

REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD VALLE DEL MOMBOY
VICERRECTORADO ACADÉMICO
FACULTAD DE INGENIERÍA
ESCUELA INGENIERIA DE COMPUTACIÓN



Plataforma Social de Componentes Open Source

Autor:

BR. ALEXANDER J. AVENDAÑO R.

Valera, Octubre 2024

REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD VALLE DEL MOMBOY
VICERRECTORADO ACADÉMICO
FACULTAD DE INGENIERÍA
ESCUELA INGENIERIA DE COMPUTACIÓN



Plataforma social de Componentes Open Source

Trabajo Especial de Grado para Optar al título de Ingeniero de Computación

AUTOR:

BR. ALEXANDER J. AVENDAÑO R.

TUTOR:

MSc. ROBERTO DI MICHELE

VALERA, OCTUBRE 2024

AGRADECIMIENTO

A mis padres; su amor y apoyo incondicional son los resultados de este trabajo, ustedes hicieron posible logro.

A mi querida hermana; Mahira, a mis tíos; Fernando, Franklin y mi primo Freddy, por el apoyo brindado.

A mi novia Natasha, por motivarme a seguir adelante hasta alcanzar todo lo que me proponga para mi futuro personal y profesional.

A mis compañeros y profesores; quienes me acompañaron durante tantos años de carrera, su solidaridad y respaldo permitieron que el aprendizaje fuese más significativo.

DEDICATORIA

Este Trabajo Especial de Grado está dedicado principalmente a mis padres; Fanny y Ramón, de igual manera a mi incondicional hermana Mahira, ustedes han sido el cimiento para yo construir que este logro.

También dedico esta meta a todos mis familiares y amigos que en el transcurso de mis estudios me apoyaron, especialmente a mi tío Fernando fuente de inspiración.

VEREDICTO



VICERRECTORADO ACADÉMICO FACULTAD DE INGENIERÍA

VEREDICTO

Nosotros, **Prof. Roberto Di Michele, Prof. Yerson González y Profa. Yajaira Segovia**, designados como miembros del Jurado Examinador del Trabajo de Grado titulado: **“PLATAFORMA SOCIAL DE COMPONENTES OPEN SOURCE”**, que presenta el **bachiller Avendaño Ramírez Alexander José**, portador de la C.I. N° **29.694.896**, nos hemos reunido para revisar dicho trabajo y después de la presentación, defensa e interrogatorio correspondiente lo hemos calificado con: **Veinte (20)** puntos, de acuerdo con las normas vigentes dictadas por el Consejo Universitario de la Universidad Valle del Momboy, referente a la evaluación de los Trabajos de Grado para optar al título de Ingeniero de Computación.

En fe de lo cual firmamos en Carvajal a los treinta (30) días del mes de octubre del dos mil veinticuatro (2024).

Profa. Yajaira Segovia
C.I: 14.148.893
JURADO

Prof. Roberto Di Michele
C.I: 19.794.45513
TUTOR

Prof. Yerson González
C.I. 14.149.542
PRESIDENTE DEL JURADO



Profa. Yumary Valecillos
C.I. 14.151.309
DECANO



Prof. Zaida Kassar
C.I. 9.175.011
**VICERRECTORA
ACADEMICA**



+58 412 2263605



www.uvm.edu.ve



universidadvalledelmomboy@uvm.edu.ve

RESUMEN

La presente investigación se planteó como objetivo desarrollar una plataforma social online que facilite el intercambio de componentes de programación open source internamente en la UVM. Este trabajo se enfoca a través de una metodología descriptiva, recopilando datos sobre los medios de búsqueda de componentes más utilizados por los usuarios y los requerimientos técnicos y de seguridad ineludibles para realizar la plataforma. Para obtener el resultado, se diseñó e implementó una plataforma colaborativa que centraliza estos recursos, mejorando la accesibilidad y fomentando el intercambio de conocimientos. La plataforma también permite la interacción activa entre estudiantes, profesores y profesionales, incentivando la creación y mejora continua de componentes de software. Los hallazgos destacaron la necesidad de contar con un entorno intuitivo, seguro y con herramientas colaborativas como comentarios, valoraciones y notificaciones en tiempo real. La propuesta contribuye significativamente al desarrollo de software colaborativo, cumpliendo con los objetivos planteados y aportando un modelo práctico para gestionar proyectos tecnológicos en entornos educativos. Las conclusiones indican que la plataforma fortalece la colaboración y el aprendizaje mutuo, mientras que las recomendaciones sugieren la inversión en mejoras continuas, alianzas estratégicas con el sector privado y la promoción activa de la plataforma de una manera interna y externa en la comunidad uvemista. Esta investigación proporciona una solución escalable y sostenible para la UVM, con un potencial impacto en la ingeniería de software y el desarrollo colaborativo.

Palabras clave: plataforma social, open source, comunidad uvemista, desarrollo colaborativo, ingeniería de software

ABSTRACT

This research work aimed to develop an online social platform that facilitates the exchange of open source programming components within the Uvemista community. The research was carried out using a descriptive methodological approach, collecting data on the most commonly used means of searching for components by users and the technical and security requirements necessary to implement the platform. As a result, a collaborative platform was designed and implemented that centralizes these resources, improving accessibility and encouraging the exchange of knowledge. The platform also allows active interaction between students, teachers and professionals, encouraging the creation and continuous improvement of software components. The findings highlighted the need for an intuitive, secure environment with collaborative tools such as comments, ratings and real-time notifications. The proposal contributes significantly to the development of collaborative software, meeting the stated objectives and providing a practical model for managing technological projects in educational environments. The findings indicate that the platform strengthens collaboration and mutual learning, while recommendations suggest investment in continuous improvements, strategic alliances with the private sector, and active promotion of the platform within and outside the community. This research provides a scalable and sustainable solution for the Uvemista community, with a potential impact on software engineering and collaborative development.

Keywords: social platform, open source, Uvemista community, collaborative development, software engineering

INDICE

AGRADECIMIENTO	3
VEREDICTO.....	5
RESUMEN	6
ABSTRACT.....	7
INDICE.....	8
INDICE DE TABLAS	13
INDICE DE FIGURAS.....	14
INTRODUCCIÓN	15
I. FASE DE PLANIFICACIÓN	17
Diagnóstico Situacional	17
Problemas de la investigación.....	18
Problema general	18
Problemas específicos.....	18
Formulación de Objetivos.....	18
Objetivo general.....	18
Objetivos específicos	19
Justificación de la Investigación	19
Teórica	19

Social.....	20
Práctico	20
Delimitación.....	20
Revisión de la literatura	21
Estudios Previos.....	21
Nacionales:.....	21
Internacionales:.....	23
Bases teóricas.....	27
Frameworks.....	27
Tendencia de crecimiento de la comunidad.....	28
Requisitos de la plataforma.....	30
Arquitectura del diseño	30
Lenguaje de programación.....	32
Bases de datos	35
Cronograma de Planificación.....	37
Operacionalización de las Variables	38
II. FASE DE IMPLEMENTACIÓN.....	50
Diseño de la investigación	50
Tipo de investigación.....	50
Nivel de la investigación.....	51

	10
Diseño de la investigación	52
Población y muestra	52
Población.....	52
Muestra	53
Diseño de instrumento de recolección de datos	53
Instrumentos.....	54
Validación y Confiabilidad de Instrumentos	54
Validez	55
Confiabilidad.....	55
Análisis de datos	56
Análisis Descriptivo.....	58
Análisis del cuestionario	58
Dimensión: Conexiones Sociales.....	58
<i>Indicador: Medios para encontrar componentes open source</i>	59
<i>Indicador: Frameworks</i>	62
<i>Indicador: Tendencia de crecimiento de la comunidad</i>	64
Análisis de las Entrevistas	67
Dimensión: Requerimientos de la plataforma.....	67
Integración de Resultados.	72
Análisis de resultados del cuestionario: Conexiones Sociales.....	72

	11
Integración con el primer objetivo.....	73
Análisis de resultados de la entrevista: Requerimientos de la plataforma.....	74
Integración con el segundo objetivo.....	74
Validación de Resultados.....	75
III. FASE DE PRESENTACIÓN.....	78
Conclusiones.....	78
Recomendaciones.....	79
Planteamiento de Propuesta.....	80
Justificación de la Propuesta.....	80
Misión.....	80
Diseño de la Propuesta.....	80
Requerimientos del Sistema.....	81
Arquitectura del Diseño.....	81
Planificación y Desarrollo de la Propuesta.....	82
Prototipo de la propuesta de la Plataforma social de componentes Open Source	83
REFERENCIAS.....	96
ANEXOS.....	101
ANEXO 1.....	102
ANEXO 2.....	106
ANEXO 3.....	109

ANEXO 4.....	112
ANEXO 5.....	114
ANEXO 6.....	116
ANEXO 7.....	132

INDICE DE TABLAS

Tabla 1. Cronograma de planificación.....	37
Tabla 2. Cuadro de operacionalización de las variables.....	38
Tabla 3. Medios para encontrar componentes Open Source.....	59
Tabla 4. Medios para encontrar componentes Open Source.....	60
Tabla 5. Medios para encontrar componentes Open Source.....	61
Tabla 6. Frameworks	62
Tabla 7. Frameworks	63
Tabla 8. Tendencia de crecimiento de la comunidad.....	64
Tabla 9. Tendencia de crecimiento de la comunidad.....	65
Tabla 10. Tendencia de crecimiento de la comunidad.....	66
Tabla 11. Análisis cualitativo de la dimensión requerimientos de la plataforma	67
Tabla 12. Análisis cualitativo de la dimensión requerimientos de la plataforma	68
Tabla 13. Análisis cualitativo de la dimensión requerimientos de la plataforma	68
Tabla 14. Análisis cualitativo de la dimensión requerimientos de la plataforma	69
Tabla 15. Análisis cualitativo de la dimensión requerimientos de la plataforma	69
Tabla 16. Análisis cualitativo de la dimensión requerimientos de la plataforma	70
Tabla 17. Análisis cualitativo de la dimensión requerimientos de la plataforma	70
Tabla 18. Análisis cualitativo de la dimensión requerimientos de la plataforma	71
Tabla 19. Análisis cualitativo de la dimensión requerimientos de la plataforma	71

INDICE DE FIGURAS

Ilustración 1. Medios para encontrar componentes Open Source.....	59
Ilustración 2. Medios para encontrar componentes Open Source.....	60
Ilustración 3. Medios para encontrar componentes Open Source.....	61
Ilustración 4. Frameworks	62
Ilustración 5. Frameworks	63
Ilustración 6. Tendencia de crecimiento de la comunidad.....	64
Ilustración 7. Tendencia de crecimiento de la comunidad.....	65
Ilustración 8. Tendencia de crecimiento de la comunidad.....	66
Ilustración 9. Login.....	83
Ilustración 10. Register	84
Ilustración 11. Pantalla principal: Para ti.....	85
Ilustración 12. Pantalla principal: Siguiendo	86
Ilustración 13. Pantalla principal: Populares	87
Ilustración 14. Pantalla principal: Buscar	88
Ilustración 15. Componente/Publicación	89
Ilustración 16. Perfil de un usuario	90
Ilustración 17. Pantalla principal: Mi perfil.....	91
Ilustración 18. Crear un componente	92
Ilustración 19. Mi componente/publicación	93
Ilustración 20. Editar Perfil.....	94
Ilustración 21. Pantalla principal: Configuraciones.....	95

INTRODUCCIÓN

El presente proyecto tiene como objetivo principal desarrollar una plataforma social comunitaria para compartir componentes de programación open source, dirigida a la comunidad uvemista. Este estudio se enmarca en la creciente importancia de los entornos colaborativos en el ámbito del desarrollo de software, donde el acceso y la reutilización de componentes open source representan una oportunidad significativa para mejorar la eficiencia y promover la innovación. En este contexto, la plataforma no solo facilitará el intercambio de conocimientos entre estudiantes, profesores y profesionales, sino que también ofrecerá un entorno propicio para el aprendizaje práctico y el crecimiento profesional.

El primer capítulo ofrece una visión general de la problemática y los antecedentes que justifican la elaboración de esta plataforma, observando las carencias actuales en la colaboración y el acceso a recursos compartidos dentro de la comunidad institucional. A continuación, se exponen los objetivos del estudio, centrados en la creación de una plataforma que responda a las necesidades de los usuarios mediante una interfaz intuitiva, segura y robusta.

El segundo capítulo presenta el enfoque metodológico utilizado en el estudio, basado en el método descriptivo, apoyado en encuestas y análisis de datos cualitativos y cuantitativos. Esto permitió identificar los medios de búsqueda más utilizados por los desarrolladores y los requerimientos técnicos y de seguridad que son necesarios para la buena marcha de la plataforma.

El tercer capítulo aborda en los resultados y conclusiones, donde se analiza el impacto de la plataforma en la comunidad uvemista y las recomendaciones para su mejora continua. A lo largo del trabajo, se demuestra cómo esta plataforma representa un avance importante en la gestión y desarrollo colaborativo de software, subrayando su relevancia para el entorno académico y profesional.

Finalmente se presenta el diseño y desarrollo de la propuesta tecnológica, detallando su proceso de ejecución mediante tecnologías modernas como Next.js, TypeScript, TailwindCSS, Express, GraphQL, Prisma y MySQL. Se abordan; requisitos del sistema, arquitectura de la solución, proceso de planificación y desarrollo, incluyendo wireframes y la organización de las funcionalidades.

I. FASE DE PLANIFICACIÓN

Diagnóstico Situacional

Desarrollar un software open source en la rama de la ingeniería de computación en el contexto actual se destaca su rol fundamental como innovación tecnológica que permite el avance de la comunidad uvemista. Sin embargo, a pesar del creciente interés y la participación en proyectos de código abierto, persisten los desafíos para una colaboración eficaz y la gestión de recursos compartidos.

Uno de los principales problemas identificados es la falta de una plataforma centralizada que facilite el intercambio de componentes de programación open source dentro de la comunidad uvemista. Lo que limita el potencial de innovación y desarrollo.

Además, la seguridad y la integridad de los componentes compartidos son preocupaciones importantes. La ausencia de estándares claros y la falta de medidas de seguridad adecuadas pueden exponer a los usuarios a riesgos de vulnerabilidades y comprometer la confianza en los recursos compartidos.

Ante este panorama, resulta imperativo desarrollar una plataforma online comunitaria que permita compartir componentes de programación open source de manera segura y colaborativa. Esta plataforma servirá como un espacio centralizado para la colaboración y el intercambio de conocimientos, además fomentará la transparencia, la accesibilidad y la seguridad en el desarrollo de software open source en la comunidad uvemista.

Por tanto, en la presente investigación se pretende dar respuesta a las siguientes preguntas:

¿Cómo desarrollar una plataforma social de componentes open source que satisfaga las necesidades de la comunidad uvemista, garantizando la seguridad y la colaboración efectiva en el intercambio de recursos compartidos?

La respuesta a esta pregunta no solo contribuirá al avance de la ingeniería de computación en la Universidad Valle del Momboy (UVM), sino que también proporcionará una solución práctica y sostenible para optimizar la gestión de recursos y fomentar la innovación en el desarrollo de software open source.

Problemas de la investigación

Problema general

¿Cómo desarrollar una plataforma online comunitaria para compartir componentes de programación open source?

Problemas específicos

¿Cuáles son los medios por los cuales la comunidad uvemista busca componentes de programación open source?

¿Cuáles son los requerimientos técnicos y de seguridad para la implementación de componentes compartidos por la comunidad?

¿Cómo desarrollar una plataforma social que permita compartir componentes open source?

¿Como implementar la plataforma social desarrollada para facilitar el intercambio de componentes open source?

Formulación de Objetivos

Objetivo general

Desarrollar una plataforma online comunitaria para compartir componentes de programación open source.

Objetivos específicos

Identificar los medios por los cuales la comunidad uvemista busca componentes de programación open source.

Definir los requerimientos técnicos y de seguridad para la implementación de componentes compartidos por la comunidad.

Desarrollar una plataforma social para compartir componentes de programación open source.

Implementar la plataforma social desarrollada para facilitar el intercambio de componentes open source.

Justificación de la Investigación

Para la comunidad uvemista el desarrollo de una plataforma social de componentes open source representa una contribución significativa al área de la ingeniería de computación, respondiendo así a la necesidad de fomentar la colaboración en espacios como la seguridad y el desarrollo de software. A tales efectos, Martínez Santos (2021) expone: “Open source se refiere a todos aquellos programas informáticos que disponen a cualquier usuario el acceso a su código de programación facilitando por parte de otros programadores ajenos a la modificación del mismo” (p.3).

Teórica

Esta investigación examinará estrategias en la ingeniería de computación fomentando la ayuda y el intercambio de recursos desde una perspectiva teórica. A través de la creación de una plataforma dedicada al intercambio de componentes de programación open source se generará información valiosa sobre las mejores prácticas en el diseño, implementación y gestión de sistemas colaborativos en entornos educativos y profesionales.

Social

En este ámbito la plataforma fortalecerá la colaboración entre estudiantes, profesores y profesionales, ofreciéndoles una herramienta para agilizar el desarrollo de componentes e incentivando la participación activa y el aprendizaje colaborativo, asimismo promoverá un sentido de pertenencia e identidad dentro de la comunidad universitaria.

Práctico

Finalmente, la investigación tiene implicaciones prácticas significativas al ofrecer una solución determinada y escalable para optimizar la gestión de recursos y activar la innovación en el desarrollo web. Esta plataforma ayudará a reducir costos y tiempo por lo que agilizará la búsqueda de componentes de programación.

Delimitación

El estudio se realizó en la Facultad de Ingeniería de la Universidad Valle del Momboy, ubicada en la Sede Estovacuy, Municipio San Rafael de Carvajal, del estado Trujillo, durante un lapso de 8 meses. Las entrevistas y encuestas fueron elaboradas para ser aplicadas a los estudiantes de la carrera de Ingeniería de Computación, como objetivo para evaluar el uso y las necesidades en torno a una plataforma social de componentes open source. Cabe destacar que esta plataforma se diseñó para facilitar el desarrollo de componentes reutilizables open source, pero debido al rápido avance tecnológico y los cambios en las tendencias del desarrollo de software, se debe señalar que los datos obtenidos podrían perder relevancia o quedar desactualizados con el tiempo.

Revisión de la literatura

Estudios Previos

El desarrollo de una plataforma social de componentes Open Source, constituyen un importante tema de investigación en propuestas de innovación educativa, por lo que es imprescindible establecer los antecedentes de la investigación. Al indagar en los estudios referenciales, se buscará establecer una base sólida que consolide este proyecto, identificando estrategias y herramientas implementadas con éxito. Esta revisión exhaustiva de la literatura existente proporcionará una comprensión más amplia de las plataformas sociales y componentes open source. Para efectos de respaldar el presente estudio se han consultado diferentes autores, evaluando su relevancia y eficacia en el ámbito educativo. Los siguientes antecedentes en el contexto nacional e internacional, hace referencia a la creciente relevancia de las plataformas sociales y el desarrollo de software open source, estudios que proporcionan una base sólida para comprender la implementación y los beneficios de plataformas similares a la propuesta en esta investigación.

Nacionales:

En este aspecto, Rojo Alfonso (2020), en su investigación intitulada: "SEGURIDAD INFORMÁTICA: TECNOLOGÍA DE DEFENSA EN PROFUNDIDAD Y PENTESTING", desarrollada en la Universidad Valle del Momboy. Tuvo como objetivo principal analizar el modelo de Defensa en Profundidad y el Pentesting como estrategias de seguridad informática, en un contexto donde la creciente demanda tecnológica plantea diversos desafíos en ciberseguridad. El enfoque de la seguridad informática es reducir estos riesgos, para asegurar la protección en un entorno global cada vez más interconectado y en constante evolución.

La investigación se desarrolló bajo una metodología de corte documental y exploratorio, empleando un cuestionario como instrumento aplicado a una muestra intencional de 37 empresas de Valera. Los resultados evidenciaron la importancia de garantizar la seguridad, confiabilidad y disponibilidad de los sistemas ante amenazas cibernéticas. Además, destacó la utilidad del Pentesting para identificar vulnerabilidades en la seguridad. Se observó que la mayoría de las PyMEs en Valera, Estado Trujillo, han enfrentado problemas de seguridad informática y carecen de métodos para proteger sus datos críticos. Por tanto, los objetivos del estudio fueron alcanzados.

El estudio también destaca la relevancia de proteger los datos de los usuarios y la seguridad en la transferencia de archivos en plataformas sociales. La investigación subraya la importancia de implementar medidas de seguridad sólidas para proteger la información de los usuarios y reducir los riesgos derivados de amenazas cibernéticas. Además, al destacar el Pentesting como una técnica eficaz para detectar vulnerabilidades, subraya la importancia de realizar evaluaciones periódicas de seguridad en la plataforma social para identificar y solucionar posibles debilidades en su infraestructura y protocolos de seguridad. Se recalca la necesidad de asegurar tanto el almacenamiento como el intercambio de archivos en plataformas de componentes open source.

Gaitán Juan (2018), en la Universidad Coronel Agustín Codazzi, desarrolló una investigación titulada “Diseño y elaboración de una página web de información publicitaria para la galletera ITALIA.C. A Ubicada en el municipio Barinas estado Barinas”, con el fin de obtener el título de Técnico Superior Universitario en Informática. El objetivo de este trabajo fue crear una página web para la galletera que promueva eficazmente sus productos y permita expandir su reconocimiento tanto a nivel nacional como internacional, abordando cuestiones sobre la viabilidad y el diseño web para mejorar su promoción.

La investigación se basó en un enfoque de campo y descriptivo. Para el que utilizó una encuesta como técnica de recolección de datos, tuvo como población un conjunto de seis personas que trabajaban directamente en los departamentos de Tecnología y Publicidad de la empresa, conformando una muestra censal.

El estudio subraya la importancia de obtener un equilibrio entre el diseño visual y el contenido para impulsar la marca y satisfacer las expectativas de los clientes. Este enfoque en la UX y en la promoción de productos resalta la necesidad de una plataforma social sólida para compartir componentes de programación open source. El estudio también sugiere que captar la atención de los usuarios requiere no solo de un enfoque en la funcionalidad de las herramientas, sino también de un diseño visual atractivo. Este equilibrio entre diseño y contenido se considera un aporte clave para el desarrollo de una plataforma social de componentes open source, ya que permite lograr una conexión efectiva con los usuarios y alcanzar los objetivos de promoción y funcionalidad.

Internacionales:

Marcos González (2021), presentó su trabajo de grado titulado "Diseño y desarrollo de una aplicación web como red social para aficionados a la automovilística" para optar al título de Ingeniero en Informática. En la Universidad Politécnica de Valencia. El objetivo principal fue crear una aplicación web que conectara a los entusiastas del motor, funcionando como una red social y ofreciendo además información sobre este tema. En esta investigación se destaca el impacto de las redes sociales, especialmente entre los jóvenes, como una herramienta ágil para acceder a información. Aunque el automovilismo está presente en redes sociales y otros medios digitales, no existía una red social específica para este mundo, lo cual impulsó el desarrollo de SocialRoad, una plataforma dedicada al automovilismo.

El estudio se basó en una investigación cualitativa, utilizando un cuestionario de 18 preguntas que recopiló 29 respuestas. Los resultados más relevantes incluyen que el 89.7% de los usuarios se encuentra en el rango de edad de 18 a 25 años, y un 58.6% son hombres. Solo un 3.4% de los encuestados reportó no usar redes sociales frecuentemente. La organización y la definición temprana de las tecnologías a utilizar fueron aspectos clave en el desarrollo del proyecto, destacando el uso de herramientas como Visual Studio Code, que facilitaron la adaptación a nuevas tecnologías. A pesar de los retos enfrentados, el proyecto fue exitoso, proporcionando al autor nuevos conocimientos útiles para su futuro profesional.

