

UNAD Emprende Móvil: Tu Vitrina Virtual al Alcance de tu Mano

Meyerson Stiven Álvarez Castro

Jhon Edison Diaz Arias

Universidad Nacional Abierta Y A Distancia UNAD

Escuela de Ciencias Básicas Tecnología e Ingeniería

Septiembre 10 de 2024

UNAD emprende móvil: tu vitrina virtual al alcance de tu mano

Meyerson Stiven Álvarez Castro

Jhon Edison Diaz Arias

Director Proyecto Aplicado.

Ing. Iván Giovanni Quesada Bonilla

Universidad Nacional Abierta Y A Distancia UNAD

Escuela de Ciencias Básicas Tecnología e Ingeniería

Ingeniería de Sistemas

Septiembre 10 de 2024

Agradecimientos

En primer lugar, queremos expresar nuestro más profundo agradecimiento a nuestros queridos padres, les debemos todo lo que somos. Su apoyo incondicional, han sido el motor que nos ha impulsado a seguir adelante. Gracias por creer en nosotros y por estar siempre a nuestro lado.

Asimismo, queremos extender nuestro sincero agradecimiento a nuestros profesores, Iván Giovanni Quesada, Daniel Guzmán y Ángela María Ospina quienes con su dedicación y conocimiento han sido pilares fundamentales en el desarrollo de nuestra formación profesional y en especial en este proyecto de grado. Agradecemos su compromiso y su enfoque riguroso que nos han ayudado a desarrollar una mentalidad crítica y analítica. Sus enseñanzas han dejado una huella imborrable en nuestra vida académica. También agradecemos su valiosa asesoría en este proyecto.

Resumen

El proyecto "UNAD Emprende Móvil: Tu vitrina virtual al alcance de tu mano" es una derivación del proyecto "Vitrina Virtual" que se lanzó con la intención de apoyar a la comunidad emprendedora de la Universidad Nacional Abierta y a Distancia (UNAD) a través de una plataforma web (Vitrina Virtual) y una aplicación móvil (UNAD Emprende Móvil), este proyecto busca ampliar la visibilidad de productos y servicios innovadores desarrollados por estudiantes, egresados y docentes de la UNAD, ayudándoles a crecer en un mercado cada vez más monopolizado por la tecnología. La herramienta es sencilla, pero ofrece una solución viable a la inexistencia del apoyo a los emprendedores.

Se utilizaron tecnologías de punta como HTML, CSS y JavaScript, junto con metodologías ágiles para su desarrollo, su interfaz es intuitiva lo que simplifica el proceso de registrar los emprendimientos y para los usuarios es fácil navegar a través de las categorías en busca de productos o servicios de su interés.

Palabras clave: Software, Android, HTML, CSS, JavaScript, UNAD, Bootstrap, APP, Vitrina Virtual, Visual Code, web.

Abstract

The "UNAD Emprende Móvil: Tu vitrina virtual al alcance de tu mano" project is a derivation of the "Vitrina Virtual" project that was launched with the intention of supporting the entrepreneurial community of the National Open and Distance University (UNAD) through a web platform (Vitrina Virtual) and a mobile application (UNAD Emprende Móvil), this project seeks to expand the visibility of innovative products and services developed by UNAD students, graduates and teachers, helping them grow in a market increasingly monopolized by technology. The tool is simple, but it offers a viable solution to the lack of support for entrepreneurs.

Cutting-edge technologies such as HTML, CSS and JavaScript were used, along with agile methodologies for its development, its interface is intuitive, which simplifies the process of registering ventures, and it is easy for users to navigate through the categories in search of products or services of your interest

Keywords: Software, Android, HTML, CSS, JavaScript, UNAD, Bootstrap, APP, Vitrina Virtual, Visual Code, web.

Tabla de Contenido

Introducción.....	13
Planteamiento Del Problema	14
Pregunta De Investigación.....	15
Justificación.....	16
Objetivos.....	17
Objetivo General.....	17
Objetivos Específicos.....	17
Marco Conceptual y Teórico.....	18
Aplicaciones móviles	18
Comercio Electrónico	19
Frontend y Backend	19
Frameworks.....	20
Marco Legal.....	21
Marco Contextual	22
Localización.....	22
Número Emprendedores	23
Contexto de los Emprendedores de la UNAD	23
Metodología.....	24
Enfoque	24

Población.....	24
Metodología de Desarrollo De Software Kanban.....	24
Principios de Kanban.	25
Fases de desarrollo Teórico del sitio web Vitrina Virtual	26
Fase De Análisis:.....	26
Fase De Diseño:	27
Fase De Desarrollo:.....	28
Fase De Pruebas:.....	29
Fase De Implementación:	30
Fase De Mantenimiento y Actualizaciones:.....	32
Catálogo De Requisitos.	33
App Vitrina Virtual.....	33
Vitrina Virtual Web.	33
Requisitos de Datos.....	34
Requisitos De Interfaz.	35
Inicio De Sesión.....	35
Cierre De Sesión.	35
Cliente Web O browser.	35
Microsoft Edge.	36
Mozilla Firefox.....	36

Safari.	36
Chrome 5.	36
Interacción cliente / aplicación.	36
Diseño y comodidad.	36
Mostrar información Acerca De Nosotros.	37
Información de emprendimientos	37
Tiempo de respuesta.	37
Diagrama De Gantt.	38
Fase De Diseño Grafico.	39
Logotipo y Paleta de Colores	39
Tipografía y Estilo de Imagen.	40
Diseño de Interfaz de Usuario (UI).	41
Fase De Desarrollo Del Código Del Software Web	46
Declaración del Documento y Encabezado Vitrina Virtual V2 (<head>).	47
Cuerpo del Documento (<body>).	47
Pie de Página (<footer>).	48
Copyright	48
Descripción de los Scripts Utilizados en el Proyecto Vitrina Virtual V2.	48
Definición y Gestión de Productos.	49
Filtrado y Renderizado Dinámico de Productos	50

Implementación de Carrusel de Imágenes	50
Funcionalidad de Pantalla Completa para Imágenes.....	51
Sistema de Categorías Para Los Emprendimientos	52
Script de Formulario de Registro	53
Fase De Desarrollo Del Del Código Del Aplicativo Móvil.....	56
Importaciones y Configuración Inicial	56
Clase Principal MainActivity.....	57
Método isConnected().....	58
Método onCreate(Bundle savedInstanceState).....	59
Métodos onResume() y onPause().....	60
Método setupWebView()	61
Método onBackPressed()	62
Clase Interna NetworkChangeReceiver	63
Fase de Pruebas.....	64
Resultados	64
Referencias Bibliográficas.....	67

