además de que los datos utilizados pueden ser estructurados, semiestructurados o no estructurados, por lo que una base de datos estructurada no suplir las necesidades de velocidad adaptiva y tampoco puede albergar todos los tipos de datos que necesitan ser recolectados.

La base de datos necesita ser de alta disponibilidad o HA (High Availability) ya que actualmente el software se presenta a los usuarios como servicio, por lo que los datos

deben estar constantemente disponibles, pues una caída de la información va a detener directamente el funcionamiento de la aplicación o el servicio que se brinde a través de ella.

Este tipo de arquitectura no estructurada no solo brinda flexibilidad, también presentan un mejor desempeño al interactuar con aplicaciones, por todas las características presentadas se determinó la opción de utilizar MongoDB, la cual se basa en datos no tabulares que se adaptan de mejor manera al desarrollo de aplicaciones y toma en cuenta las necesidades que están deben cubrir. (MongoDB, 2018)

MongoDB se usará para la creación de colecciones que contendrán todos los datos y registros de la información que estemos consumiendo de la página web, permitiendo así tener un gran registro de datos que se encuentran almacenados en la nube y que podremos consumir a conveniencia. También haciendo el uso de bases de datos no relacionales se puede escalar de mejor manera la data logrando que se adapte a las necesidades del proyecto de una forma rápida y sencilla.

1.3.2 Express

Es un framework orientado al backend de Node.js el cual es un ecosistema de Javascript bastante completo para diseñar aplicaciones web dinámicos. Este framework es flexible y rápido ofreciendo un desarrollo más rápido, siendo así muy escalable con los años (Kinsta, 2022)

Este framework nos proporciona herramientas para la integración con motores de renderización de 'Vistas' para la generación de respuestas mediante la introducción de datos.

Establece ajustes de la aplicación como puertos de usar, la localización de plantillas que se renderizan como respuesta a eventos. Añade procesamiento de peticiones dependiendo del camino definido por las rutas HTTP.

Express se usará para gestionar todas las peticiones y rutas de la aplicación, esto permitirá que secciones de código se ejecuten antes de que el usuario final reciba una respuesta y tener controlada toda la sección de vistas de la página web mediante la implementación de rutas que hacen uso de parámetros.

1.3.3 React

Es una librería de JavaScript muy usada para sitios web y móviles, React usa y tiene una gran colección de fragmentos de código que se les conoce como componentes y son reutilizables para agilizar la creación de interfaces de usuario, el cual combinado con virtual DOM permite tener una mayor velocidad de las páginas (Deyimar, 2023).

- Axios: Es una librería que permitirá realizar solicitudes de tipo HTTP y está basada en promesas es compatible con Node JS, se lo suele usar para crear solicitudes a un servidor y recibir una respuesta.
- React DOM: Es una librería para poder hacer una renderización de componentes para el navegador, teniendo esta la flexibilidad de uso en diferentes entornos para dispositivos móviles y de escritorio.
- React-S3: es una librería la cual permite subir archivos de cualquier tipo a Amazon AWS mediante el uso de sockets.

React es básicamente nuestro conjunto de interfaces de usuarios que nos permiten desarrollar y construir componentes reutilizables, entre estos componentes encontramos las vistas que pueden asociarse a los datos para ser manipulados y contar con una actualización sin necesidad de recargar una página. En base a esto el desarrollar en un ecosistema de React nos permitirá hacer uso de varias librerías las cuales nos dan alternativas para poder suplir casi cualquier tipo de necesidad que tenga el proyecto.

1.3.4 NodeJS

Node.js es un entorno multiplataforma para ser usado en la capa de servidor basado en JavaScript, el cual trabaja de forma asíncronas, sin bloqueos, simplificando así la comunicación y logrando un mejor rendimiento en comparación con otros servidores como lo es HTTP de Apache (S. Chiyana, 2021).

- NodeJs es una solución increíble para tareas intensivas de datos o análisis, para este proyecto se sacará provecho a estas características:
- Streaming de datos: NodeJs cuenta con módulos ya integrados para un buen flujo de datos tanto de lectura como de escritura, ayudando así a poner en orden los datos y distribuirlos sin ocasionar un bloqueo o interrupción.
- API REST: NodeJs cuenta con paquetes como Express que se usan para las aplicaciones web, esta clase de paquetes aceleran el proceso de integración de una API para que sea ligera y rápida.
- SPA: Las aplicaciones de una sola página se encargan de optimizar todo dentro de una sola ventana logrando así una experiencia muy parecida a la de una aplicación de escritorio, con NodeJS se pueden gestionar todo tipo de llamadas permitiendo que estas actualizaciones se realicen sin necesidad de refrescar la página.

El uso de NodeJs es fundamental por la capacidad para crear la página web de forma dinámica haciendo uso de JavaScript y su extenso gestor de paquetes de node (NPM) que nos da la posibilidad de instalar varias dependencias. También se usará para la conexión del servidor con la aplicación cliente que genera un evento para cada petición y se gestiona de forma independiente y sin bloqueos.

1.3.5 Clerk

Es un servicio de autenticación totalmente personalizable y de código abierto que da a los desarrolladores un sistema de gestión de usuarios con una interfaz bastante completa y con una API que se integra muy bien en diferentes arquitecturas de proyectos

(GalaxiesDev).

- Restricciones: Clerk permite aplicar un conjunto de restricciones para tener un mayor control de quiénes pueden generar una solicitud de ingreso al aplicativo web, estos identificadores suelen ser direcciones de correo electrónico, dominios, entre otros.

- Factores de autenticación: Clerk cuenta con varias funcionalidades personalizables para reforzar las capas de seguridad en los inicios de sesión (SMS, correo, etc.), en este caso se usará la de código de verificación de correo electrónico cuando se inicia sesión desde un nuevo dispositivo

- Gestión de usuarios: Mediante el uso que se le ha dado al aplicativo se ha podido observar el control de los usuarios y sus datos y configuraciones son bastante completas como lo es ver información de últimos logeos, registro, ver perfiles con datos, banear usuarios y eliminarlos a conveniencia.

Clerk permitirá al proyecto contar con un apartado para la gestión de usuarios independiente e implementar a la vez restricciones de logeo como lo es el dominio, métodos de inicio de sesión y una verificación de usuario para tener una capa de verificación de usuario. Aparte también nos brinda muchas más características que hacen escalable al proyecto en temas de seguridad.

1.3.6 Loader.io

Esta herramienta permite hacer pruebas de carga en páginas y aplicaciones web, es una plataforma basada en nube que una vez verificada la posesión de la aplicación se puede realizar dichas pruebas.

1.4 Metodología Scrum

Scrum es un esquema de trabajo que les permite a los equipos de desarrollo de cualquier tamaño a presentar entregas de valor para problemas complejos a través de soluciones adaptables, donde se necesita un scrum master o líder de equipo que fomente el realizar entregables de valor en periodos mediante sprints o ciclos, donde en cada ciclo el equipo y las partes interesadas revisan los avances para realizar ajustes o afinamiento de detalles de la solución y preparar un plan para el avance del siguiente sprint (Purpose of the Scrum Guide, 2020).

La finalidad de Scrum es mantener una estructura simple que ayude a desempeñar de mejor manera tareas para cumplir metas y entregar valor a las partes interesadas, mediante el apoyo de todos los integrantes y una comunicación tanto objetiva como asertiva (Purpose of the Scrum Guide, 2020).

1.4.1 Scrum Team

Los equipos están conformados por grupos pequeños de personas, la guía de Scrum dice “El Scrum Team está formado por un Scrum Master, un Product Owner y desarrolladores”, los equipos son multidisciplinarios con habilidades necesarias de crear valor en cada sprint, no se debe subestimar a los scrum team, a pesar de ser equipos pequeños tienen toda la capacidad de cumplir sus objetivos en el tiempo asignado ya que los equipos pequeños se comunican de mejor manera y son más eficientes” (Purpose of the Scrum Guide, 2020).

Scrum Master: Es el encargado de organizar y orientar al equipo en las actividades que deben enfocarse para conseguir cumplir las metas en los ciclos, este integrante debe conocer la metodología y aplicar correctamente la metodología, también debe ser el apoyo principal del desarrollador (Purpose of the Scrum Guide, 2020).

Desarrollador: Es el encargado de diseñar y poner en marcha los planes de acción para desarrollar correctamente el sprint y acercarse al sprint goal o a la meta del ciclo con

las actividades que realiza (Purpose of the Scrum Guide, 2020).

Product Owner: Debe comunicar cual es la meta del producto y mantener un papel activo en el desarrollo del proyecto, mantener comunicación constante con el equipo para afinar, modificar y corregir aspectos del desarrollo y conseguir que el producto sea transparente y comprensible (Purpose of the Scrum Guide, 2020).

El equipo interdisciplinario estará segmentado de la siguiente manera para el desarrollo de este proyecto:

- Scrum Master: Paul Morales
- Desarrollador: Leyton Valencia
- Product Owner: Ing. Daniel Diaz

1.4.2 Scrum Events

Los Scrum Events o eventos de scrum son grupos de actividades que se realizan de manera iterativa mediante sprints o iteraciones que se realizan a lo largo del proyecto con la finalidad de entregar valor en cada uno. La guía de Scrum, 2020 dice: “Los sprints son los latidos del corazón de scrum”, cada ciclo comienza donde la anterior termina y se realizan de esta manera para ser un equipo más efectivo e ir racionando el trabajo para cumplirlo en el tiempo estipulado. Cada sprint tiene:

Sprint Planning; donde entre todo el equipo de Scrum se planea un sprint completo en base a las siguientes preguntas.

- ¿Por qué es valioso un ciclo de Scrum?
- ¿Qué tareas se pueden hacer en este ciclo?
- ¿Cómo se van a realizar las actividades de este sprint?

Mediante estas preguntas es posible dar forma e idear un plan para el sprint que se debe realizar, donde una vez definida la finalidad del sprint se toma actividades específicas para realizar, además de mediante conversaciones llegar a un acuerdo del tiempo estimado en el que se debe entregar y definir cómo se identifica que el sprint se cumplió, es decir, se define un valor calculable que se debe completar.

Daily Scrums; son reuniones diarias donde se hace un análisis de los avances del equipo en relación a la meta establecida, identificar el nivel de avance y los posibles bloqueos con los que se puede encontrar el equipo, estos bloqueos pueden ser de cualquier índole, de la misma manera ayuda a ajustar el plan si fuese necesario.

Sprint Review; se realiza al final del sprint e identifica las cosas que se hicieron bien y que se pueden mejorar porque se hace un análisis de todo lo que se realizó, y que cosas podrían mejorar para el siguiente sprint, además de con todo el equipo identificar si se cumplió la meta del ciclo.

Sprint Retrospective; similar al Sprint Review se hace un análisis del proceso

pasado, pero al contrario muestra las actividades que se realizaron de mala manera y se deben corregir para el siguiente ciclo, de esa manera se llega a la meta del ciclo de mejor manera.

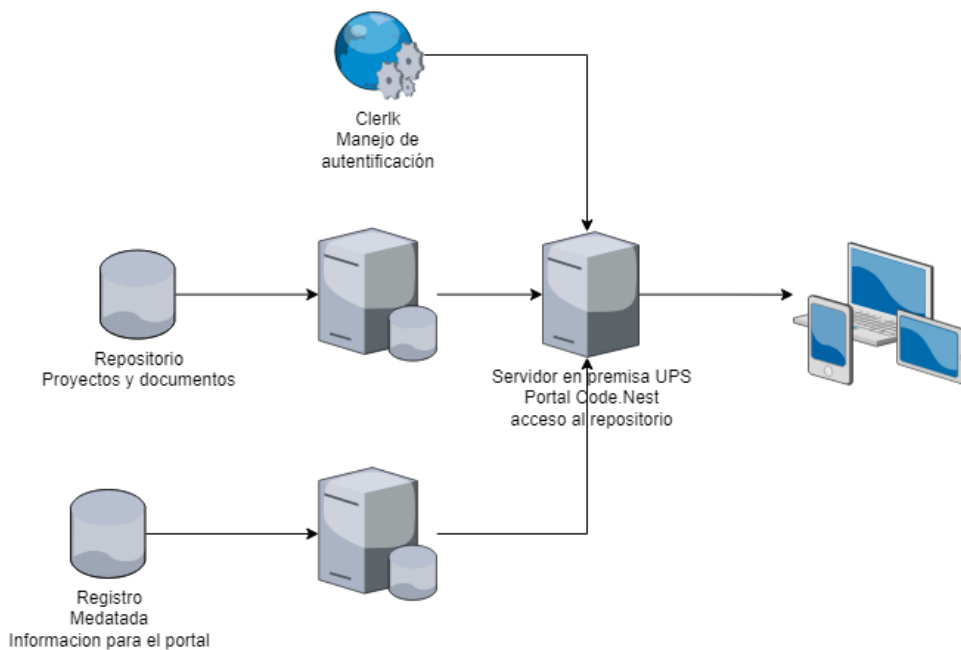
1.5 Sistemas de gestión para el repositorio de proyectos

1.5.1 On-Premise

Es posible gestionar el repositorio en un sistema físico presente en la institución, en donde se utilizan los recursos de la universidad, mediante el virtualizador que existe en la universidad crear máquinas virtuales donde se va a instalar sistemas operativos de servidores Windows server 2019 Standard donde se almacenará tanto la base de datos como el back-end del proyecto y desde donde se publicará la aplicación.

Figura 1

Arquitectura en premisa con almacenamiento en nube



Nota: Este es un diagrama de la arquitectura de un sistema on-premise.

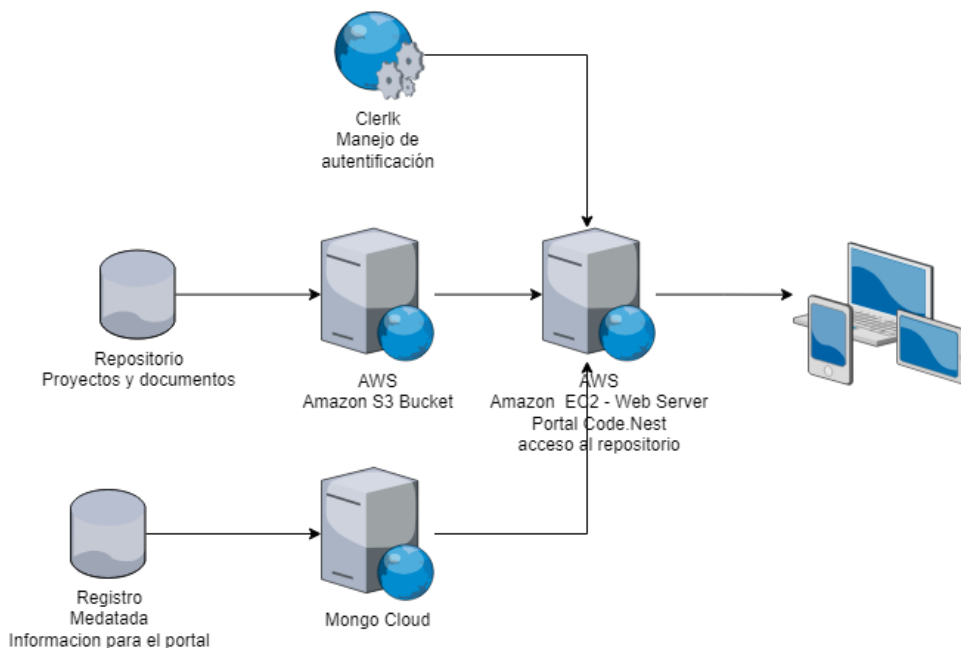
En esta arquitectura se muestra en una primera instancia, la gestión de autenticación se realiza mediante un api de Outlook 365, el repositorio se almacena en una base de datos dedicada al manejo de documentos, mientras que los metadatos e información de cada publicación se almacenará en la base de datos en Mongo DB Cloud.

1.5.2 Cloud

Una arquitectura en nube pública donde tanto el almacén de datos como la publicación de la aplicación se desplegará desde los recursos de unas de las nubes que se encuentran en el mercado, puede ser AWS, Azure, Google Cloud, donde, además de organizar un sistema para gestionar la información del repositorio de proyectos, también es posible configurar un repositorio de alta disponibilidad HA.

En cuanto a alta disponibilidad o, por sus siglas en inglés, HA o High Availability ,se refiere a la posibilidad de mantener siempre disponible la aplicación desarrollada al igual que los datos de manera que, si la infraestructura disponible llegara a fallar, una segunda arquitectura idéntica se despliegue instantáneamente a suplir la caída de la primera arquitectura.

Figura 2
Arquitectura en nube – ejemplo AWS



Nota: Este es un diagrama de la arquitectura de un sistema en nube que usa AWS.

Aquí se puede encontrar una arquitectura similar a la presentada anteriormente, donde la diferencia principal se encuentra en que el almacenamiento y la instalación del servicio se encuentra en instancias de AWS, el manejo de autenticación se mantiene en un api de Microsoft 365 y el almacenamiento de metadatos e información de las

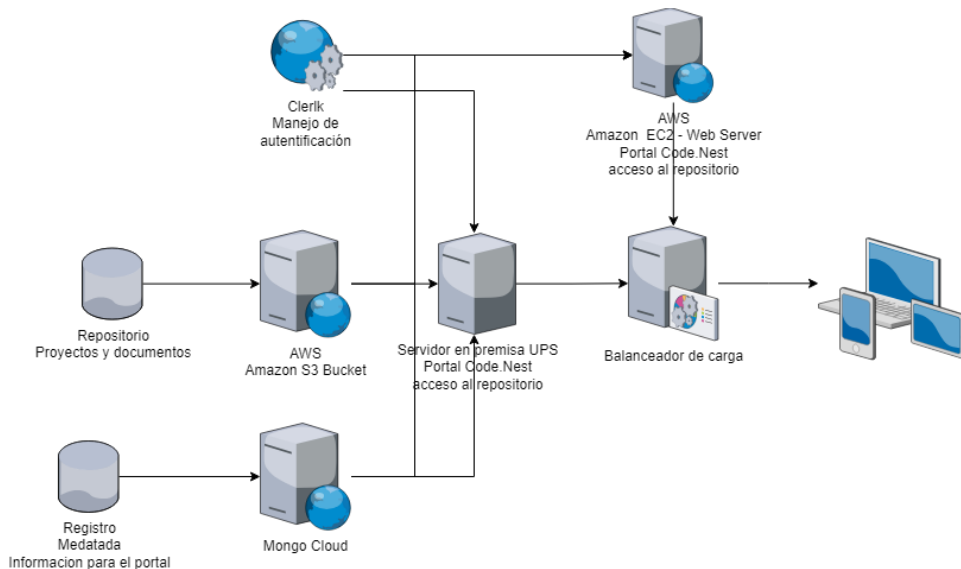
notificaciones se almacenaran en Mongo Cloud.