El trabajo de Marcos González guarda similitudes con la investigación propuesta, ya que ambos se centran en el desarrollo de plataformas web que actúan como redes sociales especializadas, conectando a comunidades con intereses específicos y ofreciendo contenido relevante. Mientras que González se enfocó en los aficionados al automovilismo, el proyecto planteado se enfoca en compartir componentes de programación. Aunque sean diferentes temáticas, ambos proyectos responden a la necesidad de ofrecer a los usuarios una fuente confiable y centralizada de información. La metodología cualitativa utilizada por González, basada en cuestionarios, también resulta valiosa para comprender las preferencias de los usuarios en el contexto de la plataforma de componentes open source. Además, la experiencia organizativa de González subraya la importancia de una clara planificación y el uso de herramientas tecnológicas adecuadas, aspectos fundamentales también para el éxito de la plataforma social.

Por otra parte, Alfredo Ramos (2020) desarrolló una investigación titulada "Tutor Inteligente Móvil para el Aprendizaje de la Experiencia de Usuario (UX) para Diseñadores Web", con la cual obtuvo su licenciatura en informática en la Universidad Mayor de San Andrés. El principal objetivo fue diseñar e implementar un tutor inteligente móvil que facilitara el aprendizaje

de UX para diseñadores web. La investigación identificó como problema central la falta de consideración de la experiencia de usuario en el desarrollo de aplicaciones web, lo que genera productos menos amigables y útiles para los usuarios finales, afectando potencialmente los ingresos de los diseñadores. Además, se señaló la carencia de material de estudio accesible y asequible sobre UX, lo que dificulta la mejora de las habilidades en este ámbito.

La metodología combinó elementos de Ingeniería de Software Educativo (MeISE) y la metodología Mobile D. Se utilizaron la observación y la indagación de emociones humanas para la recolección de datos. Con dos grupos de ocho personas, los resultados mostraron que aquellos que usaron el tutor inteligente móvil tuvieron un rendimiento significativamente mejor que los que no lo usaron. El prototipo fue bien recibido, con una tasa de aceptación del 93.75%. El material de estudio proporcionado facilitó el aprendizaje de UX y fue impulsado mediante una aplicación móvil, lo que motivó su aceptación. El diseño del aprendizaje se basó en la práctica, con niveles de dificultad y evaluaciones para optimizar el rendimiento, reduciendo el contenido teórico para hacer el proceso más dinámico.

Este estudio aporta importantes lecciones para el diseño de la plataforma social de componentes open source propuesta. La investigación de Ramos subraya la importancia de priorizar el diseño de UX en el desarrollo de aplicaciones web para crear productos más satisfactorios y accesibles. Al igual que en su trabajo, una experiencia de usuario efectiva no solo mejora la satisfacción, sino que también incrementa la retención de usuarios y la eficacia de la plataforma. La metodología centrada en la práctica y la interactividad resalta la relevancia de un enfoque basado en niveles de dificultad y evaluaciones, principios que deben aplicarse para garantizar que la plataforma social de componentes open source sea intuitiva, accesible y atractiva para su comunidad de usuarios, favoreciendo su éxito a largo plazo.

Desde otra perspectiva, en la Universidad Politécnica de Madrid, el autor Javier Guardiola (2020) realizó una investigación titulada "Plataformas Tecnológicas de Inteligencia Colectiva como Herramientas de Innovación y Emprendimiento: Caracterización y Estado del Arte". El objetivo principal de su trabajo fue evaluar el potencial de diferentes plataformas tecnológicas para el desarrollo de ideas, especialmente aquellas que inspiran el emprendimiento al abordar desafíos reales. Estas plataformas facilitan la participación de los usuarios, aprovechando la inteligencia colectiva para proponer soluciones a problemas concretos en los ámbitos económico, medioambiental y social. Cada plataforma define sus propios espacios y modos de participación, con el objetivo de crear un ecosistema digital que genere un impacto positivo.

Las técnicas de recolección de datos incluyeron entrevistas y encuestas, con una muestra de cinco respuestas en la encuesta de South Summit y 103 respuestas en CoLab.upm. El estudio concluyó que las plataformas tecnológicas crean ecosistemas donde los usuarios participan motivados por incentivos comunes, adaptando la inteligencia colectiva a cada propósito. A través de un análisis comparativo, se identificaron cinco indicadores clave: Visibilidad, Accesibilidad, Fidelización, Colaboración e Impulso de la Inteligencia Colectiva, que influyen en la participación de los usuarios. Además, se destacó cómo estas plataformas fueron efectivas para movilizar a la sociedad en la resolución de desafíos, especialmente durante la pandemia de COVID-19. Se sugiere que la evolución y la participación creciente en estas plataformas auguran un futuro prometedor en términos de innovación y emprendimiento social.

Este estudio aporta una perspectiva valiosa para la plataforma social de componentes open source propuesta. La investigación de Guardiola destaca la importancia de fomentar la participación activa y la colaboración entre los usuarios como mecanismos clave para la innovación, algo que resulta crucial para la plataforma de componentes de software. Promover la

participación activa puede generar intercambios de ideas que conduzcan a soluciones innovadoras, mientras que la colaboración fomenta el trabajo en equipo y la co-creación de proyectos, mejorando tanto la calidad como la diversidad de los componentes desarrollados. En resumen, adoptar una mentalidad centrada en la inteligencia colectiva y la colaboración puede transformar la plataforma social en un entorno dinámico y fructífero para la innovación en el desarrollo de software, alineándose con los objetivos presentados en el estudio de Guardiola.

Bases teóricas

Las bases teóricas representan el análisis sistemático de conocimientos, teorías y marcos conceptuales que fundamentan la investigación. Sirven para orientar el estudio, justificar su relevancia, formular preguntas de investigación, diseñar la metodología, interpretar resultados y contribuir al conocimiento en el campo. Estas son las bases sólidas sobre las que se desarrolla la presente investigación, proporcionando una estructura coherente y fundamentada para el contenido del mismo.

Frameworks

Un framework está compuesto de herramientas, bibliotecas y componentes que facilitan una estructura para el perfeccionamiento de software. Estas estructuras están hechas para apoyar a los desarrolladores en la realización y diseño de aplicaciones de manera más rápida y eficaz al brindar una base en que trabajar. Juan de Assembler Institute (2022) se refiere a framework como: “un marco de trabajo que tiene como objetivo facilitar la solución de problemas que pueden surgir al programar”.

Los frameworks suelen abstraer la complejidad técnica de las labores básicas en el desarrollo web, de este modo los desarrolladores pueden concentrarse en la lógica de su aplicación.

Algunos promueven la reutilización de código al proporcionar componentes predefinidos y funcionalidades probadas que pueden ser usadas en infinidad de proyectos.

Tipos de frameworks

Frameworks de Frontend: Estos frameworks están diseñados generalmente apuntando hacia los clientes, principalmente para la cimentación de interfaces prácticas al usuario, e interactivas en aplicaciones web y móviles. Ejemplos populares son React.js, Angular y Vue.js.

Frameworks de Backend: Estos frameworks están diseñados apuntando hacia el servidor y proporcionan herramientas para la configuración de API, gestión de bases de datos, autenticación, entre otras funcionalidades. Ejemplos incluyen Django (Python), Express.js (Node.js), Ruby on Rails (Ruby), entre otros.

Frameworks Full-Stack: Estos frameworks abarcan tanto el desarrollo Frontend como Backend, proporcionando una solución completa para el desarrollo de aplicaciones web. Ejemplos son Laravel (PHP), Spring Boot (Java), entre otros.

Tendencia de crecimiento de la comunidad

La Tendencia de Crecimiento, forma parte del constante cambio en una comunidad específica, debido a la participación, colaboración, dentro de esta. En el diseño de la plataforma, esta comunidad está formada por individuos que colaboran, comparten conocimientos, contribuyen con código y brindan soporte dentro del ecosistema en la creación y perfeccionamiento de código abierto.

Sin embargo, existen factores que influyen en la mencionada tendencia de crecimiento de una comunidad:

Atractivo del proyecto: Los proyectos que abordan necesidades actuales y que tienen una buena reputación son más atractivos para los nuevos contribuyentes. Si un proyecto resuelve problemas relevantes o innovadores, tiende a captar más atención.

Facilidad de contribución: La oportunidad para que los miembros más recientes puedan integrarse y contribuir es crucial. Proyectos con una barrera de entrada baja, guías claras para contribuir y procesos de incorporación amigables logran atraer y retener más colaboradores.

Comunicación y colaboración: La efectividad de la comunicación a lo interno de la comunidad perturba significativamente su crecimiento. Las comunidades que fomentan un ambiente inclusivo, brindan recursos a los miembros nuevos miembros y suscitan la interacción entre sus participantes, suelen experimentar un crecimiento más rápido.

Calidad del código y documentación: Un código base bien mantenido y documentado, junto con el seguimiento de prácticas de alta calidad en el desarrollo, es esencial para atraer a nuevos contribuyentes. La calidad percibida del proyecto tiene un impacto directo en su capacidad de retener colaboradores.

Eventos y actividades comunitarias: En cuanto a esta materia, como conferencias, talleres y hackathons puede impulsar el incremento de la comunidad, brindando a los miembros la oportunidad que se conozcan, colaboren y compartan conocimientos, lo que refuerza la cohesión y el compromiso.

Consecuentemente, es fundamental para la comunidad, entender el avance y el éxito a largo plazo de la plataforma. Una comunidad activa y en crecimiento no solo contribuye con nuevas ideas y corrige errores, sino que también amplía la base de datos, que contiene los usuarios y aumenta la adopción del proyecto. Para asegurar un crecimiento sostenible, es importante implementar estrategias como la promoción del proyecto, la mejora constante de la

documentación, la participación en eventos relevantes y la simplificación de la contribución en el proceso. Estas acciones no solo atraerán a más miembros, sino que también garantizarán un entorno de colaboración más dinámico y próspero.

Requisitos de la plataforma

Son las especificaciones y características que deben cumplir para complacer las necesidades y expectativas de los usuarios, para garantizar el funcionamiento correcto. De igual manera, los requerimientos para desarrollar la plataforma de componentes open source se centran en la seguridad, el modelo del diseño y el lenguaje de programación.

Arquitectura del diseño

Consiste en planificar y crear un esquema que oriente el perfeccionamiento de la plataforma de forma eficiente y efectiva. Los requisitos específicos pueden incluir modularidad, la escalabilidad, el rendimiento y el mantenimiento del sistema. La arquitectura de esta plataforma social de componentes plantea utilizar la metodología Kanban para planificar y organizar el desarrollo, los wireframes para visualizar la interfaz de usuario, el design system para garantizar coherencia y consistencia visual, usando los diagramas de entidad-relación, necesarios en el modelaje la estructura en la base de datos.

Metodología Kanban

La metodología Kanban se implementa mediante tableros Kanban, como un enfoque visual de gestión de proyectos que facilita a los equipos la perfecta visual de sus flujos de trabajo y carga de trabajo. Estos tableros organizan el trabajo en columnas que representan diferentes etapas del proceso, como Trabajo pendiente, En progreso y Terminado, y las tareas individuales avanzan, por medio de las columnas hasta su finalización, representadas por tarjetas visuales en el tablero. (Martins, 2024). Funciona de la siguiente manera: primero, se divide el trabajo en tarjetas o notas,

cada una representando una tarea o elemento de trabajo. Estas tarjetas se mueven en el tablero mediante diferentes columnas, que representan estados o etapas del trabajo, como "pendiente", "en progreso" y "terminado". El equipo puede ver fácilmente qué tareas están en qué estado y quién está trabajando en cada una. Mediante que el trabajo progresa, las tarjetas se mueven de una columna a otra, lo que proporciona una visualización clara del flujo de trabajo y facilita la ubicación de áreas donde se necesita más atención. Esto permite una gestión más eficiente del trabajo y una colaboración efectiva entre los miembros del equipo.

Wireframes

Wireframe es una representación visual de la estructura y su funcionalidad de una página web o aplicación móvil en las primeras etapas del inicio de desarrollo. Estos wireframes tienen varios propósitos, como garantizar que el sitio cumpla con los diseños establecidos, centrarse en la facilidad de uso, prepararse para el crecimiento del proyecto y brindar comentarios e iteraciones. Además, los wireframes son valiosos en cuanto al desarrollo se refiere, al disponer a todos los participantes, mejorías en la comprensión del equipo de desarrollo y ahorrar tiempo y dinero al evitar cambios costosos a futuro en el proceso, (Lucidchart, s.f.).

Los wireframes son una herramienta esencial al momento de diseñar y en la construcción de la web y de aplicaciones, ya que proporcionan una representación visual clara y esquemática de su diseño y funcionalidad de un producto digital.

Design System

Es un conjunto de componentes de diseño, pautas y principios que garantizan la coherencia visual y funcional en toda la plataforma. La arquitectura del diseño debe incluir la definición y documentación que especifique los estilos, la tipografía, los colores, los iconos y otros elementos

de la interfaz de usuario. Este funciona como un "ADN" que guía a todo el grupo en una misma dirección. Donde la clave radica en su dinamismo, ya que evoluciona conforme crece el producto, lo que garantiza coherencia y eficiencia en cada fase del desarrollo (Parra, 2021).

Es crucial establecer un lenguaje consistente en la interfaz, incluyendo aspectos como el tono de comunicación y la UX. Esto implica mantener la coherencia en todos los aspectos del producto, desde el diseño visual hasta la terminología utilizada. La coherencia no solo afecta al equipo de diseño, incluso a otros departamentos, como el desarrollo, donde la consistencia es fundamental, en la implementación de estos componentes, que mejoran considerablemente la eficiencia y la reutilización. Por ello, mantener la coherencia en el diseño y desarrollo de productos es esencial para garantizar una experiencia de usuario unificada y efectiva.

Esquema entidad-relación

Es una representación visual de los datos y sus respectivas relaciones en una base de datos. Esta herramienta ayuda a los desarrolladores a comprender el diseño y por supuesto, la lógica de los datos que serán almacenados y manipulados por la aplicación. Lucidchart (s.f.) argumenta que

Los ER se relacionan con los diagramas de estructura de datos (DSD) y a menudo se combinan con los diagramas de flujo de datos (DFD) para mostrar el flujo de información en procesos o sistemas. Utiliza símbolos como rectángulos, diamantes y líneas para representar entidades, relaciones y atributos, reflejando la estructura gramatical con entidades como sustantivos y relaciones como verbos.

Lenguaje de programación

Es una herramienta que facilita la comunicación entre los programadores y las computadoras. A través de un conjunto de reglas y una sintaxis específica, permite que los

desarrolladores expresen instrucciones claras para que la computadora ejecute tareas concretas (Cimas Gustavo, 2024).

En este caso el lenguaje de programación propuesto para la plataforma social de componentes open source será primordialmente JavaScript y TypeScript, puesto que se usarán frameworks de JavaScript como: NextJS, Tailwind CSS, Express.js, entre otros.

JavaScript

Es un lenguaje de programación fundamental en el desarrollo web, alineado con el estándar ECMAScript. Siendo este un lenguaje de alto nivel, dinámico e interpretado, teniendo un papel clave junto con HTML y CSS para la creación de sitios web. Sus características incluyen la orientación a objetos, el uso de prototipos, un enfoque imperativo y una tipificación débil, lo que le permite ser flexible y ampliamente compatible con todos los navegadores modernos sin necesidad de complementos adicionales. (Armetrics, 2022).

JavaScript ha transformado la experiencia web al permitir que las páginas sean más dinámicas y receptivas a la interacción del usuario. JavaScript ha evolucionado para ser empleado tanto en el frontend como en el backend. A medida que el lenguaje ha avanzado, los desarrolladores han creado bibliotecas, marcos y prácticas de programación para exprimir al máximo su utilidad.

TypeScript

Es una extensión de JavaScript que introduce tipado en la sintaxis, mejorando la robustez y escalabilidad del código (AWS, s.f.). Typescript amplía las capacidades de ECMAScript al agregar características adicionales como tipado fuerte, anotaciones y módulos. Al ser un superconjunto de JavaScript, todo el código Javascript es válido en TypeScript, pero no al revés. Para que el código TypeScript sea legible por los navegadores web, es necesario transpilarlo a

JavaScript. Este proceso de transpiración permite que el código sea compatible con diferentes versiones de ECMAScript, lo que asegura el funcionamiento en diversos entornos de navegador.

NextJS

Es un framework basado en React, diseñado para desarrollar sitios y aplicaciones web altamente optimizados. Se destaca por su capacidad para generar aplicaciones estáticas que mejoran la experiencia del usuario y con muy buen SEO. Además, es reconocido por ofrecer una excelente experiencia para los desarrolladores, proporcionando todas las herramientas necesarias para crear aplicaciones listas para producción, con características integradas que simplifican el proceso de desarrollo. (Kinsta, 2022)

Algunas de sus principales características son:

Renderizado híbrido: ofreciendo renderizado del lado del cliente (CSR) y del lado del servidor (SSR), permitiendo que las páginas se rendericen en el servidor antes de ser enviadas al navegador, lo que mejora el rendimiento y la SEO.

Enrutamiento Dinámico: Proporciona un enrutamiento dinámico que permite la generación de rutas de manera flexible y dinámica, lo que simplifica la creación de aplicaciones con diversas rutas.

Preprocesamiento de Datos: Permite el procesamiento previo de datos antes de renderizar las páginas, lo cual agiliza enormemente la carga de información desde fuentes externas, como APIs o bases de datos antes de que los datos se envíen al cliente.

Optimización Automática: Next.js incluye características de optimización automática, como el fragmentado de código, la carga anticipada de rutas, la optimización de imágenes y el renderizado en paralelo, que mejoran la experiencia del usuario.

Soporte para TypeScript: Ofrece un sólido soporte para TypeScript, lo que permite desarrollar aplicaciones con tipos estáticos y obtener ventajas durante el desarrollo.

API Routes: Permite la creación de API routes para manejar solicitudes HTTP, lo que facilita el desarrollo de backend junto con el código del frontend en la misma aplicación.

Escalabilidad: Next.js es altamente escalable y se puede utilizar para construir aplicaciones desde pequeños proyectos hasta grandes plataformas empresariales.

TailwindCSS

Huet (2022) declara “en pocas palabras, es un framework CSS que da prioridad a la utilidad sobre el propio estilo”. Este se diferencia de otros frameworks al no proporcionar una serie de componentes predefinidos, sino clases de ayuda para estructurar y estilar rápidamente diseños personalizados. A diferencia de otros frameworks, Tailwind ofrece una flexibilidad gracias a su diversidad de clases CSS con enfoques específicos, lo que facilita el estilo de elementos con solo agregar clases.

Express.js

Es un framework web ligero y flexible para Node.js, utilizado para el desarrollo de aplicaciones web y APIs. Es uno de los frameworks más populares dentro del ecosistema de Node.js. debido a su simplicidad y versatilidad, es mantenido por la Fundación Node.js. Este proporciona un sistema de enrutamiento flexible y características básicas que permiten a los desarrolladores extender su funcionalidad según las necesidades de la aplicación (Kinsta, 2022).

Bases de datos

Es una colección organizada de datos almacenada electrónicamente, que puede contener diversos tipos de datos. Para la gestión de estos datos, se utiliza un software llamado sistema de

administración de bases de datos, el cual permite almacenar, recuperar y modificar la información (AWS, s.f.).

Tipos:

Relacionales: Utilizan un modelo relacional para organizar los datos, estos se organizan en tablas con filas y columnas, y se establecen relaciones entre las tablas a través de claves primarias y foráneas. Ejemplos incluyen MySQL, PostgreSQL, Oracle, entre otros.

Bases de Datos NoSQL: Utilizan modelos de datos no relacionales para almacenar y organizar los datos. Estos incluyen bases de datos de documentos, de columnas, de grafos y clave-valor. Ejemplos incluyen MongoDB, Redis y Neo4j.

Cronograma de Planificación

Tabla 1. Cronograma de planificación

Cronograma de Planificación								
Actividades	Fechas							
Desarrollo de la primera fase del informe								
Diagnóstico Situacional	13/02/24							
Problemas de la investigación y formulación de objetivos		15/02/24						
Justificación			Del 18 al 23 de febrero de 2024					
Delimitaciones				24/02/24				
Estudios Previos					Del 26 de febrero al 11 de marzo			
Operacionalización de la variable						12/03/24		
Cronograma de planificación							13/03/24	
Bases Teóricas								Del 13 al 15 de marzo de 2024

Operacionalización de las Variables

Tabla 2. Cuadro de operacionalización de las variables

<i>Cuadro de operacionalización de la variable</i>				
Objetivo General: Desarrollar una plataforma online comunitaria para compartir componentes de programación open source.				
Objetivos Específicos	Variables	Dimensión	Indicadores	Instrumento
Identificar los medios por los cuales la comunidad uvemista busca componentes de programación open source.	Capital Social	Conexiones Sociales	- Medios para encontrar componentes open source - Frameworks - Tendencia de crecimiento de la comunidad.	Cuestionario
Definir los requerimientos técnicos y de seguridad para la implementación de componentes compartidos por la comunidad.	Plataforma social de componentes	Requerimientos de la plataforma	- Casos de uso - Necesidades del usuario	Entrevista
Desarrollar una plataforma social que permita compartir componentes open source.		Propuesta	- Procedimiento del diseño de la plataforma. - Desarrollo de la plataforma	
Implementar la plataforma social desarrollada para facilitar el intercambio de componentes open source.				

Nota: Muestra de la operacionalización de la variable
Fuente: Elaboración propia (2024)

II. FASE DE IMPLEMENTACIÓN

En este estudio, se abordan todos los elementos esenciales del marco metodológico, los cuales engloban un conjunto de acciones destinadas a describir y analizar a fondo el problema en cuestión. Estas acciones se llevan a cabo mediante procedimientos específicos que incluyen técnicas de observación, recopilación y análisis de datos. A su vez, se define el proceso de la investigación y los pasos requeridos para llevarla a cabo, identificando el método que conduce a la solución del problema planteado.

Al definir el tipo de investigación, el diseño y la elección de la población y la muestra, se consideran los objetivos del estudio. Además, se establecen las técnicas e instrumentos para la recolección de datos, evaluando su validez y confiabilidad, y se detalla el procesamiento y análisis de los datos de acuerdo con el procedimiento de investigación.

Diseño de la investigación

Tipo de investigación

El tipo de investigación mixta, es un enfoque que combina procesos sistemáticos, empíricos y críticos, integrando tanto datos cuantitativos como cualitativos. Esta metodología no solo implica la recolección y análisis de ambos tipos de datos, sino también su integración para tomar decisiones conjuntas que conduzcan a una meta-interferencia, es decir, una interpretación general basada en toda la información recopilada. Con el objetivo de lograr una comprensión más profunda y completa del fenómeno en estudio (Hernández Sampieri et al., 2018).

Cuando se integran datos cuantitativos y cualitativos, los investigadores puede obtener una comprensión más completa del fenómeno estudiado, de esta forma, ofrece una visión más profunda que trasciende las métricas numéricas y se adentra a las experiencias y motivaciones de los usuarios. La precisión de obtener los datos y la interpretación correcta de los comportamientos y

dinámicas comunitarias son cruciales, aumentando así, la confianza en la validez y fiabilidad de los hallazgos. Dado que la comunidad y la tecnología están en constante evolución, la flexibilidad es crucial para captar y entender los cambios en tiempo real. De este modo, la investigación mixta resulta un enfoque acertado para fortalecer la plataforma y transmitir de manera clara su propósito.