Lista De Tablas

Tabla 1 <i>Zonas donde se llevará a cabo el proyecto aplicado.</i>	22
Tabla 2 <i>Análisis teórico del desarrollo del software.</i>	26
Tabla 3 <i>Diseños y prototipos del software.</i>	27
Tabla 4 <i>Fase de desarrollo del software</i>	29
Tabla 5 <i>Pruebas funcionales de usabilidad y compatibilidad del software</i>	30
Tabla 6 <i>Implementación del software</i>	31
Tabla 7 <i>Mantenimiento y actualizaciones periódicas</i>	32
Tabla 8 <i>Requisitos mínimos App vitrina virtual</i>	33
Tabla 9 <i>Requisitos mínimos vitrina virtual web</i>	33
Tabla 10 <i>Requisitos de emprendimientos en datos de programación</i>	34
Tabla 11 <i>Fuentes tipográficas de la página web</i>	40

Lista de Figuras

Figura 1 <i>Tablero Kanban del proyecto aplicado</i>	25
Figura 2 <i>Paleta de colores de la UNAD</i>	39
Figura 3 <i>Diseños de Usuario</i>	41
Figura 4 <i>Diseño gráfico Índice</i>	42
Figura 5 <i>Diseño Gráfico Sobre Nosotros</i>	43
Figura 6 <i>Diseño gráfico emprendimientos</i>	44
Figura 7 <i>Diseño Gráfico Haz Parte</i>	45
Figura 8 <i>Código índice software web</i>	46
Figura 9 <i>Plantilla de emprendimientos en JavaScript</i>	49
Figura 10 <i>Función renderProducts</i>	50
Figura 11 <i>Código carrusel de ítems</i>	51
Figura 12 <i>Full screen código principal</i>	52
Figura 13 <i>Código de categorías de la vitrina virtual web</i>	53
Figura 14 <i>Formulario de registro de emprendimientos</i>	54
Figura 15 <i>Script automático de código de emprendimientos</i>	55
Figura 16 <i>Importe de librerías Android Studio</i>	57
Figura 17 <i>Clase MainActivity Android Studio</i>	58
Figura 18 <i>Boolean de conexión isConnected</i>	59
Figura 19 <i>Método onCreate principal</i>	60
Figura 20 <i>Metodo onResume de segundo plano</i>	61
Figura 21 <i>Código setupWebView configuraciones de la app</i>	62
Figura 22 <i>Código onBackPressed</i>	63

Figura 23 <i>Prueba de rendimiento en GTmetrix</i>	64
Figura 24 <i>Ventana ver más con redes sociales</i>	65
Figura 25 <i>Índex vista desde la app</i>	66

Introducción

En un mundo cada vez más digitalizado, la necesidad de plataformas accesibles que promuevan el emprendimiento se vuelve crucial. La Universidad Nacional Abierta y a Distancia (UNAD) ha identificado esta necesidad y ha desarrollado el proyecto "UNAD Emprende Móvil: Tu Vitrina Virtual al Alcance de tu Mano". Esta iniciativa busca ofrecer un espacio digital que facilite la visibilidad y promoción de los productos y servicios generados por su comunidad.

La vitrina virtual se erige como una solución innovadora para los emprendedores de la UNAD, permitiéndoles exhibir sus iniciativas de manera efectiva. A través de una aplicación móvil y web, se busca no solo conectar a los emprendedores con un público más amplio, sino también fomentar el intercambio de conocimientos y experiencias, fortaleciendo así el ecosistema emprendedor.

El desarrollo de esta aplicación se basa en principios de accesibilidad e inclusión, garantizando que todos los miembros de la comunidad UNADISTA, sin importar sus limitaciones tecnológicas, puedan beneficiarse de las oportunidades que esta plataforma ofrece. Utilizando tecnologías modernas y metodologías ágiles, el proyecto se estructura en fases que aseguran un desarrollo eficiente y de alta calidad.

Planteamiento Del Problema

En la actualidad, los emprendedores de la Universidad Nacional Abierta y a Distancia (UNAD) enfrentan desafíos significativos para dar a conocer y comercializar sus productos y servicios en un mercado cada vez más digitalizado y competitivo tecnológicamente. A pesar de la diversidad y el potencial de los emprendimientos desarrollados por los estudiantes, egresados y miembros de la comunidad UNADISTA, la falta de visibilidad y accesibilidad a plataformas de promoción reguladas, promocionadas y administradas directamente por la universidad limita sus oportunidades de crecimiento y expansión en este nuevo mercado digital.

En este contexto, la propuesta de una aplicación móvil como responsive de la vitrina virtual aborda directamente estas limitaciones, ofreciendo una plataforma adicional y accesible que facilita la conexión entre los emprendedores UNADISTA y un público más amplio, la iniciativa de esta aplicación móvil trae consigo una reestructuración del proyecto vitrina virtual, haciendo que la página web sea actualizada y mejorada considerablemente, trayendo consigo mejores opciones que la primera versión. Esta iniciativa no solo busca amplificar la visibilidad y el acceso a los emprendimientos, sino que también pretende fortalecer las redes de colaboración y apoyo dentro de la comunidad universitaria y con el sector empresarial, contribuyendo así al crecimiento económico y al fortalecimiento del ecosistema emprendedor de la UNAD.

Pregunta De Investigación

¿Cómo puede la implementación de una aplicación móvil de la vitrina virtual específicamente diseñada para la comunidad UNADISTA mejorar la visibilidad, el acceso a mercados y el intercambio de conocimientos entre los emprendedores, contribuyendo así al crecimiento y sostenibilidad de sus proyectos en el entorno digital actual?

Justificación

La implementación de una App de la vitrina virtual dedicada a los emprendimientos de la comunidad UNADISTA se justifica por la necesidad creciente de ofrecer un espacio digital multiplataforma dentro de la universidad que facilite la visibilidad, promoción y conexión de los proyectos emprendedores con un público más amplio. En el entorno cada vez más digitalizado que se vive, donde la competencia por la atención del consumidor es intensa y fundamental, contar con múltiples herramientas proporcionadas por la misma universidad que les permita a los emprendedores destacar y acceder a oportunidades de mercado, sintiendo el acompañamiento y apoyo de la UNAD.

Este proyecto no solo busca ofrecer una solución tecnológica para la promoción de productos y servicios desarrollados por los miembros de la UNAD, sino que también aspira a fomentar una cultura de emprendimiento e innovación dentro de la comunidad universitaria. Al brindar un espacio para el intercambio de conocimientos y experiencias, la vitrina virtual se convierte en una herramienta clave para el desarrollo profesional y personal de los emprendedores, promoviendo la colaboración y el apoyo mutuo entre los miembros de la comunidad, fomentando más esa cultura emprendedora, en donde todos pueden dejar el miedo a empezar su emprendimiento.