1.5.3 Híbrido

Resumiendo, los dos puntos anteriores es posible hacer una arquitectura híbrida o también llamada multi-nube donde no solo los datos o servicios se encuentren desplegados en sistemas distintos y separados, sino que también cuenta con capacidades de HA donde, a pesar de que caiga la arquitectura en premisa, se despliegue automáticamente una segunda arquitectura en nube

Figura 3

Arquitectura híbrida con alta disponibilidad



Nota: Este es un diagrama de la arquitectura híbrida o multi-nube.

Aquí se puede observar una mezcla entre la arquitectura on-premise y cloud, donde se crean dos instancias idénticas para publicar la aplicación y el almacenamiento alimenta tanto al servidor en premisa como al servidor en nube, esta arquitectura necesita un balanceador de carga para identificar el estado de salud de las instancias y redireccionar el tráfico a la instancia más saludable.

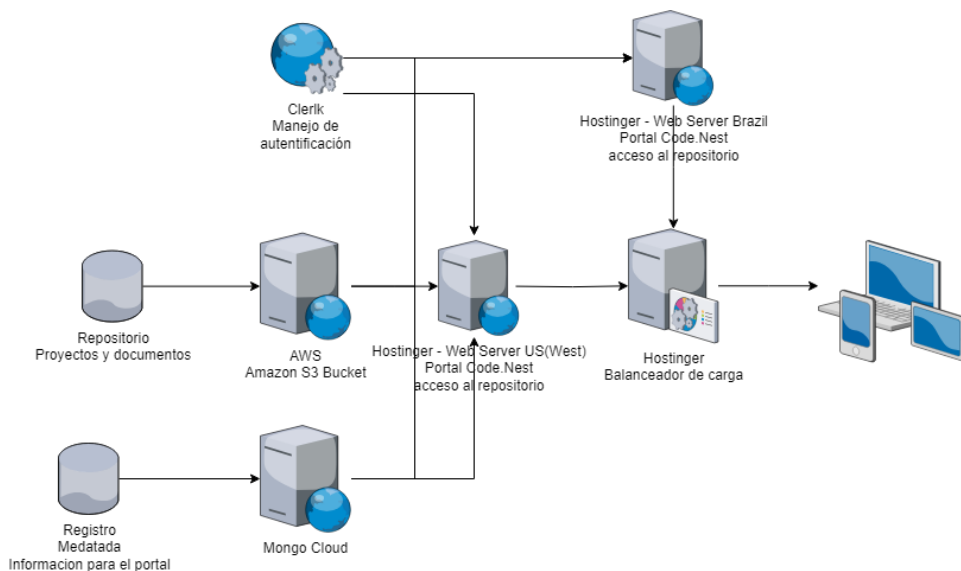
1.5.4 Mediante terceros

Cuando se realiza contratos con terceros para el almacenamiento de datos y hosteo de la aplicación no solo se escala la necesidad de infraestructura, sino que también se cuenta características de seguridad dependiendo del proveedor, de la misma manera le va a brindar capacidades de HA ya que en proveedores como Hostinger que van años en este negocio cuentan con varios servidores que van a mantener siempre activas las aplicaciones de sus clientes.

También el hostear aplicaciones mediante terceros no solo brinda capacidades de disponibilidad y seguridad, sino que también brinda acceso global al repositorio, dando la oportunidad de que los estudiantes accedan desde cualquier punto como los laboratorios, hogares o dispositivos móviles, esto incluye certificaciones de seguridad necesarias para la página.

Figura 4

Arquitectura de alta disponibilidad por región - Hostinger Cloud



Nota: Este es un diagrama de la arquitectura con la cual Hostinger genera una alta disponibilidad en sus servicios.

Esta arquitectura, aunque similar a la que se presentó en el punto anterior no solo se basa en el estado de salud de dos servidores, sino que depende del número de servidores presentes en la región que se encuentren disponibles para presentar la aplicación, de la misma manera las bases de datos y el manejo de autenticación se realiza en servicios en nube.

1.6 Arquitecturas para un sistema de gestión de repositorio de proyectos

Cliente – Servidor: Las arquitecturas cliente servidor son sumamente conocidas por su versatilidad y flexibilidad, segmenta de diferentes maneras la aplicación de usuario para que las aplicaciones sean ligeras y rápidas, además de gestionar de manera unificada los datos, pues estos se almacenan en el servidor, mientras que las sesiones con las que interactúan los usuarios solo son una conexión al servidor para mejorar el desempeño de la aplicación y la seguridad de los datos (Davey & Tatnall, 2002).

Para Davey y Tatnall (2002) la arquitectura viene en diferentes presentaciones, donde el servidor es el encargado de realizar casi todo, de manera que el cliente solo se encarga de la presentación de estos datos, y los componentes se presentan en las siguientes capas:

- Manejo de datos
- Lógica de la aplicación
- Presentación

Según Gartner, presentado por Davey y Tatnall (2002), los modelos en los que se puede distribuir estas capas dependen del nivel en el que se involucra el cliente en el funcionamiento de la aplicación, a continuación, se listan de menor a mayor participación del cliente:

- Presentación distribuida;
 - El servidor se encarga tanto del manejo de datos, la lógica y parte de la presentación.

- Mientras que el cliente solo se encarga de la la parte complementaria de la presentación necesaria para el correcto funcionamiento de la aplicación.
- Presentación remota;
 - El servidor se encarga del manejo de datos y la lógica de la aplicación.
 - El cliente se encarga de la presentación de la app en su totalidad.
- Lógica distribuida;
 - La lógica de la aplicación se distribuye entre el servidor y el cliente, de manera que algunos módulos y cálculos se realizan en el cliente para comunicarlos con el servidor.
 - El manejo de datos se alberga completamente en el servidor.
 - La presentación se maneja por completo de forma remota en el cliente.
- Manejo de datos remoto;
 - El manejo de datos se encuentra por completo en el servidor
 - La presentación y lógica de la aplicación se manejan por completo en el cliente mientras que el acceso del cliente a los datos almacenados en el servidor se realiza de forma remota.
- Base de datos distribuido;
 - El cliente es el encargado de manejar la mayor carga de trabajo puesto que además de albergar tanto la presentación como lógica de la aplicación también es el encargado de parte del manejo de datos
 - Mientras que el servidor solo se encarga de manejar el segmento de datos faltante.

1.7 Métricas de calidad de software para un Sistema de Repositorio.

En un concepto, la calidad se refiere a las expectativas y cualidades que puede buscar un cliente y es justamente por esto que para producir software de valor y calidad que cumpla con las expectativas se hace uso de diferentes métricas de software (Arias et al., 2023).

Las diferentes categorías de métricas pueden centrarse en estos puntos:

- Productividad del desarrollador
- Rendimiento del Software
- Defectos y seguridad
- Experiencia de usuario (UX)

Métricas de productividad

- Tiempos de entrega
- Cantidad de código
- Número de versiones del software

Métricas de rendimiento

- Tiempos de respuesta
- Fiabilidad
- Escalabilidad
- Disponibilidad

Métricas de Defectos

- Número de Vulnerabilidades de seguridad
- Porcentaje de detección de defectos
- Tiempos medios de detección y reparación de fallos

Métricas de usabilidad (UX)

- Pruebas de usabilidad cuantificadas
- Monitoreo de la sesión del usuario

1.8 Métodos de seguridad aplicados a sistemas de repositorios

Sabemos gracias a Tejena-Macías (2018) que indica que hoy en día los ataques cibernéticos han aumentado de manera acelerada, tanto en volumen como en sofisticación, las amenazas son diversas y en diferentes formas, por lo que siempre se debe tener mapeadas estrategias para analizar y responder frente a las amenazas presentes, previniendo, controlando y reduciendo los riesgos que se encuentren enlazados las vulnerabilidades de su información.

Existen leyes que se han creado con la finalidad de proteger los datos de los ciudadanos, en marzo del presente año del desarrollo de este documento se puso en vigencia la Ley Orgánica de Protección de Datos Personales, donde se desplegaron un grupo de políticas para salvaguardar la información personal de usuarios y clientes, además de involucrar más a los usuarios a identificar y conocer de qué manera se están tratando sus datos, esto les da la responsabilidad a los gestores de páginas donde se manejan datos delicados si llega a suceder un tipo de ataque donde estos datos sean vulnerados de cualquier manera, es decir, si estos datos fueron divulgados, robados o encriptados.

De la misma manera existen los ataques de denegación de servicios y ataques de denegación de servicios distribuidos, los cuales son ataques de capa 4 que tienen como finalidad tirar los servicios levantados mediante solicitudes de alto volumen de datos repetitivas que tienen como finalidad sobrecargar la capacidad del servidor, estos pueden realizarse de un solo dispositivo o de varios dispositivos simultáneamente generando un ataque distribuido DDOS.

Los siguientes métodos de seguridad están implícitos programados en el sistema o se prestan en el servicio de host Hostinger, servicio que será contratado para cubrir tanto requerimientos de disponibilidad, como de seguridad.

1.8.1 Método 1: Protección contra NoSQL Injection

En la actualidad existe un gran número de amenazas, pero la más común en páginas web es la inyección de código SQL y NoSQL, para romper seguros de autenticación y consulta de datos mediante las solicitudes que hace el cliente en la base, con este tipo de ataques se compromete la información de los almacenes de datos, información sensible que se debe tratar con cuidado, puesto que, en la actualidad los datos de usuarios son de mucho valor, por lo que es normal que estén en peligro de ser vulnerados.

1.8.2 Método 2: Firewall perimetral

Este dispositivo es un equipo robusto encargado de proteger a los servidores donde se albergan los servicios y datos, con el fin de detectar y detener un amplio abanico de amenazas, con políticas, configuraciones y monitoreo la telemetría del estado de una red y de los dispositivos que se encuentran en ella, de esta manera se consigue las siguientes características protección:

- Protección contra Ataques DOS y DDOS
- Protección contra Ataques de Spoofing, este tipo de ataques busca suplantar tanto la ip como el DNS con un sitio alternativo para robar datos mediante esta suplantación de aplicación.
- IPS: Intrusion Prevention System, o en español sistema de prevención de intrusiones.
- Supervisión del estado de salud de los servidores donde se encuentran los servicios

Estos dispositivos son costosos, pero uno de los beneficios de hostear la página

por terceros es conseguir estos beneficios, puesto que el proveedor como Hostinger cuenta con este tipo de protecciones (¿Qué medidas de seguridad usa Hostinger? | Centro de ayuda de Hostinger, s. f.).

1.8.3 Módulos de seguridad avanzados

“A nivel de aplicaciones existen diversas protecciones que aseguran la mejor protección posible” (Seguridad | Centro de ayuda de Hostinger, s. f.) como:

- mod_security
- Suhosin PHP hardening
- PHP open_basedir protection y otros.

1.8.4 Protección antimalware en terminales y servidores

Protección contra virus y amenazas conocidas, a nivel de servidor, este tipo de protecciones se consigue mediante el escaneo de agentes para identificar y responder frente a archivos y documentos malignos para evitar amenazas como ransomware, spyware y demás tipos de malware conocido y desconocido (Seguridad | Centro de ayuda de Hostinger, s. f.).

1.8.5 Procedimientos y prácticas

No basta solo con contar con agentes de protección, los puntos más importantes son los procedimientos y prácticas necesarias para proteger el servicio y datos, Centro de ayuda de Hostinger especifica lo siguiente:

- Un equipo SOC dedicado.
- Respalda la información mediante procesos y políticas internas.
- Búsqueda de vulnerabilidades continua mediante pruebas de intrusión.
- Impulsar la búsqueda de errores y crear políticas de DLP.
- Estandarizar la codificación para mitigar futuras vulnerabilidades, creando

aplicaciones más seguras.

- Considerar el uso de MFA donde sea posible aplicarlo.

1.8.6 Integridad de los datos

Anteriormente ya expusimos la importancia de protección de datos, por lo que el Centro de ayuda de Hostinger declara que el proveedor realiza las siguientes acciones para la protección de información:

- Procurar la actualización constante
- Usar algoritmos que cifren la información
- Mantener la respectiva copia de seguridad de la información almacenada.
- Analizar el código de cada aplicación de manera constante para crear una aplicación más robusta.

1.8.7 Certificaciones

Centro de ayuda de Hostinger realiza un disclaimer por no contar con una certificación específica, pero detalla que cuenta con diferentes prácticas para proteger su información, se presenta en los siguientes puntos:

- Hostinger no cuenta con la certificación ISO 27001 o un equipo SOC dedicado, pero sí cuentan con varias prácticas de seguridad para garantizar la integridad de la información.

CAPÍTULO II ANÁLISIS Y DISEÑO

2.1 Encuestas al cuerpo estudiantil (Docentes y estudiantes)

Este proyecto se inició con la finalidad de ayudar y ser útil para estudiantes y docentes, con esta premisa antes de definir el comportamiento y funcionalidad del presente trabajo de titulación, se encuestó a varios alumnos y docentes con la finalidad de tomar recomendaciones, la perspectiva de varios puntos de vista.

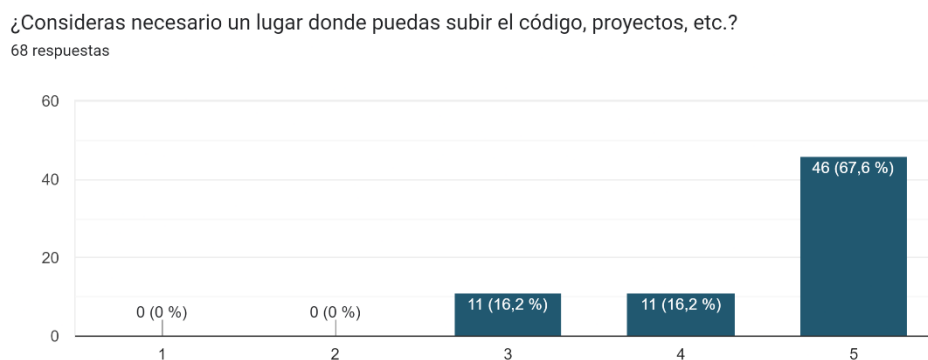
La entrevista en cuestión se realizó vía Microsoft Forms para que sea cómoda de realizar y no tome demasiado tiempo y es totalmente anónima para proteger la identidad de las personas encuestadas, a continuación, se va a presentar de manera tabulada y con un análisis del resultado, al momento del registro de resultado se tiene un total de 68 respuestas almacenadas equivalentes al 24% de estudiantes de una carrera que en total son 280 alumnos.

2.1.1 Interpretación de respuestas

1) ¿Consideras necesario un lugar donde puedas subir el código, proyectos, etc.?

Figura 5

Gráfico de respuestas a pregunta #1



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #1.

Se puede apreciar que para un 67,6% de la muestra de estudiantes es necesario un espacio seguro donde puedan almacenar sus trabajos y proyectos técnicos independiente de las plataformas y modos actuales de guardar información.

2) ¿Te gustaría subir tus proyectos a la nube?

Figura 6

Gráfico de respuestas a pregunta #2



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #2.

Para el 94.1% de encuestados este espacio o repositorio además de ser seguro debe estar en nube y poder acceder vía internet.

3) ¿Porque no consideras importante subir los proyectos a la nube?

Esta pregunta se encuentra oculta en la encuesta, una vez marcado no en la pregunta anterior, se habilita para comprender por qué no es una buena idea crear este repositorio.

Las respuestas fueron y cito:

- “Por qué, pueda que coincidan y estudiantes lo utilicen en forma de copia” <anónimo>
- “porque pienso que en caso de ser algo muy importante, lo tenga siempre bajo mi poder” <anónimo>
- “Se puede tener respaldos físicos” <anónimo>
- “Por qué puede existir plagio” <anónimo>

Todos los comentarios permiten entregar un mejor trabajo, de esta manera es posible tener un panorama amplio como se comentó en un principio, sin embargo, se debe analizar estos comentarios por partes:

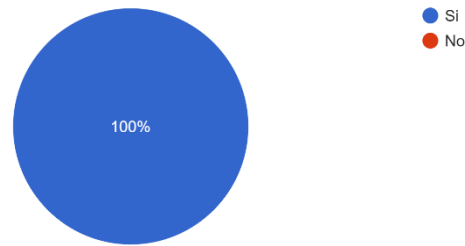
- Usar el repositorio para realizar plagio de código
 - La preocupación de que el presente repositorio se use a manera de plagio esta injustificado, ya que en internet existen códigos creados y compartidos por millones de personas que ya desarrollaron las mismas actividades en el pasado, y no es diferente a un repositorio de artículos académicos, en dichas bibliotecas también hay información sobre las clases que dictan los profesores y sería impensable no documentar estos escritos y avances.
 - El hecho de que haya miles de códigos compartidos por internet hace sumamente complicado identificar copia en códigos, de hecho, el repositorio puede ser una solución para atacar esta actividad, al tener toda la información centralizada y a la mano de estudiantes y profesores se brinda la posibilidad de crear a manera de proyecto programas para identificar intentos de copia, plagio y similitud en la manera de codificar de los estudiantes.
- Tener la información bajo el poder del usuario final en caso de ser algo importante
 - Es comprensible esta preocupación por ello se implementará la oportunidad de subir proyectos tanto públicos como privados, para que el usuario pueda respaldar su información y solo sea disponible para el alumno o docente a manera de backup no debe compartir sus datos de manera obligatoria.

4) ¿Consideras importante que puedas compartir tus proyectos con otros compañeros de la institución de algún momento?

Figura 7

Gráfico de respuestas a pregunta #4

¿Consideras importante que puedas compartir tus proyectos con otros compañeros de la institución de algún momento?
68 respuestas



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #4.

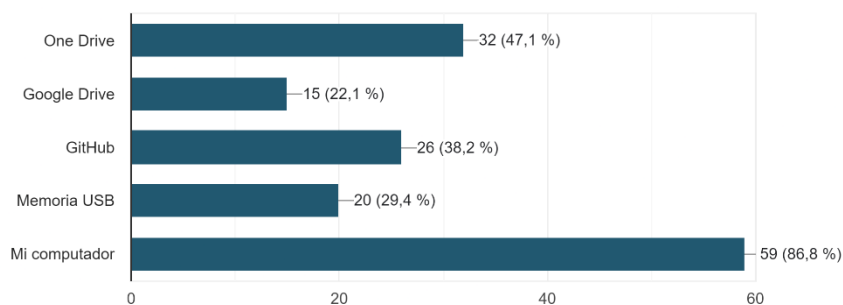
Aquí se consiguió una respuesta unánime, todas las personas encuestadas consideran que es importante compartir sus proyectos con sus compañeros.