El enfoque de investigación mixta resulta valioso en este estudio ya que permite una comprensión completa, validación de resultados, exploración de complejidades y adaptación a cambios en la comunidad y la tecnología. El método mixto enfrenta la complejidad del entorno investigativo, permitiendo a los investigadores obtener una comprensión más completa y precisa de los fenómenos analizados en esta investigación.

Nivel de la investigación

El enfoque exploratorio es adecuado cuando se busca investigar un fenómeno o problemas del cual existe poca información. Siendo un tipo de estudio que se realiza sobre temas u objetivos poco conocidos o escasamente investigados. Su propósito es ofrecer una primera aproximación al objeto de estudio, proporcionando una comprensión inicial. Los resultados obtenidos no profundizan en el tema, sino que brindan un conocimiento superficial que sirve como base para investigaciones futuras más detalladas (Fidias, 2006).

En esta investigación, se examinan las maneras en que la comunidad uvemista busca componentes de programación open source, además de los requerimientos técnicos y de seguridad necesarios para implementar componentes compartidos por la comunidad. El enfoque exploratorio en esta investigación ayuda a identificar necesidades y problemas, explorar distintas metodologías y evaluar la usabilidad, la escalabilidad, la seguridad, y la flexibilidad de diversas soluciones, lo cual, facilita la adaptación de las plataformas sociales de código abierto a las necesidades particulares de dicha comunidad, de esta manera promover la participación activa, además ayuda

a generar hipótesis y preguntas, lo que ayuda a dirigir y enfocar el trabajo futuro asegurando que la investigación resulta relevante para la comunidad uvemista.

Diseño de la investigación

Este se apoya en estrategias de procedimientos donde el diseño le facilita al investigador las acciones necesarias para lograr los objetivos planteados. Según Arias (2006), es “La estrategia general que adopta el investigador para responder al problema planteado. En atención al diseño, la investigación se clasifica en documental, de campo y experimental” (p.27).

El diseño que se implementará en esta investigación se basará en un estudio de campo. El cual se lleva a cabo directamente en el lugar donde se presenta el problema. Este se centra en obtener información de los sujetos involucrados o del entorno específico, lo que permite obtener datos más reales y contextuales. Mediante esta aproximación, se busca una comprensión más directa y precisa del fenómeno en estudio. (Arias, 2006). Esta investigación será de campo ya que permitirá abordar resultados a través de recolección de datos mediante encuestas y cuestionarios.

Población y muestra

Población

Según González y Salazar (2008), se considera “Un conjunto de individuos (todos los habitantes de un país). Además, se estudia la estructura interna, la dinámica y su distribución sobre el espacio de las poblaciones humanas y las leyes que rigen estos fenómenos.” (p.11). Este se refiere al conjunto de personas o elementos seleccionados para investigar y comprender sus características. Este grupo es clave para que los resultados sean aplicables y se utiliza como inicio para analizar los datos recopilados, los cuales pueden provenir tanto de individuos como de objetos. Además, permite realizar estimaciones al recolectar información. La población puede

clasificarse como finita o infinita: es finita si se puede determinar su tamaño, mientras que es infinita cuando es tan grande que no es posible cuantificarla.

Muestra

Es una porción de la población total, compuesta por individuos u objetos seleccionados de manera científica, por lo tanto, es representativa de la población en general. Su función es proporcionar información sobre el grupo completo, permitiendo hacer generalizaciones sin necesidad de estudiar a todos los elementos de la población (González y Salazar, 2008). La muestra es una parte de la población y se considera representativa cuando refleja de manera precisa las diversas características de esta. Para que los resultados obtenidos sean generalizables, la muestra debe ser lo más precisa posible y tener el menor sesgo posible.

Por lo tanto, la población de este estudio está compuesta por los estudiantes de la carrera Ingeniería de Computación de la UVM, quienes son la comunidad uvemista de desarrolladores posibles para la plataforma.

Diseño de instrumento de recolección de datos

Las técnicas de recolección de datos son fundamentales para obtener información relevante que permita responder preguntas de investigación, validar hipótesis y tomar decisiones informadas., estas incluyen formas diversas de obtener información, cada técnica se selecciona según los objetivos específicos de la investigación, lo que subraya la importancia de elegir el método adecuado (Arias, 2006).

Se ha optado por la encuesta como técnica de recolección, la cual puede generar una amplia variedad de estadísticas útiles, facilitando el análisis y la interpretación de los datos. Esto permite identificar patrones y relaciones que no serían evidentes de otro modo, aportando una visión más detallada y precisa sobre el fenómeno en estudio (Marytere, 2023).

Instrumentos

Son herramientas clave para recolectar y almacenar información relevante en una investigación. Es responsabilidad de los investigadores diseñar instrumentos adecuados a los indicadores del estudio, asegurando que recojan los datos necesarios de manera eficiente y precisa. Como señala Arias, existen diversos instrumentos que permiten obtener una visión más completa del objeto de estudio.

Un cuestionario es una técnica de recolección de datos cuantitativos que consiste en una serie de preguntas organizadas en un orden específico (Ortega, 2019). Es una herramienta usada para recopilar información, generalmente numérica, de las personas. En la presente investigación se llevó a cabo un cuestionario que arrojó 18 respuestas.

Se puede visualizar el cuestionario dirigido a los estudiantes de la carrera de Ingeniería de Computación de la UVM en el anexo 1.

También se realizaron entrevistas a varios estudiantes seleccionados, la entrevista es una técnica que se basa en una conversación cara a cara entre el entrevistador y el entrevistado para obtener información relevante sobre un tema específico (Arias, 2006). En este caso, las entrevistas permitieron explorar en profundidad las percepciones de los participantes sobre la plataforma, enfocándose en sus expectativas, funcionalidades deseadas e inquietudes de seguridad. Estas opiniones cualitativas complementan los datos cuantitativos del cuestionario, proporcionando una visión más completa de las necesidades de los usuarios para un diseño efectivo de la plataforma.

Validación y Confiabilidad de Instrumentos

Ambos son aspectos muy importantes en la evaluación de cualquier instrumento, ya sea cuestionario, una prueba psicométrica o cualquier herramienta utilizada para la recolección de datos.

Validez

Esta se refiere a la precisión con la que un instrumento mide la variable que busca evaluar (Hernández Sampieri et al., 2019). Lo cual es muy importante para que los instrumentos sean precisos y relevantes para la variable que se estudia, si un instrumento no es válido los resultados que produce pueden ser engañosos o irrelevantes lo que puede conducir a conclusiones erróneas o acciones inapropiadas. En este caso el instrumento fue revisado por un panel de dos (02) expertos en metodologías y proyectos tecnológicos, quienes realizaron una evaluación individual y propusieron correcciones al instrumento. Posteriormente, firmaron el modelo final, aprobando su aplicación y el análisis de resultados (Anexo 3).

Confiabilidad

Se refiere a su capacidad para producir los mismos resultados cuando se aplica repetidamente al mismo sujeto u objeto (Hernández Sampieri et al., 2019). Es decir, mide la consistencia de un instrumento en circunstancias similares. Existen diversas técnicas para evaluar la confiabilidad. El coeficiente alfa de Cronbach es uno de los métodos más comunes para medir la consistencia interna, este se calcula de la siguiente manera:

$$r_{rr} = \frac{K}{K - 1} \left[1 - \frac{\sum Si^2}{St^2} \right]$$

Donde k es el número de ítems

r_{rr} = Coeficiente de confiabilidad

K = Número de ítems

1 = Constante

$\sum Si^2$ = Sumatoria de la varianza de los ítems

St^2 = Varianza de los puntajes totales.

Luego de recopilar toda la información del cuestionario, se llevaron a cabo los cálculos de confiabilidad correspondientes, obteniendo un valor de 0.67, esto indica que el instrumento tiene una alta confiabilidad (Anexo 4).

Análisis de datos

El procedimiento y el análisis de datos son componentes esenciales en cualquier investigación, proporcionando la estructura y la comprensión necesarias para extraer conclusiones significativas. Se clasificaron los datos recopilados mediante un cuestionario por indicador y se analizaron utilizando estadística descriptiva. Los resultados se presentaron en tablas de frecuencia y gráficos porcentuales para evaluar la fiabilidad de la información obtenida.

La estadística descriptiva se enfoca en presentar datos a través de tablas y gráficos para resumir su comportamiento, sin hacer inferencias, ya que los datos provienen solo de una parte de la población (Hernández, 2016).

El proceso de la investigación comenzó con la identificación del problema de investigación y la formulación de los objetivos del estudio y se justificó su relevancia desde un contexto teórico, metodológico, práctico y social. También se delimitó el alcance de la investigación.

Seguidamente, se realizó una revisión bibliográfica detallada para fundamentar el marco teórico de la investigación. Se examinaron antecedentes, teorías y modelos pertinentes, además de estructurar la operacionalización de la variable de estudio.

Posteriormente, se diseñaron los instrumentos de recolección de datos, que incluyeron cuestionarios y entrevistas.

Una vez recopilada la información, se realizó un análisis detallado para extraer patrones y tendencias significativas. Para las preguntas cerradas, se ejecutó un análisis cuantitativo mediante estadísticas descriptivas para resumir las respuestas y calcular medidas de tendencia central y

dispersión. Esto facilitó una comprensión general de las respuestas de los participantes en cuanto a su frecuencia y distribución.

Se efectuó un análisis cualitativo para las preguntas abiertas empleando técnicas de categorización, durante el cual se examinaron detenidamente las respuestas para identificar temas comunes, patrones emergentes y opiniones recurrentes. Estos temas se agruparon en categorías por pregunta, para simplificar su análisis y presentación.

Posteriormente, se compararon y contrastaron los hallazgos de ambos análisis para obtener una comprensión holística de las percepciones y necesidades de los participantes con relación a la plataforma. Se destacaron las principales conclusiones y se proporcionaron interpretaciones basadas en los datos recopilados.

Después de ordenar, clasificar y tabular los datos obtenidos, se hizo un resumen de las observaciones, obteniendo las respuestas necesarias tanto del cuestionario como de las entrevistas. Posteriormente, se interpretaron los resultados, extrayendo conclusiones y recomendaciones relevantes.

Finalmente, se analizaron los resultados de los hallazgos en relación con el diseño y desarrollo de la plataforma, así como su relevancia para investigaciones futuras en el ámbito de la ingeniería de computación.

Durante este procedimiento, se revisaron cuidadosamente los datos obtenidos, enfocándose en los aspectos clave del estudio para extraer conclusiones que respondieran a los objetivos planteados. Se utilizaron técnicas estadísticas y herramientas visuales para explorar patrones, tendencias y relaciones entre las variables analizadas, lo que facilitó una comprensión más detallada de los hallazgos. Asimismo, se puso especial énfasis en la coherencia y consistencia de los resultados, asegurando su fiabilidad y validez dentro del contexto del estudio.

Análisis Descriptivo

El enfoque se aplicó para exponer y sintetizar la información de manera clara y concisa, permitiendo una comprensión inicial de los patrones y tendencias observadas en los datos. El análisis fue fundamental para identificar las principales características de la muestra y para analizar la distribución y variabilidad de las variables examinadas. Se eligió esta técnica estadística por su habilidad para ofrecer una visión general de los datos sin realizar inferencias sobre la población en su totalidad, lo que resultó apropiado dada la naturaleza exploratoria de la investigación. El análisis descriptivo permitió una interpretación inicial de los datos y estableció las bases para un análisis más detallado en etapas posteriores del estudio.

Análisis del cuestionario

Se enfoca en comprender cómo interactúan los usuarios con los componentes open source y la plataforma social. Se examinan las respuestas relacionadas con la búsqueda de componentes, el uso de frameworks, y la participación en la comunidad. Estos datos proporcionan información clave sobre las necesidades y expectativas de los usuarios, fundamentales para el desarrollo de la plataforma.

Dimensión: Conexiones Sociales

Indicador: Medios para encontrar componentes open source

Tabla 3. Medios para encontrar componentes Open Source

Ítem	Categoría	Fa	Fr (%)
1. ¿Qué tan seguido busca componentes open source para agilizar el desarrollo de su proyecto?	Siempre	3	16.7%
	Casi siempre	8	44.4%
	Algunas veces	6	33.3%
	Casi nunca	1	5.6%
	Nunca	-	-
	Total	18	100%

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Qué tan seguido busca componentes open source para agilizar el desarrollo de su proyecto?

18 respuestas

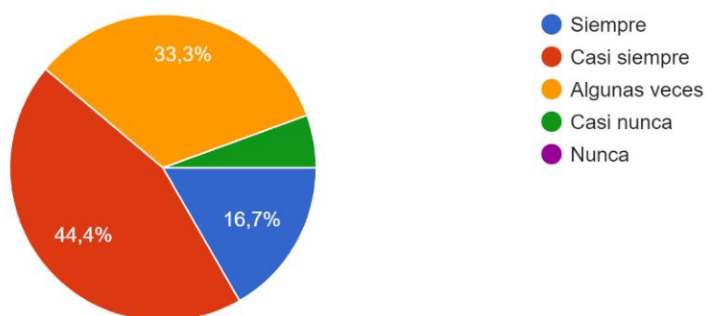


Ilustración 1. Medios para encontrar componentes Open Source

En el indicador de los Medios para encontrar componentes Open Source se observa que un 44,4% y 33,3% de las personas encuestadas, expresan que Casi siempre o Algunas veces buscan componentes Open Source para agilizar el desarrollo de sus proyectos.

Tabla 4. Medios para encontrar componentes Open Source

Ítem	Categoría	Fa	Fr (%)
2. ¿Con qué regularidad consulta plataformas sociales de componentes open source para intercambiar información?	Siempre	-	-
	Casi siempre	5	27.8%
	Algunas veces	10	55.6%
	Casi nunca	3	16.7%
	Nunca	-	-
	Total	18	100%

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Con qué regularidad consulta plataformas sociales de componentes open source para intercambiar información?

18 respuestas

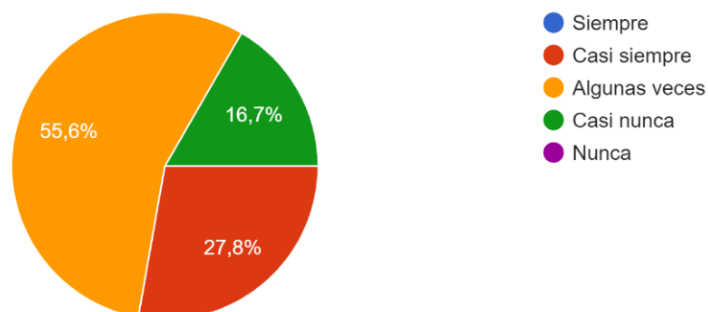


Ilustración 2. Medios para encontrar componentes Open Source

En el indicador de los Medios para encontrar componentes Open Source se observa que un 55.6% de las personas encuestadas, expresan que Algunas veces consultan plataformas sociales de componentes para intercambiar información.

Tabla 5. Medios para encontrar componentes Open Source

Ítem	Categoría	Fa	Fr (%)
3. ¿Qué tan seguido tu entorno social te comparte información sobre componentes open source?	Siempre	2	11.1%
	Casi siempre	4	22.2%
	Algunas veces	5	27.8%
	Casi nunca	6	33.3%
	Nunca	1	5,6%
	Total		18

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Qué tan seguido tu entorno social te comparte información sobre componentes open source?

18 respuestas

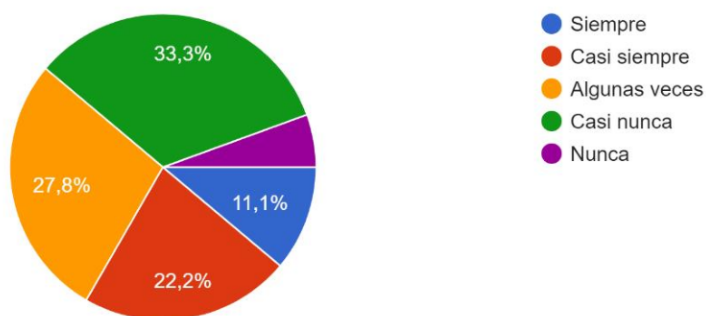


Ilustración 3. Medios para encontrar componentes Open Source

En el indicador de los Medios para encontrar componentes Open Source se observa que un 33.3% y 27.8% de las personas encuestadas, expresan que Casi nunca o algunas veces su entorno social les comparte información sobre componentes Open Source.

Indicador: Frameworks

Tabla 6. Frameworks

Ítem	Categoría	Fa	Fr (%)
4. ¿Sueles encontrar con facilidad componentes open source del framework que necesitas?	Siempre	2	11.1%
	Casi siempre	4	22.2%
	Algunas veces	11	61.1%
	Casi nunca	1	5.6%
	Nunca	-	-
	Total		18

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Sueles encontrar con facilidad componentes open source del framework que necesitas?
18 respuestas

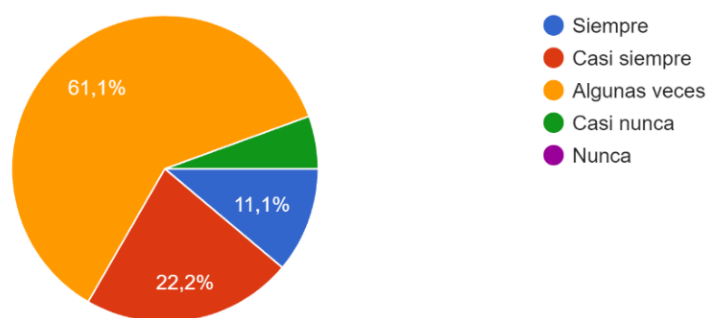


Ilustración 4. Frameworks

En el indicador de los Frameworks se observa que un 61.1% de las personas encuestadas, expresan que Algunas veces consiguen con facilidad componentes open source del framework que necesitan.

Tabla 7. Frameworks

Ítem	Categoría	Fa	Fr (%)
5. ¿Con qué frecuencia utilizas frameworks populares (React, Angular, Vue, etc) en tus proyectos de desarrollo?	Siempre	9	50%
	Casi siempre	4	22.2%
	Algunas veces	2	11.1%
	Casi nunca	2	11.1%
	Nunca	1	5.6%
	Total	18	100%

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Con qué frecuencia utilizas frameworks populares (React, Angular, Vue, etc) en tus proyectos de desarrollo?

18 respuestas

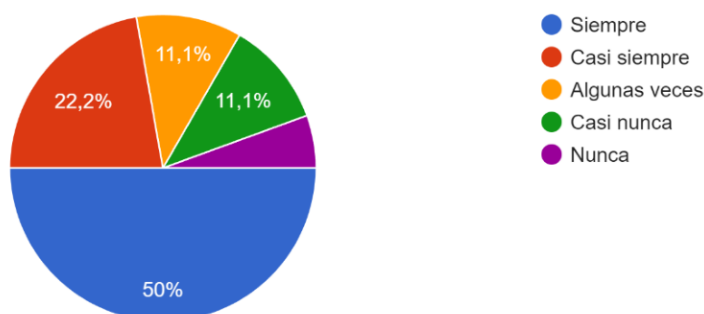


Ilustración 5. Frameworks

En el indicador de los Frameworks se observa que un 50% de las personas encuestadas, expresan que Siempre utilizan frameworks populares (React, Angular, Vue, etc) en sus proyectos de desarrollo.

Indicador: Tendencia de crecimiento de la comunidad

Tabla 8. Tendencia de crecimiento de la comunidad

Ítem	Categoría	Fa	Fr (%)
6. ¿Con qué frecuencia has utilizado una plataforma social de componentes open source en los últimos seis meses?	Siempre	1	5.6%
	Casi siempre	4	22.2%
	Algunas veces	11	61.1%
	Casi nunca	2	11.1%
	Nunca	-	-
	Total	18	100%

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Con qué frecuencia has utilizado una plataforma social de componentes open source en los últimos seis meses?

18 respuestas

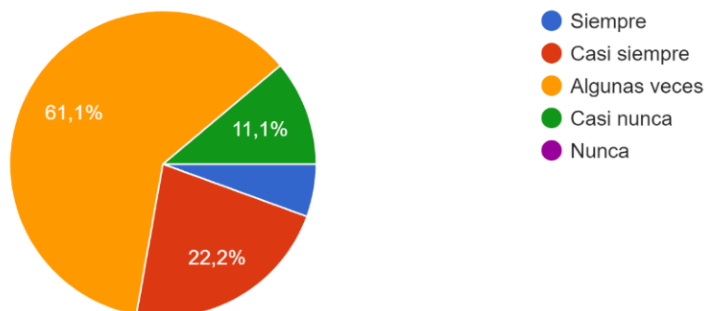


Ilustración 6. Tendencia de crecimiento de la comunidad

En el indicador de las Tendencias de crecimiento de la comunidad se observa que un 61.1% de las personas encuestadas, expresan que Algunas veces han utilizado una plataforma social de componentes open source en los últimos seis meses.

Tabla 9. Tendencia de crecimiento de la comunidad

Ítem	Categoría	Fa	Fr (%)
7. ¿Qué tan a menudo utilizarías la plataforma social de componentes open source en el futuro?	Siempre	4	22.2%
	Casi siempre	8	44.4%
	Algunas veces	6	33.3%
	Casi nunca	-	-
	Nunca	-	-
	Total		18

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Qué tan a menudo utilizarías la plataforma social de componentes open source en el futuro?

18 respuestas

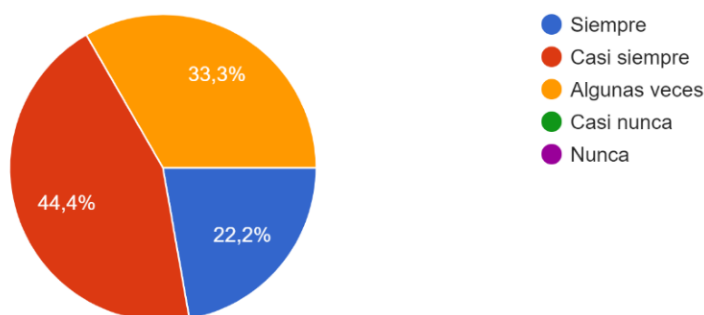


Ilustración 7. Tendencia de crecimiento de la comunidad

En el indicador de las Tendencias de crecimiento de la comunidad se observa que un 44.4% y 33.3% de las personas encuestadas, expresan que Casi siempre o algunas veces utilizarían la plataforma social de componentes open source en el futuro.

Tabla 10. Tendencia de crecimiento de la comunidad

Ítem	Categoría	Fa	Fr (%)
8. ¿Qué tan seguido compartirías componentes de esta plataforma con tus amigos y colegas?	Siempre	3	16.7%
	Casi siempre	8	44.4%
	Algunas veces	7	38.9%
	Casi nunca	-	-
	Nunca	-	-
	Total		18

Nota: Opinión de los estudiantes de la carrera de Ingeniería de Computación

Fuente: Universidad Valle del Momboy (2024)

¿Qué tan seguido compartirías componentes de esta plataforma con tus amigos y colegas?

18 respuestas

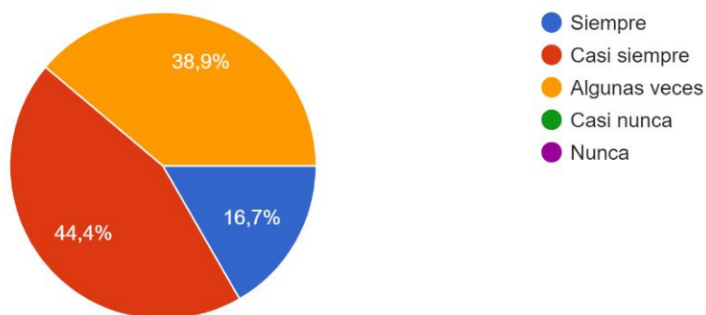


Ilustración 8. Tendencia de crecimiento de la comunidad

En el indicador de las Tendencias de crecimiento de la comunidad se observa que un 44.4% y 38.9% de las personas encuestadas, expresan que Casi siempre o Algunas veces compartirían componentes de esta plataforma con sus amigos y colegas.