Objetivos

Objetivo General

Desarrollar una App de la vitrina virtual haciendo uso de las tecnologías de desarrolló WEB y móvil, para los emprendedores de la comunidad UNADISTA que permita promocionar productos y servicios.

Objetivos Específicos

Analizar detalladamente los requerimientos funcionales y no funcionales de la App vitrina virtual de acuerdo con la planeación del proyecto y estudios preliminares.

Diseñar el Front End para cumplir los estándares y requerimientos identificados que subdivide el menú de la App de la vitrina.

Codificar la App según el diseño especificado en el Front End, con las funcionalidades necesarias para los emprendedores.

Realizar pruebas exhaustivas de la App para garantizar su correcto funcionamiento, seguridad de datos y tráfico en la red.

Implementar la APP de la vitrina virtual bajos los estándares de aplicativo móvil y accesibilidad, normas educativas UNADISTAS, seguridad y protección de datos.

Marco Conceptual y Teórico

Aplicaciones móviles

Desde la perspectiva del Diccionario de la Real Academia Española (RAE), el término "aplicación" se define como un "programa preparado para una utilización específica, como el pago de nóminas, el tratamiento de textos, etc." Esta definición proporciona una base para entender las aplicaciones como software diseñado para realizar tareas concretas a través de dispositivos electrónicos.

Expandiendo esta definición, una aplicación en el contexto de la tecnología móvil puede describirse como un tipo de software diseñado específicamente para operar en dispositivos portátiles como teléfonos inteligentes, tabletas y otros dispositivos similares. Estas aplicaciones están optimizadas para aprovechar las características únicas de estos dispositivos, incluyendo su portabilidad, conectividad y capacidades de interfaz táctil.

El desarrollo de aplicaciones móviles ha evolucionado considerablemente con el avance de la tecnología, adaptándose a la creciente demanda de soluciones que permitan a los usuarios gestionar diversas actividades de manera eficiente desde cualquier lugar. Estos programas, a menudo descritos como "ligeros", están diseñados para proporcionar funcionalidades específicas sin consumir excesivos recursos del dispositivo, facilitando así una experiencia de usuario ágil y efectiva.

En este contexto, las aplicaciones móviles se han convertido en herramientas esenciales para la vida cotidiana, impulsando la innovación y redefiniendo la forma en que interactuamos con la tecnología digital. A medida que la tecnología avanza, la frontera de lo que puede lograrse

con estas aplicaciones sigue expandiéndose, abriendo nuevas posibilidades para el desarrollo y la integración de funciones cada vez más complejas y sofisticadas.

Comercio Electrónico

El Comercio Electrónico es definido en la ley 527 de 1999 en el artículo 2º ítem b) como “toda operación comercial de suministro o intercambio de bienes o servicios; todo acuerdo de distribución; toda operación de representación o mandato comercial; todo tipo de operaciones financieras, bursátiles y de seguros; de construcción de obras; de consultoría; de ingeniería; de concesión de licencias; todo acuerdo de concesión o explotación de un servicio público; de empresa conjunta y otras formas de cooperación industrial o comercial; de transporte de mercancías o de pasajeros por vía aérea, marítima y férrea, o por carretera”. (Congreso de Colombia, 1999).

La vitrina virtual actúa como una multiplataforma de comercio electrónico, debido a que cuenta con una app móvil y un sitio web. Está diseñada específicamente para facilitar a los emprendedores de la UNAD publicar sus productos y servicios de sus respectivos emprendimientos, la app de la vitrina permite tanto a los clientes como emprendedores tener un acceso rápido y eficiente desde cualquier parte.

Frontend y Backend

El frontend se refiere a la parte de la aplicación que los usuarios interactúan directamente. Incluye la interfaz gráfica y todas las funcionalidades visibles, lo que hace que la experiencia del

usuario sea intuitiva y atractiva. Las tecnologías más utilizadas en el frontend son HTML para la estructura, CSS para el estilo y JavaScript para la interactividad (Crespo, 2007).

Por otro lado, el backend se ocupa de la gestión de datos y lógica del servidor. Esto incluye la gestión de bases de datos y la ejecución de operaciones que permiten que la aplicación funcione correctamente. Lenguajes como PHP, Python y Ruby son comunes en esta área, permitiendo la conexión con bases de datos y la manipulación de información (García, 2007).

Frameworks

El uso de frameworks como Bootstrap permite a los desarrolladores crear aplicaciones responsivas y adaptativas de manera más eficiente. Bootstrap, por ejemplo, proporciona un conjunto de herramientas y componentes predefinidos que facilitan el diseño y desarrollo de aplicaciones web modernas (BOSCO, 2019). Esto no solo acelera el proceso de desarrollo, sino que también garantiza que la aplicación sea estéticamente coherente y funcional en diversos dispositivos.

Marco Legal

Ley 2108 de 2021 – Internet como un servicio público esencial. Establece a internet como un servicio público de telecomunicaciones con carácter esencial para asegurar la prestación de servicios. (Ministerio DE Tecnologías).

Ley 2069 de 2020 - Promueve impulsar el emprendimiento en el país. Esta ley establece un marco regulatorio que propicia el crecimiento y consolidación de los emprendedores.

Ley 1581 de 2012 - Ley General de Protección de Datos Personales. Esta ley tiene como objetivo el desarrollar un derecho constitucional de las personas a reconocer y rectificar la información de estos mismos que se halla recopilado en bases de datos o internet.

Ley 527 de 1999 - Ley de Comercio Electrónico. Esta ley define el reglamento sobre el acceso y el uso de los datos del comercio electrónico, también de las firmas digitales y establece las normas de servicios electrónicos.

Decreto 587 de 2016 - Reglamenta la adquisición de bienes o servicios adquiridos a través de comercio electrónico. Este reglamento presenta las condiciones y procedimiento para revisión de pagos, que puede ser solicitada por los consumidores, está en especial se centra en los pagos realizados por medios digitales con tarjetas de crédito o débito.

Decreto 2499 de 2012 - Se declara retención en la fuente para actividades de la industria de software. Los contribuyentes a desarrollo de software con residencia en Colombia están obligados a presentar declaración de impuesto sobre renta.

Marco Contextual

Localización

El proyecto Vitrina Virtual está delimitado para atraer y apoyar a los emprendedores de la zona sur y centro sur de la UNAD. La zona sur está compuesta por los departamentos del Tolima, Huila, Caquetá y putumayo, por otro lado, la zona centro sur abarca los departamentos del Valle, Cauca y Nariño. Aunque la UNAD cuente con más zonas en todo el país el proyecto inicialmente abarcará solo estas zonas, pero en un futuro se podría aumentar la exposición a todas las zonas de la UNAD para atraer un mayor número de emprendedores.