5) ¿En dónde guardas actualmente tus proyectos?

Figura 8

Gráfico de respuestas a pregunta #5

¿En dónde guardas actualmente tus proyectos?
68 respuestas



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #5.

El 86.8% de los estudiantes guardan sus proyectos en su computadora personal, si bien es una manera útil de guardar información, es limitada y los proyectos pueden olvidarse con el pasar del tiempo.

6) ¿Cuál de las siguientes opciones consideras óptimas para subir un proyecto a la web?

Figura 9

Gráfico de respuestas a pregunta #6



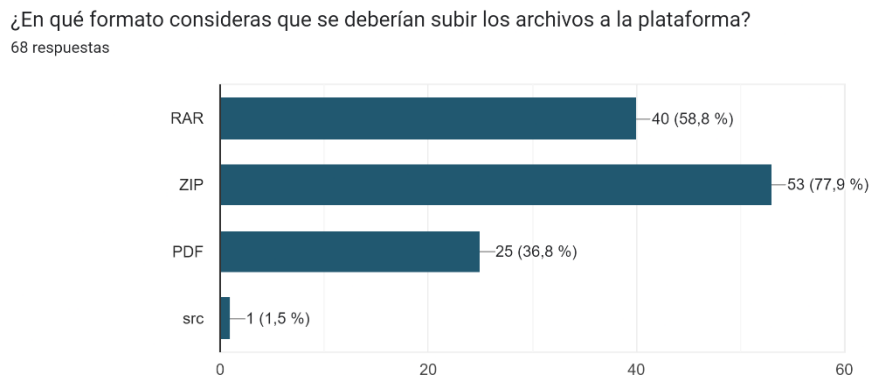
Nota: Gráfico tabulado de encuesta de estudiantes pregunta #6.

El 75% de las personas encuestadas prefieren Seleccionar o arrastrar los documentos para subir la información a la web, es comprensible porque es un método sencillo y fácil de utilizar.

7) ¿En qué formato consideras que se deberían subir los archivos a la plataforma?

Figura 10

Gráfico de respuestas a pregunta #7



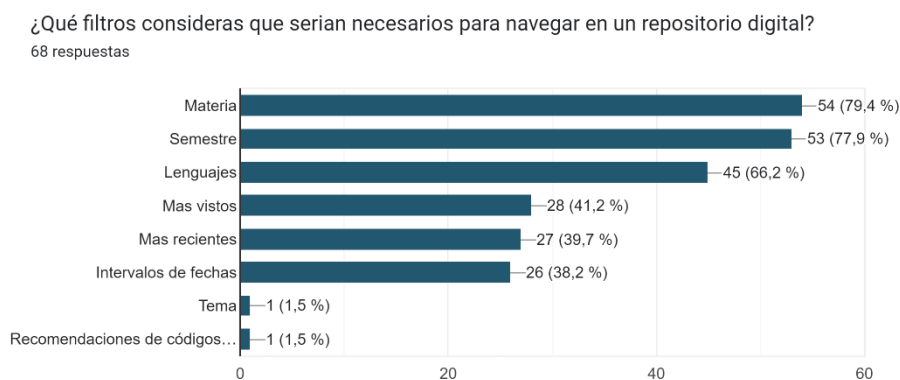
Nota: Gráfico tabulado de encuesta de estudiantes pregunta #7.

Los formatos preferidos para las personas encuestadas son ZIP con un 77.9% y RAR con el 58,8 %.

8) ¿Qué filtros consideras que serían necesarios para navegar en un repositorio digital?

Figura 11

Gráfico de respuestas a pregunta #8



Nota: Grafico tabulado de encuesta de estudiantes pregunta #8.

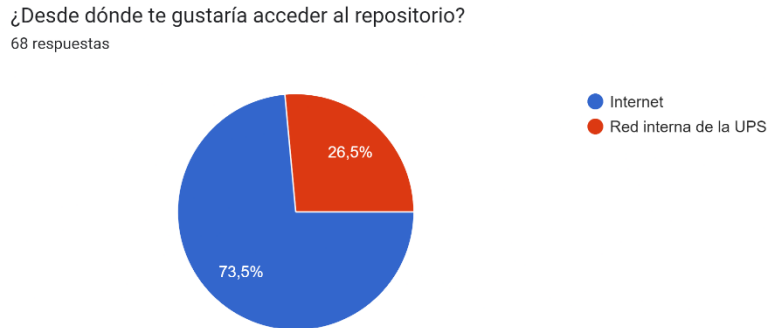
Se consideraron los siguientes filtros para ser seleccionables en el repositorio digital y permitan la navegación en la página web, de los 3 primeros superan el 50% de las respuestas y son:

- Materia
- Semestre
- Lenguaje

9) ¿Desde dónde te gustaría acceder al repositorio?

Figura 12

Gráfico de respuestas a pregunta #9



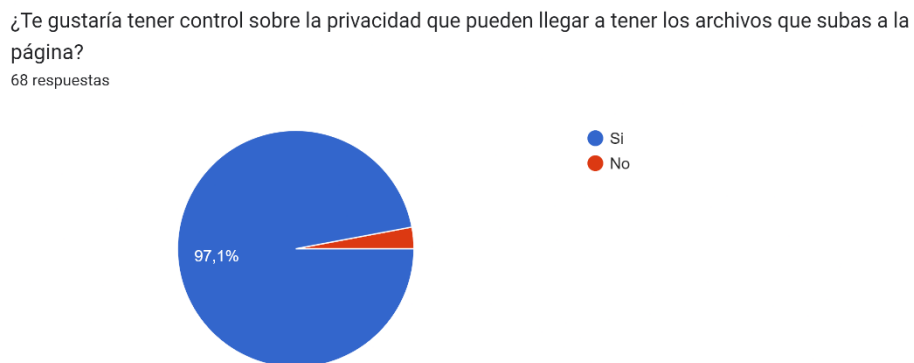
Nota: Gráfico tabulado de encuesta de estudiantes pregunta #9.

El 73% de personas buscan que el repositorio esté disponible a través de internet, es una buena opción, esto permitirá el avance y trabajo tanto dentro como fuera de la universidad.

10) ¿Te gustaría tener control sobre la privacidad que pueden llegar a tener los archivos que subas a la página?

Figura 13

Gráfico de respuestas a pregunta #10



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #10.

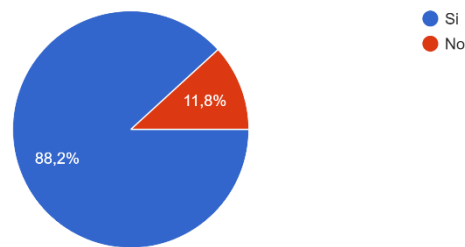
Es una respuesta comprensible, siempre es importante tener el control del nivel de privacidad que tienen las publicaciones de todos los usuarios en el repositorio.

11) ¿Crees importante un factor de doble autenticación para el inicio de sesión o validación del registro en una aplicación web?

Figura 14

Gráfico de respuestas a pregunta #11

¿Crees importante un factor de doble autenticación para el inicio de sesión o validación del registro en una aplicación web?
68 respuestas



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #11.

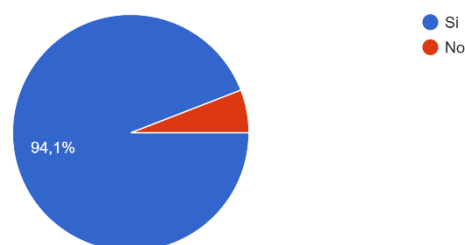
Esta opción es sumamente importante, contar con un doble factor de autenticación para proteger la información y los perfiles de cada usuario, sobre todo ahora que los ataques de robo de credenciales son cotidianos.

12) ¿Te gustaría poder interactuar de alguna manera con los autores de otros proyectos o publicaciones?

Figura 15

Gráfico de respuestas a pregunta #12

¿Te gustaría poder interactuar de alguna manera con los autores de otros proyectos o publicaciones?
68 respuestas



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #12.

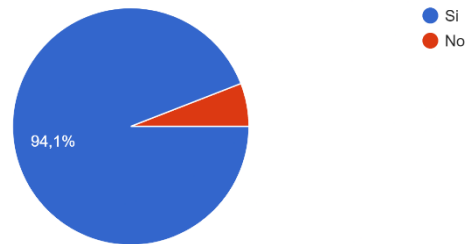
La comunicación entre creadores es un punto clave para el avance y la mejora de proyectos.

13) ¿Te gustaría contar con una sección donde puedas dejar sugerencias o comentarios acerca de tu experiencia en el aplicativo web?

Figura 16

Gráfico de respuestas a pregunta #13

¿Te gustaría contar con una sección donde puedas dejar sugerencias o comentarios acerca de tu experiencia en el aplicativo web?
68 respuestas



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #13.

La comunicación es clave, tanto entre usuarios como entre usuarios y los administradores del sitio web, para mejorar constantemente el comportamiento y las funcionalidades.

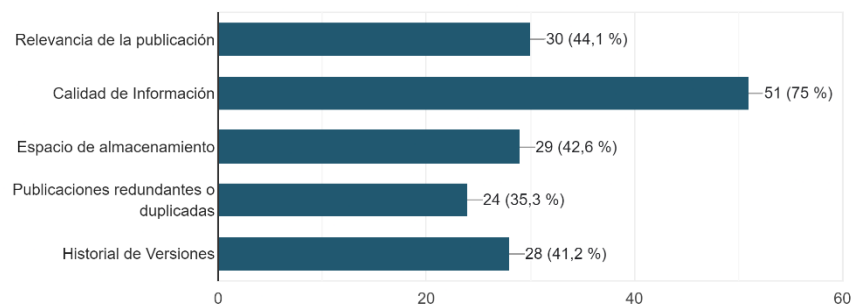
14) ¿Cuál de los siguientes métodos consideras adecuado para depurar a largo plazo la información que se encuentra en una página web?

Figura 17

Gráfico de respuestas a pregunta #14

¿Cuál de los siguientes métodos consideras adecuado para depurar a largo plazo la información que se encuentra en una página web?

68 respuestas



Nota: Gráfico tabulado de encuesta de estudiantes pregunta #14.

Los métodos para la depuración de información siempre fue una de las dudas más grandes del proyecto, sin embargo, la opción más seleccionada fue depurando a través de la calidad de información con un 75%

15) ¿Tienes alguna idea que pueda aportar al proyecto? o alguna sugerencia sobre una funcionalidad? Por favor detállanosla a continuación.

Las sugerencias presentadas por los encuestados fueron:

- Creo que una sugerencia, es que subamos los códigos en forma pdf ya que en zip es muy fácil descargarse y ya utilizarlo sin analizarlo primero
- Yo pienso que otro filtro importante, sería el área del que trata el proyecto.
- Sería genial crear repositorios de código de la carrera.
- Se podría añadir una especie de calificación (por estrellas, por ejemplo) para calificar los proyectos que se hayan subido y aquellos de 5 estrellas que puedan observarse en un apartado para ello
- Que se pueda acceder mediante secciones a los diferentes lenguajes, y puede haber una explicación del código junto con una prueba echa.
- Par de vídeos o sugerencias donde expliquen cómo organizar los proyectos para que todo sea más rápido
- Búsqueda personalizada
- Que exista una carpeta para cada materia y que se guarde automáticamente los trabajos del que trata el tema de la materia.

- Un inicio con una bandeja llena de códigos y sugerencias de proyectos o cosas que puedan ser interesante para el usuario.
- Librerías y más funciones

2.2 Requerimientos funcionales y no funcionales

2.2.1. Requerimientos funcionales

Se tomaron requerimientos clave para prototipo en producción que se va a desarrollar con puntos que se deben solventar en un inicio para un repositorio virtual, nuestra intención al desarrollar este sistema es que tanto los proyectos que se encuentren en el repositorio como el propio sistema tenga un mejor nivel de desarrollo promoviendo más desarrollos open source que no solo dependan de una organización o persona, sino que mejora constantemente con los aportes de varios programadores, por ello los requerimientos que vamos a cubrir serán los siguientes.

CÓDIGO	NOMBRE	DESCRIPCIÓN
RF001	Autenticación de Usuarios	Establecer un proceso de logeo con doble autenticación exclusivamente para usuarios del dominio de la UPS.
RF002	Gestión de Publicaciones	Implementar un panel que permita al usuario controlar las publicaciones que tiene y su privacidad de manera efectiva.
RF003	Búsquedas Personalizadas	Ofrecer a los usuarios la posibilidad de utilizar filtros para mejorar la eficiencia en la búsqueda de archivos y publicaciones.
RF004	Carga y descarga de Archivos ZIP/RAR	Permitir a los usuarios cargar y descargar información en formatos .zip o .rar, manteniendo la estructura original de los datos.
RF005	Comentarios Interactivos en Publicaciones	Facilitar la interacción entre estudiantes permitiendo comentarios en las publicaciones para compartir opiniones y brindar retroalimentación constructiva.

Tabla 2: Requerimientos funcionales

2.2.2. Requerimientos no funcionales

Para los requerimientos funcionales nos basamos en los requerimientos más solicitados en la mayoría de páginas y repositorios web.

CÓDIGO	NOMBRE	DESCRIPCIÓN
RNF001	Tiempos de Respuesta Rápidos	El sistema debe responder rápido al cargar y descargar archivos, con un límite de espera específico.
RNF002	Manejo Concurrente de Usuarios	La plataforma debe soportar muchos usuarios al mismo tiempo sin volverse lenta.
RNF003	Cifrado de Datos en Repositorio	La información guardada debe estar segura mediante cifrado, tanto cuando se envía como cuando está guardada.
RNF005	Interfaz Intuitiva	La forma en que interactúas con el sistema debe ser fácil de entender y aprender, especialmente para nuevos usuarios.
RNF006	Compatibilidad Multiplataforma	La plataforma debe funcionar bien en diferentes navegadores y dispositivos para que sea accesible y fácil de usar para todos.
RNF007	Disponibilidad de la pagina	El sistema deberá garantizar que los usuarios puedan acceder y recuperar información desde cualquier sitio en cualquier momento.
RNF008	Controles de Acceso	Solo las personas autorizadas deben poder acceder a la información almacenada, con medidas de seguridad adecuadas.

Tabla 3: Requerimientos no funcionales

2.3 Historias de usuario

Se creó una sección de historias de usuarios para poder desglosar los pasos para el correcto funcionamiento de cada requerimiento, así mismo como las posibles variantes que pueda generar cada usuario para ser validadas.

2.3.1 Historia de usuario: RF001 - Autenticación de Usuarios

CÓDIGO	RF001
ACTOR	Estudiante, docente
Flujo Normal	<ol style="list-style-type: none">1. El usuario ingresa a la sección de logeo2. El sistema responde cargando la interfaz correspondiente3. El usuario accede a través de Outlook, y debe ingresar los datos y el sistema solicitará permisos de la cuenta para crear su cuenta.4. El sistema valida las credenciales de acceso y permite el ingreso a la sesión respectiva
Flujo Alternativo	4.a El sistema no encuentra las credenciales del usuario y emite un mensaje de información

Tabla 4: Desglose de historia - Autenticación de Usuarios

2.3.2 Caso de uso: RF002 - Gestión de Publicaciones

CODIGO	RF002
ACTOR	Estudiante, docente
Flujo Normal	<ol style="list-style-type: none">1. El usuario ingresa a un panel de gestión2. El sistema despliega un listado de todas las publicaciones hechas por la sesión iniciada.3. El usuario podrá seleccionar las diferentes acciones por cada publicación como bloquear acceso, eliminar las publicaciones.4. El sistema ejecutara el comando correspondiente dependiendo de la acción seleccionada previamente.
Flujo Alternativo	2.a El usuario no tiene publicaciones y el sistema despliega un mensaje de que el usuario aun no a publicado nada.

Tabla 5: Desglose de historia - Gestión de Publicaciones

2.3.3 Caso de uso: RF003 – Búsquedas Personalizadas

CÓDIGO	RF003
ACTOR	Estudiante, docente
Flujo Normal	<ol style="list-style-type: none"> 1. El usuario ingresa a la sección de publicaciones 2. El usuario selecciona los diferentes filtros a conveniencia 3. El sistema despliega las publicaciones en base a los filtros seleccionados sin recargar el portal
Flujo Alternativo	3.a El sistema no encuentra publicaciones de acuerdo a los filtros seleccionados y da un mensaje indicando que no se ha encontrado nada.