Análisis de las Entrevistas

El propósito de este análisis es explorar en profundidad las percepciones y opiniones de los participantes sobre la plataforma social de componentes open source, centrándose en aspectos como las características deseadas, las necesidades específicas de los usuarios y las sugerencias para mejorar la plataforma. Este análisis cualitativo permite comprender mejor las experiencias y expectativas de los usuarios, identificar patrones emergentes y obtener información detallada que complementa los datos cuantitativos recopilados a través del cuestionario.

Dimensión: Requerimientos de la plataforma

Tabla 11. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Características relevantes	Las características relevantes, como la usabilidad, seguridad, y escalabilidad, son fundamentales para el desarrollo de la plataforma social de componentes open source. La usabilidad permite una navegación eficiente, fomentando la participación de la comunidad. La seguridad protege los datos y genera confianza entre los usuarios. La escalabilidad garantiza que la plataforma pueda crecer y adaptarse a nuevas funcionalidades y usuarios sin comprometer el rendimiento. Estas características aseguran un entorno colaborativo y sostenible, alineado con las necesidades de los desarrolladores.	Las respuestas reflejan que los desarrolladores esperan una plataforma social de componentes open source con una amplia variedad de componentes reutilizables, bien documentados y disponibles bajo licencias abiertas. Valoran la documentación clara, compatibilidad con múltiples lenguajes, y una comunidad activa que fomente la colaboración. Además, requieren herramientas avanzadas de búsqueda y categorización, una interfaz fácil de usar, y sistemas de retroalimentación como calificaciones y comentarios. Finalmente, esperan soporte adicional mediante foros, casos de uso y manuales que faciliten la aplicación práctica de los componentes.

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 12. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Funcionalidades esperadas	<p>Las funcionalidades esperadas definirán la experiencia del usuario y la efectividad del intercambio de componentes. Se espera que estas características promuevan la colaboración y el aprendizaje entre los miembros de la comunidad, facilitando la creación y el acceso a recursos de programación. Además, una adecuada implementación de estas funcionalidades garantizará una interacción fluida y un entorno de trabajo eficiente, lo que contribuirá al crecimiento y sostenibilidad de la plataforma a largo plazo.</p>	<p>Las respuestas muestran que los desarrolladores esperan un motor de búsqueda avanzado, con recomendaciones personalizadas y categorización clara. Valoran un sistema de etiquetas y valoraciones, ejemplos prácticos, guías detalladas, y un manual de usuario. También destacan la importancia de un modo oscuro y claro, la posibilidad de contactar al autor del componente, y una interfaz intuitiva para mejorar la experiencia en la plataforma.</p>

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 13. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Capacidad de personalización	<p>La personalización permite a los usuarios ajustar su experiencia según sus necesidades y gustos. Esta flexibilidad fomenta una mayor satisfacción y compromiso, facilitando el uso efectivo de la plataforma. Además, al permitir la personalización, fomenta un entorno que se adapta a los estilos de trabajo y necesidades de cada usuario, lo que aumenta la aceptación y el éxito de la plataforma. en la comunidad.</p>	<p>Las respuestas indican que la capacidad de personalización varía en importancia entre los usuarios. Algunos consideran fundamental personalizar su perfil, incluyendo temas visuales y esquemas de color, así como configurar notificaciones y agregar enlaces a GitHub. Sin embargo, otros ven la personalización como secundaria y no esencial, destacando que un sistema básico de likes o favoritos también sería suficiente. En general, la personalización es valorada, pero no es prioritaria para todos.</p>

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 14. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Sistema de Notificaciones	<p>El sistema de notificaciones es fundamental para el desarrollo de la plataforma, ya que mejora la comunicación y la interacción entre los usuarios. Al mantener a los miembros informados sobre actualizaciones, colaboraciones y eventos relevantes, se fomenta un entorno activo y participativo. Además, un sistema de notificaciones eficaz puede ayudar a aumentar la retención de usuarios y facilitar el intercambio de conocimientos, lo que resulta en una comunidad más cohesiva y comprometida en torno a la plataforma.</p>	<p>Las respuestas indican que los usuarios desean un sistema de notificaciones personalizable que les permita elegir qué alertas recibir. Prefieren notificaciones sobre actualizaciones en proyectos seguidos, y alertas por correo y push. También valoran la actividad de usuarios que siguen, publicaciones de nuevos componentes, así como recomendaciones personalizadas. En general, buscan un sistema que ofrezca flexibilidad y relevancia en las notificaciones</p>

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 15. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Soporte o recursos adicionales	<p>El soporte o recursos adicionales son esenciales, ya que proporcionan a los usuarios las herramientas y la asistencia necesarias para maximizar su experiencia. Además, un buen sistema de soporte contribuye a resolver dudas y problemas rápidamente, lo que mejora la satisfacción del usuario y fomenta una comunidad activa y colaborativa. Estos recursos son clave para garantizar el éxito y la sostenibilidad de la plataforma a largo plazo.</p>	<p>Las respuestas indican que los desarrolladores esperan un soporte integral que incluya documentación exhaustiva para cada componente. Buscan tutoriales, guías de uso y documentación detallada de la API. Además, valoran los foros de discusión y la posibilidad de contactar a los autores, así como acceso a proyectos de referencia y casos de uso. En general, desean recursos accesibles que faciliten su proceso de desarrollo</p>

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 16. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Aspectos de seguridad	Implementar medidas de seguridad robustas, es crucial para prevenir accesos no autorizados y ataques cibernéticos. Además, una sólida infraestructura de seguridad fomenta la confianza de los usuarios, incentivando su participación y colaboración en la plataforma. Al priorizar la seguridad, se asegura un entorno seguro y confiable que respalda el crecimiento y la sostenibilidad del proyecto.	Las respuestas indican que los desarrolladores valoran varios aspectos de seguridad al utilizar o contribuir a una plataforma social de componentes open source. Destacan la importancia de auditorías de seguridad regulares, y la evaluación de dependencias de terceros para protegerse contra el código malicioso. También enfatizan la importancia de proteger la privacidad de los datos personales y asegurar la autenticidad de los usuarios. En general, buscan una plataforma que asegure la seguridad y privacidad en sus interacciones y contribuciones.

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 17. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Moderación y control de calidad	La moderación y el control de calidad son esenciales para mantener el orden y la confiabilidad de los componentes compartidos en la plataforma. Un sistema de moderación asegura que los contenidos subidos cumplan con los estándares establecidos, asegurándose de que no se propague código malicioso o contenido inapropiado. Además, un control de calidad efectivo fomenta la confianza entre los usuarios, garantizando que los componentes disponibles sean útiles, seguros y estén bien documentados, lo que contribuye al éxito y sostenibilidad del proyecto.	Las respuestas destacan la necesidad de un proceso de revisión inicial de los componentes, revisiones de código, y un sistema de valoración y comentarios. También consideran importantes normas claras, y sistemas para reportar contenido inapropiado. Muchos proponen que los usuarios asuman el rol de moderadores para garantizar la calidad. En general, buscan un entorno que priorice la calidad y relevancia de los recursos compartidos.

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 18. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Mejora de la colaboración entre desarrolladores	Esta es fundamental para el éxito del proyecto, ya que fomenta un entorno de trabajo en equipo que potencia la innovación y la creatividad. Facilitar la interacción y el intercambio de conocimientos entre los miembros de la comunidad favorece la resolución eficaz de problemas y al desarrollo de soluciones más robustas. Al implementar herramientas y funcionalidades que favorezcan la colaboración, se logra maximizar el potencial colectivo de los desarrolladores, lo que resulta en un avance significativo en la calidad y efectividad de los componentes de programación open source compartidos en la plataforma.	Las respuestas sugieren que podría mejorar la colaboración y el intercambio de conocimientos entre desarrolladores al facilitar el intercambio de componentes reutilizables y permitir la discusión y colaboración en mejoras. Proponen un sistema de comentarios, además de eventos virtuales como hackathons. La documentación de calidad, un sistema de valoración, y la posibilidad de contactar a los autores son fundamentales para fomentar la retroalimentación y el aprendizaje en un ambiente amigable.

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Tabla 19. Análisis cualitativo de la dimensión requerimientos de la plataforma

Categoría	Datos	Interpretación
Adaptación a las necesidades y evolución	La adaptación a las necesidades y evolución es crucial, ya que permite responder de manera ágil a los cambios en las demandas de los usuarios y las tendencias del mercado. Al considerar la flexibilidad y la escalabilidad en el diseño, se asegura que la plataforma no solo satisfaga las necesidades actuales, sino que también se ajuste a futuras exigencias. Esta capacidad de evolución es esencial para mantener la relevancia y la competitividad del proyecto a largo plazo, garantizando que los componentes open source compartidos sigan siendo útiles y accesibles para la comunidad.	Las respuestas sugieren que debe adaptarse a las necesidades cambiantes de la comunidad mediante el monitoreo constante de tendencias y tecnologías emergentes. La implementación de encuestas regulares para recoger feedback. Además, es importante que la comunidad ofrezca retroalimentación, facilite el contacto entre desarrolladores, e informe sobre componentes en desuso y avisos de incompatibilidad. Esto asegurará la relevancia y utilidad de la plataforma a largo plazo.

Fuente: Entrevistas a estudiantes de la carrera de Ingeniería de Computación

Integración de Resultados.

Análisis de resultados del cuestionario: Conexiones Sociales

Los resultados revelan que la comunidad uvemista busca de manera activa componentes Open Source para mejorar su desarrollo. Sin embargo, la consulta de plataformas sociales para intercambiar información indica una participación moderada, lo que sugiere que, aunque los miembros reconocen la utilidad de estos recursos, no los utilizan de manera constante.

Además, la baja interacción social con respecto al intercambio de información sobre componentes Open Source refleja una oportunidad de mejora en las conexiones sociales de la comunidad.

Promover el diálogo y la colaboración en la comunidad, así como crear espacios donde se comparta información sobre componentes, podría fortalecer estas conexiones y mejorar la experiencia colectiva en el uso de estos recursos.

Los resultados indican que la comunidad uvemista enfrenta desafíos moderados al buscar componentes Open Source específicos para sus frameworks, lo que sugiere que la accesibilidad y disponibilidad de estos recursos podría mejorarse. Sin embargo, la preferencia marcada por frameworks populares refleja una tendencia hacia herramientas que ofrecen robustez y un amplio soporte comunitario, lo cual facilita la localización de recursos en comparación con frameworks menos conocidos.

Sin embargo, la situación también revela una oportunidad para potenciar el acceso a componentes en entornos de desarrollo populares. Mejorar la organización y la visibilidad de estos recursos, así como promover la creación de repositorios más accesibles y dinámicos, podría mejorar la experiencia de los desarrolladores y promover una colaboración más eficiente en la comunidad.

Los resultados reflejan un uso moderado de plataformas sociales de componentes Open Source por parte de la comunidad, lo que sugiere que, aunque estas herramientas son reconocidas, su adopción y utilización son inconsistentes. No obstante, la voluntad de utilizar estas plataformas en el futuro sugiere un interés positivo y una posible apertura hacia su integración en el desarrollo.

El deseo de compartir componentes con colegas sugiere una cultura colaborativa emergente, lo que podría ser clave para impulsar la participación constante en estas plataformas. En conjunto, estos hallazgos apuntan a una oportunidad para incentivar el compromiso de la comunidad con estas herramientas, promoviendo su utilidad y facilitando el intercambio de recursos y conocimientos entre los desarrolladores. Esto podría resultar en un crecimiento sostenible y un aumento en la interacción dentro de la comunidad en el tiempo.

Integración con el primer objetivo

La comunidad uvemista muestra un interés importante en la búsqueda de componentes Open Source, aunque la utilización de plataformas sociales para ello no muy común. Lo cual destaca lo importante que es mejorar la interacción en la comunidad.

La búsqueda de componentes relacionados con frameworks populares indica que, si bien los desarrolladores buscan robustez y soporte, existe una oportunidad para aumentar la disponibilidad de recursos específicos. Fomentar una mejor organización y visibilidad en estos frameworks facilitaría su acceso.

El interés por usar plataformas sociales en el futuro y compartir recursos con colegas sugiere la posibilidad de crear una cultura colaborativa. Para aumentar el compromiso de la comunidad, es crucial mejorar el acceso a la información y a los componentes, lo que podría fomentar la interacción y un crecimiento sostenible.

Análisis de resultados de la entrevista: Requerimientos de la plataforma

Las entrevistas indican que los desarrolladores buscan una plataforma social de componentes Open Source que ofrezca una amplia gama de recursos bien documentados y accesibles. Se destaca la relevancia de un motor de búsqueda avanzado y una interfaz intuitiva, además de la opción de contactar a los autores.

La personalización es valorada, aunque su relevancia varía entre los usuarios. También se requiere un sistema de notificaciones personalizables que proporcione alertas relevantes y actualizaciones sobre proyectos seguidos.

En cuanto al soporte, los desarrolladores esperan documentación exhaustiva, tutoriales y foros de discusión. La seguridad es una preocupación prioritaria, con un enfoque en auditorías y protección de datos.

Finalmente, se menciona la necesidad de un proceso de revisión de componentes y la promoción de un entorno colaborativo. Para mantener la relevancia de la plataforma, se sugiere realizar encuestas regulares y monitorear tendencias emergentes. Estos elementos son clave para mejorar los conocimientos de los desarrolladores y fomentar un uso activo de la plataforma.

Integración con el segundo objetivo

Los hallazgos de las entrevistas apuntan a varios requerimientos técnicos y de seguridad cruciales para la implementación de una plataforma social de componentes Open Source, orientada a facilitar el intercambio colaborativo en la comunidad.

Los desarrolladores requieren un motor de búsqueda avanzado y una interfaz fácil de usar que simplifique la búsqueda de componentes. La documentación clara y exhaustiva es esencial, complementada por tutoriales y foros que respalden el aprendizaje. Además, la personalización,

aunque no es prioritaria para todos, es apreciada y debería contar con un sistema de notificaciones personalizables que mantenga a los usuarios al tanto de actualizaciones importantes.

La seguridad es crucial, por lo que se recomienda realizar auditorías periódicas y establecer revisiones iniciales para asegurar la calidad de los componentes, protegiendo los datos personales y verificando la autenticidad de los usuarios de la plataforma.

Fomentar un entorno colaborativo es vital; los desarrolladores valoran la retroalimentación. La comunidad debe involucrarse de manera constante en la moderación de componentes, asegurando así un estándar de calidad.

Validación de Resultados.

La investigación realizada por Rojo Alfonso (2020) en la UVM subraya la importancia de garantizar la seguridad y confiabilidad de los sistemas de información en el contexto de la ciberseguridad. Este hallazgo es coherente con los resultados de la presente investigación, que destaca la preocupación de los desarrolladores por la seguridad en el intercambio de componentes open source. Es esencial implementar auditorías regulares y un proceso de revisión para garantizar la calidad de los componentes, mencionada en las entrevistas, se alinea con la recomendación de Rojo sobre Es fundamental establecer medidas de seguridad robustas para salvaguardar la información de los usuarios.

Asimismo, el estudio de Gaitán Juan (2018) resalta la importancia de la experiencia del usuario y la presentación visual en plataformas digitales. Esta idea complementa los hallazgos de la presente investigación, que identifican la necesidad de una interfaz intuitiva y un motor de búsqueda avanzado para facilitar la localización de componentes. La interrelación entre el diseño y el contenido, mencionada por Gaitán, sugiere que una plataforma social efectiva no solo debe centrarse en la funcionalidad técnica, sino también en cómo se presentan los recursos, lo que puede

impactar positivamente en la participación de los usuarios y en la creación de una comunidad colaborativa.

Los hallazgos de la investigación indican una necesidad evidente de desarrollar un espacio centralizado para usuarios interesados en el ámbito de la programación open source, similar a la búsqueda de Marcos González (2021) por una red social dedicada a los aficionados al automovilismo. Ambas investigaciones muestran que los usuarios valoran la posibilidad de conectarse y compartir conocimientos en un entorno especializado, lo que refuerza la idea de que el diseño de plataformas debe responder a intereses específicos de las comunidades.

A su vez, subrayan la relevancia de la experiencia de usuario (UX) en el diseño de la plataforma. La investigación de Ramos Alfredo (2020) resalta que una mala UX puede llevar a productos menos funcionales. De manera paralela, nuestros resultados indican que los usuarios priorizan una interfaz amigable y accesible, lo que implica que descuidar la UX podría resultar en una menor participación y retención de usuarios. Esta coincidencia enfatiza la necesidad de integrar principios de diseño centrados en el usuario desde el inicio del desarrollo.

La importancia de promover la participación activa y la colaboración se ven reflejados en los resultados obtenidos, alineándose con las conclusiones de Guardiola Javier (2020). Los hallazgos sugieren que los usuarios no solo buscan consumir contenido, sino también interactuar y co-crear, lo que refleja el interés de involucrarse en el proceso de desarrollo. Este punto de vista implica que la plataforma debe facilitar mecanismos que incentiven la colaboración y el intercambio de ideas, reforzando así el potencial de innovación que puede surgir de una comunidad activa.

Determinando así, que los resultados coinciden con los antecedentes de esta investigación al resaltar la importancia de la seguridad informática, la experiencia del usuario en el diseño y

desarrollo de la plataforma, su relevancia en el desarrollo de software y la importancia de fomentar la participación y colaboración activa de los usuarios en el desarrollo de proyectos tecnológicos. Esto indica que las preocupaciones y demandas de la investigación están relacionadas.

III. FASE DE PRESENTACIÓN

Conclusiones.

La investigación cumplió con los objetivos establecidos al abordar cada aspecto clave de la problemática planteada. En primer lugar, se identificaron los medios utilizados por la comunidad uvemista para buscar componentes de programación Open Source, revelando una búsqueda activa, aunque con una participación moderada en plataformas sociales para intercambiar información. Esto evidencia una oportunidad para fomentar mayor colaboración en la comunidad.

Asimismo, se definieron los requerimientos técnicos y de seguridad necesarios para la implementación de componentes compartidos, destacándose la relevancia de facilitar el acceso a recursos bien organizados y mejorar la visibilidad de estos en la plataforma. En cuanto al desarrollo, la creación de la plataforma social de componentes Open Source alcanzó las expectativas de la comunidad al brindar un entorno intuitivo, con funciones de búsqueda avanzada, opciones de personalización, y un sistema de notificaciones personalizables.

Finalmente, la plataforma fue implementada, mostrando un potencial de crecimiento moderado pero prometedor, reflejando la disposición de los desarrolladores a compartir y colaborar en proyectos. Esto confirma la viabilidad y necesidad de la plataforma para mejorar la experiencia de los desarrolladores y a su vez, promover un intercambio colaborativo en la comunidad uvemista.

Recomendaciones.

La universidad debería incentivar la participación en la plataforma, animando a estudiantes, profesores y profesionales a compartir sus conocimientos. Esto fortalecería el capital social de la comunidad uvemista y promovería redes de colaboración y aprendizaje.

Es fundamental que la UVM continúe invirtiendo en el desarrollo y mejora de la plataforma, incorporando nuevas funcionalidades y mejoras basadas en las necesidades y retroalimentación de los usuarios. Esto asegurará que la plataforma siga siendo relevante y útil para la comunidad uvemista a largo plazo, promoviendo su sostenibilidad y crecimiento.

Se recomienda establecer alianzas estratégicas con empresas del sector privado que puedan beneficiarse de la plataforma y contribuir a su desarrollo y promoción. Estas alianzas pueden facilitar recursos adicionales, experiencia técnica y oportunidades de colaboración que impulsen la innovación y el crecimiento de la plataforma.

La UVM debe promover la plataforma y sus beneficios entre la comunidad académica, y el sector privado de la región. Esto puede lograrse a través de campañas de marketing, eventos de lanzamiento y participación en conferencias y talleres, resaltando su valor para el desarrollo colaborativo de software e innovación.

Por último, es aconsejable llevar a cabo revisiones periódicas sobre el rendimiento y la satisfacción de los usuarios con la plataforma, para detectar posibles mejoras y ajustar las características a las demandas de la comunidad. La retroalimentación por parte de los usuarios es fundamental para guiar el desarrollo futuro de la plataforma.

Planteamiento de Propuesta

La presente propuesta tiene la finalidad de desarrollar una plataforma social de componentes open source destinada a la comunidad uvemista. Este desarrollo surge de la necesidad de crear un espacio colaborativo y seguro donde estudiantes, profesores y profesionales puedan compartir, buscar y utilizar componentes de programación open source de manera eficiente. A partir de los resultados de la investigación, se ha detectado una alta demanda de una plataforma fácil de usar, accesible, segura y con funcionalidades de colaboración.

Justificación de la Propuesta

Los hallazgos del estudio demostraron que la comunidad está muy activa y dependiente de componentes open source, así como que tienen una tendencia a trabajar juntos en plataformas sociales. Sin embargo, también se observó una limitación en el intercambio de información en el entorno social. Buscando cubrir estas necesidades, se proporciona una plataforma que centraliza los recursos, fomenta la colaboración y mejora la eficiencia en el desarrollo web.

Misión

Desarrollar una plataforma social de componentes open source que facilite el intercambio, la colaboración y la innovación en el desarrollo de software dentro de la comunidad uvemista, asegurando un entorno seguro y eficiente para todos sus usuarios.

Diseño de la Propuesta

La plataforma será desarrollada utilizando una combinación de tecnologías modernas y probadas que garantizan un rendimiento óptimo, escalabilidad y seguridad.

Requerimientos del Sistema

- Funcionales:

- Registro y autenticación de usuarios.
- Búsqueda y filtrado de componentes open source.
- Publicación y gestión de componentes por parte de los usuarios.
- Sistema de comentarios y valoraciones.
- Notificaciones y actualizaciones en tiempo real.

- No funcionales:

- Seguridad y protección de datos.
- Alta disponibilidad y escalabilidad.
- Rendimiento y tiempos de respuesta óptimos.
- Interfaz de usuario intuitiva y accesible.

Arquitectura del Diseño

La arquitectura de la plataforma se basará en una estructura modular y escalable, utilizando las siguientes tecnologías:

- Frontend:

- Next.js
- Typescript
- TailwindCSS

- Backend:

- Express.js
- GraphQL
- Prisma
- **Base de Datos:**
- MySQL

Planificación y Desarrollo de la Propuesta

El desarrollo de la plataforma seguirá un enfoque ágil, dividiéndose en varias fases:

- Fase 1: Análisis y Diseño

- Recolección de requisitos.
- Diseño de la arquitectura del sistema.
- Creación de wireframes y prototipos.

- Fase 2: Desarrollo del Backend

- Configuración del entorno de desarrollo.
- Implementación de la API GraphQL con Express.js y Prisma.
- Configuración de la base de datos MySQL.

- Fase 3: Desarrollo del Frontend

- Configuración del proyecto Next.js.
- Implementación de las interfaces de usuario con TailwindCSS.
- Integración con la API GraphQL.

- Fase 4: Implementación y Despliegue

- Configuración del entorno de producción.
- Despliegue de la plataforma.
- Monitoreo y mantenimiento continuo.

Prototipo de la propuesta de la Plataforma social de componentes Open Source

UVMDev
UVMDev te ayuda a encontrar y compartir componentes de desarrollo web Open Source.