Tabla 1

Zonas donde se llevará a cabo el proyecto aplicado.

ZONA SUR	Centros	Neiva
		Florencia
		Ibagué
		Líbano
		Mariquita
		La Plata
		San Vicente del Caguán
		Valle del Guamuez
		Pitalito
		Puerto Asís
ZONA CENTRO SUR	Centros	Cali
		El Bordo
		Palmira
		Pasto

Popayán
Santander de Quilichao
Tumaco

Nota. Esta tabla muestra todos los centros de la zona sur y centro sur. *Fuente.* Autoría propia

Número Emprendedores

La vitrina virtual inicia alojando 6 emprendimientos y 1 servicio, pero se espera que a mayor exposición y mejoras de esta misma se empiece a atraer más emprendedores de la UNAD, por lo que se espera que este número incremente paulatinamente.

Contexto de los Emprendedores de la UNAD

En la universidad nacional abierta y a distancia (UNAD), la comunidad unadista enfrenta grandes desafíos debido al bajo apoyo de la misma universidad, a pesar de que se tiene un espíritu emprendedor en la UNAD no se tiene un espacio específico para promover y promulgar a los emprendimientos de nuestros emprendedores, esto limita significativamente la capacidad de llevar estos emprendimientos al mercado y que empiecen a tener visualización por parte de los posibles clientes que cada día se inclinan más por el comercio electrónico.

Metodología

Enfoque

Para el desarrollo del proyecto "UNAD Emprende Móvil: Tu Vitrina Virtual al Alcance de tu Mano" se tomó un enfoque cuantitativo, que permite analizar estadísticamente la cantidad de usuarios y emprendimientos que se suben a la vitrina.

Población

La población objetivo para este proyecto incluye a todos los miembros de la comunidad UNADISTA ubicados en las zona sur y centro sur, específicamente estudiantes actuales, egresados, profesores y personal administrativo que estén involucrados en emprendimientos o interesados en el ecosistema emprendedor.

Esta población abarca una diversa gama de disciplinas y sectores, reflejando la amplia variedad de proyectos emprendedores que pueden beneficiarse de la vitrina virtual. Además, se considera también a los potenciales clientes e inversores dentro de estas zonas geográficas que busquen conectar con los emprendimientos presentados en la plataforma.

Metodología de Desarrollo De Software Kanban

La metodología Kanban es un enfoque ágil y visual para la gestión de proyectos que se centra en la mejora continua, la flexibilidad y la entrega incremental. Esta metodología es especialmente adecuada para el proyecto "UNAD Emprende Móvil", ya que permite una gestión eficiente de las tareas y una adaptación rápida a los cambios en los requisitos del proyecto.

Principios de Kanban.

Visualización del Trabajo: Se utilizará un tablero Kanban para representar visualmente todas las tareas del proyecto. Cada tarea será representada por una tarjeta que se moverá a través de diferentes columnas que reflejan el estado del trabajo, como "Por Hacer", "En Progreso" y "Completado".

Limitación del Trabajo en Progreso: Se establecerán límites en la cantidad de tareas que pueden estar en progreso al mismo tiempo. Esto ayudará a evitar cuellos de botella y asegurará que el equipo se concentre en completar las tareas antes de comenzar nuevas.

Gestión del Flujo: El objetivo es optimizar el flujo de trabajo, asegurando que las tareas se muevan de manera eficiente a través del tablero. Se monitorizarán las métricas de rendimiento para identificar áreas de mejora.

Mejora Continua: Se fomentará una cultura de retroalimentación constante y adaptación. Se realizarán reuniones periódicas para revisar el progreso y discutir posibles mejoras en el proceso.

Figura 1

Tablero Kanban del proyecto aplicado

TABLERO KANBAN				
Nombre de la tarea	Fecha de inicio	Fecha de finalización	Asignado	Estado
Análisis y diseño			Todos	Terminado
Selección de tecnologías y herramientas a utilizar	22.03.2024	05.04.2024	Todos	Terminado
Creación del tablero Kanban para la gestión del proyecto	22.03.2024	30.03.2024	Edison	Terminado
Revisión de bases de datos y antecedentes	22.03.2024	30.03.2024	Stiven	Terminado
Selección de fuentes, imágenes y estilos	07.04.2024	15.04.2024	Todos	Terminado
Prototipo estructural Front End	15.04.2024	21.04.2024	Todos	Terminado
Trasado estratégico para la recolección de emprendimientos	15.04.2024	05.05.2024	Stiven	Terminado
Desarrollo de la plataforma			Todos	Terminado
Diseño final de la App de la vitrina en base al prototipo del Front End	22.04.2024	05.05.2024	Todos	Terminado
Programación de la infraestructura de la App	14.05.2024	14.06.2024	Todos	Terminado
Configuración de la App para múltiples versiones	01.06.2024	22.06.2024	Todos	Terminado
Selección y adecuación del Hosting	21.06.2024	30.06.2024	Todos	Terminado
Pruebas y despliegue				Terminado
Optimización y ajustes del Hosting	01.07.2024	07.06.2024	Todos	Terminado
Pruebas y monitoreo	07.07.2024	22.07.2024	Todos	Terminado
Identificación y corrección de errores de la App identificados en las pruebas	15.07.2024	30.07.2024		Terminado
Despliegue escalado de la vitrina en el Hosting	01.08.2024	15.08.2024	Todos	Terminado
Creación del manual de usuario	02.08.2024	16.08.2024	Todos	Terminado
Cierre del proyecto		22.08.2024	Todos	Terminado

Nota. Tareas por realizar durante el desarrollo del proyecto. Tomado de autoría propia.

Fases de desarrollo Teórico del sitio web Vitrina Virtual

Fase De Análisis:

En esta fase, el objetivo principal es analizar y diseñar la aplicación "UNAD Emprende Móvil" para la promoción de los emprendimientos de la comunidad UNADISTA y se realizará un diagnóstico preliminar para identificar las necesidades esenciales y posibles soluciones que satisfagan los requerimientos fundamentales del proyecto. Se definirán lineamientos sobre cómo se asegurará la protección de la información y la integridad de los procesos dentro de la aplicación.

Tabla 2

Análisis teórico del desarrollo del software.