Tabla 6: Desglose de historia - Búsquedas Personalizadas

2.3.4 Caso de uso: RF004 – Carga y descarga de Archivos ZIP/RAR

CÓDIGO	RF004
ACTOR	Estudiante, docente
Flujo Normal	<p>Carga de documentos</p> <ol style="list-style-type: none"> 1. El usuario ingresa a la sección para subir publicaciones 2. El sistema desplegará el formulario con todos los campos obligatorios y no obligatorios para crear una publicación. 3. El usuario tendrá que llenar todos los campos y subir toda la información necesaria. 4. El sistema tendrá que subir los archivos a AWS y a MongoDB Cloud con el mapeo de los diferentes campos que se llenaron en el formulario 5. El sistema redirige al usuario a la página de publicaciones donde debería reflejar la publicación recién creada. <p>Descarga de documentos</p> <ol style="list-style-type: none"> 1. El usuario ingresa a la sección de publicaciones 2. El usuario selecciona la publicación que le cause más interés. 3. El sistema despliega las publicaciones con toda la información con la que fue creada y subida. 4. El usuario contará visualmente con toda la información y a la vez podrá descargar el documento correspondiente en dos botones visibles. 5. El sistema se encargará de descargar el archivo correspondiente desde una URL pública del archivo almacenado en AWS a la que se dirigen los botones mencionados anteriormente.
Flujo Alternativo	<p>Carga de documentos</p> <p>5.a El sistema no puede terminar de cargar la publicación correspondiente y emite un mensaje de error del motivo.</p> <p>Descarga de documentos</p> <p>3.a El sistema no encuentra publicaciones con el mapeo correspondiente en MongoDB y da un mensaje indicando que no se ha encontrado nada.</p>

	5.a El sistema tiene un error al intentar conectar con la base de AWS que se encarga de brindarnos un enlace y se procede a representar datos por default.
--	--

Tabla 7: Desglose de historia - Carga y descarga de Archivos ZIP/RAR

2.3.5 Caso de uso: RF005 – Comentarios Interactivos en Publicaciones

CÓDIGO	RF005
ACTOR	Estudiante, docente
Flujo Normal	<ol style="list-style-type: none"> 1. El usuario accede a alguna publicación de su interés. 2. El sistema despliega todos los datos de la publicación. 3. El usuario observa los comentarios anteriores. 4. El usuario coloca su opinión en el formulario ya desplegado en la página. 5. El usuario selecciona una calificación en base a su comentario. 6. El usuario presiona el botón envía y se actualiza inmediatamente la sección de comentarios. 7.
Flujo Alternativo	3.a El sistema no encuentra comentarios previos y muestra un mensaje

Tabla 8: Desglose de historia - Comentarios Interactivos en Publicaciones

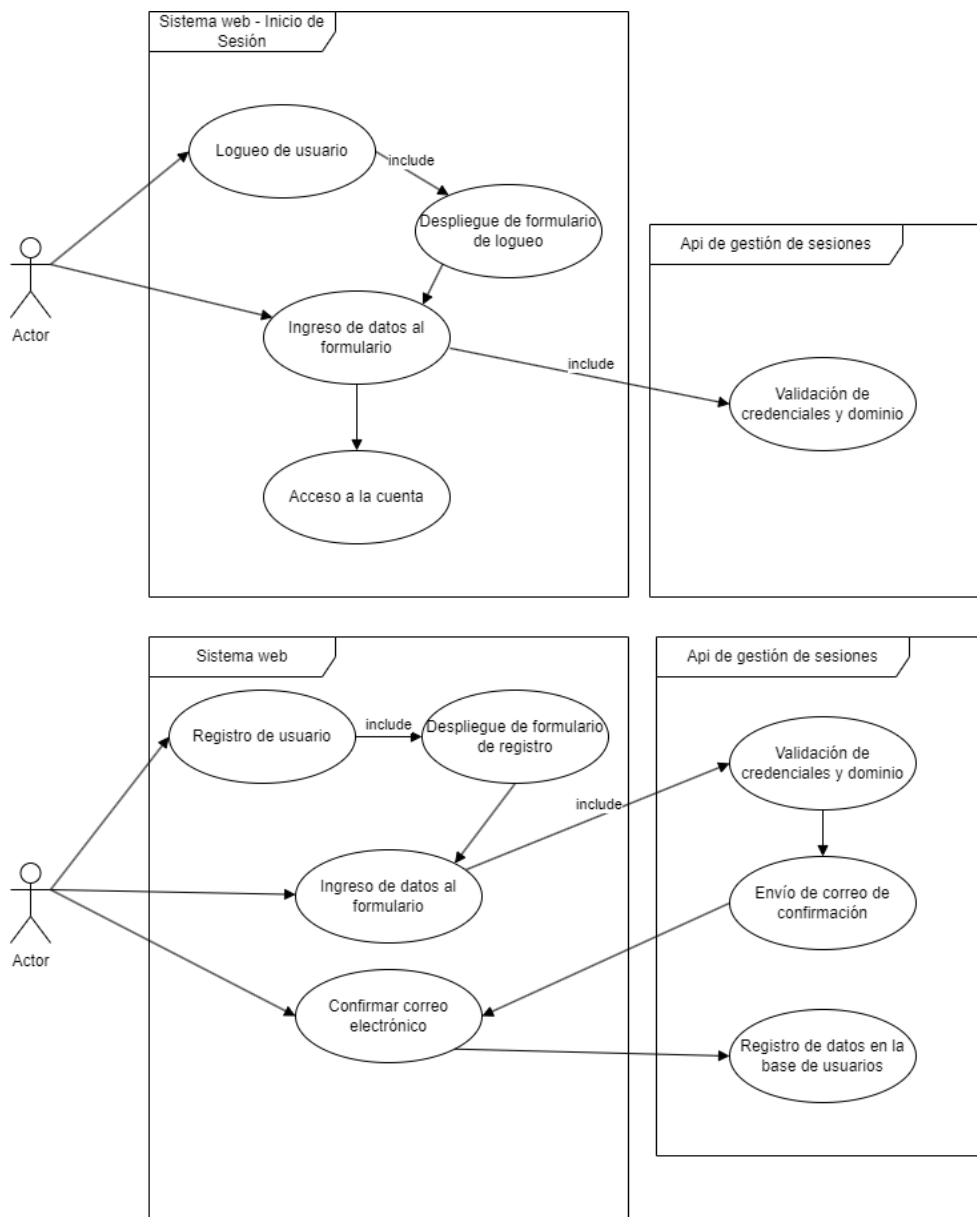
2.4 Caso de uso

Los casos de uso representan el comportamiento y nivel de interacción que tienen los usuarios con el repositorio de información a través de la página que gestiona cada una de las bases de datos, todos los diagramas están acompañados de una descripción que complementa el diagrama en cada caso.

2.4.1 Caso de uso: RF001 - Autenticación de Usuarios

Figura 18

Diagrama de autenticación de usuario



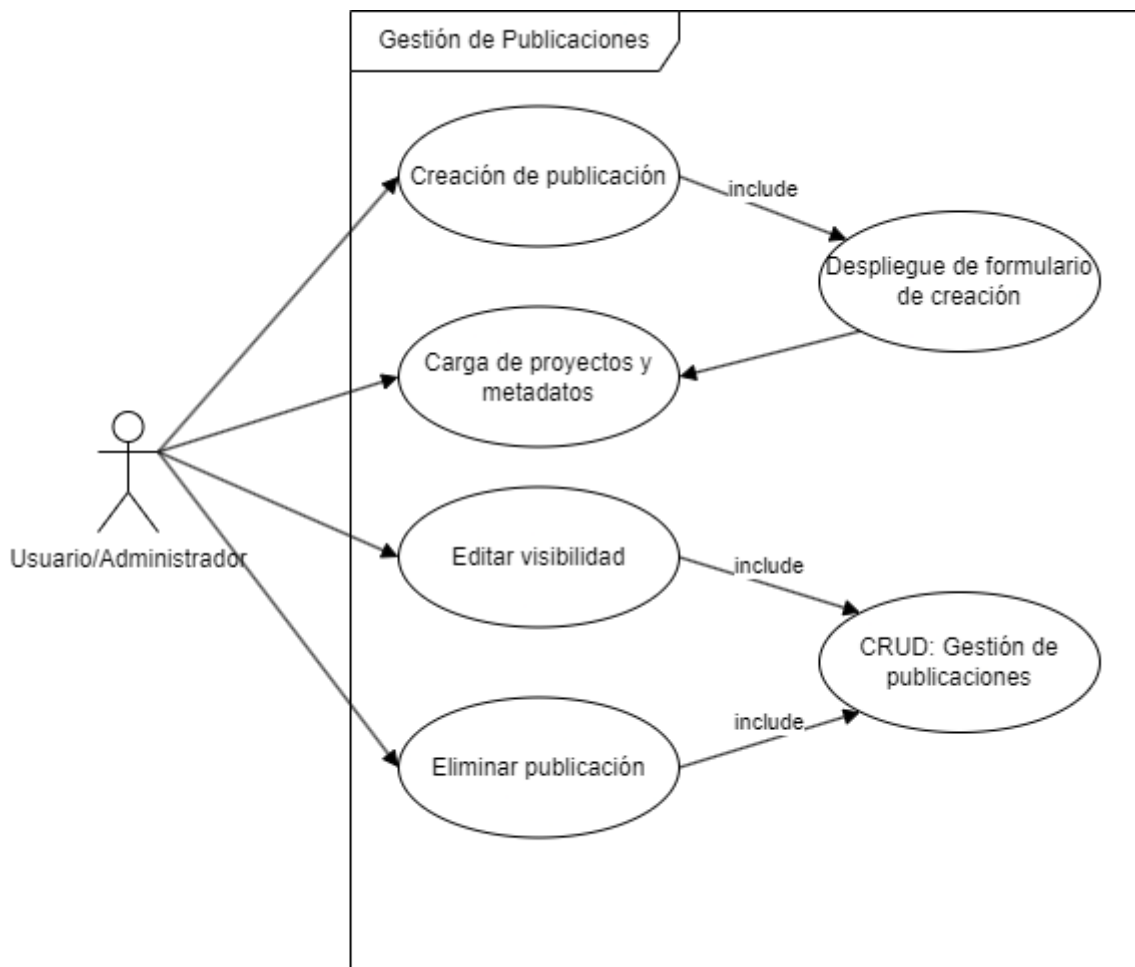
Nota: Grafico de diagrama de caso de uso de la autenticación de usuario en la página.

Para el usuario es sencillo iniciar sesión o registrarse, Clerk presenta un formulario de autenticación de Microsoft 365 tanto para el registro como para el inicio de sesión, donde el usuario/administrador debe ingresar sus credenciales, si este no está registrado debe confirmar su correo electrónico para el doble factor de autenticación, una vez almacenados sus datos en la base podrá ingresar con normalidad.

2.4.2 Caso de uso: RF002 - Gestión de Publicaciones

Figura 19

Diagrama d gestión de Publicaciones



Nota: Diagrama de caso de uso de las configuraciones que pueden tener las publicaciones.

La gestión de publicaciones de cara al usuario es sumamente sencilla, la subida de cada publicación o carga privada al repositorio se maneja en dos partes;

A. Formulario de publicación.

B. Tabla crud de edición de privacidad y eliminación de cargas

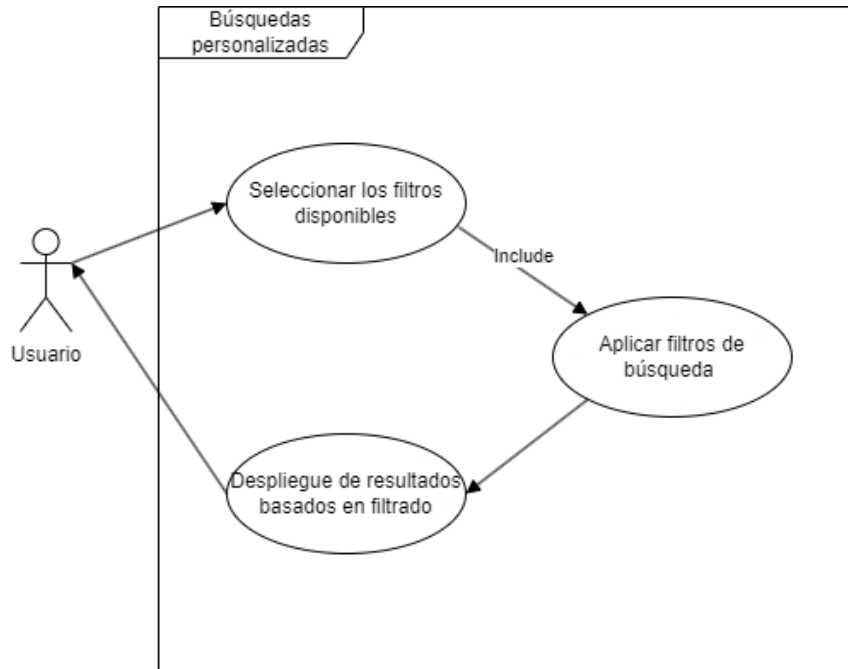
En el formulario cada usuario debe subir el archivo rar/zip junto con toda la metadata e información adicional que se pueda añadir a esta carga, de la misma manera el nivel de privacidad.

Con la tabla es posible modificar la privacidad de cada publicación propia y eliminarlas por cualquier motivo, eliminar una publicación es una tarea riesgosa, ya que una vez eliminada la carga esta no se podrá recuperar.

2.4.3 Caso de uso: RF003 – Búsquedas Personalizadas

Figura 20

Diagrama de búsquedas personalizadas



Nota: Diagrama de caso de uso de los filtros que tienen las publicaciones.

Buscando manejar procesos transparentes para el usuario, el usuario solo debe seleccionar y establecer de los filtros disponibles, el sistema realizará el proceso de filtrado y presentará las publicaciones resultantes de esta búsqueda.

2.4.4 Caso de uso: RF004 – Carga y descarga de Archivos ZIP/RAR

Figura 21

Diagrama de carga y descarga de Archivos ZIP/RAR



Nota: Diagrama de caso de las descargas y subida de archivos en el repositorio.

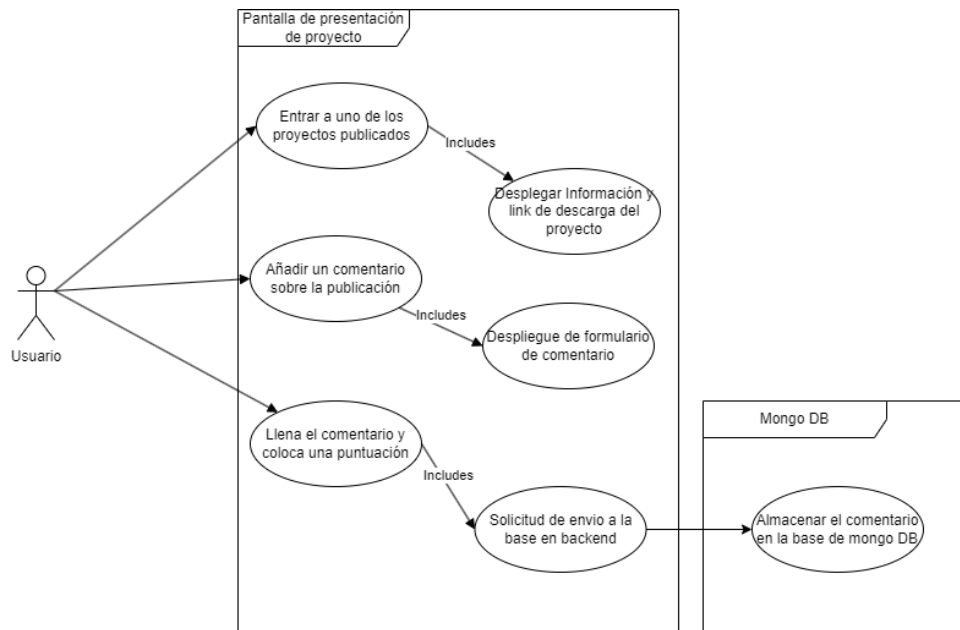
La subida de proyectos se realiza por la página, el sistema se encarga de asegurar la información y subirla al repositorio de proyectos de AWS.

De la misma manera como en el punto anterior, el usuario se encuentra a un solo clic, si la carga al repositorio es pública puede realizar la descargar, el sistema se va a realizar la descarga; una copia de la información del repositorio.

2.4.5 Caso de uso: RF005 – Comentarios Interactivos en Publicaciones

Figura 22

Diagrama de comentarios interactivos en publicaciones



Nota: Diagrama de caso de uso sobre las opciones que tienen la pantalla de visualización de una publicación.

Comentar sobre una publicación es sencillo, el usuario debe seleccionar la publicación en la que está interesado, ahí se encuentra tanto la información de cada publicación como el formulario en la zona inferior, donde el usuario debe escribir su comentario y añadir una calificación en estrellas sobre la información presente en el repositorio.

2.5 Diagrama lógico de la base de datos

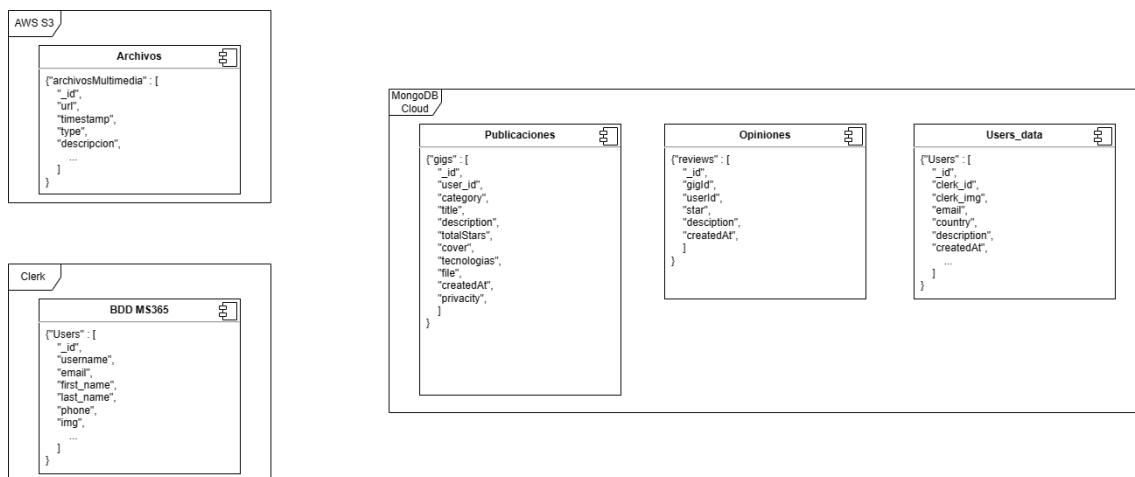
La arquitectura para almacenar la información fue una de las incógnitas que más tiempo tomó al momento de concretar el repositorio por lo que para contar con bases de datos segmentadas que alimenten desde diferentes fuentes al sistema, se segmenta la información en 3 ejes.

- A. Repositorio de proyectos (S3 Bucket AWS)
- B. Metadata de proyectos (MongoDB Cloud)
- C. Base de usuarios gestionada por M365 (Api Clerck)

En la figura presente presentarnos de qué manera están, cada eje tiene su propia fuente tanto con los atributos, los datos de usuario y el repositorio de archivos.

Figura 23

Diagrama lógico de la base de datos



Nota: Diagrama sobre la distribución que tienen las bases de datos no relacionales.

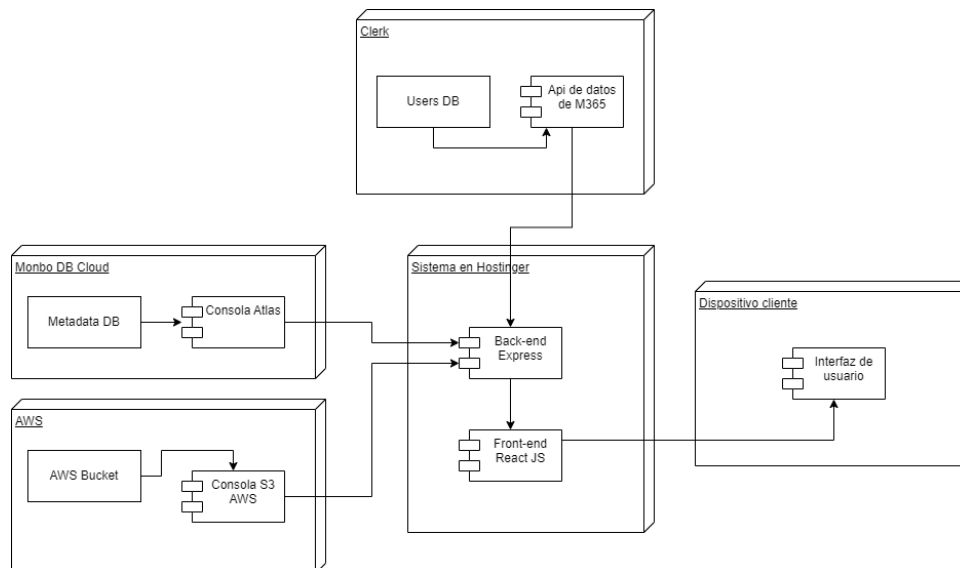
CAPÍTULO III CONSTRUCCION Y RESULTADOS

3.1 Diagrama de despliegue

En la figura presente se desglosa los componentes que arman la aplicación, se cuenta con las siguientes piezas:

- A. Clerk – Fuente de datos de usuarios.
- B. S3 AWS – Fuente de datos, repositorio de archivos.
- C. Mongo Cloud – Fuente de metadatos e intermediario entre Clerk y S3 AWS
- D. Sistema levantado en Hostinger – Sistema desarrollado para gestionar la información del repositorio virtual.
- E. Dispositivos de cada usuario – Este componente representa a el acceso de cada usuario al repositorio que se puede establecer en cualquier navegador web.