Inicia sesión

Correo electrónico
Placeholder

Contraseña
Placeholder

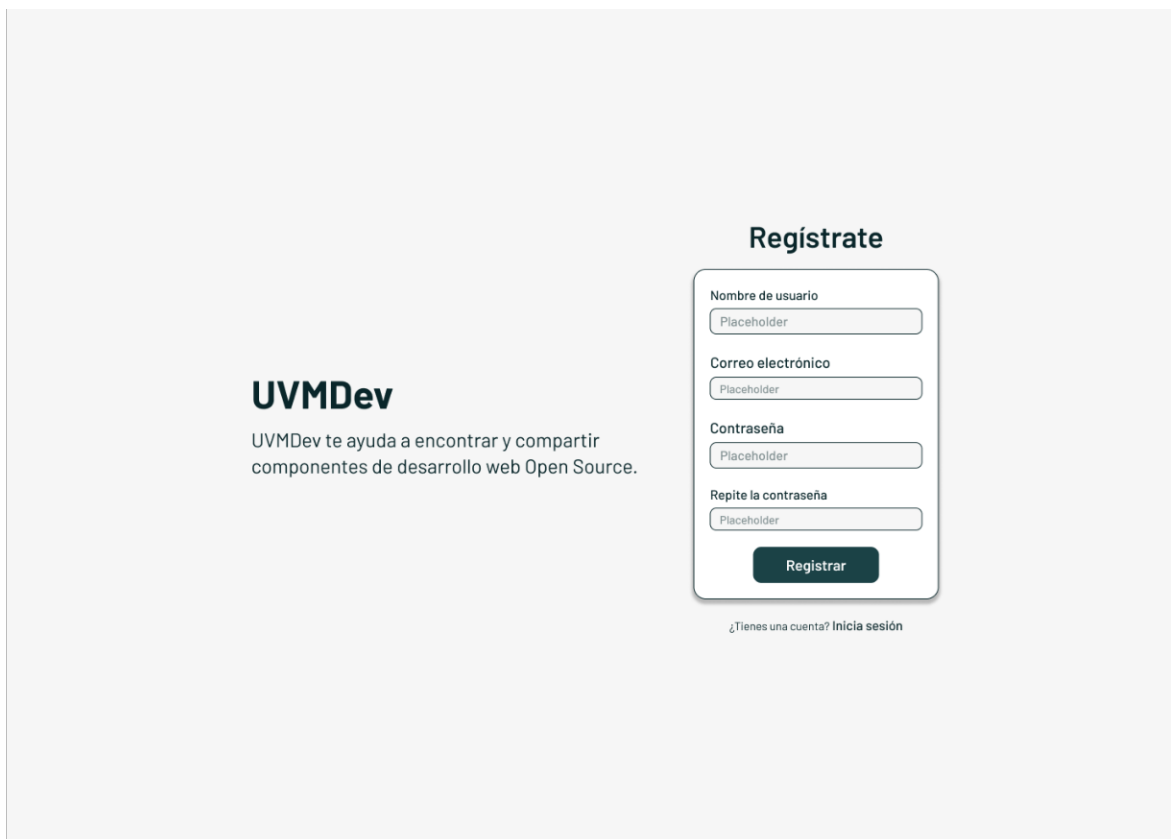
Iniciar sesión

¿Olvidaste tu contraseña?

¿No tienes una cuenta? [Regístrate](#)

Ilustración 9. Login

Fuente: Avendaño (2024)



The image shows a registration form for UVMDev. On the left, the logo 'UVMDev' is displayed in a bold, dark font, with the tagline 'UVMDev te ayuda a encontrar y compartir componentes de desarrollo web Open Source.' below it. On the right, a white rounded rectangle contains the registration fields. The title 'Regístrate' is centered at the top of this box. Below the title are four input fields, each with a 'Placeholder' label: 'Nombre de usuario', 'Correo electrónico', 'Contraseña', and 'Repite la contraseña'. A dark green button labeled 'Registrar' is positioned below the password fields. At the bottom of the registration box, there is a link that says '¿Tienes una cuenta? Inicia sesión'.

Ilustración 10. Register

Fuente: Avendaño (2024)

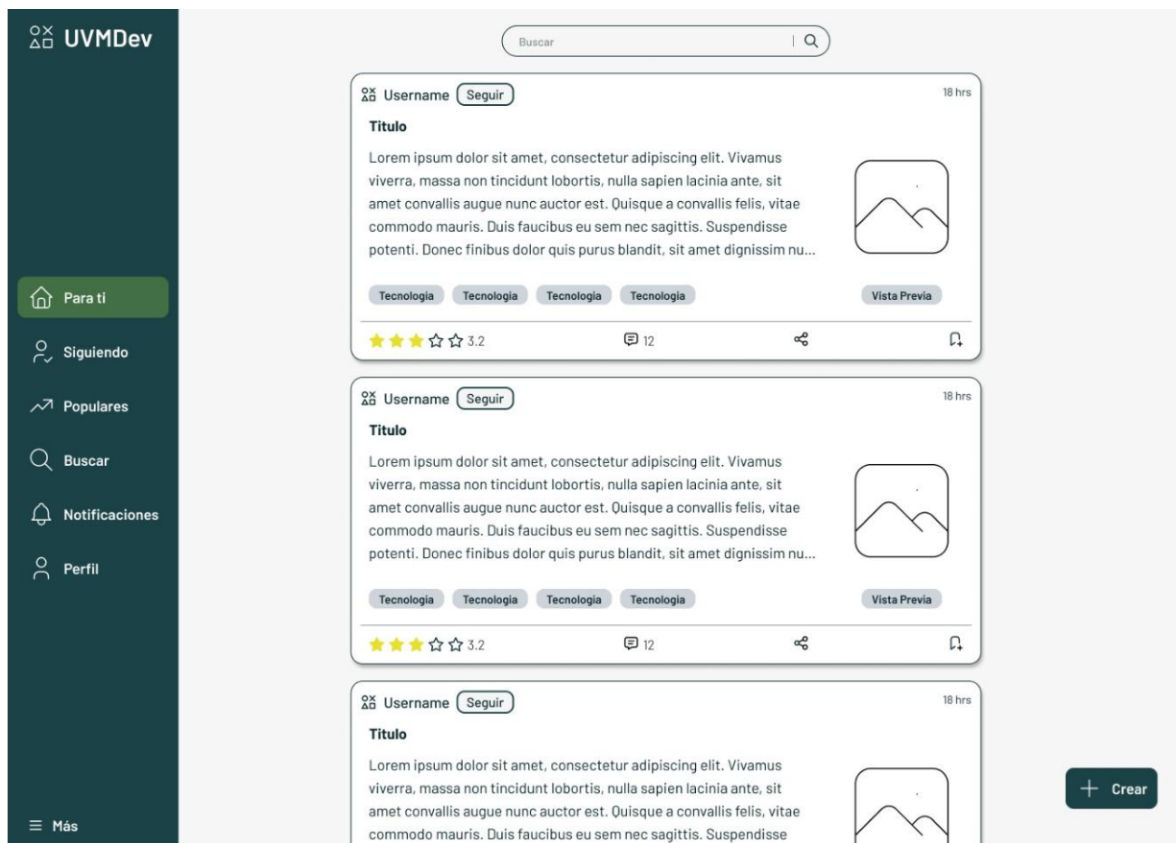


Ilustración 11. Pantalla principal: Para ti

Fuente: Avendaño (2024)

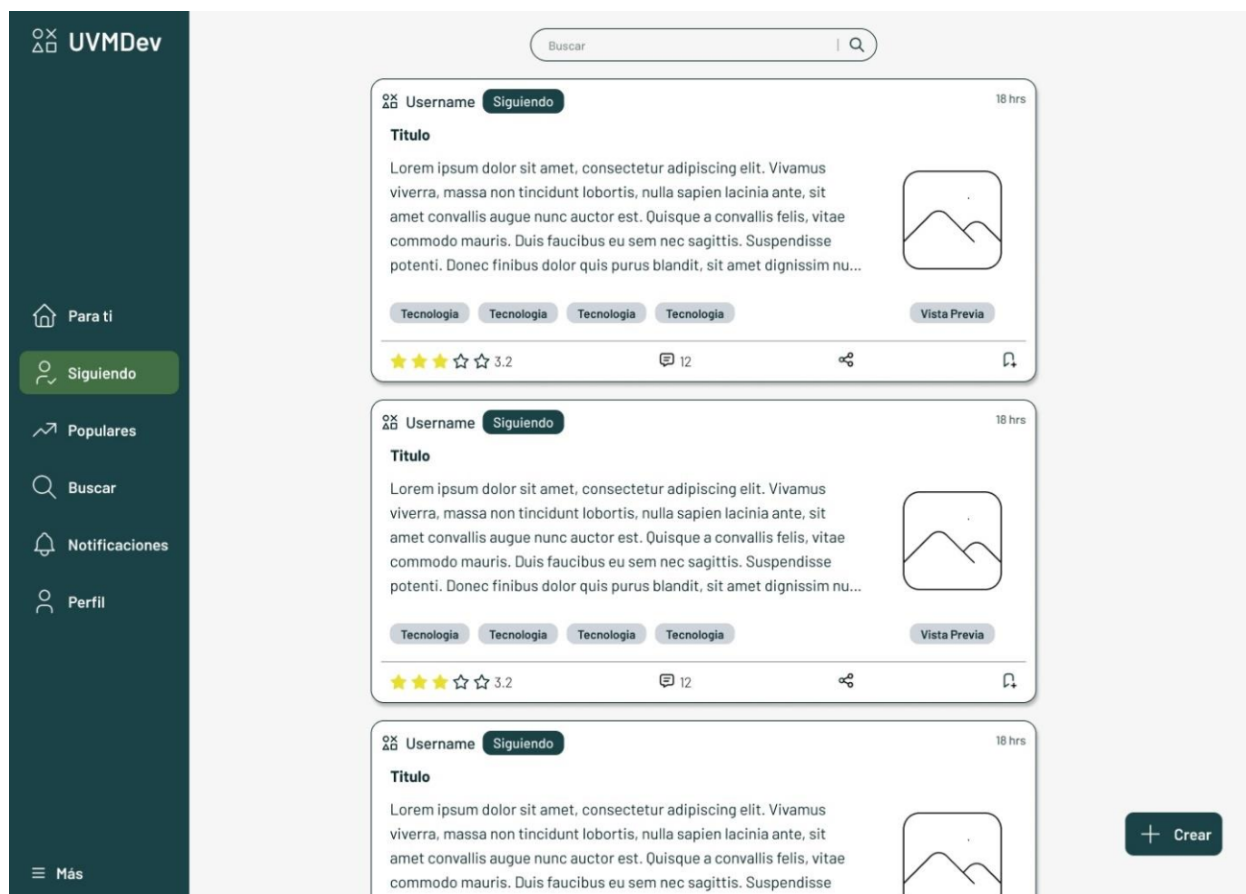


Ilustración 12. Pantalla principal: Siguiendo

Fuente: Avendaño (2024)

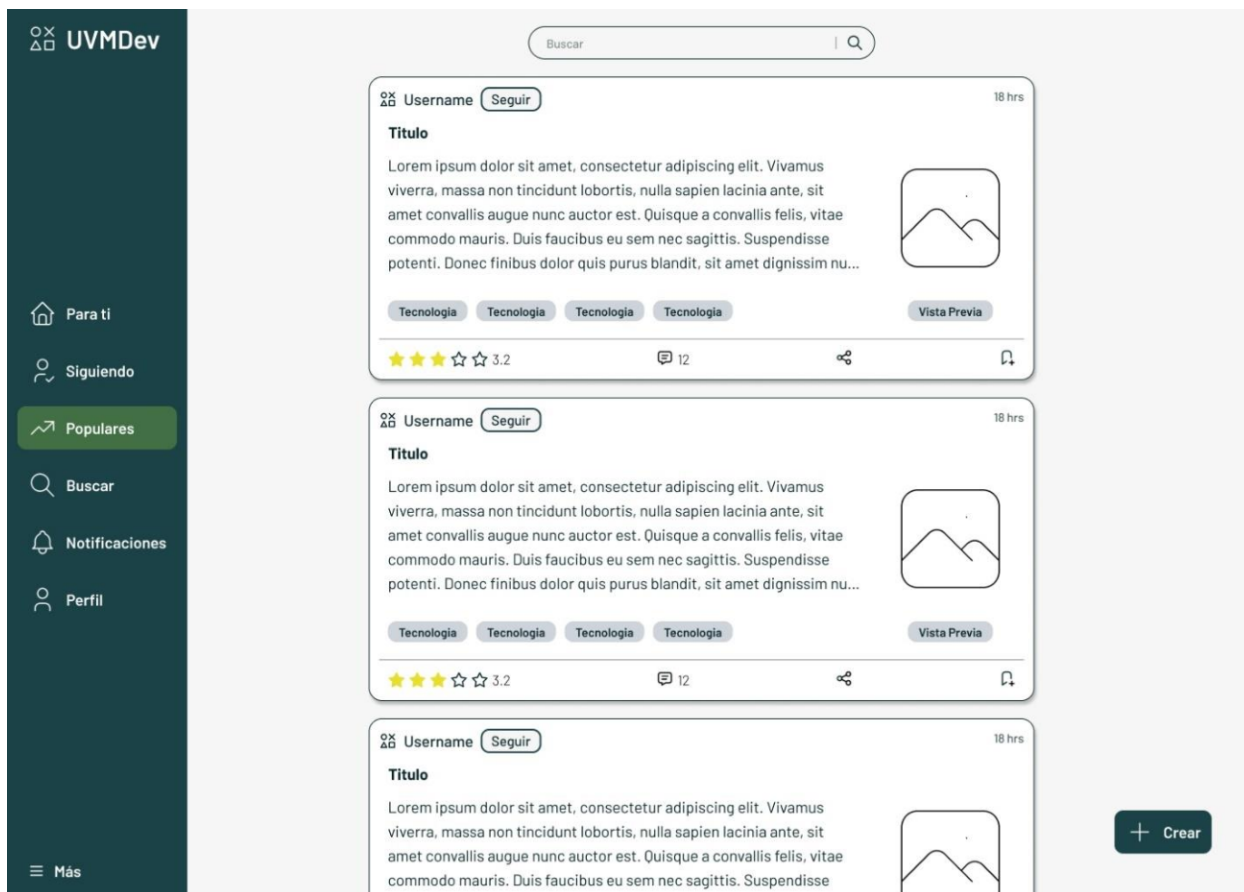


Ilustración 13. Pantalla principal: Populares

Fuente: Avendaño (2024)

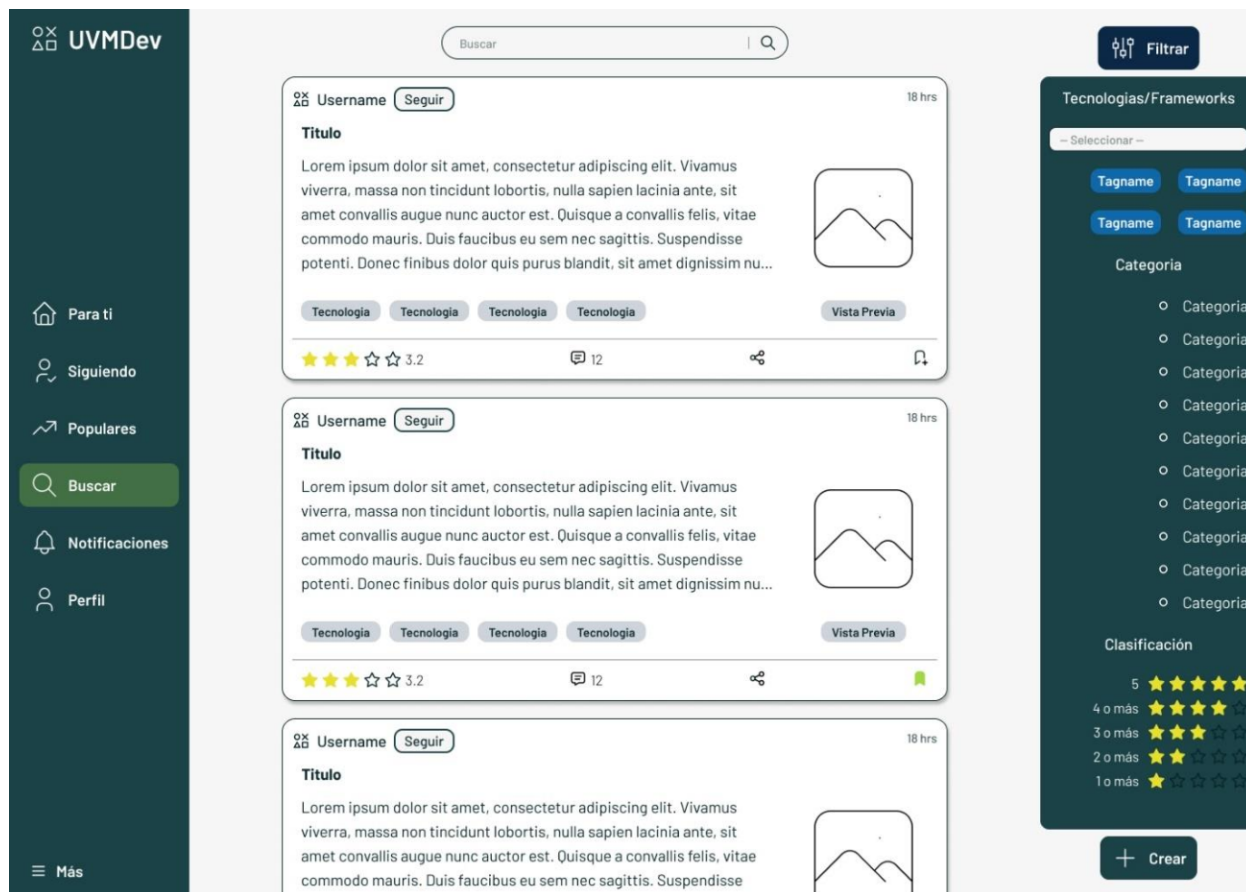


Ilustración 14. Pantalla principal: Buscar

Fuente: Avendaño (2024)

The screenshot displays a web application interface for UVMDev. On the left is a dark green sidebar with navigation options: 'Para ti', 'Siguiendo', 'Populares', 'Buscar', 'Notificaciones', 'Perfil', and 'Más'. The main content area features a post with the following elements:

- Title:** 'Titulo'
- Description:** A paragraph of Lorem Ipsum text.
- Metadata:** 'Desarrollado por: Username', 'Fecha de publicación: 01/03/2024', and a 'Stack' section with six 'Tagname' tags.
- Image:** A placeholder image showing a mountain range.
- Rating:** A star rating of 3.5 and a 'Compartir' button.
- Code Snippets:** Two identical code blocks, each with a 'Nombre del archivo' label and a 'Descargar archivo' button. The code is:


```
import { Button } from "@components/ui/button"

export function ButtonDemo() {
  return <Button>Button</Button>
}
```

On the right, a 'Comentarios' sidebar shows 10 comments, each with a user profile icon, 'Username', and a timestamp (e.g., 'Ahora', '20 min', '1 h', '2 h', '8 h'). At the bottom of the sidebar is a text input field with the placeholder 'Deja tu comentario aqui' and a submit icon.

Ilustración 15. Componente/Publicación

Fuente: Avendaño (2024)

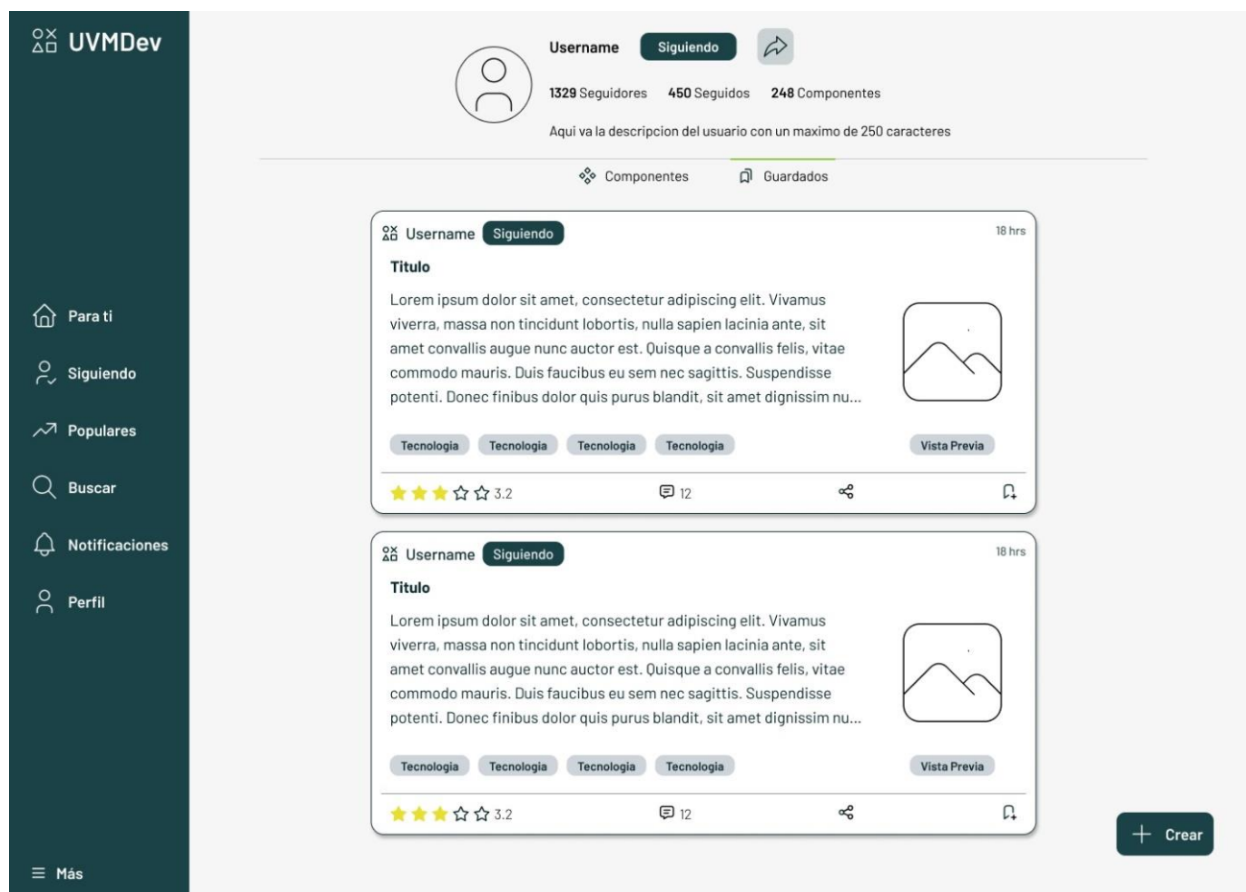


Ilustración 16. Perfil de un usuario

Fuente: Avendaño (2024)