ASPECTO ANALIZADO	DESCRIPCIÓN TÉCNICA
Requerimientos Funcionales	Se identificaron las funcionalidades clave necesarias para la "Vitrina Virtual", tales como la creación y gestión de perfiles de emprendedores, la publicación de productos/servicios, y la capacidad de realizar transacciones seguras en línea.
Requerimientos No Funcionales	Se establecieron requisitos no funcionales, incluyendo la seguridad del sistema, la escalabilidad para soportar un aumento en el número de usuarios, y la eficiencia en el tiempo de carga de las páginas para una experiencia de usuario fluida.
Análisis de Seguridad	Se evaluaron los riesgos potenciales de seguridad para la protección de datos sensibles de los emprendedores y usuarios. Se diseñaron políticas de autenticación y

	encriptación de datos para garantizar la confidencialidad e integridad de la información.
Análisis de Usabilidad	Se realizaron estudios preliminares de usabilidad para determinar cómo los usuarios interactuarán con la plataforma. Esto incluyó la creación de perfiles de usuario (personas) y escenarios de uso para guiar el diseño de la interfaz.
Análisis de Compatibilidad	Se investigaron las mejores tecnologías para el desarrollo del frontend y backend. Se seleccionaron tecnologías basadas en JavaScript (como React para frontend y Node.js para backend) debido a su flexibilidad, rendimiento y comunidad de soporte.
Análisis de Integridad de Datos	Se revisaron los requisitos de compatibilidad para garantizar que la "Vitrina Virtual" funcione correctamente en todos los navegadores web principales y dispositivos móviles.

Nota. Esta tabla muestra los análisis teóricos realizados en el desarrollo del proyecto del software. *Fuente.* Autoría propia

Fase De Diseño:

En esta fase, se crearán wireframes y prototipos de la interfaz de usuario. Estos diseños ayudarán a visualizar cómo interactuarán los usuarios con la aplicación y asegurarán que se satisfagan las necesidades identificadas en la fase de análisis. Se definirán los elementos de navegación, la disposición de los contenidos y la estética general de la vitrina virtual.

Tabla 3

Diseños y prototipos del software.

ASPECTO DE DISEÑO	DESCRIPCIÓN TÉCNICA
Arquitectura de Información	Se estructuró la información de manera lógica y coherente para facilitar la navegación y el acceso a las distintas funcionalidades de la plataforma.
Diseño de Interfaz de Usuario (UI)	Se definió la estética visual de la plataforma, seleccionando paletas de colores, tipografías, y estilos de botones que fueran coherentes con la identidad de la UNAD y atractivos para los usuarios. Se priorizó la creación de una interfaz limpia y moderna que facilite la interacción del usuario.
Elementos de Navegación	Se diseñaron menús y barras de navegación intuitivas para garantizar que los usuarios puedan acceder fácilmente a todas las secciones del sitio. Se incluyeron enlaces de navegación internos para mejorar la localización dentro de la plataforma.
Diseño Adaptativo y Responsivo	Se implementó un diseño responsivo que permite que la "Vitrina Virtual" se adapte automáticamente a diferentes tamaños de pantalla y dispositivos, asegurando una experiencia de usuario consistente tanto en ordenadores como en dispositivos móviles.

Nota. Esta tabla muestra los diferentes puntos y diseños realizado. *Fuente.* Autoría propia

Fase De Desarrollo:

La fase de desarrollo se llevará a cabo en Sprints utilizando la metodología Kanban. Cada sprint incluirá tareas específicas relacionadas con la implementación de las funcionalidades definidas en el diseño. Se utilizarán tecnologías como HTML, CSS, JavaScript para el frontend y de nuevo JavaScript para el backend. Durante esta fase, las tareas se moverán a través del tablero Kanban según su estado de avance.

Tabla 4*Fase de desarrollo del software*

ASPECTO DE DESARROLLO	DESCRIPCIÓN TÉCNICA
Metodología de Desarrollo	Se utilizó la metodología ágil Kanban para gestionar el flujo de trabajo. Esta metodología permitió un enfoque flexible y adaptable, gestionando tareas en un tablero visual que muestra el estado de cada tarea (pendiente, en progreso, en revisión, completada), facilitando la colaboración y priorización continua.
Desarrollo Frontend	Se desarrolló la interfaz de usuario utilizando tecnologías como HTML, CSS y JavaScript. Se emplearon frameworks modernos como React para componentes dinámicos y reutilizables, mejorando la eficiencia del desarrollo y la experiencia del usuario con interacciones fluidas y rápidas.
Optimización y Performance	Se implementaron técnicas de optimización como la carga diferida (lazy loading) de imágenes y scripts, minificación de archivos CSS y JavaScript, y caching de datos en el servidor para mejorar la velocidad de carga del sitio y proporcionar una experiencia de usuario más rápida y eficiente.

Nota. Esta tabla muestra cómo se desarrolló el software del proyecto. *Fuente.* Autoría propia

Fase De Pruebas:

Después de completar las funcionalidades, se realizarán pruebas exhaustivas para garantizar que la aplicación funcione correctamente. Esto incluirá pruebas funcionales, de usabilidad y seguridad. Se buscará identificar y corregir errores antes de la implementación final.

Tabla 5*Pruebas funcionales de usabilidad y compatibilidad del software*

ASPECTO DE PRUEBAS	DESCRIPCIÓN TÉCNICA
Pruebas Funcionales	Se realizaron pruebas funcionales para verificar que cada característica y funcionalidad del sitio web opere conforme a los requisitos especificados. Estas pruebas incluyeron la verificación de la navegación y publicación de productos.
Pruebas de Usabilidad	Se llevaron a cabo pruebas de usabilidad con usuarios reales para evaluar la facilidad de uso de la plataforma. Se midió la eficiencia en completar tareas clave, como la búsqueda de productos, uso de enlaces de redes sociales, así como la satisfacción del usuario en términos de experiencia general y diseño de interfaz.
Pruebas de Compatibilidad	Se llevaron a cabo pruebas de compatibilidad para garantizar que el sitio web funcione correctamente en diferentes navegadores web (Chrome, Firefox, Safari, Edge) y dispositivos (escritorios, tabletas, móviles). Se verificó que la interfaz de usuario se adaptara correctamente y que todas las funcionalidades estuvieran operativas.

Nota. Esta tabla muestra los diferentes tipos de pruebas a los cuales se debe someter el software.

Fuente. Autoría propia

Fase De Implementación:

Una vez que todas las pruebas hayan sido superadas con éxito, se procederá a implementar la aplicación en un entorno de producción. Esto incluirá la configuración del servidor y la base de datos, asegurando que todo esté listo para el lanzamiento al público.