Figura 24
Diagrama de despliegue



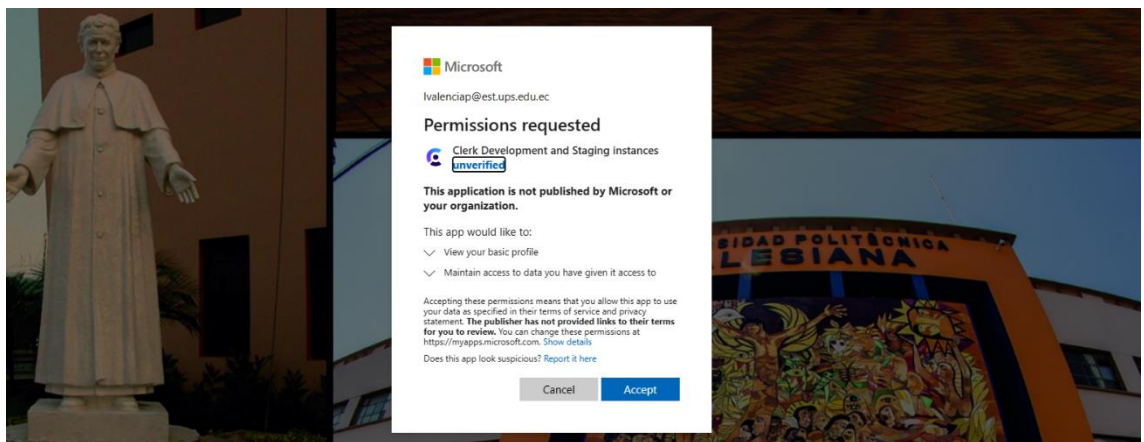
Nota: Diagrama de la distribución de tecnologías que han sido usadas en la aplicación.

3.1.1 Clerk – Fuente de datos de usuarios.

El primer componente con el que interactúa el usuario es Clerk, básicamente es un api que permite conectar con los usuarios de la UPS sin necesidad de hacer un registro desde cero, a partir de que un usuario se inicie sesión mediante la API se despliegan varias opciones adicionales en la página y es aquí donde ya se empieza a relacionar más con MongoDB Cloud.

Figura 25

Imagen de permisos requeridos por Clerk sobre cuenta UPS



Todo se maneja mediante la librería y sus funciones que permiten dar uso a la data pública del usuario y sus respectivas validaciones.

Ilustración 26

Código se llamado a librería de Clerk

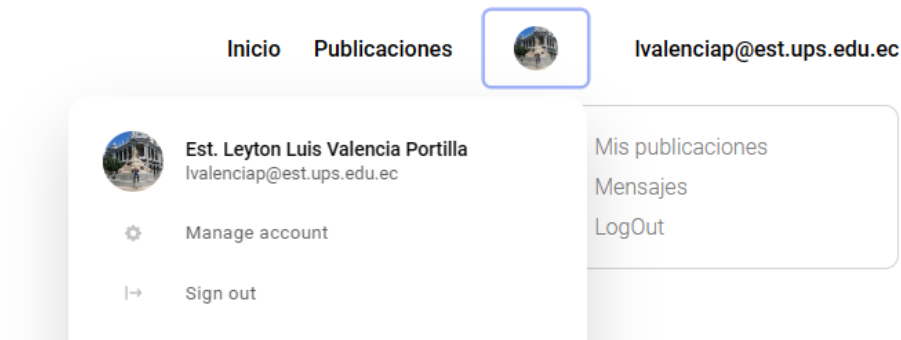
```
import { UserButton, useUser } from "@clerk/clerk-react";  
function Desktop_menu() {  
  
  const { isLoading, isSignedIn, user } = useUser();
```

Mongo Cloud – Fuente de metadatos e intermediario entre Clerk y S3 AWS
MongoDB traerá varias colecciones de datos que permiten darle funcionalidad a la página y estas mismas acciones que puede realizar el usuario se guardarían en base a su ID el

cual solo se encuentra visible en el panel de control de Clerk y se crea automáticamente cuando el usuario hace su primer logeo en la página.

Figura 27

Componente Clerk de usuario logeado



Mongo probablemente es el pilar más grande que permite interconectar las diferentes herramientas utilizadas en el proyecto, tanto AWS S3 y Clerk funcionan dentro del código de JSX, mediante funciones asíncronas permiten mapear correctamente las acciones que hace el usuario realiza como interacciones en publicaciones o subir información a la página.

3.1.2 S3 AWS – Fuente de datos, repositorio de archivos.

AWS: Es una función asíncrona que hace una conexión y envía un archivo a la nube de archivos, solo sirve para enviar datos, pero no se hace ningún mapeo en AWS sobre de quién es ese dato. Por eso es importante enviar automáticamente la URL pública a MongoDB cuando se sube un archivo.

Figura 28

Función asíncrona de AWS S3 para subir archivos

```
const onFileUpdate = async (file) => {
  const config = {
    bucketName: 'upsnet2',
    region: 'us-east-1',
    accessKeyId: 'AKIAQRCVFP2ZR7N0FPT',
    secretAccessKey: 'X3ZV38ReL0YR125FiB74Kfpe7l1px0aLZRuDA44',
  };
  try {
    const data = await S3FileUpload.uploadFile(file, config);
    const url = data.location;
    return url;
  } catch (err) {
    // Manejar el error según sea necesario
    console.error(err);
    throw err; // Propagar el error para que sea capturado por el bloque catch en handleUpload
  }
};
```

Para lograr subir información a la página es donde interactúan las 3 herramientas directamente, porque lo que estamos subiendo se enlazará al usuario logeado, se subirá un archivo a AWS y solo recibiremos la URL la cual se almacenará en MongoDB Cloud mediante un request.

Figura 29

Objeto con meta data de publicaciones en MongoDB Cloud

```
{
  _id: ObjectId('658cdd4b9dc3be063236c4a7')
  userId: "6563926685ca9e3dd3ce06ee"
  cat: "Desarrollo Móvil"
  title: "App de Noticias con Flutter"
  desc: "Este proyecto implica la creación de una aplicación móvil de noticias ..."
  totalStars: 37
  starNumber: 13
  cover: "https://upsnet2.s3.amazonaws.com/OIG.jpg"
  tools: Array (2)
  file: "https://upsnet2.s3.amazonaws.com/FlutterNewsApp.zip"
  privacy: true
  level: 2
  area: "Mobile Development"
  createdAt: 2023-12-18T02:51:04.543+00:00
  updatedAt: 2024-01-02T04:58:31.914+00:00
  __v: 0
}
```

3.2 Pruebas y resultados

La sección de pruebas se va a dividir en dos partes, siendo la primera, pruebas de carga o estrés donde se va a realizar requerimientos de conexión al sitio y corroborar que va a sostener las conexiones de todos los estudiantes de la Universidad Politécnica Salesiana de la Carrera de Computación, que son alrededor de 600 estudiantes, mientras que la segunda sección se va a enfocar en pruebas de caja negra sobre cada requerimiento funcional para evaluar el comportamiento en cada proceso.

3.2.1 Pruebas de carga

Sabemos que la carrera cuenta con alrededor de 300 estudiantes en la Carrera de Computación del Campus Sur, por lo que tomamos en primera instancia consideramos que al menos la mitad de los estudiantes podrían realizar conexiones simultaneas a la aplicación.

Como se documentó en la primera unidad, la herramienta donde se va a realizar las pruebas es *Loader.io* que nos va a ayudar a realizar las pruebas de carga de manera automática mediante solicitudes de bots, esta prueba también va a reconfirmar el funcionamiento del WAF que utiliza Hostinger para mitigar ataques DOS y DDOS que podrían buscar afectar el funcionamiento del sitio web.

Con los datos previos se procedió a verificar nuestra aplicación en la herramienta a través de un token en los documentos de nuestro aplicativo para dar paso a las pruebas, una vez validemos nuestro sistema se procede a desplegar las pruebas de carga y estrés.

Para la primera prueba se realizaron 300 conexiones en el transcurso de un minuto, donde se consiguió un 100% de conexiones exitosas en el tiempo ejecutado, esto se traduce en que si la mitad de los estudiantes en el intervalo de un minuto se llegaron a conectar de manera simultánea el sitio respondería de manera favorable.

En la segunda prueba se ejecutó buscando encontrar el límite posible que pueda trabajar la página, por lo que se realizaron 300 conexiones simultaneas a lo largo de un segundo de manera continua hasta completar un minuto a 300 conexiones por segundo dando un total de 17997 conexiones simultaneas alrededor de un minuto donde todas ellas se conectaron de manera correcta, pero el WAF reaccionó frente a este nivel de exigencia, en el bloque de resultados se presenta un desglose mucho más granular y el actuar del WAF frente a las pruebas.

3.2.2 Resultados: Pruebas de carga

Prueba 1 - 300 usuarios en 1 minuto: Los resultados de esta prueba fueron 100% favorables con los siguientes datos.

- Tiempos de respuesta
 - Porcentaje: 114 ms
 - Min/Max: 110 / 433 ms
- Conteos de respuestas
 - Exitosas: 300
 - Errores Timeout: 0
 - Errores de red: 0
- Anchos de banda
 - Envío: 65.63 KB
 - Respuesta: 582.71 KB
- Redirecciones
 - Validos: 300
 - Inválidos: 0
- Tiempos de respuesta

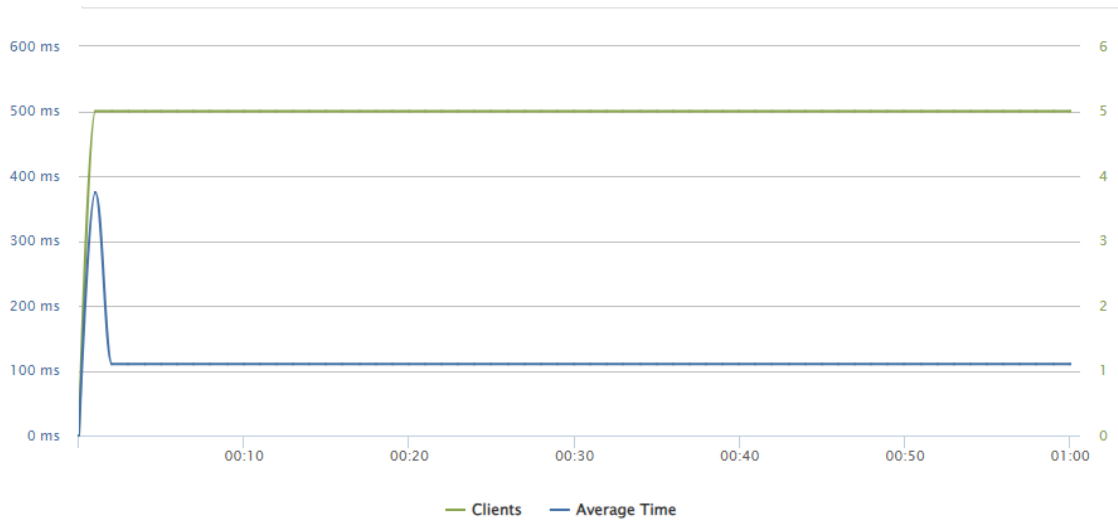
En el gráfico de tiempos de respuesta se muestran dos trazas, la primera de color verde el cual es la suma del tiempo de respuestas de los usuarios que en este segundo son 5, mientras que la línea azul es el promedio el tiempo de respuesta de los usuarios, como podemos ver las trazas son estables y se mantiene en una media de 114ms.

- Eje vertical uno: Tiempo de respuesta en milisegundos 0ms – 600ms.
- Eje vertical dos: Número de usuarios conectados 0 usuarios – 6 usuarios.
- Eje horizontal: Tiempo de la prueba en segundos duración 1 minuto.

- Traza verde: sumatoria de tiempo de respuesta por segundo.
- Traza azul: promedio de tiempo de respuesta.

Figura 30

Tiempo de respuesta en milisegundos frente al número de usuarios y minutos de conexión.



Nota: Grafico de tiempos de respuesta de los clientes en la pagina.

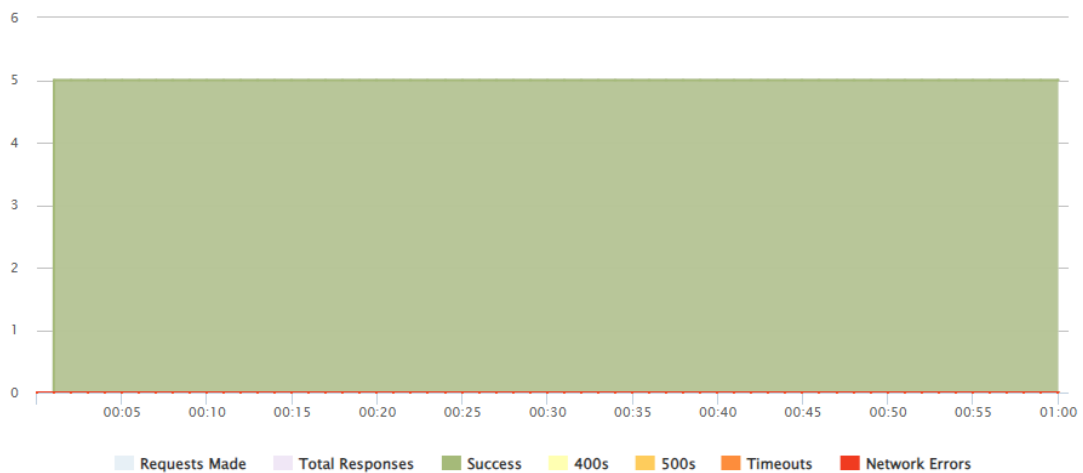
- Detalles de conexión

Aquí se muestra una gráfica del detalle de conexiones donde se muestra las conexiones exitosas y de distintos errores, delay, timeout, errores de red, los datos presentes son una traza verde que equivale a usuarios conectados de manera exitosa.

- Eje vertical uno: Respuestas obtenidas 0 respuestas – 6 respuestas.
- Eje horizontal: Tiempo de la prueba en segundos duración 1 minuto.
- Traza verde: Respuestas exitosas por segundo.
- Traza roja (poco visible): Respuestas fallidas por segundo.

Figura 31

Gráfico de detalles de conexión por usuario



Nota: Gráfico de los eventos de la página donde no se encuentran errores y el código se ejecuta sin inconvenientes.

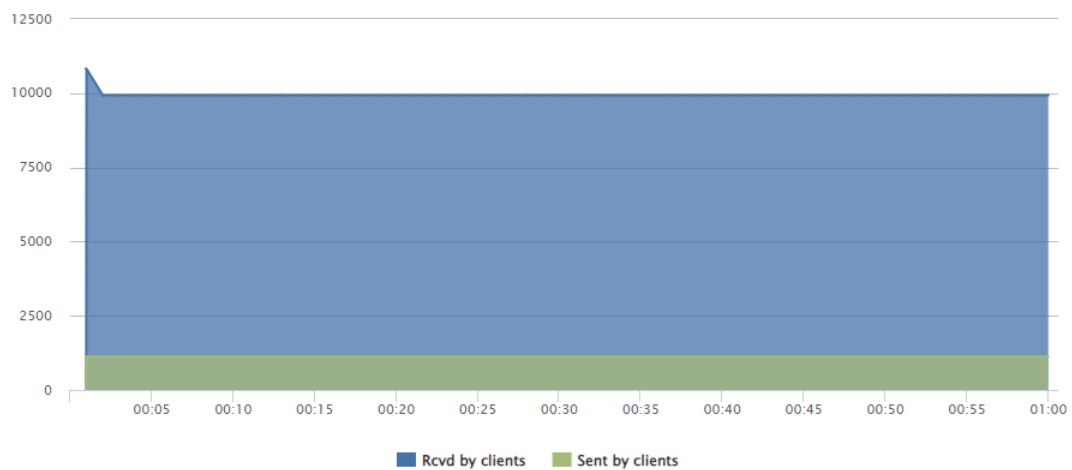
- Uso de ancho de banda

Se presenta el ancho de banda enviado por cliente en color verde y recepción automática de señal por cliente en color azul.

- Eje vertical uno: Ancho de banda de respuesta de clientes 0 – 12500.
- Eje horizontal: Tiempo de prueba en segundos, duración 1 minuto.
- Traza azul: Ancho de banda de respuesta automática de clientes.
- Traza verde: Ancho de banda de envío a clientes.

Figura 32

Gráfico de uso de ancho de banda contra tiempo en minutos



Nota: Se presenta el ancho de banda enviado por cliente en color verde y recepción automática de señal por cliente en color azul.

Prueba 2 - 300 usuarios por segundo durante 1 minuto: Los resultados de esta prueba fueron en su mayoría favorables con los siguientes datos, aquí se necesita contextualizar el trasfondo de la prueba puesto que también tiene como finalidad probar la funcionalidad del WAF de Hostinger ya que se busca llevar al límite al sitio, probar el despliegue automático de autenticar si estos son humanos mediante recaptcha, esta prueba fue exitosa ya que llegado el segundo 46 se activa la verificación de que el usuario no es un bot, por lo que las ultimas solicitudes dieron como resultado error de red.