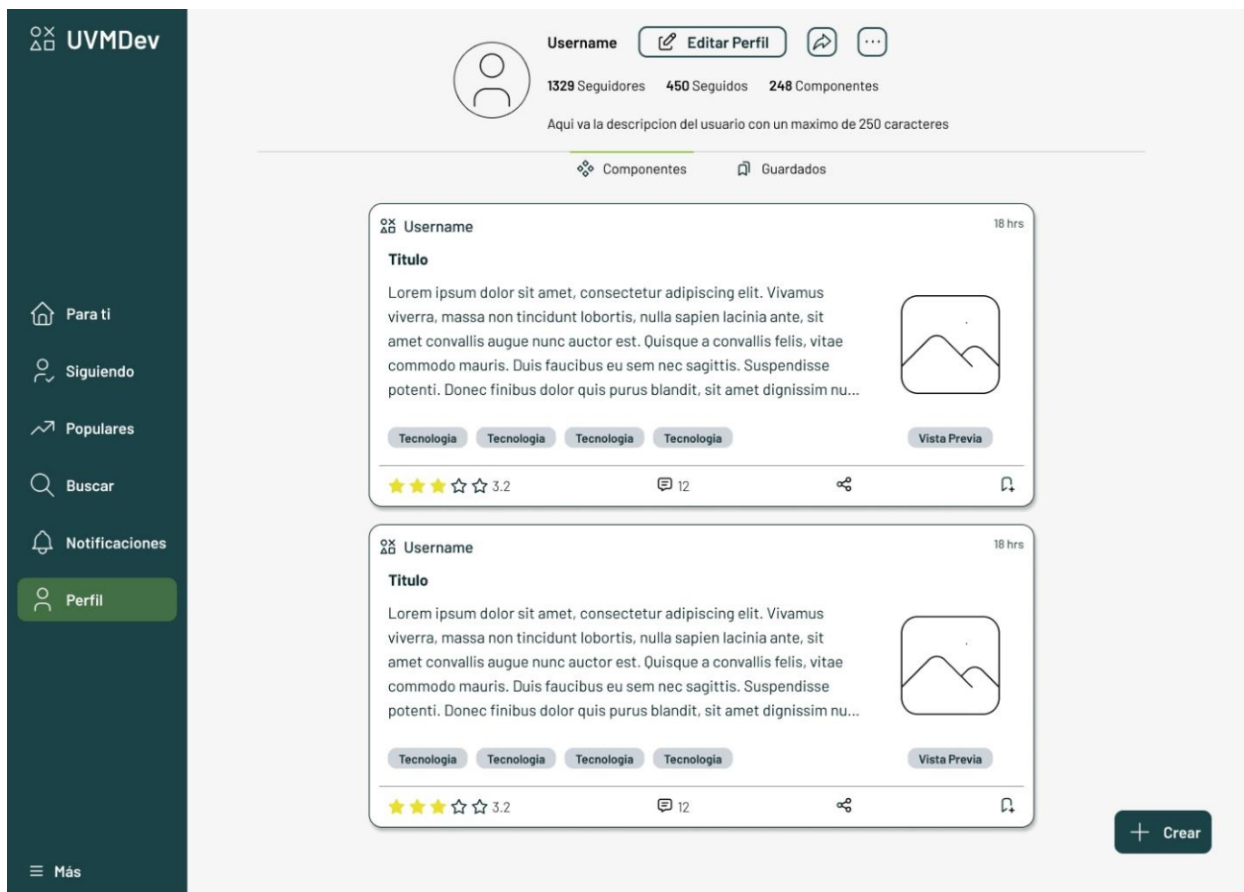


Ilustración 17. Pantalla principal: Mi perfil


Fuente: Avendaño (2024)

UVMDev

Comparte tu componente

Título

Descripción

 **Stack**

Nombre del archivo

Para ti
Siguiendo
Populares
Buscar
Notificaciones
Perfil
Más

Ilustración 18. Crear un componente

Fuente: Avendaño (2024)

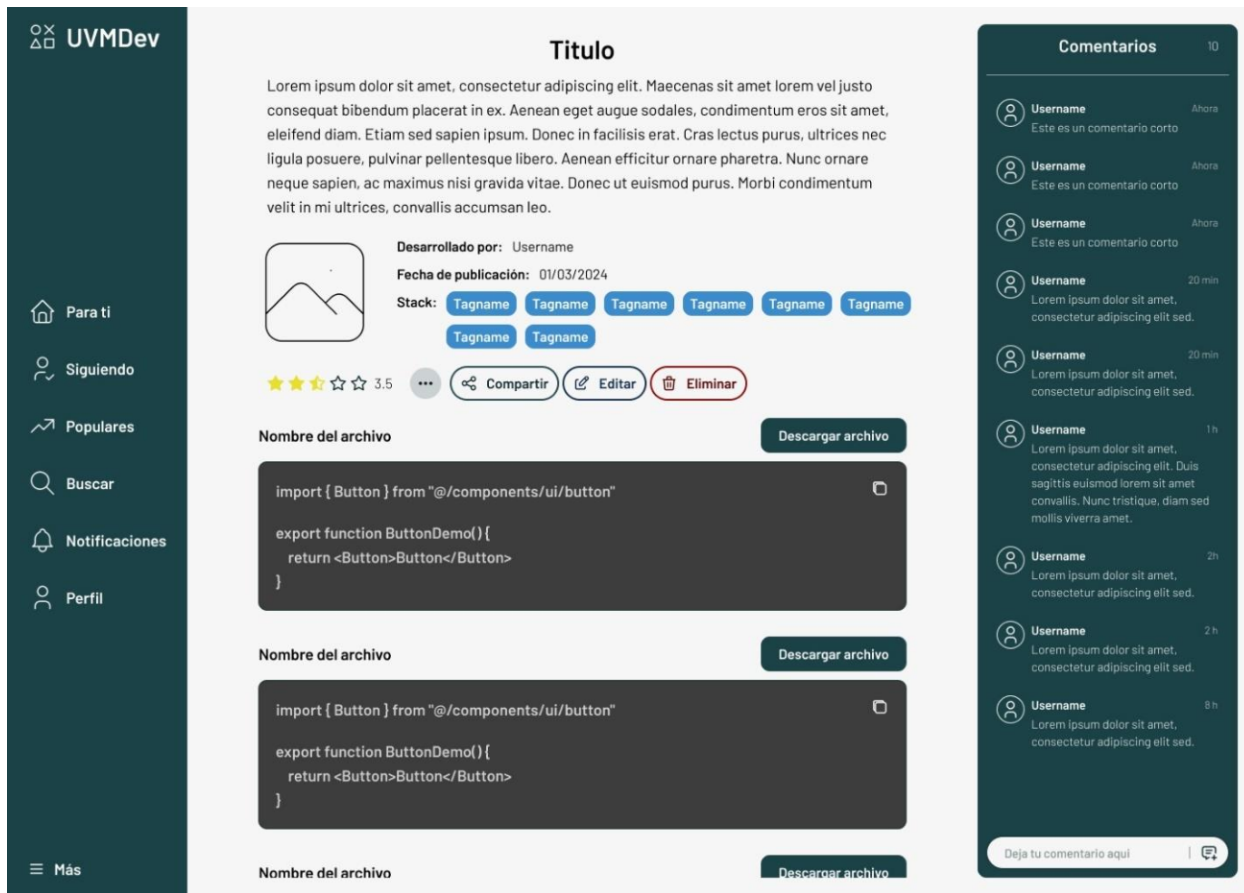


Ilustración 19. Mi componente/publicación

Fuente: Avendaño (2024)

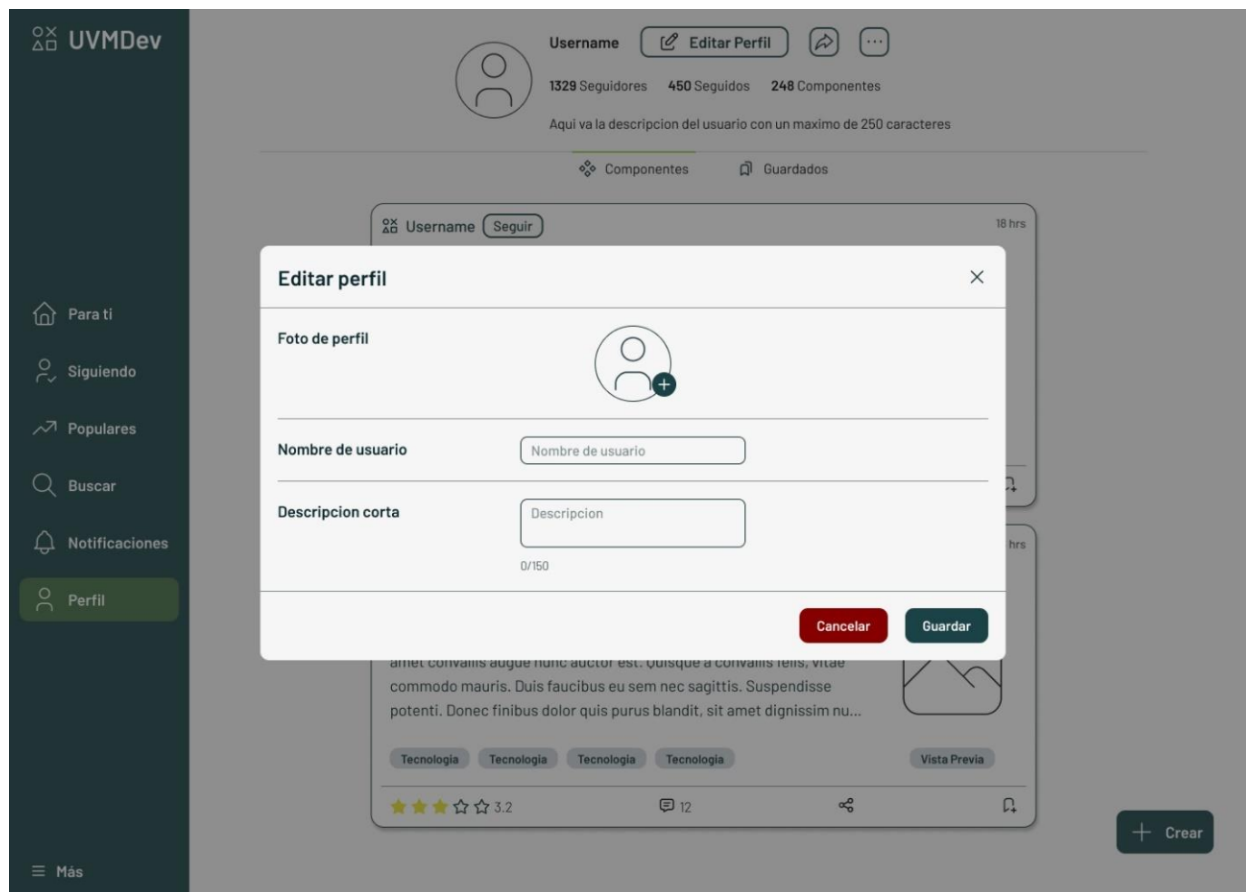


Ilustración 20. Editar Perfil

Fuente: Avendaño (2024)

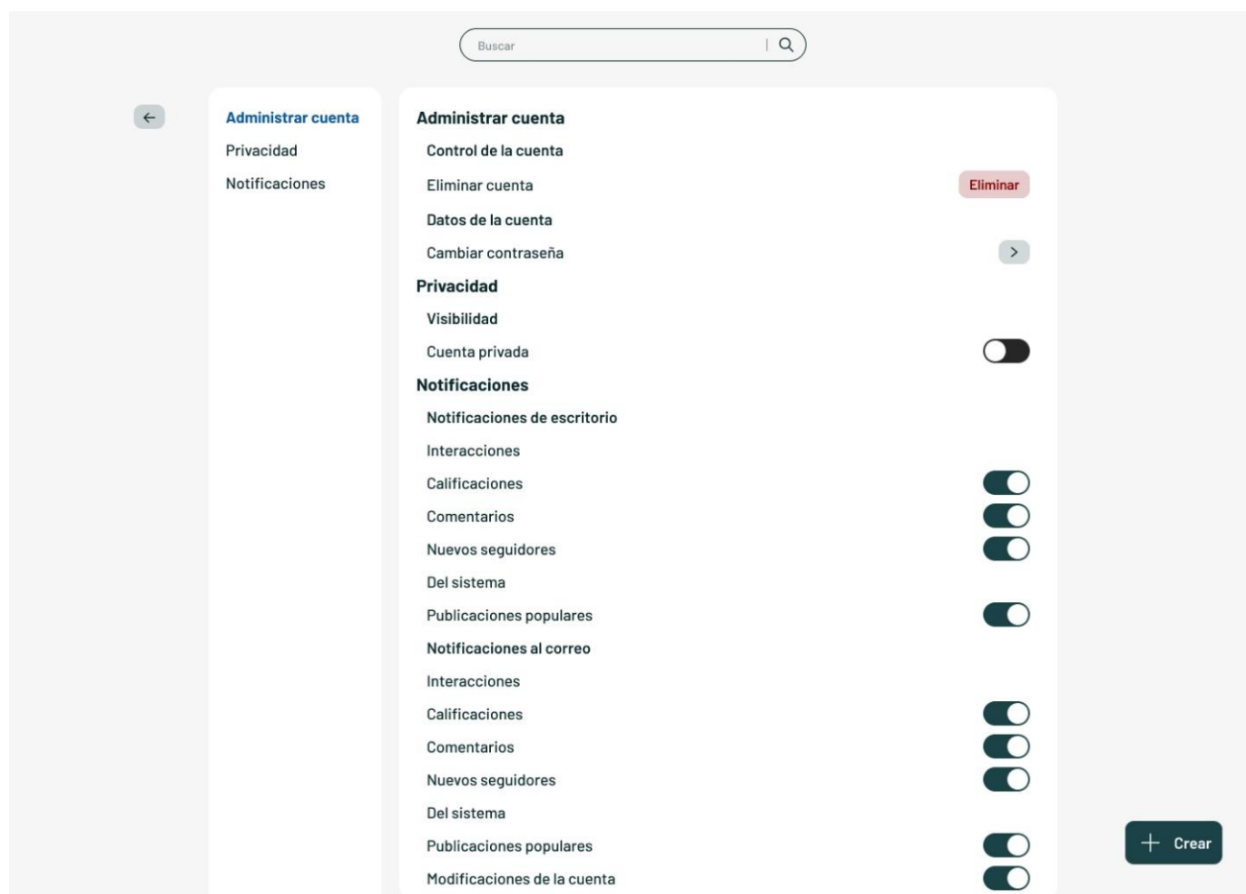


Ilustración 21. Pantalla principal: Configuraciones

Fuente: Avendaño (2024)

REFERENCIAS

Arimetrics. (s/f). "JavaScript". <https://www.arimetrics.com/glosario-digital/javascript>

Amazon Web Services, Inc. (2023). "¿Qué es JavaScript?". <https://aws.amazon.com/es/what-is/javascript/>.

Amazon Web Services, Inc. (2023). "¿Qué es una base de datos?". <https://aws.amazon.com/es/what-is/database/>.

Cimas Gustavo (2024). "Qué es un lenguaje de programación: Guía para principiantes". <https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/>

Fidias G. Arias (2006). El proyecto de investigación. 6ªed. Editorial Episteme C.A.

GAITAN JUAN. (2018). "Diseño y elaboración de una página web de información publicitaria para la galletera ITALIA C.A. (tesis para optar al título de Técnico Superior Universitario en Informática). <https://es.scribd.com/document/400888004/Proyecto-de-Investigacion-Gaitan>.

Guardiola Javier (2020). "Plataformas Tecnológicas de Inteligencia Colectiva como Herramientas de Innovación y Emprendimiento; Caracterización y Estado del Arte". (Tesis para optar por el título de Ingeniería Industrial). UNIVERSIDAD POLITÉCNICA DE MADRID. ESPAÑA. https://oa.upm.es/63452/1/TFG_JAVIER_GUARDIOLA_SAURA.pdf

Gonzales, R. y Salazar, F. (2008). ASPECTOS BÁSICOS DEL ESTUDIO DE MUESTRA Y POBLACIÓN PARA LA ELABORACIÓN DE LOS PROYECTOS DE INVESTIGACIÓN

Disponible en: <https://recursos.salonesvirtuales.com/assets/bloques/Raisirys-Gonz%C3%A1lez.pdf>

Hernández Guillermo. (2019). Metodología de código abierto Open Source Methodology

<https://ojs.docentes20.com/index.php/revista-docentes20/article/download/24/47/106>

Herraiz Marcos (2021). “Diseño y desarrollo de una aplicación web como Red social para aficionados a la automovilística”. (Tesis para optar por el título de Ingeniería Informática)

Universidad Politécnica de Valencia.

[https://riunet.upv.es/bitstream/handle/10251/173828/Herraiz%20-](https://riunet.upv.es/bitstream/handle/10251/173828/Herraiz%20-%20Diseno%20y%20desarrollo%20de%20una%20aplicacion%20web%20como%20red%20social%20para%20aficionados%20a%20la%20automov....pdf;jsessionid=F09D479934D935A9BF15968DB11D9902?sequence=1)

[%20Diseno%20y%20desarrollo%20de%20una%20aplicacion%20web%20como%20red%20social%20para%20aficionados%20a%20la%20automov....pdf;jsessionid=F09D479934D935A9BF15](https://riunet.upv.es/bitstream/handle/10251/173828/Herraiz%20-%20Diseno%20y%20desarrollo%20de%20una%20aplicacion%20web%20como%20red%20social%20para%20aficionados%20a%20la%20automov....pdf;jsessionid=F09D479934D935A9BF15968DB11D9902?sequence=1)

[968DB11D9902?sequence=1](https://riunet.upv.es/bitstream/handle/10251/173828/Herraiz%20-%20Diseno%20y%20desarrollo%20de%20una%20aplicacion%20web%20como%20red%20social%20para%20aficionados%20a%20la%20automov....pdf;jsessionid=F09D479934D935A9BF15968DB11D9902?sequence=1)

Huet Pablo. (2022). “Qué es Tailwind CSS y por qué deberías usarlo”.

<https://openwebinars.net/blog/que-es-tailwind-css-y-por-que-deberias-usarlo/>.

Hernández sampieri (2006). Metodología de la investigación.

[https://jalintonreyes.files.wordpress.com/2013/05/sampieri-5a-edicic3b3n-roberto-et-al-](https://jalintonreyes.files.wordpress.com/2013/05/sampieri-5a-edicic3b3n-roberto-et-al-metodologic3ada-de-la-investigacic3b3n.pdf)

[metodologic3ada-de-la-investigacic3b3n.pdf](https://jalintonreyes.files.wordpress.com/2013/05/sampieri-5a-edicic3b3n-roberto-et-al-metodologic3ada-de-la-investigacic3b3n.pdf)

Hernández Posada Jaime. (2016). "ELEMENTOS BÁSICOS DE ESTADÍSTICA DESCRIPTIVA Para el análisis de datos". Fundación Universitaria Luis Amigo Transversal 51 A 67 B 90. Medellín, Antioquia, Colombia.

Hernández Sampieri, R.; Fernández, C. y Baptista, P. (2019). Metodología de la Investigación. 7ª ed. Editorial Mc Graw Hill. Interamericana S.A.

Juan (2022). "Que es un Framework en programación y sus principales usos". Recuperado de: <https://assemblerinstitute.com/blog/framework-programacion/>

Kinsta. (2022). "¿Qué es Express.js? Todo lo que Debes Saber". <https://kinsta.com/es/base-de-conocimiento/que-es-express/>.

Kinsta. (2022). "¿Qué es Next.js? Un Vistazo al Popular Framework de JavaScript". <https://kinsta.com/es/base-de-conocimiento/next-js/>

Lucid Software Inc. (2024). "Qué es un diagrama entidad-relación". <https://www.lucidchart.com/pages/es/que-es-un-diagrama-entidad-relacion#:~:text=Un%20diagrama%20entidad-relaci%C3%B3n%2C%20tambi%C3%A9n,s%C3%AD%20dentro%20de%20un%20sistema>

Martínez Santos. (2023). ¿QUÉ ES EL OPEN SOURCE Y CÓMO PUEDE AYUDARTE? <https://openexpoeurope.com/es/open-source-puede-ayudarte/>

Martins Asana. (2024). “¿Qué es la metodología Kanban y cómo funciona?”.

<https://asana.com/es/resources/what-is-kanban>

Marylere, N. (2023). Técnicas de recolección de datos: Qué son y cuáles existen. Recuperado de:

[https://www.questionpro.com/blog/es/tecnicas-de-recoleccion-de-](https://www.questionpro.com/blog/es/tecnicas-de-recoleccion-de)

[datos/#:~:text=T%C3%A9cnicas%20de-,recolecti%C3%B3n,-](https://www.questionpro.com/blog/es/tecnicas-de-recoleccion-de-datos/#:~:text=T%C3%A9cnicas%20de-,recolecti%C3%B3n,-)

[de%20datos%3A%20Qu%C3%A9](https://www.questionpro.com/blog/es/tecnicas-de-recoleccion-de-datos/#:~:text=T%C3%A9cnicas%20de-,recolecti%C3%B3n,-de%20datos%3A%20Qu%C3%A9)

Ortega, C. (2019). ¿Qué es un cuestionario? Recuperado de:

<https://www.questionpro.com/blog/es/que-es-un-cuestionario/>

Rojó Alfonso (2020). “SEGURIDAD INFORMATICA: TECNOLOGIA DE DEFENSA EN

PROFUNDIDAD Y PENTESTING”, (Tesis para optar por el título de Ingeniero en Computación).

Universidad Valle del Momboy.

<https://repositorio.uvm.edu.ve/server/api/core/bitstreams/8d10b857-bb23-47fb-a7bc->

[6e1116304009/content](https://repositorio.uvm.edu.ve/server/api/core/bitstreams/8d10b857-bb23-47fb-a7bc-6e1116304009/content)

Ramos Alfredo (2020). “Tutor Inteligente Móvil para el Aprendizaje de la Experiencia de Usuario

(UX) para Diseñadores web”. (Tesis para optar por el título de Licenciado en Informática)

Universidad Mayor de San Andrés).

<https://repositorio.umsa.bo/bitstream/handle/123456789/27853/T->

[3644.pdf?sequence=1&isAllowed=y](https://repositorio.umsa.bo/bitstream/handle/123456789/27853/T-3644.pdf?sequence=1&isAllowed=y)

V. Parra Manuel. (2021). “¿Qué es un Design System? Pasos, ejemplos y herramientas para crear uno”.

https://profile.es/blog/que-es-design-system-ejemplo/#%C2%BFQue_es_un_sistema_de_diseno_o_Design_System

ANEXOS

ANEXO 1
CUESTIONARIO

CUESTIONARIO

Esta encuesta busca evaluar los hábitos y preferencias de los desarrolladores en la búsqueda y uso de componentes open source. A través de preguntas que utilizan una escala de Likert, se indagará sobre la frecuencia de búsqueda de componentes, la consulta de plataformas sociales y la facilidad para encontrar recursos según el framework utilizado. Las respuestas recopiladas ayudarán a identificar áreas de mejora para la plataforma social de componentes open source propuesta y a comprender mejor las tendencias en el uso de recursos colaborativos.

Medios para conseguir componentes open source

1. ¿Qué tan seguido busca componentes open source para agilizar el desarrollo de su proyecto?
 - Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca

2. ¿Con qué regularidad consulta plataformas sociales de componentes open source para intercambiar información?
 - Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca

3. ¿Qué tan seguido tu entorno social te comparte información sobre componentes open source?
- Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca

Frameworks

4. ¿Sueles encontrar con facilidad componentes open source del framework que necesitas?
- Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca
5. ¿Con qué frecuencia utilizas frameworks populares (React, Angular, Vue, etc) en tus proyectos de desarrollo?
- Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca

Tendencia de crecimiento

6. ¿Con qué frecuencia has utilizado una plataforma social de componentes open source en los últimos seis meses?
- Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca
7. ¿Qué tan a menudo utilizarías la plataforma social de componentes open source en el futuro?
- Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca
8. ¿Qué tan seguido compartirías componentes de esta plataforma con tus amigos y colegas?
- Siempre
 - Casi siempre
 - Algunas veces
 - Casi nunca
 - Nunca

ANEXO 2
ENTREVISTA

ENTREVISTA

Preguntas de la entrevista

1. ¿Qué tipo de características esperarías encontrar en una plataforma social de componentes open source para que la consideres útil y relevante para tus necesidades de desarrollo? Describe tus expectativas.
2. ¿Qué funcionalidades específicas te gustaría ver en una plataforma social de componentes open source para mejorar tu experiencia como usuario? Describe las características que consideras más importantes para ti.
3. ¿Qué tan importante es para ti la capacidad de personalización y configuración en una plataforma social de componentes open source? Describe qué aspectos te gustaría poder personalizar según tus preferencias y necesidades.
4. ¿Qué tipo de sistema de notificaciones o alertas considerarías útil en una plataforma social de componentes open source para mantenerte informado sobre nuevas actualizaciones, comentarios o actividades relevantes?
5. ¿Qué tipo de soporte o recursos adicionales esperarías encontrar en una plataforma social de componentes open source para ayudarte en tu proceso de desarrollo? Menciona cualquier herramienta, documentación o comunidad de soporte que consideres útil.
6. ¿Qué aspectos de seguridad consideras más importantes al utilizar o contribuir a una plataforma social de componentes open source? Detalla tus preocupaciones sobre seguridad.
7. ¿Qué medidas de moderación o control de calidad esperarías que implemente una plataforma social de componentes open source para garantizar la fiabilidad y la

relevancia de los recursos compartidos? Detalla tus expectativas sobre la gestión del contenido.