Tabla 6*Implementación del software*

ASPECTO DE IMPLEMENTACIÓN	DESCRIPCIÓN TÉCNICA
Despliegue del Frontend	El frontend, construido con tecnologías como React, fue compilado y desplegado en el servidor de producción. Se utilizaron técnicas de optimización como minificación y compresión de archivos CSS y JavaScript para mejorar los tiempos de carga y reducir el uso de ancho de banda.
Migración de Datos	Se llevó a cabo una migración de datos desde entornos de desarrollo a producción, asegurando la integridad de los datos y su correcta transformación al formato requerido por la base de datos de producción. Se realizaron verificaciones posteriores a la migración para confirmar que los datos se hayan transferido correctamente.
Pruebas en el Entorno de Producción	Antes del lanzamiento público, se realizaron pruebas finales en el entorno de producción para validar que todas las funcionalidades estuvieran operativas y que el sitio web se comportara de manera óptima bajo condiciones de tráfico real. Se llevaron a cabo pruebas de estrés para evaluar el rendimiento en condiciones de alta carga.
Documentación y Capacitación	Se creó una documentación exhaustiva sobre la configuración y operación del sistema para el equipo de operaciones. Además, se ofreció capacitación al personal encargado del soporte y mantenimiento del sitio, asegurando que comprendieran los procedimientos de despliegue.

Nota. Esta tabla muestra cómo se realizó la implementación del software al público. *Fuente.* Autoría propia

Fase De Mantenimiento y Actualizaciones:

Después del lanzamiento, se establecerá un plan de mantenimiento para realizar actualizaciones periódicas, corregir errores y agregar nuevas funcionalidades basadas en la retroalimentación de los usuarios. Se fomentará una cultura de mejora continua para adaptarse a las necesidades cambiantes de la comunidad emprendedora.

Tabla 7

Mantenimiento y actualizaciones periódicas

ASPECTO DE MANTENIMIENTO Y ACTUALIZACIÓN	DESCRIPCIÓN TÉCNICA
Mantenimiento Preventivo	Se planificaron tareas de mantenimiento preventivo para actualizar regularmente el software del servidor, librerías y dependencias del proyecto a sus versiones más recientes, minimizando riesgos de seguridad y asegurando la estabilidad del sistema.
Pruebas de Reversión y Despliegue	Cada cambio o actualización se somete a pruebas exhaustivas en un entorno de staging que replica el entorno de producción. En caso de problemas con una actualización, se tiene preparado un plan de reversión (rollback) para restaurar rápidamente el estado anterior sin afectar a los usuarios.
Mejora Continua y Optimización	Se implementa un proceso de mejora continúa basado en el análisis de métricas de rendimiento y la retroalimentación de los usuarios.

Nota. Esta tabla muestra como realizar mantenimiento a al software. *Fuente.* Autoría propia

Catálogo De Requisitos.

A continuación, se proporciona una descripción del comportamiento del sistema que se va a desarrollar.

App Vitrina Virtual.

Tabla 8

Requisitos mínimos App vitrina virtual

Sistema Operativo	Android 5.0 (Lollipop) o más reciente.
Resolución de pantalla:	<ul style="list-style-type: none"> • 1024 x 768 píxeles y una profundidad de color mínima de 65,000 colores. • 1280 x 1024 píxeles.

Nota. Esta tabla muestra los requisitos mínimos del dispositivo para ejecutar la app. *Fuente.* Autoría propia

Vitrina Virtual Web.

Tabla 9

Requisitos mínimos vitrina virtual web

Navegadores	<ul style="list-style-type: none"> • Microsoft Edge 9 o más reciente. • Mozilla Firefox versión 4 en adelante. • Safari 5. • Chrome 5.
-------------	--

Resolución de pantalla:	<ul style="list-style-type: none"> • 1024 x 768 píxeles y una profundidad de color mínima de 65,000 colores. • 1280 x 1024 píxeles.
-------------------------	---

Nota. Esta tabla muestra los requisitos mínimos ejecutar la vitrina virtual. *Fuente.* Autoría propia

Requisitos de Datos.

Los requisitos de datos, también conocidos como requisitos de contenido, requisitos conceptuales o requisitos de almacenamiento de información, responden a la pregunta de qué información debe almacenar y gestionar la aplicación.

Tabla 10

Requisitos de emprendimientos en datos de programación

Datos	Nombre del Emprendimiento.
	Descripción Corta Del Emprendimiento.
Categorías	alimentos
	moda
	calzado
	tecnologías
	belleza
	higiene
	asesorías
	medicina
	mantenimiento

Nota. Esta tabla muestra los datos básicos de los emprendimientos publicados. *Fuente.* Autoría propia

Requisitos De Interfaz.

Los requisitos de interfaz, también conocidos como requisitos de interacción del usuario, responden a la pregunta de cómo el usuario interactuará con la aplicación.

Inicio De Sesión

Solo los desarrolladores o administradores del software deberán realizar el proceso de inicio de sesión, esto para acceder directamente al hosting del sitio Web y realizar actualizaciones o mejoras al software directamente, los demás usuarios podrán acceder directamente desde la Web a todos los servicios públicos del software.

Cierre De Sesión.

Una vez los administradores o desarrolladores hayan iniciado sesión, esta se cerrará ya sea por la acción del propio desarrollador o mediante un mecanismo automático basado en un tiempo de inactividad predefinido por el hosting.

Cliente Web O browser.

La aplicación podrá utilizarse con los navegadores más conocidos de la actualidad entre ellos encontramos:

Microsoft Edge.

Microsoft Edge desde la versión (44.190419) para Windows, Microsoft Edge desde la versión (100.0.1185.50) para Android.

Mozilla Firefox.

Mozilla Firefox desde la versión (100.0.1) para Windows, Mozilla Firefox desde la versión (100.1.2) para Android, Mozilla Firefox desde la versión (56.0.1) para MacOS.

Safari.

Safari 18 (Beta), Safari 17.6, Safari 17.5, Safari 17.4, Safari 17.3.

Chrome 5.

Google Chrome desde la versión (98.0.4758.82) para Windows, Google Chrome desde la versión (102.0.5005.59) para Android, Google Chrome desde la versión (106.0.5249.103) para MacOS.

Interacción cliente / aplicación.

La aplicación deberá ser intuitiva, con un tiempo de aprendizaje máximo de cinco minutos para el usuario. Debe ofrecer una interacción adecuada mediante mensajes informativos, orientados a proporcionar una excelente experiencia al usuario final.

Diseño y comodidad.

El diseño de las interfaces debe ser responsive para asegurar una visualización adecuada en diversos dispositivos, como computadoras personales, tablets y teléfonos inteligentes. Estas

interfaces deben utilizar colores suaves, que sean agradables a la vista, y emplear fuentes claramente legibles.

Responsive Web Design (RWD); dado que utilizamos HTML5, CSS3 y Bootstrap, el diseño responsivo es fundamental. Nos aseguramos de que nuestra página web se acomode y funcione bien en todos los dispositivos, desde teléfonos hasta tablets y ordenadores. El sistema de grid de Bootstrap son herramientas clave en este proceso.

Mostrar información Acerca De Nosotros.

Informa sobre el uso de la página web o móvil, presenta una breve descripción de la página.