- Tiempos de respuesta
 - Porcentaje: 117 ms
 - Min/Max: 109 / 2014 ms
- Conteos de respuestas
 - Exitosas: 14306
 - Errores Timeout: 0
 - Errores de red: 3691
- Anchos de banda
 - Envío: 3.46 MB
 - Respuesta: 27.32 MB
- Redirecciones
 - Validos: 14480
 - Inválidos: 0
- Tiempos de respuesta

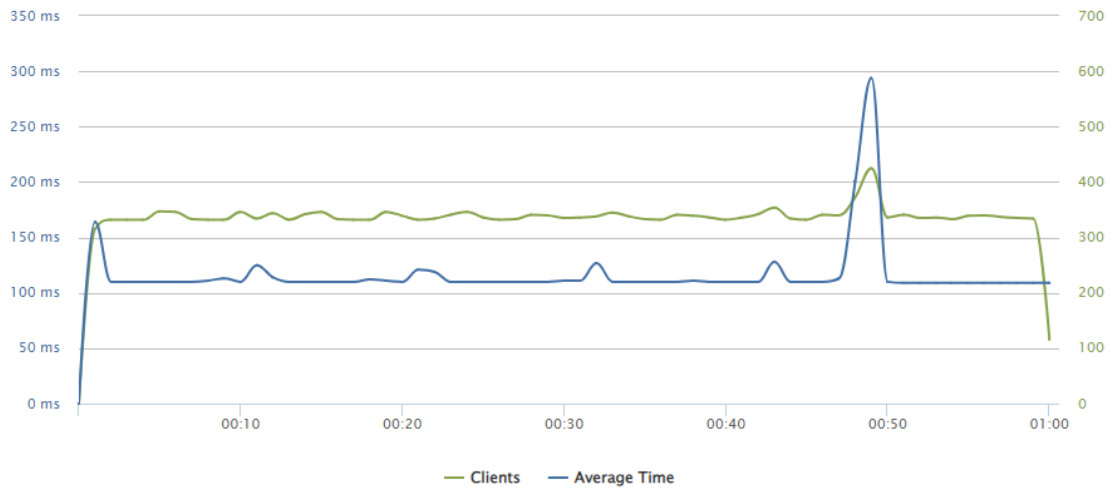
En la figura 34 tenemos una tabla de tiempos de respuesta, se muestran dos trazas, la primera de color verde el cual es la suma del tiempo de respuestas de los usuarios, mientras que la línea azul es el promedio del tiempo de respuesta de los usuarios.

- Eje vertical uno: Tiempo de respuesta en milisegundos 0ms – 350ms.
- Eje vertical dos: Número de usuarios conectados 0 usuarios – 700 usuarios.
- Eje horizontal: Tiempo de la prueba en segundos duración 1minuto.
- Traza verde: sumatoria de tiempo de respuesta por segundo.

- Traza azul: promedio de tiempo de respuesta.

Figura 33

Gráfico de tiempos de respuesta en ms contra segundos, contra número de usuarios



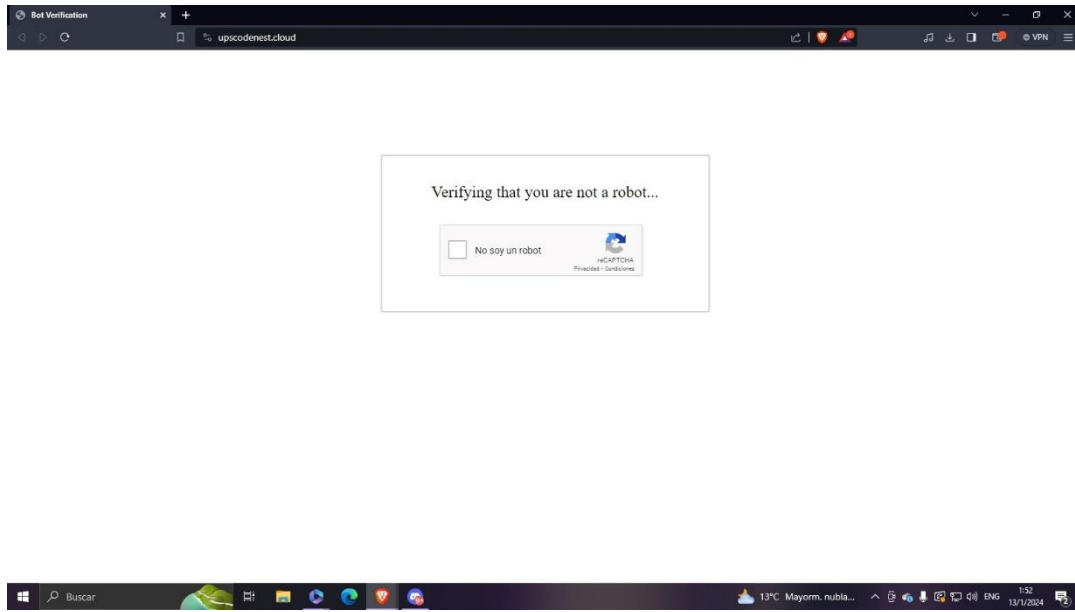
Nota: Diagrama de la distribución de tecnologías que han sido usadas en la aplicación.

- Despliegue de recaptcha

Así el WAF de Hostinger despliega el recaptcha para evitar más conexiones de bots a la aplicación web.

Figura 34

Captura de reacción del WAF frente a los bots.



Nota: Imagen sobre el cuadro contra bots cuando se detectan muchas conexiones en simultaneo a la página.

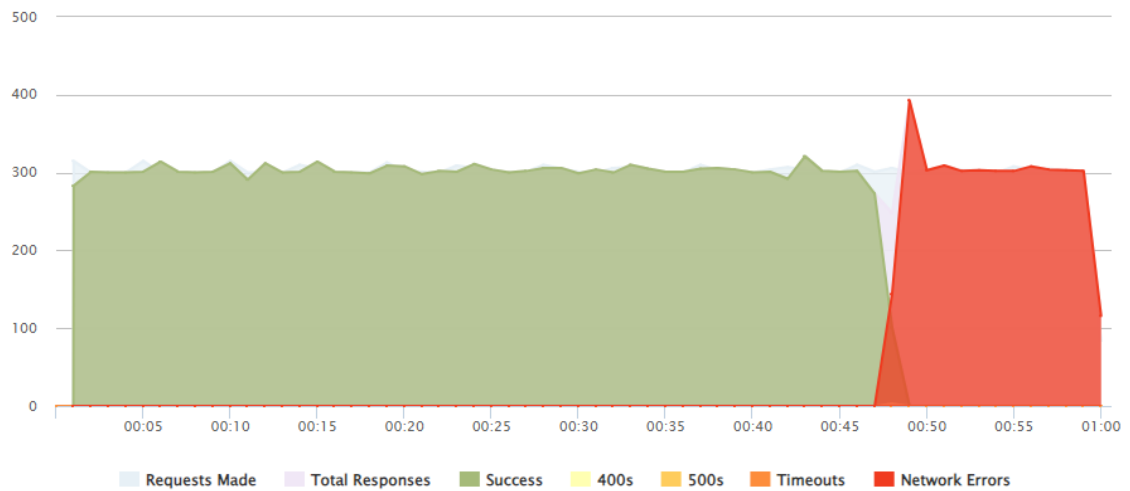
- Detalles de conexión

Aquí se muestra una gráfica del detalle de conexiones donde se muestra las conexiones exitosas y de distintos errores, delay, timeout, errores de red, estos errores se dieron porque el WAF de Hostinger activó la protección contra DOS.

- Eje vertical uno: Respuestas obtenidas 0 respuestas – 6 respuestas.
- Eje horizontal: Tiempo de la prueba en segundos duración 1 minuto.
- Traza verde: Respuestas exitosas por segundo.
- Traza roja: Respuestas fallidas por segundo.

Figura 35

Gráfico de solicitudes correctas contra tiempo en minutos



Nota: Gráfico de conexiones correctas durante un intervalo de tiempo contra un posible ataque DOS.

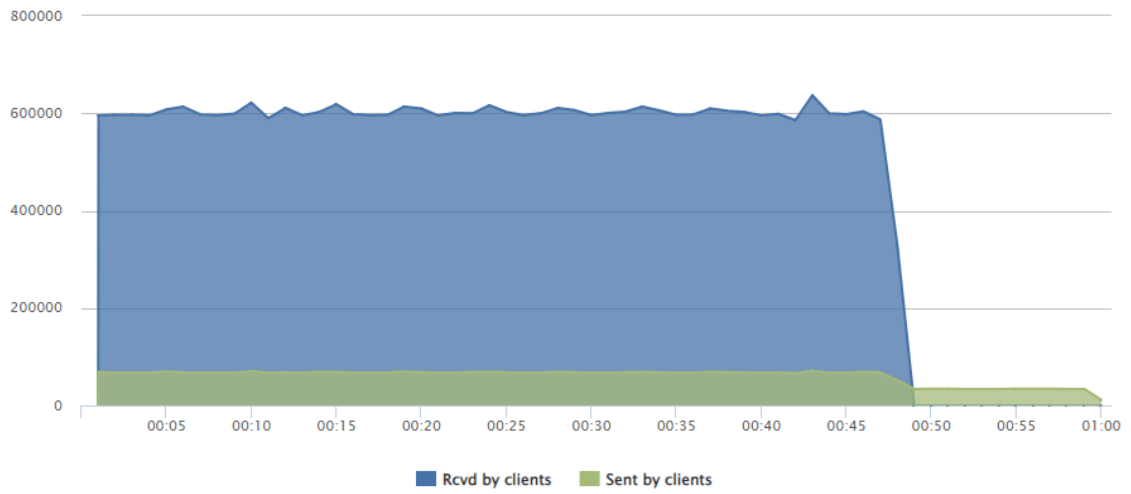
- Uso de ancho de banda

Se presenta el ancho de banda enviado por cliente en color verde y recepción automática de señal por cliente en color azul.

- Eje vertical uno: Ancho de banda de respuesta de clientes 0 – 12500.
- Eje horizontal: Tiempo de prueba en segundos, duración 1 minuto.
- Traza azul: Ancho de banda de respuesta automática de clientes.
- Traza verde: Ancho de banda de envío a clientes.

Figura 36

Gráfico de ancho de banda de usuarios contra tiempo en minutos



Nota: Gráfico sobre las conexiones de los clientes y los paquetes recibidos en su ingreso.

3.2.3 Pruebas de caja negra

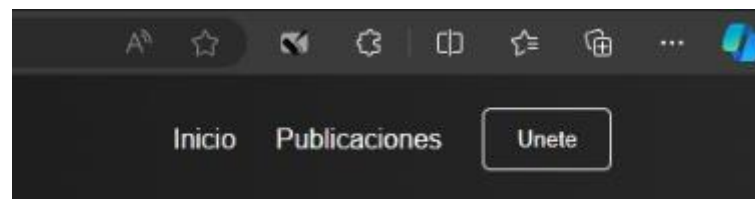
En el presente módulo se van a realizar pruebas de caja negra, las cuales van a revalidar el comportamiento del sistema frente a los diferentes requerimientos funcionales.

- RF001 - Autenticación de Usuarios

El usuario selecciona el botón de Únete para inicio de sesión.

Figura 37

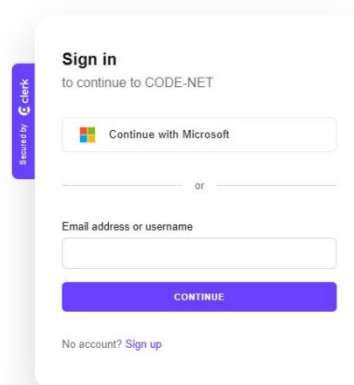
Botón inicio de sesión



EL sistema responde cargando la interfaz correspondiente

Figura 38

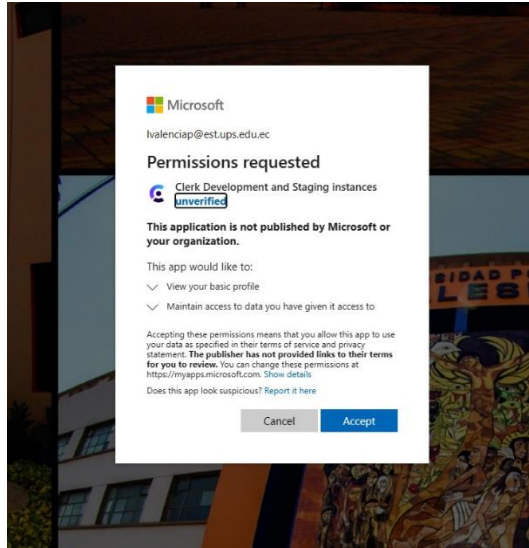
Cuadro de ingreso de credenciales



El usuario accede a través de Outlook, y debe ingresar los datos y el sistema solicitará permisos de la cuenta para crear su cuenta.

Figura 39

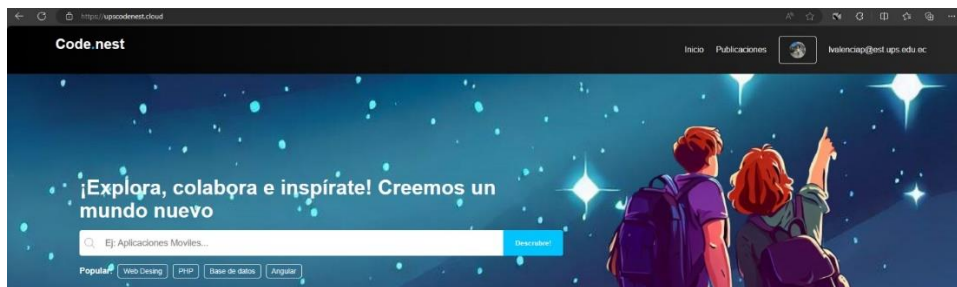
Permisos requeridos de la aplicación Clerk



El sistema valida las credenciales de acceso y permite el ingreso a la sesión respectiva.

Figura 40

Inicio de la página web

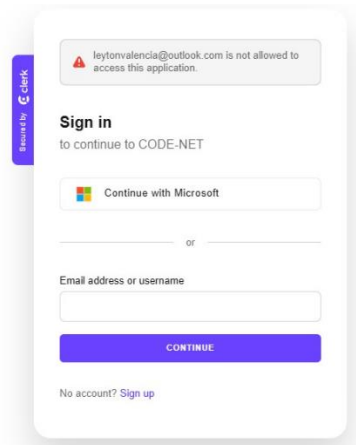


Flujo alternativo

4.a. El sistema no encuentra las credenciales del usuario y emite un mensaje de información

Figura 41

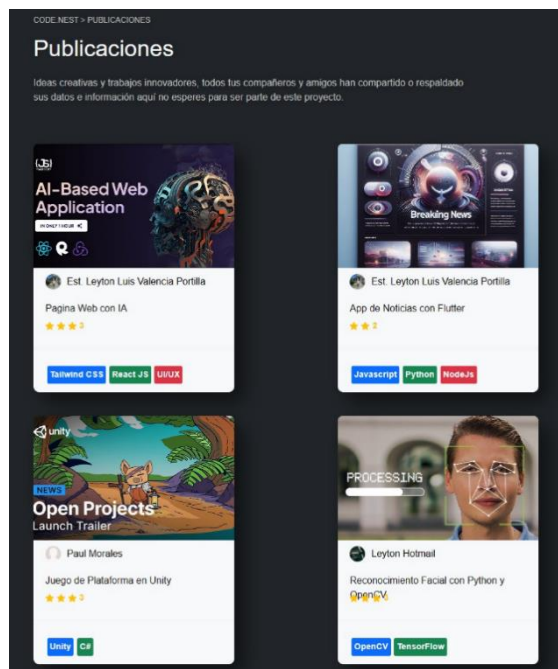
Validación de ingreso de dominios no permitidos



- RF002 - Gestión de Publicaciones
El usuario ingresa a un panel de gestión

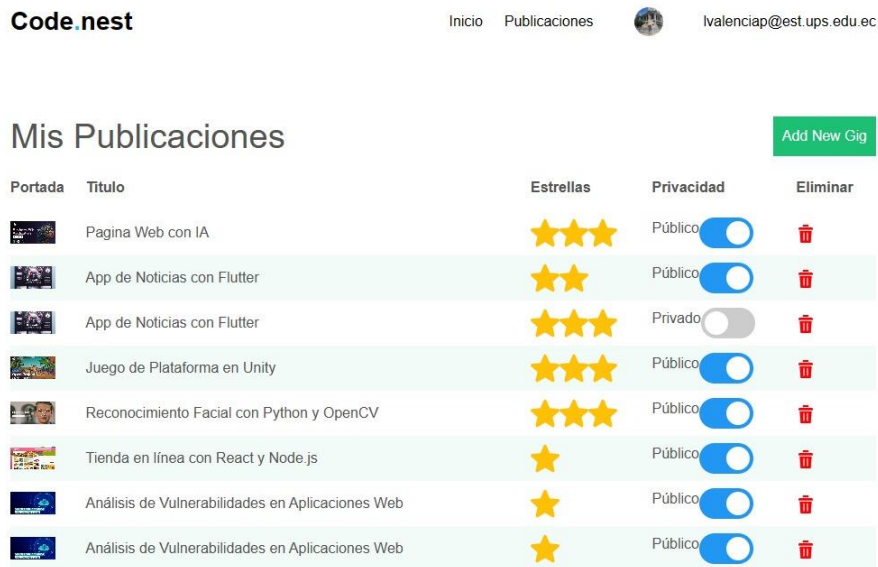
Figura 42

Sección de publicaciones en la página web



El sistema despliega un listado de todas las publicaciones hechas por la sesión iniciada.

Figura 43
Panel de administración de publicaciones

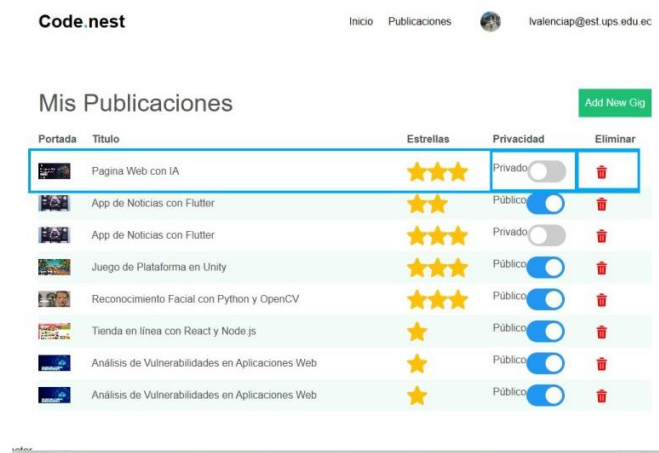


The screenshot shows the 'Code.nest' interface with the user 'lvalenciap@est.ups.edu.ec' logged in. The 'Mis Publicaciones' section displays a table of publications. Each row includes a thumbnail, title, star rating, privacy status (Publico/Privado), and an 'Eliminar' button.