8. ¿Cómo crees que una plataforma social de componentes open source podría mejorar la colaboración y el intercambio de conocimientos entre los desarrolladores de software? Proporciona ejemplos o ideas específicas.
9. ¿Cómo crees que una plataforma social de componentes open source podría adaptarse mejor a las necesidades cambiantes y la evolución de la comunidad de desarrolladores a lo largo del tiempo? Proporciona ideas sobre cómo la plataforma podría mantenerse relevante y útil a largo plazo.

ANEXO 3
VALIDEZ

UNIVERSIDAD VALLE DEL MOMBOY
VICERRECTORADO ACADÉMICO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE COMPUTACIÓN



CONSTANCIA DE VALIDACIÓN DEL INSTRUMENTO

Quien Suscribe: **Yerson José González Pacheco**, titular de la cédula de identidad No: 14149542 de profesión: **Ingeniero de Computación**, hace constar por medio de la presente, que luego de leer, analizar e interpretar el instrumento de recolección de información, elaborado para dar cumplimiento a los objetivos de la investigación titulada: **Plataforma social de Componentes Open Source**, que presenta el Br. Alexander José Avendaño Ramírez, titular de la cédula de identidad No.: V-29.694.896, considero que el mismo reúne las condiciones necesarias en cuanto a pertinencia, relación variable-dimensión-indicador-ítems, congruencia y estilo de redacción adecuado de los ítems.

En consecuencia, el referido instrumento es válido para los fines previamente establecidos.

Constancia que se expide en la ciudad de Valera, a los 30 días del mes de Septiembre del año 2024.

Firma

AUTOR: Br. Alexander Avendaño

TUTOR: Ing. Roberto Di Michele

UNIVERSIDAD VALLE DEL MOMBOY
VICERRECTORADO ACADÉMICO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE COMPUTACIÓN



CONSTANCIA DE VALIDACIÓN DEL INSTRUMENTO

Quien Suscribe: **Edgardo José Paolini Quiroz**, titular de la cédula de identidad No: 13897564 de profesión: **Ingeniero en computación**, hace constar por medio de la presente, que luego de leer, analizar e interpretar el instrumento de recolección de información, elaborado para dar cumplimiento a los objetivos de la investigación titulada: **Plataforma social de Componentes Open Source.**, que presenta el Br. Alexander José Avendaño Ramírez, titular de la cédula de identidad No.: V-29.694.896, considero que el mismo reúne las condiciones necesarias en cuanto a pertinencia, relación variable-dimensión-indicador-ítems, congruencia y estilo de redacción adecuado de los ítems.

En consecuencia, el referido instrumento es válido para los fines previamente establecidos.

Constancia que se expide en la ciudad de Valera, a los 30 días del mes de Septiembre del año 2024.

Firma

AUTOR: Br. Alexander Avendaño

TUTOR: Ing. Roberto Di Michele

ANEXO 4
CONFIABILIDAD

CÁLCULO DE CONFIABILIDAD

Escala de confiabilidad

Rangos	Magnitud
0,81 a 1,00	Muy alta
0,61 a 0,80	Alta
0,41 a 0,60	Moderada
0,21 a 0,40	Baja
0,01 a 0,20	Muy baja

Nota: Revela el nivel de confiabilidad según el valor del coeficiente obtenido

Fuente: Ruiz (2013)

Coefficiente de Alfa de Cronbach.

Numero de ítems	8
Sumatoria de las varianzas de los ítems	5,98
Varianza total del instrumento	14,32
Coeficiente de confiabilidad del instrumento	0,67
Confiabilidad	0,67 (Alta)

CONFIABILIDAD ALFA DE CRONBACH										
ENCUESTADOS	ITEMS								SUMA	
	1	2	3	4	5	6	7	8		
E1	4	4	4	3	5	4	5	3	32	
E2	4	3	2	3	5	3	4	3	27	
E3	3	3	3	5	5	3	3	3	28	
E4	4	3	5	3	3	4	4	4	30	
E5	4	3	3	5	5	5	5	5	35	
E6	3	2	2	3	5	3	3	3	24	
E7	4	3	2	3	4	4	4	3	27	
E8	5	4	2	4	5	3	4	4	31	
E9	4	3	4	3	5	3	4	4	30	
E10	5	4	5	4	5	4	5	4	36	
E11	4	2	2	4	2	3	4	5	26	
E12	3	3	3	3	3	3	3	3	24	
E13	3	4	3	3	5	3	4	5	30	
E14	4	2	2	4	4	2	3	4	25	
E15	5	4	4	3	4	3	3	4	30	
E16	3	3	1	3	2	2	5	4	23	
E17	2	3	3	3	4	3	3	3	24	
E18	3	3	4	2	1	3	4	4	24	
VARIANZA	0,645062	0,432099	1,222222	0,570988	1,555556	0,506173	0,54321	0,506173		
SUMATORIA DE VARIANZAS	5,981481481									
VARIANZA DE LA SUMA DE ITEMS	14,3									
Coeficiente de confiabilidad del instrumento								0,67		
Numero de ítems del instrumento								8		
Sumatoria de las varianzas de los ítems								5,98148		
Varianza total del instrumento								14,32		

ANEXO 5
CARTA DE ACEPTACIÓN

UNIVERSIDAD VALLE DEL MOMBOY
VICERRECTORADO ACADÉMICO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE COMPUTACIÓN



APROBACIÓN DEL TUTOR

En mi carácter de Tutor del Trabajo Especial del Grado Titulado: **PLATAFORMA SOCIAL DE COMPONENTES OPEN SOURCE** realizado por el bachiller Alexander José Avendaño Ramírez C.I. 29.694.896, para optar al título de **Ingeniero de Computación**, considero que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido ante la presentación pública y la evaluación por parte del jurado que se asigne.

Atentamente,

MSc. Roberto Di Michele
C.I. 19.794.455

A los 11 días del mes de octubre de 2024.

ANEXO 6
MANUAL DE USUARIO

Manual de usuario de la Plataforma Social de Componente Open Source

MANUAL DE USUARIO DE UVM DEV HOUSE

Plataforma Social de Componentes Open Source

Desarrollado por:

Alexander Avendaño

Inicio de sesión

Esta es la primera vista que encuentra un usuario al ingresar a la plataforma, en donde deberá ingresar sus datos si ya tiene una cuenta creada, de lo contrario deberá pulsar el boton “**Regístrate**” para ir a crear una nueva cuenta, en esta vista ingresa sus datos correspondientes y accede a la plataforma.

UVM Dev House
UVM Dev House te ayuda a encontrar y compartir componentes de desarrollo web Open Source.

Iniciar sesion

Correo Electronico
correo@gmail.com

Contraseña
Ingresa tu contraseña

Iniciar Sesion

¿Olvidaste tu contraseña?

¿No tienes una cuenta? [Regístrate](#)

Registro

En esta vista es donde el usuario podrá registrarse en la plataforma rellenando los datos solicitados en el formulario, una vez que haya creado el usuario, le redirigirá al inicio de sesión donde deberá colocar sus datos de inicio de sesión y podrá acceder a la plataforma.

UVM Dev House
UVM Dev House te ayuda a encontrar y compartir componentes de desarrollo web Open Source.

Regístrate

Nombre de Usuario

Correo Electronico

Contraseña

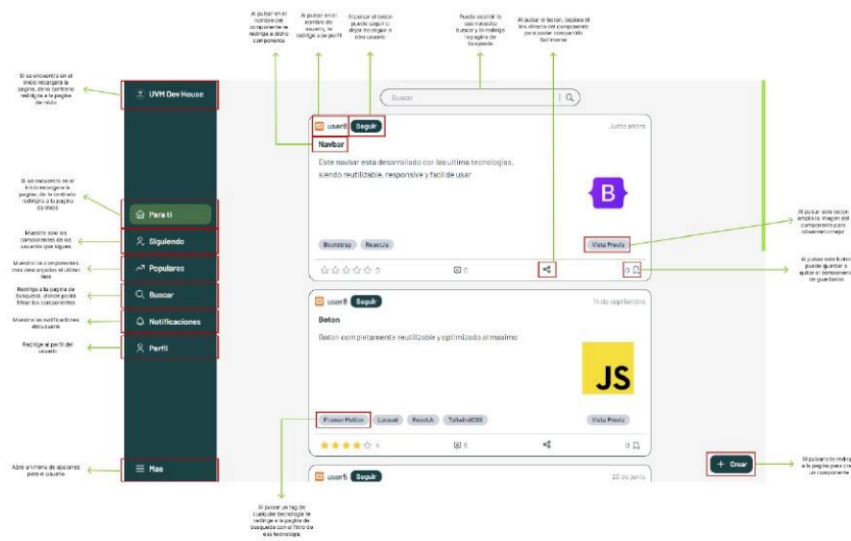
Repite la Contraseña

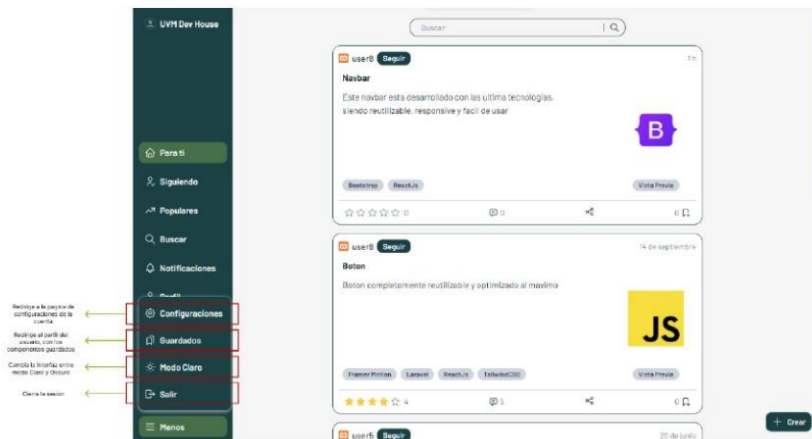
Registrar

[¿Tienes una cuenta? Inicia sesión](#)

Inicio

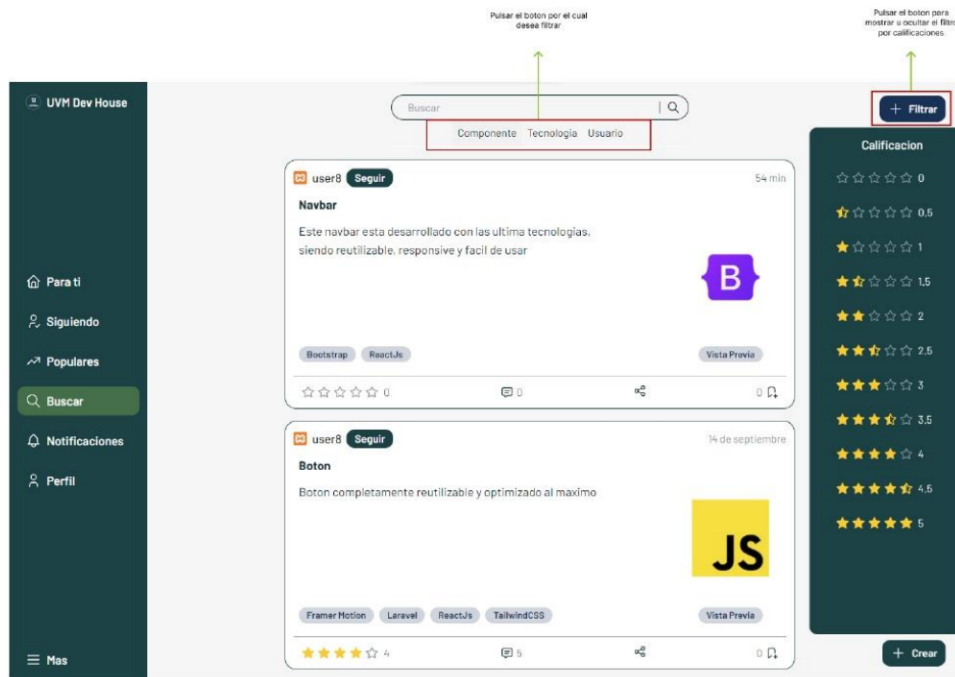
Ésta es la página principal que verá el usuario, donde se encuentran todos los componentes, el usuario puede desplazarse por la plataforma libremente, ir a diferentes secciones como seguidores y populares, realizar diversas funciones, incluso hacer búsquedas e ir a crear un componente.





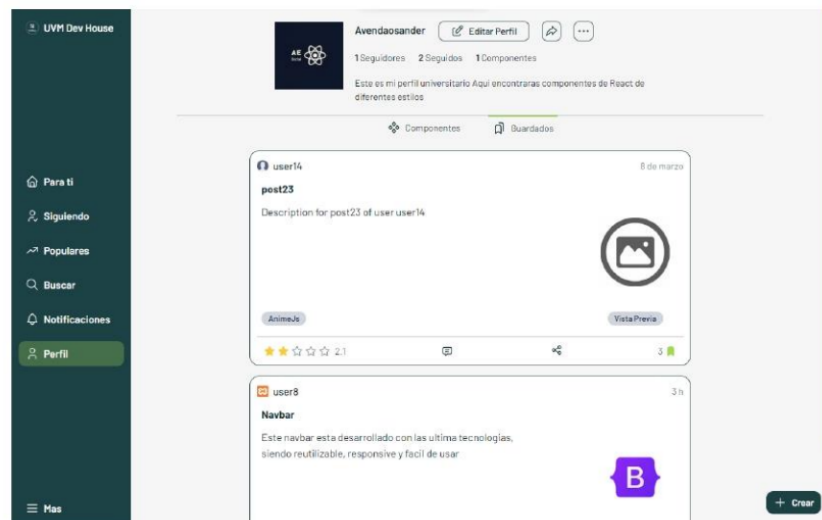
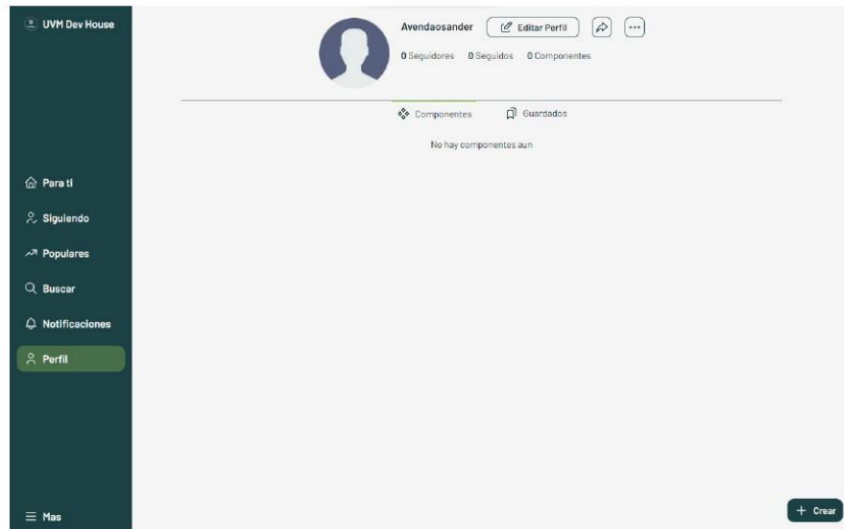
Buscar

En esta vista el usuario puede realizar búsquedas de componentes y filtrar los resultados por componente, tecnología o usuarios, incluyendo un filtro de componentes por calificaciones.



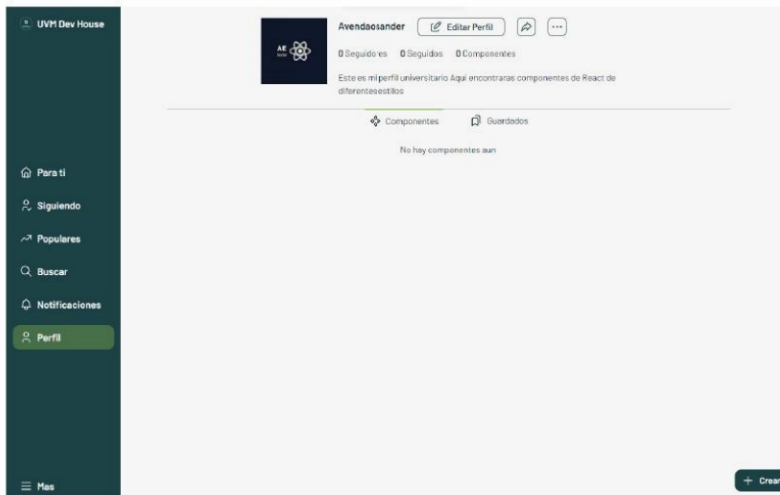
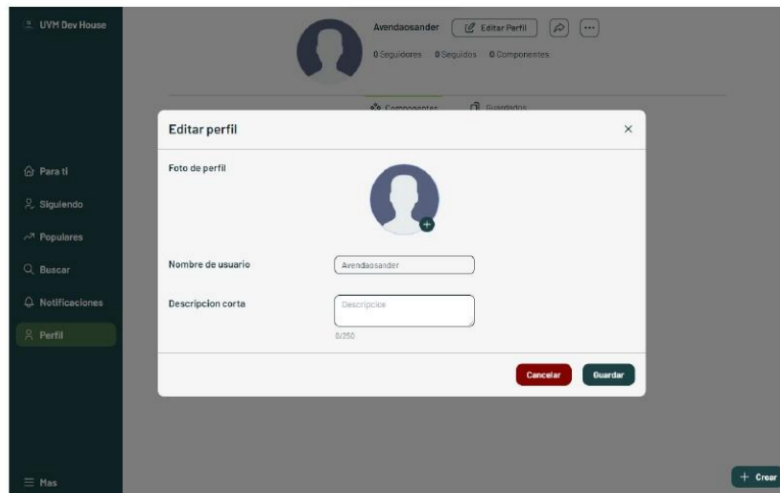
Perfil

Esta es la vista del perfil del usuario, donde puede observar cuantos seguidores tiene, cuantos usuarios sigue, ver sus componentes creados y guardados, al igual que puede editar su perfil.



Editar perfil

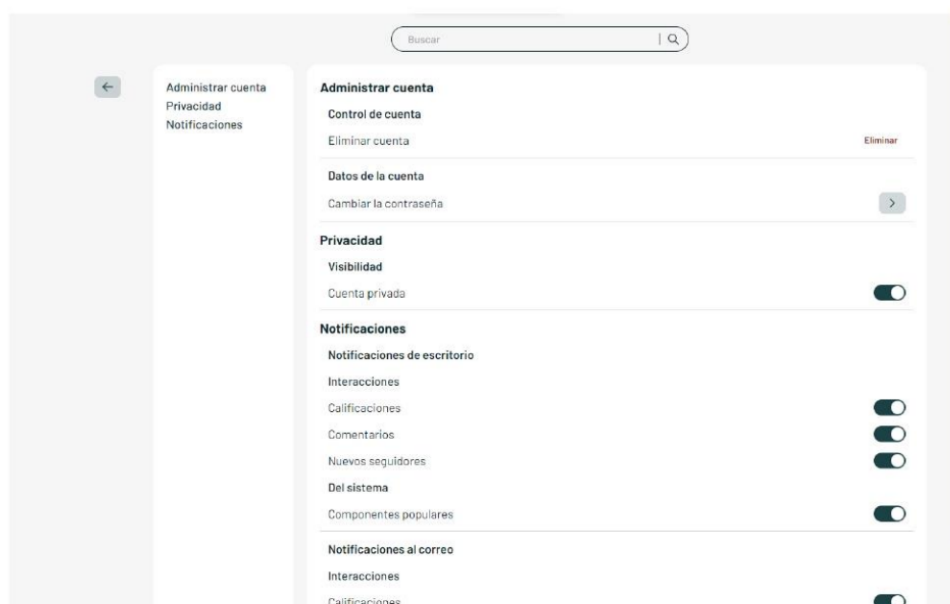
Al pulsar el botón de editar perfil, aparecerá este modal con un formulario para editar los datos modificables, permitiendo editar la foto de perfil, nombre de usuario y descripción. Una vez guardes los cambios el modal se cerrará y los cambios se reflejan de inmediato.



Configuraciones

En esta página se podrá realizar diversos ajustes, tales como: Aceptar o rechazar las notificaciones de algún tipo, cambiar la contraseña o eliminar la cuenta.

Para recuperar la contraseña, le será enviado un mensaje con un código a su correo electrónico asociado a su cuenta, y solo deberá copiarlo y pegarlo en la página a la que se le redirigirá, para confirmar su autenticación y procederá a cambiar su contraseña.



Crear un componente

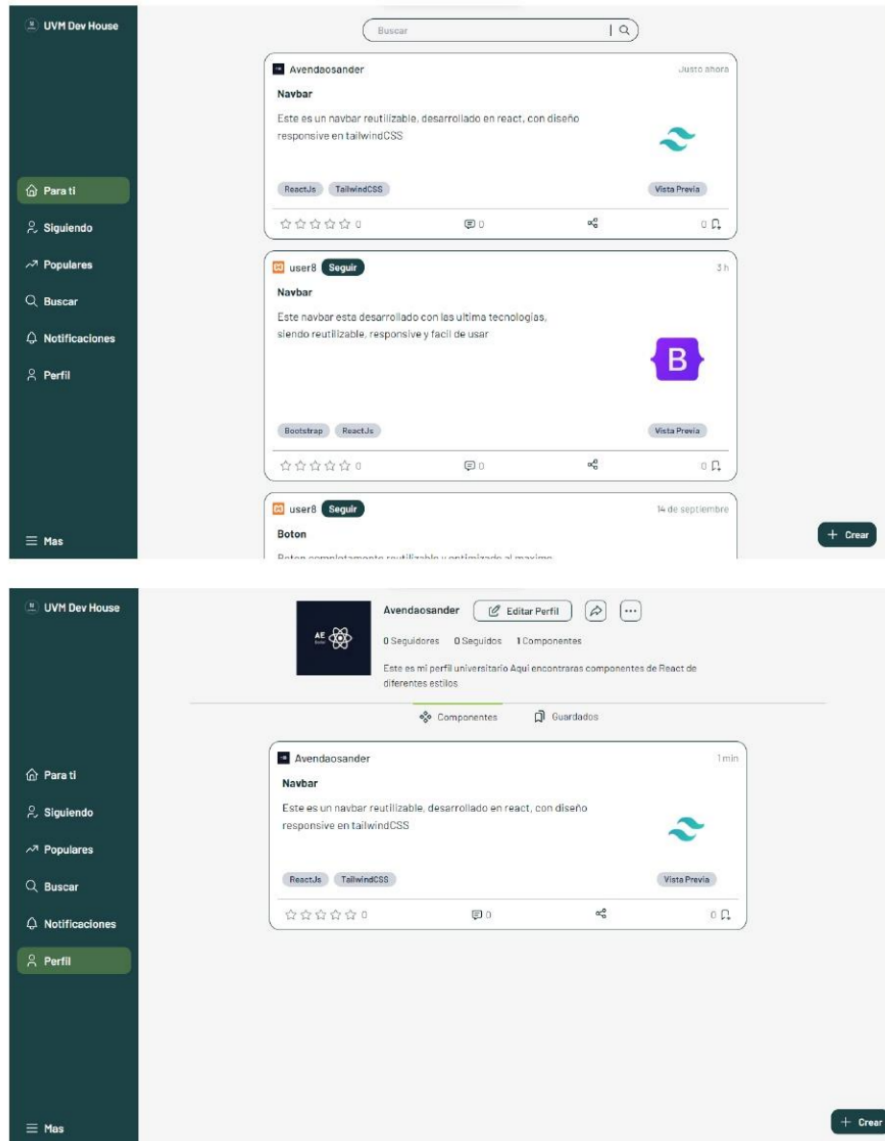
Esta es una de las vistas más importantes, donde puedes crear un componente, para ello debes rellenar todos los campos del formulario:

- Título
- Descripción
- Imagen del componente
- Tecnologías usadas (Al menos 1)
- Archivos (Al menos 1)

The screenshot shows the 'Comparte tu componente' form with several annotations. The form fields are: Título, Descripción, and un campo de texto con un icono de enlace. Below these are 'Stack' (with a dropdown menu), 'Tecnologías' (with a search input and a dropdown), and 'Archivos' (with a 'Subir archivo' button and a 'Publicar' button). Annotations with arrows point to: 1) 'Título', 2) 'Descripción', 3) 'Imagen del componente' (represented by a camera icon), 4) 'Tecnologías' dropdown, 5) 'Tecnologías' search input, 6) 'Archivos' section, and 7) 'Publicar' button. To the right of the form, there are three text boxes: 'Puedes seleccionar las tecnologías que quieras, pero tu componente deberá tener al menos 1 tecnología', 'Puedes añadir imágenes a tu componente', and 'Puedes añadir todos y cada uno de los archivos que quieras, siempre en formato de archivo de código fuente'.