Información de emprendimientos

Muestra emprendimientos relevantes acerca de la vitrina virtual incluyendo información del emprendimiento en el cual el usuario puede contactarse con el vendedor por diferentes medios digitales como: Facebook, Instagram y WhatsApp).

Tiempo de respuesta.

El tiempo de respuesta de la interfaz web de la aplicación debe ser inferior a 3 segundos, sin considerar los retardos de red.

Fase De Diseño Grafico

En la Fase de Diseño Gráfico del proyecto "Vitrina Virtual", se contrató a un diseñador gráfico profesional para desarrollar la identidad visual de la plataforma web. Este diseñador fue responsable de crear un concepto visual que reflejara los colores y objetivos de la "Vitrina Virtual" y la UNAD, asegurando que la plataforma fuera atractiva, coherente y funcional para sus usuarios.

El trabajo final del diseñador, documentado en el PDF entregado, incluyó varios elementos clave:

Logotipo y Paleta de Colores

Se diseñó un logotipo distintivo que representa la esencia de la comunidad UNAD y sus emprendedores. La paleta de colores seleccionada fue pensada para ser vibrante y moderna, facilitando la conexión emocional y el reconocimiento visual de la plataforma.

Figura 2

Paleta de colores de la UNAD



Nota. Paleta de colores usados en las principales páginas de la UNAD. Tomado de autoría propia.

Tipografía y Estilo de Imagen

Se eligieron fuentes legibles y modernas para los textos, garantizando una lectura fácil tanto en dispositivos móviles como en pantallas de computadoras. Además, se estableció un estilo de imagen que enfatiza la profesionalidad y la innovación, alineándose con la misión de apoyar y promover a los emprendedores de la UNAD.

Tabla 11

Fuentes tipográficas de la página web

Angsana New	Angsa.ttf
franklin-gothic-cufonfonts	fgwo____.ttf
	FRANKGO.ttf
	Franklin Gothic Book
	Regular.ttf
	Franklin Gothic Condensed.ttf
	FranklinGothic.ttf
	FranklinGothicITALIC.ttf

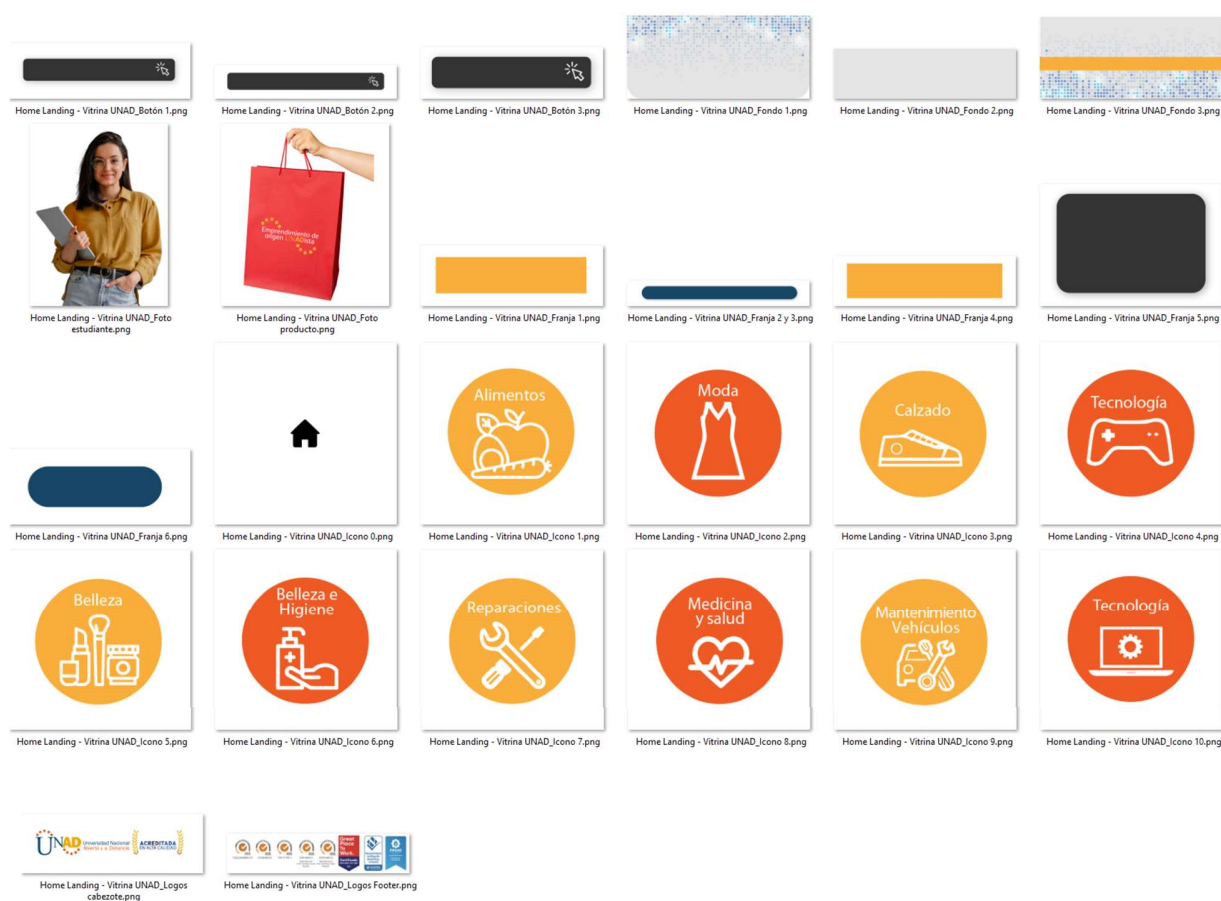
Nota. Esta tabla muestra los diferentes estilos de tipografías usados en el software. *Fuente.* Autoría propia

Diseño de Interfaz de Usuario (UI)

Se crearon diseños para la interfaz de usuario que incluían la disposición de los elementos visuales, como botones, menús y banners, asegurando una navegación intuitiva y una experiencia de usuario agradable.

Figura 3

Diseños de Usuario



Nota. Diferentes interfaces de usuario. Tomado de autoría propia.

Figura 4

Diseño gráfico Índice



Nota. Diseño gráfico del índice de la página web. Tomado de autoría propia.

Figura 5

Diseño Gráfico Sobre Notros

UNAD Universidad Nacional Abierta y a Distancia

ACREDITADA

Sobre Nosotros Emprendimientos ¡Haz parte!

Sobre Nosotros

Vitrina Virtual UNAD

Vitrina Comercial Unadista nace con la necesidad de brindarte un espacio digital donde todos los Unadistas que tienen una empresa, negocio o emprendimiento puedan tener un anuncio o promoción en línea, sin necesidad de tener necesariamente una PÁGINA WEB.

En vitrina comercial Unadista esperamos brindarte la mejor de las experiencias en servicio, crecer con contigo es nuestra prioridad.