Portada	Título	Estrellas	Privacidad	Eliminar
	Pagina Web con IA	★★★★	Público <input checked="" type="checkbox"/>	
	App de Noticias con Flutter	★★★	Público <input checked="" type="checkbox"/>	
	App de Noticias con Flutter	★★★★	Privado <input type="checkbox"/>	
	Juego de Plataforma en Unity	★★★★	Público <input checked="" type="checkbox"/>	
	Reconocimiento Facial con Python y OpenCV	★★★★	Público <input checked="" type="checkbox"/>	
	Tienda en línea con React y Node.js	★	Público <input checked="" type="checkbox"/>	
	Análisis de Vulnerabilidades en Aplicaciones Web	★	Público <input checked="" type="checkbox"/>	
	Análisis de Vulnerabilidades en Aplicaciones Web	★	Público <input checked="" type="checkbox"/>	

El usuario podrá seleccionar las diferentes acciones por cada publicación como bloquear acceso, eliminar las publicaciones.

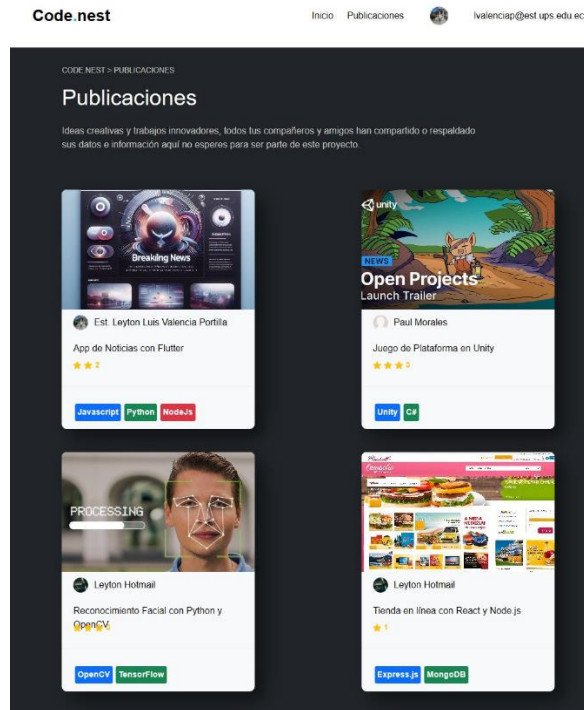
Figura 44
Opciones disponibles en la gestión de publicaciones



This screenshot is identical to Figure 43 but includes a red rectangular box highlighting the 'Privacidad' and 'Eliminar' columns for the first publication, 'Pagina Web con IA'. The privacy toggle is currently set to 'Privado' (off), and the delete icon is visible.

El sistema ejecutara el comando correspondiente dependiendo de la acción seleccionada previamente.

Figura 45
Sección de publicaciones luego de ejecutar una acción



Flujo alternativo

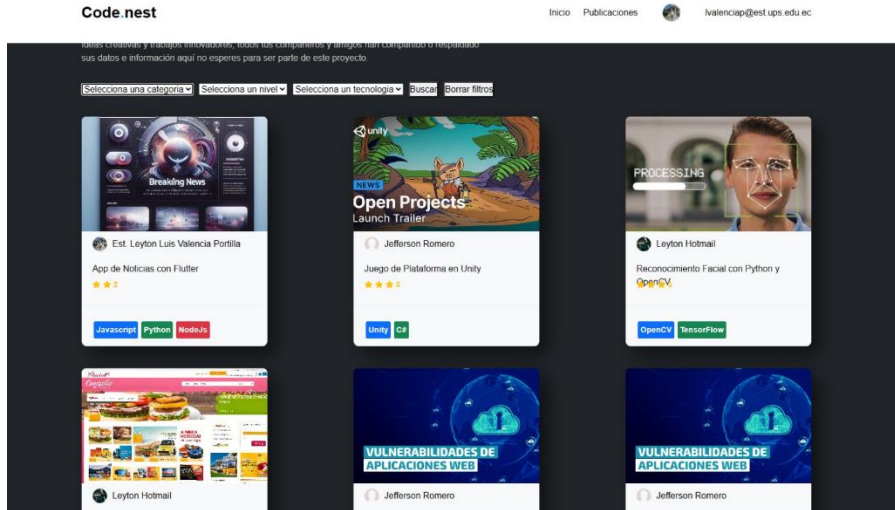
2.a El usuario no tiene publicaciones y el sistema despliega un mensaje de que el usuario aún no ha publicado nada todavía.

Figura 46
Mensaje de validación cuando no se tienen publicaciones creadas



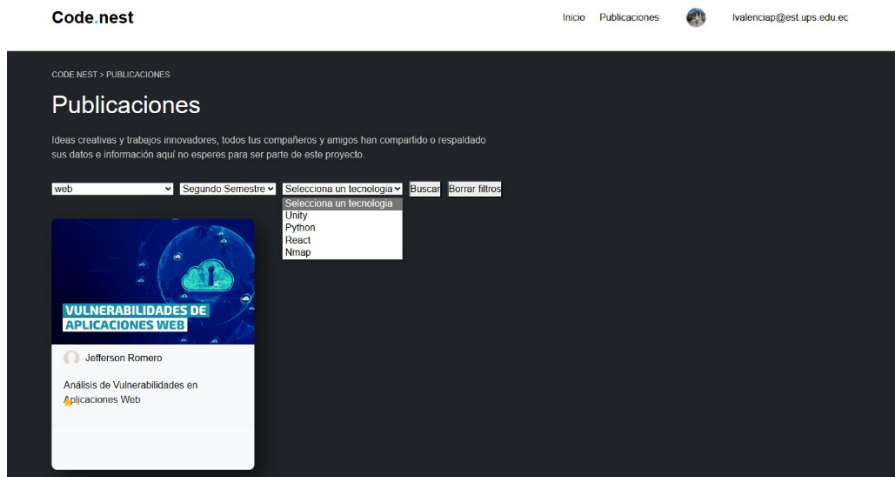
- RF003 – Búsquedas Personalizadas
El usuario ingresa a la sección de publicaciones

Figura 47
Filtros de sección de publicaciones



El usuario selecciona los diferentes filtros a conveniencia

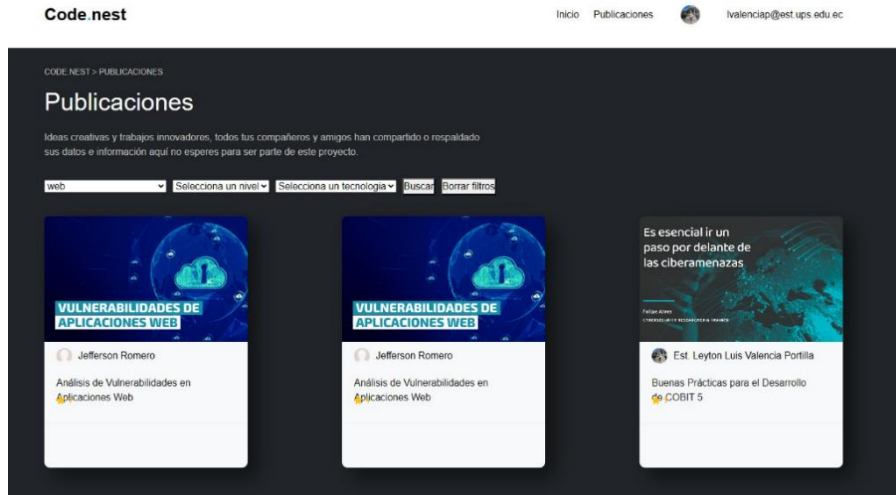
Figura 48
Opciones en filtros de publicaciones



El sistema despliega las publicaciones en base a los filtros seleccionados sin recargar el portal

Figura 49

Publicaciones en base a filtros usados

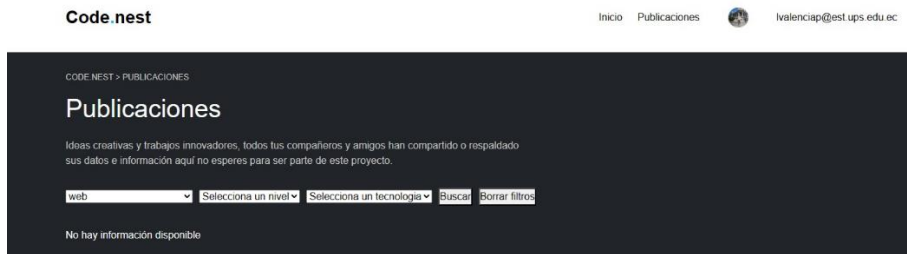


Flujo alternativo

3.a El sistema no encuentra publicaciones de acuerdo con los filtros seleccionados y da un mensaje indicando que no se ha encontrado nada.

Figura 50

Mensaje de validación cuando no se encuentra una publicación



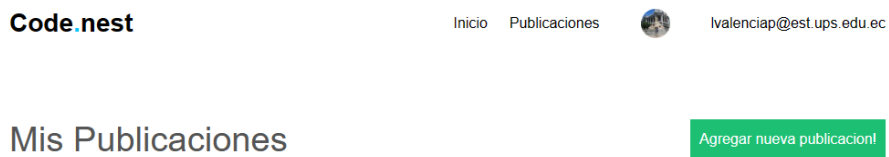
- RF004 – Carga y descarga de Archivos ZIP/RAR

Carga de documentos

- a. El usuario ingresa a la sección para subir publicaciones

Figura 51

Ingreso a crear nueva publicación



- b. El sistema desplegará el formulario con todos los campos obligatorios y no obligatorios para crear una publicación.

Figura 52
Formulario de nueva publicación

The screenshot shows the 'Code.nest' website interface for adding a new publication. The page title is 'Agregar nueva publicación!'. The form is organized into two columns. The left column contains the following fields: 'Title' (text input with placeholder 'e.g. I will do something I'm really good at'), 'Category' (dropdown menu with 'Design' selected), 'Cover Image' (file selection button 'Choose File' and 'No file chosen'), 'Upload Images' (file selection button 'Choose Files', 'No file chosen', and a green 'Upload' button), 'Archivo Rar o ZIP' (file selection button 'Choose File' and 'No file chosen'), and 'Description' (text area with placeholder 'Brief descriptions to introduce your service to customers'). The right column contains: 'area' (text input with placeholder 'e.g. One-page web design'), 'Privacidad' (dropdown menu with 'Publico' selected), 'level' (text input), and 'Add Features' (text input with placeholder 'e.g. page design' and a green 'add' button). At the bottom of the form is a large green 'Create' button. The top navigation bar includes 'Inicio', 'Publicaciones', a user profile icon, and the email 'hbalenciap@est.ups.edu.ec'.

- c. El usuario tendrá que llenar todos los campos y subir toda la información necesaria.

Figura 53

Formulario de nueva publicación completo

Agregar nueva publicación!

Title
Internal Control Using COBIT® 5

Category
Web Development

Cover Image
Choose File | img1.png

Upload Images
Choose Files | 2 files | Upload

Archivo Rar o ZIP
Choose File | Gtx+750+(6).zip

Description
Garantiza la eficiencia y calidad del desarrollo de software con prácticas esenciales.
-Control de Versiones (Git)
-Automatización de Pruebas
-Metodologías Ágiles (Scrum, Kanban)
-Documentación Actualizada
-Revisiones de Código
-Seguridad Integrada
-Despliegue Continuo (CI/CD)
-Gestión de Dependencias (npm, pip, Maven)
-Monitoreo y Registro
-Feedback Continuo
Estas prácticas promueven la colaboración, la flexibilidad y la mejora constante, asegurando un desarrollo sostenible y eficiente.

area
Calidad de Software

Privacidad
Publico

level
6

Add Features
e.g. page design
COBIT x Buenas Prácticas x

Create

- d. El sistema tendrá que subir los archivos a AWS y a MongoDB Cloud con el mapeo de los diferentes campos que se llenaron en el formulario.

Figura 54

Imagen del objeto creado en MongoDB luego de subir los datos del formulario

```
{
  "_id": ObjectId('65a8b6e3cf7a11a904a51...'),
  "userId": "65a5eca8b8f8d06482a7e15a",
  "cat": "web",
  "title": "Buenas Prácticas para el Desarrollo de COBIT 5",
  "desc": "Garantiza la eficiencia y calidad del desarrollo de software con práct...",
  "totalStars": 1,
  "starNumber": 1,
  "cover": "https://upsnet2.s3.amazonaws.com/img1.png",
  "tools": Array (3),
  "images": Array (2),
  "file": "https://upsnet2.s3.amazonaws.com/Gtx+750+(6).zip",
  "privacy": true,
  "level": 6,
  "area": "Calidad de Software",
  "createdAt": 2024-01-18T05:28:03.384+00:00,
  "updatedAt": 2024-01-18T05:28:03.384+00:00,
  "__v": 0
}
```


Figura 55

Imagen del archivo en AWS luego de subir el formulario

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	img2.jpg	jpg	January 18, 2024, 00:27:41 (UTC-05:00)	163.2 KB	Standard
<input type="checkbox"/>	img3.jpg	jpg	January 18, 2024, 00:27:41 (UTC-05:00)	313.7 KB	Standard
<input type="checkbox"/>	Gtx+750+(6).zip	zip	January 18, 2024, 00:27:38 (UTC-05:00)	50.7 MB	Standard
<input type="checkbox"/>	img1.png	png	January 18, 2024, 00:27:38 (UTC-05:00)	930.5 KB	Standard

- e. El sistema redirige al usuario a la página de publicaciones donde debería reflejar la publicación recién creada.

Figura 56

Publicación creada con el formulario completo

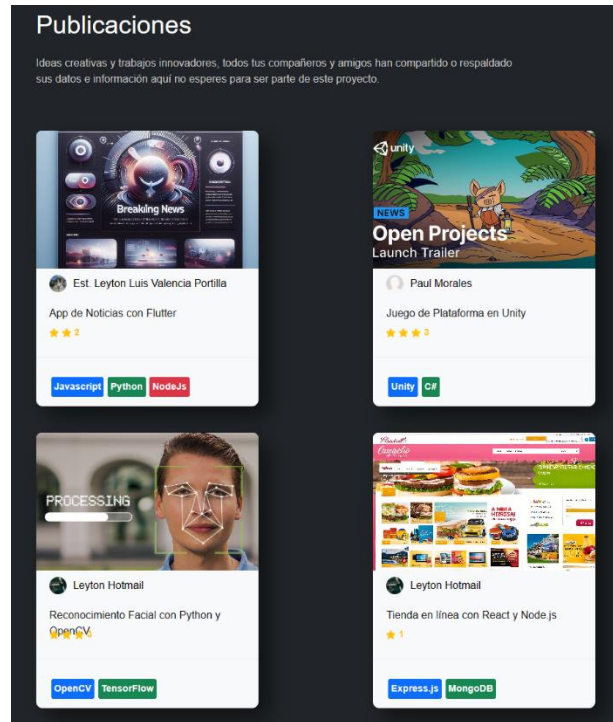


Descarga de documentos

- a. El usuario ingresa a la sección de publicaciones

Figura 57

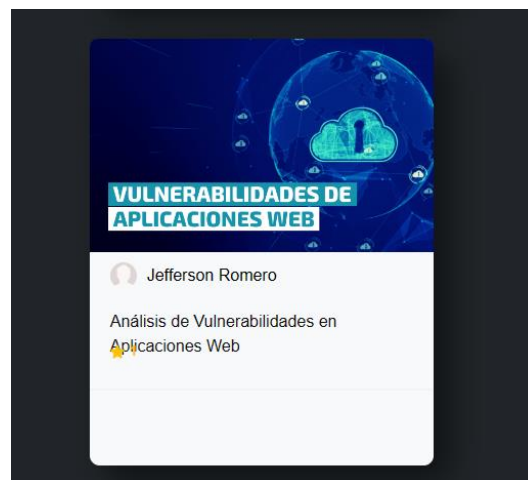
Sección de todas las publicaciones



- b. El usuario selecciona la publicación que le cause más interés.

Figura 58

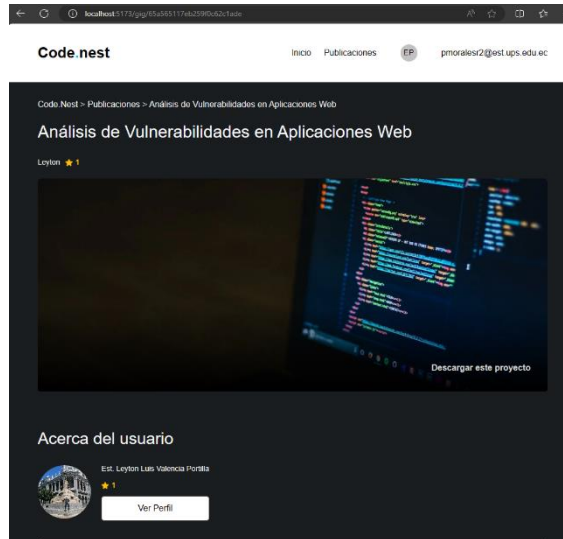
Componente de publicación en sección de publicaciones



- c. El sistema despliega las publicaciones con toda la información con la que fue creada y subida.

Figura 59

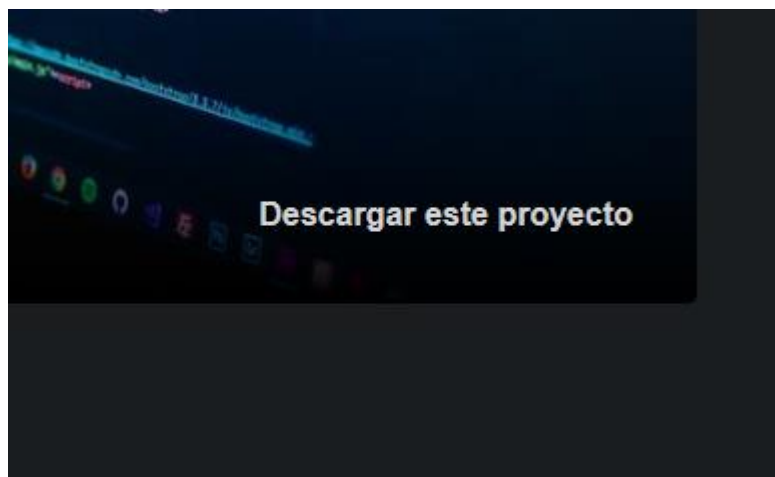
Vista principal cuando se ingresa a una publicación



- d. El usuario contará visualmente con toda la información y a la vez podrá descargar el documento correspondiente en dos botones visibles.