The screenshot shows the 'Comparte tu componente' form with filled data. The 'Título' field contains 'React'. The 'Descripción' field contains 'Este es un ejemplo de un componente de React con un estado responsable de manejar el estado'. The 'Stack' dropdown is set to 'Tecnologías'. The 'Tecnologías' search input contains 'ReactJS'. The 'Archivos' section shows 'Subir archivo' and 'Publicar' buttons. The 'Publicar' button has a red notification icon.

Una vez creas el componente te lleva a la página principal y podrás verlo, al igual que en tu perfil.



Componente

Esta es la página de un componente, donde podrás encontrar toda la información del componente, calificarlo, copiar el código o descargarlo, y comentarlo.

UVM Dev House

Navbar

Este es un navbar reutilizable, desarrollado en react, con diseño responsive en tailwindCSS

Desarrollado por: [Avendoosander](#)

Fecha de publicación: 10 min

Stack: [React.js](#) [TailwindCSS](#)

Calificación: ★★★★☆ 3 calificaciones

[Calificar](#) [Compartir](#) [Editar](#) [Eliminar](#)

1727877361436-Button.tsx [Descargar archivo](#)

```
import clsx from "clsx"
import React from "react"
import { twMerge } from "tailwind-merge"
import { Variants } from "../variants"

interface Props extends React.ButtonHTMLAttributes<HTMLButtonElement> {
  variant?: "solid" | "outline" | "flat" | "ghost"
  color?: "primary" | "secondary" | "destructive"
  size?: "sm" | "md" | "lg"
  marginX?: "none" | "auto"
}

export default function Button({ variant, color, size, marginX, ...props }: Props) {
  return (
    <button
      className={clsx(
        "border rounded px-4 py-2 text-sm font-medium",
        variant === "solid" ? "border-transparent" : "border border-gray-300",
        color === "primary" ? "text-primary" : "text-gray-700",
        size === "sm" ? "text-xs" : "text-sm",
        marginX === "auto" ? "margin-left: auto; margin-right: auto" : ""
      )}
      {...props}
    >
      {props.children}
    </button>
  )
}
```

1727877361431-Navbar.tsx [Descargar archivo](#)

```
"use client"
import React, { useEffect, useState } from "react"
import Button from "../Button"
import {
  BellIcon,
  BookmarksIcon,
```

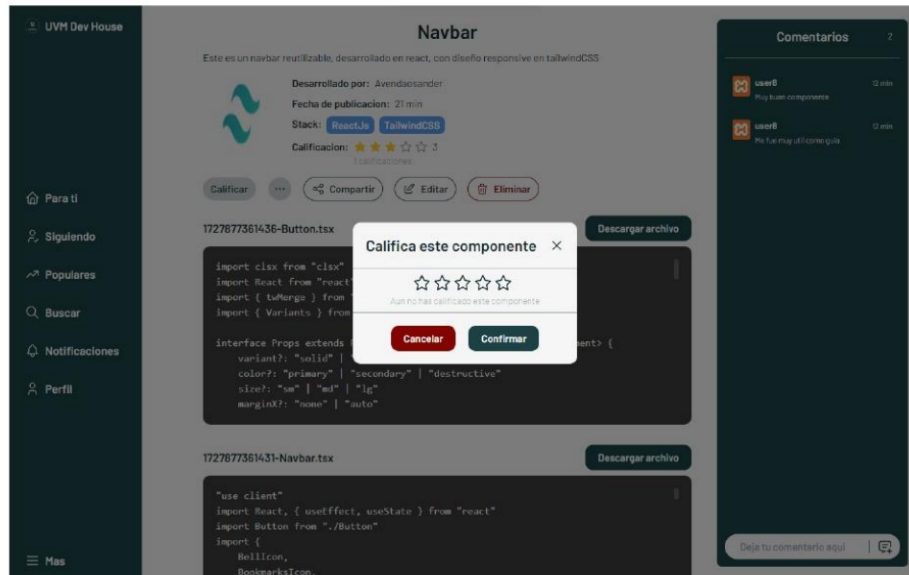
Comentarios 2

[user8](#) 1 min
Muy buen componente

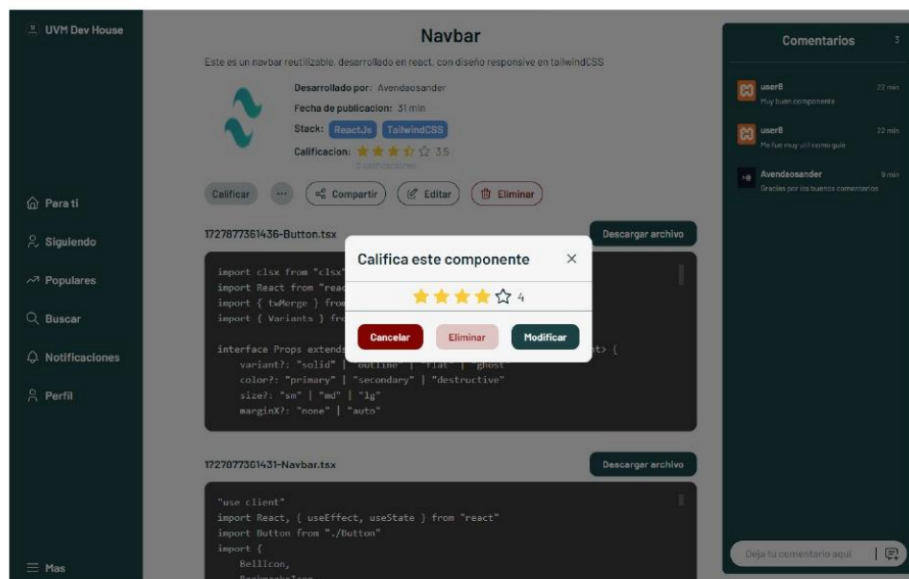
[user8](#) Justo ahora
Me fue muy útil como guía

Deja tu comentario aquí

Puede calificar el componente, si ya lo ha calificado puede eliminar la calificación o modificarla



The screenshot shows a component page for 'Navbar' with a rating modal open. The modal title is 'Califica este componente' and it displays five empty stars. Below the stars, it says 'Aun no has calificado este componente'. There are two buttons: 'Cancelar' (red) and 'Confirmar' (green). The background shows the component's code and a sidebar with navigation options like 'Para ti', 'Siguiendo', 'Populares', 'Buscar', 'Notificaciones', and 'Perfil'.



The screenshot shows the same component page for 'Navbar'. The rating modal now displays four yellow stars and one empty star, with the number '4' next to them. The buttons are now 'Cancelar' (red), 'Eliminar' (pink), and 'Modificar' (green). The background content is identical to the previous screenshot.

Puede realizar comentarios

The screenshot shows a component page for 'Navbar' on a platform called 'UVM Dev House'. The page includes a description, author information (Aveadosander), publication date (22 min), stack (React.js, TailwindCSS), and a rating of 3.5 stars. Below the metadata are two code snippets with 'Descargar archivo' buttons. The first snippet is for 'Button.tsx' and the second is for 'Navbar.tsx'. On the right side, there is a 'Comentarios' (Comments) sidebar with three comments from 'user8' and 'Aveadosander'. At the bottom of the sidebar is a text input field with the placeholder 'Deja tu comentario aquí' and a submit button.

Si el usuario es el autor del componente podrá editarlo o eliminarlo.

The screenshot shows the 'Edita tu componente' (Edit your component) modal for the 'Navbar' component. The modal has a close button (X) in the top right corner. It contains the following fields:

- Título:** Navbar
- Descripción:** Este es un navbar reutilizable, desarrollado en react, con diseño responsive en tailwindCSS. Below this is a character count '80/500'.
- Stack:** Tecnologías. Below this are tags for 'React.js' and 'TailwindCSS'.
- Archivos:** A list of files: '1727877381438-Button.tsx' and '1727877381431-Navbar.tsx', each with a delete icon.
- Subir archivo:** A button with a plus sign and the text 'Subir archivo'.
- Buttons:** 'Cancelar' (Cancel) and 'Modificar' (Modify) buttons at the bottom.

 The background shows the same component page as the previous screenshot, but it is dimmed.

The screenshot shows a web interface for a code-sharing platform. The main content area displays a component titled "Navbar" with the following details:

- Este es un navbar reutilizable, desarrollado en react, con diseño responsive en tailwindCSS
- Desarrollado por: Avendoosander
- Fecha de publicación: 38 min
- Stack: ReactJS, TailwindCSS
- Calificación: 3 calificaciones, 3.5

Below the component details are buttons for "Calificar", "Compartir", "Editar", and "Eliminar". A confirmation dialog is overlaid on the screen with the text "Esta seguro de eliminar este componente?" and "Cancelar" and "Confirmar" buttons.

The interface also includes a left sidebar with navigation options: "Para ti", "Siguiendo", "Populares", "Buscar", "Notificaciones", and "Perfil". The right sidebar shows a "Comentarios" section with three comments from "user8" and "Avendoosander".

```
import clsx from 'clsx'
import React from 'react'
import { twMerge } from 'tailwind-merge'
import { VariantProps } from 'class-variance-authority'

interface Props extends React.ButtonHTMLAttributes<React.ElementType> {
  variant?: 'solid' | 'outline' | 'flat' | 'ghost'
  color?: 'primary' | 'secondary' | 'destructive'
  size?: 'sm' | 'md' | 'lg'
  marginX?: 'none' | 'auto'
}

export default function Button({
  variant = 'solid',
  color = 'primary',
  size = 'md',
  marginX = 'none',
  ...props
}: Props) {
  return (
    <button
      className={twMerge(
        'inline-block',
        'border',
        'border-transparent',
        'border-width: 2px',
        'border-radius: 0.5em',
        'padding: 0.5em 1em',
        'font-size: 0.9em',
        'font-weight: bold',
        'text-decoration: none',
        'transition: border-color 0.15s ease-in-out, background-color 0.15s ease-in-out, color 0.15s ease-in-out',
        variant === 'ghost' ? 'border: none' : '',
        variant === 'solid' ? `background-color: ${color}` : '',
        variant === 'outline' ? `border-color: ${color}` : '',
        variant === 'flat' ? `background-color: ${color}, border-color: ${color}` : '',
        variant === 'ghost' ? `color: ${color}` : ''
      )}
      {...props}
    >
      {props.children}
    </button>
  )
}
```

```
"use client"
import React, { useEffect, useState } from "react"
import Button from "../Button"
import {
  BellIcon,
  BookmarkIcon,
```

ANEXO 7
MANUAL TECNICO

Manual técnico de la Plataforma Social de Componente Open Source

MANUAL TÉCNICO DE UVM DEV HOUSE

Plataforma Social de Componentes Open Source

Desarrollado por:

Alexander Avendaño

Introducción

La presente plataforma social de componentes open source tiene como objetivo facilitar el intercambio y la colaboración entre desarrolladores dentro de la comunidad uvemista. Utilizando tecnologías como Next.js, TypeScript, TailwindCSS, Express, GraphQL, Prisma y MySQL, se ha creado una infraestructura que permite a los usuarios compartir, buscar y colaborar en componentes de software de manera eficiente. La plataforma centraliza los recursos open source, optimizando la experiencia de los usuarios al promover la innovación, el aprendizaje y el trabajo en equipo dentro de un entorno seguro y accesible.

Tecnologías utilizadas:

Next.js: Framework de React para el desarrollo de aplicaciones web, utilizado para construir la interfaz de usuario.

TypeScript: Lenguaje de programación que extiende JavaScript, utilizado para mejorar la robustez del código.

TailwindCSS: Framework de CSS para el diseño visual y la estilización de la interfaz.

ExpressJs: Framework de Node.js para la construcción del backend y la creación de API RESTful.

Express.js: Framework minimalista para Node.js que facilita la creación de aplicaciones web y APIs, utilizado en el backend del proyecto.

GraphQL: Lenguaje de consulta para API utilizado para optimizar la comunicación entre el frontend y backend.

Prisma: ORM utilizado para interactuar con la base de datos y gestionar las consultas en el backend.

MySQL: Sistema de gestión de bases de datos utilizado para almacenar y gestionar la información de la plataforma.

Requisitos del Sistema

Software necesario:

- Sistema operativo: Compatible con Linux, macOS, o Windows.
- Node.js: Versión 20.10 o superior.
- NPM: Se recomienda usar NPM versión 10.2.3 o superior.
- Base de datos: MySQL para el almacenamiento de datos, se recomienda la versión 8.0.31.
- Express.js: Framework para el backend, integrado con GraphQL.
- Prisma: ORM utilizado para interactuar con la base de datos.

Instalación

Para la instalación se debe escoger una carpeta destino para almacenar el proyecto, luego abrir la terminal desde esa carpeta y ejecutar el siguiente comando:

- `git clone https://github.com/Avendaosander/Plataforma-social.git`

Se clonará el repositorio que contiene 2 carpetas (frontend y server), en ambas deberá instalar la dependencia necesaria ejecutando el siguiente comando:

- `npm install`

Seguido de eso, hay que configurar las variables de entorno en ambas carpetas, para ello hay que crear un archivo `.env.local` en el frontend y otro archivo `.env` en el server, en estos archivos se debe modificar “`http://localhost:4000`” por el dominio establecido en el servidor que utilices, al igual que “`http://localhost:3000`” por el dominio del frontend que uses.

Variables de entorno (frontend)

API_ROUTE_GRAPHQL=http://localhost:4000/graphql

NEXT_PUBLIC_API_ROUTE_SUBSCRIPTION=http://localhost:4000/subscription

NEXT_PUBLIC_API_ROUTE_AVATAR=http://localhost:4000/uploads/avatar/

NEXT_PUBLIC_API_ROUTE_PREVIEW=http://localhost:4000/uploads/preview/

NEXT_PUBLIC_API_ROUTE_FILE=http://localhost:4000/uploads/files/

NEXT_PUBLIC_API_ROUTE_UPDATE_FILE=http://localhost:4000/uploads/updatefil

e/

NEXTAUTH_URL=http://localhost:3000

NEXTAUTH_SECRET='TG_PS_OS'

*NEXT_PUBLIC_VAPID_PUBLIC_KEY='BOiJ5WPeQjgEs6X5NEIGqKmhrdRaMG8ada
fP1aZ1dhPP16FhI4JytJ12E7zF3GcYQhX68KmIFrKPKUk4IDQb7Gg'*

Variables de entorno (server)

DATABASE_URL="mysql://root:PASSWORD@localhost:PORT/plataforma-social"

FRONTEND_URL="http://localhost:3000/"

PORT=4000

HASH_SALT=10

SECRETKEY=TG_PS_OS

*PUBLIC_VAPID_KEY=BOiJ5WPeQjgEs6X5NEIGqKmhrdRaMG8adafP1aZ1dhPP16F
hI4JytJ12E7zF3GcYQhX68KmIFrKPKUk4IDQb7Gg*

PRIVATE_VAPID_KEY=ZrLev6NVgXi5_hqE_hSnTmXb_BwuSSa33_7MMQ4Aptk

Migración de la base de datos:

Para migrar la base de datos se debe ejecutar desde la terminal de la carpeta *server* los siguientes comandos:

- `npm run generate`
- `npm run migrate`

Por último, para el uso de la plataforma se debe ejecutar los siguientes comandos desde la terminal de ambas carpetas (frontend y server):

Para compilar el proyecto: (frontend y server)

- `npm run build`

Para desarrollo: (frontend y server)

- `npm run dev`

Para producción: (frontend y server)

- `npm run start`

Estructura del proyecto (frontend)

- ***/app***: Contiene las rutas de la aplicación desarrollada en Next.js. Aquí se definen las páginas del frontend, con rutas dinámicas y estáticas que se corresponden a las URL del sitio. Los archivos dentro de esta carpeta siguen el sistema de enrutamiento basado en archivos de Next.js.
 - ***./api***: Contiene las rutas y controladores de la API para manejar solicitudes desde el frontend hacia el backend. Se utiliza principalmente para la implementación de **Express.js** en el servidor backend, integrando los servicios de GraphQL.
-

- *./auth/[...nextauth]*: Contiene el archivo de configuración para el servicio de autenticación.
 - *./components*: Incluye todos los componentes reutilizables de la interfaz de usuario (UI). Esta se subdivide en carpetas para una mejor organización, cada carpeta corresponde a la página donde se encuentra el componente. Cada componente está diseñado para ser modular y reutilizable en diferentes partes de la aplicación, como botones, formularios, listas, etc.
 - *./home*: Esta es la carpeta principal, ya que es la página de inicio el cual contiene las diferentes paginas correspondientes de la plataforma, incluyendo el layout de toda la plataforma.
 - *./lib*: Esta carpeta contiene la lógica del frontend y las configuraciones esenciales para manejar el cliente de Apollo, la sesión, los esquemas de GraphQL y otros archivos de utilidad.
 - *./graphql*: Contiene los esquemas organizados para las queries y mutaciones de GraphQL, estructurados en diferentes archivos para mantener la organización de los CRUD.
 - *./types*: Contiene todos los types necesarios para el proyecto, asegurando que los datos manejados sean tipados correctamente en el frontend.
 - *./webpush*: Contiene el archivo necesario para suscribirse al servicio de web-push, utilizado para manejar las notificaciones push en el frontend.
-

- */apollo-wrapper.tsx*: Componente que envuelve la aplicación con el cliente Apollo para manejar las consultas y mutaciones de GraphQL.
 - */session-wrapper.tsx*: Componente que gestiona el estado de la sesión del usuario.
 - */SessionProvider.tsx*: Proveedor de sesión para la autenticación de usuarios y manejo de datos relacionados con la sesión.
 - */client.ts*: Cliente de Apollo Client configurado para manejar las consultas y mutaciones de GraphQL en el frontend.
 - */logic.ts*: Archivo que contiene las funciones de lógica utilizadas en la aplicación para manejar procesos de negocio y cálculos lógicos necesarios.
 - */login*: Contiene la página de inicio de sesión.
 - */register*: Contiene la página para registrarse.
 - */store*: Contiene los diferentes archivos de store para almacenar datos globales usando zustand
 - */public*: Archivos estáticos que son accesibles directamente desde el navegador, como imágenes, íconos, y otros recursos que no requieren procesamiento dinámico.
 - *sw.js*: Este es el archivo de configuración de las notificaciones Push
-

Estructura del proyecto (server)

- /prisma: Esta carpeta contiene todo lo relacionado con Prisma, que se usa como ORM para la base de datos MySQL. Aquí se definen los modelos y las migraciones:
 - /migrations: Carpeta que almacena las migraciones de la base de datos, las cuales permiten mantener la estructura de la base de datos actualizada con los modelos definidos.
 - schema.prisma: Archivo principal de Prisma donde se definen los modelos de datos y las relaciones entre tablas de la base de datos.
 - /src: El directorio principal donde se encuentra el código fuente del backend.
 - /graphql: Carpeta que contiene los esquemas de GraphQL y resolvers que definen cómo manejar las consultas, mutaciones y suscripciones en el servidor.
 - /helpers: Incluye archivos de utilidad y funciones que ayudan a realizar tareas comunes, como seeding de datos o configuración de web-push.
 - /lib: Archivos reutilizables que definen constantes y funciones comunes.
 - /routes: Aquí se encuentran las rutas del backend que manejan las diferentes peticiones HTTP, organizadas en controladores.
 - /controllers: Carpeta con los controladores que manejan la lógica de las rutas del backend.
 - /index.ts: Archivo principal que define las rutas del backend.
 - /types.ts: Archivo que define los tipos utilizados en el servidor.
-

- uploads: destinada a almacenar archivos que los usuarios suben al servidor, como avatares, previews y documentos.

Servicios Externos

- API Web-Push:

Para integrar notificaciones push en la plataforma, se utiliza la API Web-Push. Este servicio permite enviar notificaciones a los usuarios incluso cuando la aplicación no está activa.

La configuración del servicio implica los siguientes pasos:

1. Registro del Service Worker: Se debe registrar un service worker en el cliente, que manejará las notificaciones. Este archivo debe contener la lógica para recibir y mostrar las notificaciones.
 2. Suscripción del Usuario: Al acceder a la aplicación, se solicita al usuario permiso para recibir notificaciones. Una vez otorgado, se crea una suscripción que incluye información sobre el endpoint y las claves necesarias para enviar notificaciones.
 3. Servidor de Envío: En el backend, se utiliza la librería *web-push* para enviar notificaciones a los usuarios suscritos. Esto incluye la configuración de los parámetros de la notificación (título, cuerpo, icono, etc.) y la autenticación con las claves de VAPID.
 4. Manejo de Errores: Se deben implementar mecanismos para manejar posibles errores durante el envío de las notificaciones, asegurando que la experiencia del usuario no se vea afectada.
-

- Gmail SMTP para envío de emails:

La plataforma utiliza el servicio SMTP de Gmail para el envío de correos electrónicos, como notificaciones de registro o recuperación de contraseña. La configuración del correo en el backend se realiza mediante la librería Nodemailer, y se lleva a cabo de la siguiente manera:

1. Instalación de Nodemailer: Se instala la librería Nodemailer en el proyecto utilizando npm o yarn.

2. Configuración del Transportador: Se crea un transportador SMTP con las credenciales de una cuenta de Gmail. Esto incluye el correo electrónico y la contraseña de la cuenta que se utilizará para enviar los correos.

3. Definición del Correo: Al preparar el correo, se especifican detalles como el remitente, destinatario, asunto y cuerpo del mensaje. También se pueden agregar archivos adjuntos si es necesario.

4. Envío y Manejo de Respuestas: Se envía el correo utilizando el método *sendMail* de Nodemailer, y se implementa la lógica para manejar respuestas exitosas o errores durante el proceso de envío, garantizando que el usuario reciba retroalimentación sobre el estado de su solicitud.

Estas integraciones permiten mejorar la interacción con los usuarios, facilitando la comunicación y asegurando que reciban notificaciones importantes relacionadas con su uso de la plataforma.