Figura 60

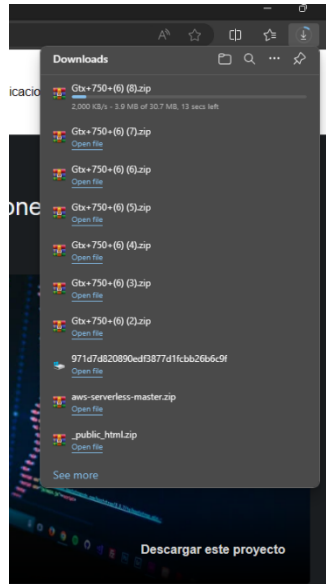
Botón de descargar proyecto



- e. El sistema se encargará de descargar el archivo correspondiente desde una URL pública del archivo almacenado en AWS a la que se redirigen los botones mencionados anteriormente.

Figura 61

Imagen referencia cuando se descarga el .zip



- 5.a El sistema no puede terminar de cargar la publicación correspondiente y emite un mensaje de error del motivo.

Figura 62

Validación con respecto al tamaño del archivo subido al formulario

Cover Image

Choose File | img2.jpg

Upload Images

Choose Files | img1.png

El archivo de la portada supera el límite de 10 MB

uploading

Archivo Rar o ZIP

Choose File | Gtx+750+(6) (3).zip

3.a El sistema no encuentra publicaciones con el mapeo correspondiente en MongoDB y da un mensaje indicando que no se ha encontrado nada.

Figura 63

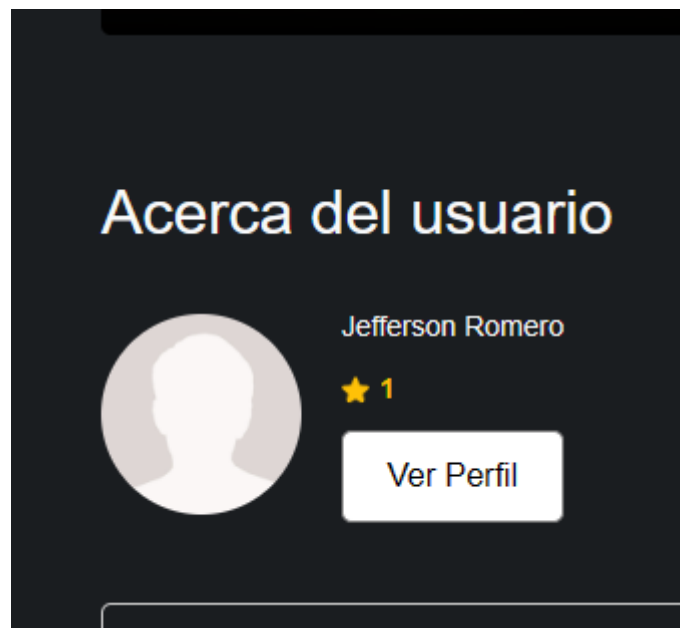
Mensaje de control cuando no se logra cargar la información desde el servidor backend.



5.a El sistema tiene un error al intentar conectar con la base de AWS que se encarga de brindarnos un enlace para representar el dato, se cambian estos valores por unos de defecto.

Figura 64

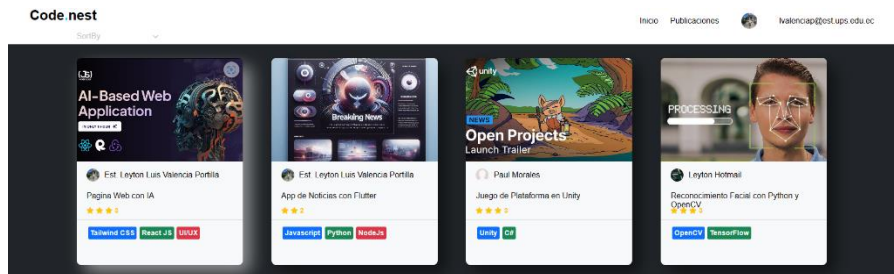
Validación para no cargar una imagen del usuario cuando no se ha elegido aun



- RF005 – Comentarios Interactivos en Publicaciones
 - a. El usuario accede a alguna publicación de su interés

Figura 65

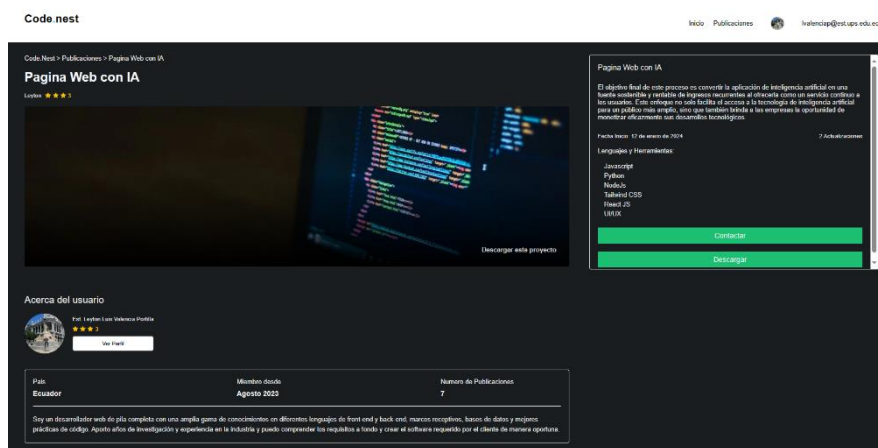
Listado de publicaciones con sus tecnologías.



- b. El sistema despliega todos los datos de la publicación

Figura 66

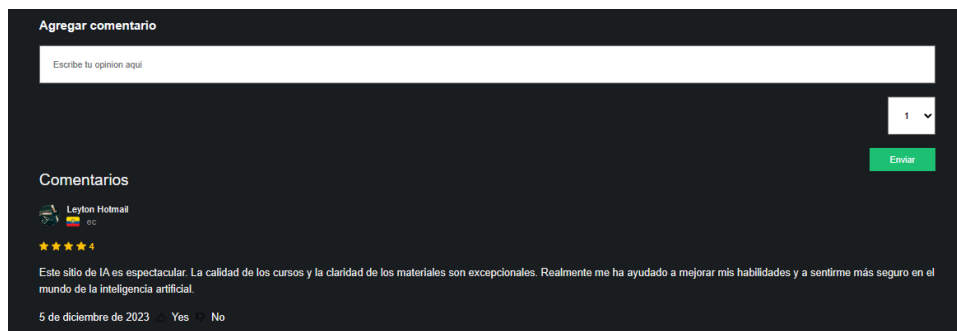
Vista completa de página de publicación



- c. El usuario observa los comentarios anteriores

Figura 67

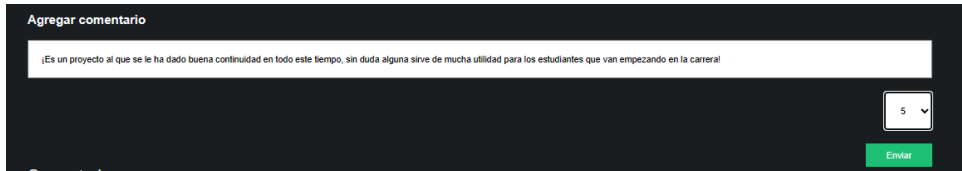
Sección de comentarios en publicación



d. El usuario coloca su opinión en el formulario ya desplegado en la página

Figura 68

Campos de componentes para publicar un comentario

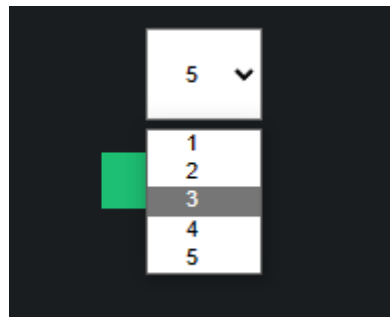


A screenshot of a form titled "Agregar comentario". It features a text input field with the placeholder text: "¡Es un proyecto al que se le ha dado buena continuidad en todo este tiempo, sin duda alguna sirve de mucha utilidad para los estudiantes que van empezando en la carrera!". To the right of the input field is a dropdown menu currently showing the number "5". Below the dropdown is a green button labeled "Enviar".

e. El usuario selecciona una calificación en base a su comentario

Figura 69

ComboBox de puntuación en el comentario

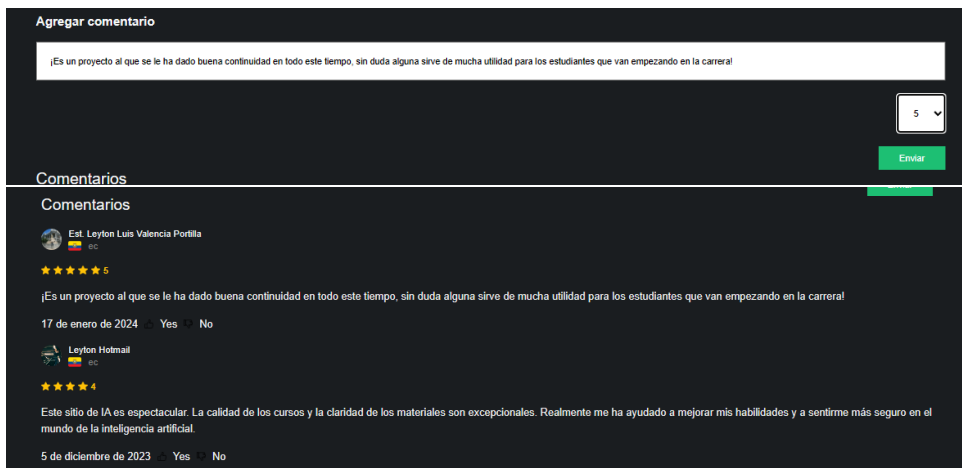


A close-up of the rating dropdown menu. The menu is open, showing a list of numbers from 1 to 5. The number "3" is highlighted with a grey background. A green square is visible to the left of the menu.

f. El usuario envía el comentario y se actualiza inmediatamente la sección

Figura 70

Visualización de comentarios general

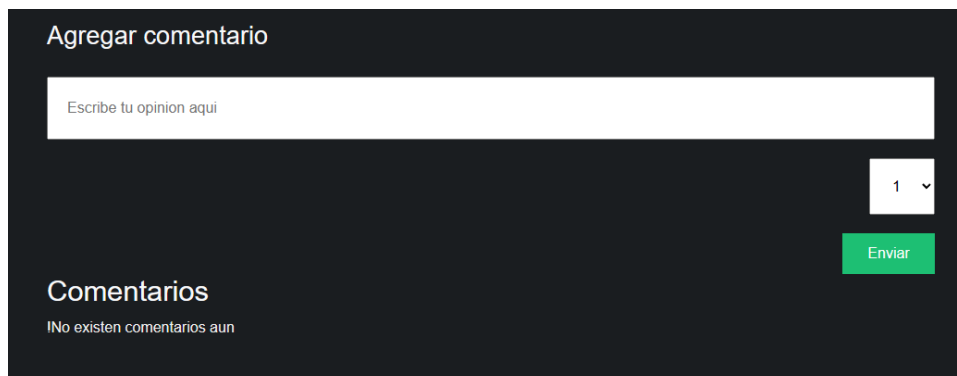


A screenshot showing the "Agregar comentario" form at the top and a "Comentarios" section below. The form has the same placeholder text and rating dropdown as in Figure 68. The "Comentarios" section displays two user reviews. The first review is from "Est. Leyton Luis Valencia Portilla" (ec) with a 5-star rating and the same placeholder text, dated "17 de enero de 2024" with "Yes" and "No" options. The second review is from "Leyton Holmail" (ec) with a 4-star rating and the text "Este sitio de IA es espectacular. La calidad de los cursos y la claridad de los materiales son excepcionales. Realmente me ha ayudado a mejorar mis habilidades y a sentirme más seguro en el mundo de la inteligencia artificial.", dated "5 de diciembre de 2023" with "Yes" and "No" options.

3.a El sistema no encuentra comentarios previos y muestra un mensaje

Figura 71

Mensaje de validación cuando no se tienen comentarios en la publicación



The image shows a dark-themed user interface for adding a comment. At the top, the text "Agregar comentario" is displayed. Below it is a large white text input field with the placeholder text "Escribe tu opinion aqui". To the right of the input field is a small white box containing the number "1" and a downward-pointing arrow, indicating a dropdown menu. Below the input field is a green button with the text "Enviar". Underneath the input field, the section "Comentarios" is visible, with the message "No existen comentarios aun" below it.

CONCLUSIONES

Después de haber realizado todo el estudio previo a toda la construcción de este trabajo de titulación encontramos varios puntos que podrían ser un tema a discusión del presente proyecto, no solo a nivel técnico, también existe un cierto nivel de resistencia de los docentes ante un repositorio, sin embargo, el desarrollo y varios replanteamientos se dio paso a la creación del sistema.

Finalizando el diseño nos encontramos con que el proyecto debía tener un ciclo de vida bien fijado, pues en este punto se descartaron algunos requerimientos que podrían ser tomados como tesis de futuros alumnos que trabajen sobre este sistema, como una posible reubicación de hosting del repositorio, al igual que el lugar de almacenamiento de las bases de datos, esto dado porque tenemos la visión y la confianza en nuestros docentes que el presente proyecto seguirá creciendo con próximos estudiantes.

Este proyecto de tesis ha culminado con la creación de un repositorio de excelente calidad, destacándose por su implementación en un framework tan usado como React y la integración de servicios cloud. La elección estratégica de APIs ha mejorado y optimizado significativamente las funcionalidades del repositorio potenciando al máximo la experiencia de usuario. El seguir esta arquitectura permite la compartición eficiente de datos entre los diferentes servicios a la vez que se mantienen en anonimato preservando la privacidad y confidencialidad de usuarios.

En la culminación de un proyecto de este tipo representa un paso importante en la trayectoria académica y profesional donde se tienen que aplicar una amplia gama de conocimientos adquiridos a lo largo de la carrera y donde se puede experimentar el crecimiento personal y confianza de enfrentar desafíos informáticos ambiciosos.

RECOMENDACIONES

Se recomienda pensar en la utilidad que brinda un repositorio para centralizar la información, esto da pie a proyectos que permitan detectar de mejor manera el plagio en código de estudiantes, comenzar un proceso de enriquecimiento del repositorio y también sin mirar más allá, usarlo con el fin más simple con el que fue creado, compartir información.

En el desarrollo del trabajo de titulación nos encontramos con tecnologías nuevas e innovadoras que deberían ser socializadas y tratadas por los docentes, la carrera se puede sentir anticuada no solo en desarrollo web, sino también en infraestructura y networking, el uso de nubes públicas, desarrollo y uso de APIs, crear proyectos que crezcan con el tiempo son temas críticos que se tratan de manera casi nula en la carrera, por lo que invitamos de la manera más cordial a pensar en innovar, con la gran capacidad de transmitir conocimientos que tienen los docentes.

El enfoque que se tiene al evaluar la profundidad de nuevas tecnologías que pueden funcionar en conjunto como una arquitectura impacta en gran manera al desarrollo y mantenimiento de los sistemas. Gracias a esto se pueden elaborar propuestas valiosas llenas de conocimientos sobre la escalabilidad, flexibilidad y factibilidad de integración de estas herramientas.

BIBLIOGRAFÍA

- ÁLVAREZ, M. A., FUNDADOR DE DESARROLLOWEB.COM. (2023, 20 SEPTIEMBRE). QUÉ ES MVC. DESARROLLOWEB.COM. RECUPERADO 29 DE OCTUBRE DE 2023, DE [HTTPS://DESARROLLOWEB.COM/ARTICULOS/QUE-ES-MVC.HTML](https://desarrolloweb.com/articulos/que-es-mvc.html)
- CENTRO DE AYUDA DE HOSTINGER. (S. F.). ¿QUÉ MEDIDAS DE SEGURIDAD USA HOSTINGER? [HTTPS://SUPPORT.HOSTINGER.COM/ES/ARTICLES/1583287-QUE-MEDIDAS-DE-SEGURIDAD-USA-HOSTINGER](https://support.hostinger.com/es/articles/1583287-que-medidas-de-seguridad-usa-hostinger)
- CLOUDINARY. (2023, 7 AGOSTO). CLOUDINARY REVIEW: CUSTOMER EXPERIENCES AND SUCCESS STORIES. [HTTPS://CLOUDINARY.COM/CUSTOMERS=](https://cloudinary.com/customers=)
- DAVEY, B., & TATNALL, A. (2002). TOOLS FOR CLIENT SERVER COMPUTING. SOFTWARE ENGINEERING: EDUCATION AND PRACTICE, 1996. PROCEEDINGS. INTERNATIONAL CONFERENCE. [HTTPS://DOI.ORG/10.1109/SEEP.1996.534011](https://doi.org/10.1109/SEEP.1996.534011)
- GÓMEZ, P. M. (2023, 17 ABRIL). POR QUÉ USAR MONGODB. OPENWEBINARS.NET. [HTTPS://OPENWEBINARS.NET/BLOG/POR-QUE-USAR-MONGODB/](https://openwebinars.net/blog/por-que-usar-mongodb/)
- KINSTA. (2022, 19 DICIEMBRE). ¿QUÉ ES EXPRESS.JS? TODO LO QUE DEBES SABER. KINSTA®. [HTTPS://KINSTA.COM/ES/BASE-DE-CONOCIMIENTO/QUE-ES-EXPRESS/](https://kinsta.com/es/base-de-conocimiento/que-es-express/)
- MONGODB. (2018). TOP 5 CONSIDERATIONS WHEN EVALUATING NOSQL DATABASES [CONJUNTO DE DATOS]. EN WHITE PAPER. [HTTPS://WWW.MONGODB.COM/SCALE/NOSQL-DATABASES-PROS-AND-CONS](https://www.mongodb.com/scale/nosql-databases-pros-and-cons)
- REACT. (S. F.). NATIVE AUTHENTICATION WITH EXPO ROUTER & CLERK. [HTTPS://GALAXIES.DEV](https://galaxies.dev). [HTTPS://GALAXIES.DEV/REACT-NATIVE-AUTHENTICATION-CLERK](https://galaxies.dev/react-native-authentication-clerk)
- SCRUMGUIDES.ORG. (2020). PURPOSE OF THE SCRUM GUIDE. [HTTPS://SCRUMGUIDES.ORG/DOCS/SCRUMGUIDE/V2020/2020-SCRUM-GUIDE-US.PDF](https://scrumguides.org/docs/scrumguide/v2020/2020-scrum-guide-us.pdf)
- TEJENA-MACÍAS, M. A. (2018). ANÁLISIS DE RIESGOS EN SEGURIDAD DE LA INFORMACIÓN. POLO DEL CONOCIMIENTO. [HTTPS://DOI.ORG/10.23857/PC.V3I4.809](https://doi.org/10.23857/pc.v3i4.809)