Vicerrectoría de Innoyación y emprendimiento

UNAD

Datos de Contacto:
 Línea gratuita nacional: 01 8000 115223
 Teléfonos: 001-375 9500 - En Bogotá D.C.
 Colombia
 P.O. Box 661 3443700
 Sede Principal: Calle 14 Sur # 14 - 23
 Barrio Restrepo, Bogotá - Colombia.

Gobierno:
 Ministerio de Educación Nacional de Colombia
 Gobierno de Colombia - Gosco
 Colombia Compra Eficiente
 Fondo Emprender
 C.C.T.C.
 Dinamio: Ciudad
 Colombia Aprende
 Gobierno Digital

Partners de Interés:
 Agencia de la UNAD
 Centros y Sejes administrativas
 Portal de Mías
 Setio Editorial
 Investigación
 Medios y Meditaciones Comunicacionales
 Portal de Eventos

Normatividad:
 Aprende Institucional
 Estatuto General
 Estatuto Docente
 Estatuto de Bienestar Institucional
 Estatuto de Contratación
 Reglamento estudiantil

Universidad Nacional Abierta y a Distancia UNAD de Colombia - © Copyright UNAD 2023
 Institución de Educación Superior sujeta a inspección y vigilancia por el Ministerio de Educación Nacional

Síguenos en:

Certificaciones

Nota. Diseño gráfico sobre nosotros de la página web. Tomado de autoría propia.

Figura 6

Diseño gráfico emprendimientos



Nota. Diseño gráfico sobre emprendimientos de la página web. Tomado de autoría propia.

Figura 7

Diseño Gráfico Haz Parte



Nota. Diseño gráfico sobre haz parte de la página web. Tomado de autoría propia.

Declaración del Documento y Encabezado Vitrina Virtual V2 (<head>)

<!doctype html>: Declara el tipo de documento y la versión de HTML. <html> es el elemento raíz del documento HTML. <head> contiene metadatos y enlaces a recursos externos como hojas de estilo y scripts. <meta charset="UTF-8"> define la codificación de caracteres.

<meta name="viewport" content="width=device-width, initial-scale=1.0"> establece el tamaño de la ventana de visualización en dispositivos móviles. <link rel="stylesheet" href="estilos/estilos.css"> enlaza a una hoja de estilos CSS externa.

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet"> enlaza a Bootstrap para estilos adicionales. <title> define el título de la página que aparece en la pestaña del navegador.

Cuerpo del Documento (<body>)

<div class="container px-0"> contiene la estructura del encabezado (<header>). <nav class="navbar navbar-expand-lg fixed-top bg-white"> elemento de navegación que incluye un logo y un menú desplegable para dispositivos móviles. <div class="navbar-logo"> contiene el logo de la página. <button class="navbar-toggler"> botón para mostrar el menú en dispositivos móviles.

<div class="offcanvas offcanvas-end"> menú desplegable (offcanvas) con enlaces de navegación. <div class="container-fluid py-5 mb-5 hero-header"> sección principal de la página con un encabezado de tipo "hero" que incluye un título y un botón de retroceso. <div class="row text-center"> contiene el título principal.

`<div class="container py-1 c mt-1 d-flex justify-content-end">` botón para retroceder.
`<div class="tab-content" id="myTabContent">` contiene el contenido dinámico y scripts para la funcionalidad. `<div id="tecnologias-products" class="container d-flex flex-wrap justify-content-center"></div>` espacio para los productos relacionados con la tecnología. `<script>` enlaces a archivos JavaScript necesarios para la funcionalidad de la página.

Pie de Página (<footer>)

`<div class="container-fluid footer pt-5 mt-5 hero-footer">`: Contiene el pie de página con información de contacto, enlaces gubernamentales, enlaces de interés, y normatividad.

`<div class="row g-5">`: Contiene varias columnas con diferentes secciones de información.

Copyright

`<div class="container-fluid copyright bg-white py-4">`: Sección final con derechos de autor e información adicional.

`<div class="row text-center">`: Contiene el texto de copyright y enlaces a redes sociales.

Descripción de los Scripts Utilizados en el Proyecto Vitrina Virtual V2

A partir de los hallazgos, se procedió a una optimización significativa del código utilizando JavaScript. Este proceso no solo mejoró el rendimiento general de la aplicación, sino que también permitió una reducción sustancial en la cantidad de líneas de código, manteniendo la misma funcionalidad o incluso ampliándola. Un enfoque clave de esta fase fue la introducción de un sistema de categorías más robusto, que facilitó la navegación y mejoró la experiencia del usuario.

Definición y Gestión de Productos

El script define una estructura de datos para manejar un catálogo de productos. Cada producto se representa como un objeto que incluye detalles como nombre, descripción, imágenes principales y adicionales, información adicional (como eslóganes y datos de contacto), y enlaces a redes sociales. Esta estructura permite gestionar fácilmente la información y facilita la actualización de los productos.

Figura 9

Plantilla de emprendimientos en JavaScript

```
318 //Plantilla
319 {
320   id: 0,
321   name: "",
322   description: "",
323   image: "",
324   category: "",
325   additionalImages: [
326     "",
327     "",
328     ""
329   ],
330   additionalInfo: {
331     slogan: "",
332     productType: "",
333     city: "",
334     email: "",
335     phone: "",
336     address: ""
337   },
338   socialMedia: {
339     facebook: "",
340     instagram: "",
341     whatsapp: ""
342   },
343 },
```

Nota. Diseño gráfico sobre haz parte de la página web. Tomado de autoría propia.

Filtrado y Renderizado Dinámico de Productos

La función `renderProducts(category)` se encarga de renderizar los productos según su categoría en la interfaz del usuario. Utilizando JavaScript y consultas al DOM, se filtran los productos por categoría y se genera un HTML dinámico que se inserta en el contenedor correspondiente de la página. Esto permite a los usuarios ver solo los productos que pertenecen a la categoría seleccionada, mejorando la experiencia de navegación.

Figura 10

Función renderProducts

```

377 function renderProducts(category) {
378   const container = document.querySelector(`#${category}-products`);
379   const filteredProducts = products.filter(
380     (product) => product.category === category
381   );
382   let html = "";
383   filteredProducts.forEach((product) => {
384     const carouselIndicators = product.additionalImages
385       .map( (_, index) => `
386         <button type="button" data-bs-target="#carousel-${product.id}"
387           data-bs-slide-to="${Math.floor(index / 2)}" ${index < 2 ? 'class="active" aria-current="true"' : ""}
388           aria-label="Slide ${Math.floor(index / 2) + 1}"></button>
389       `)
390       .filter( (_, index) => index % 2 === 0) // Filtra para solo incluir un indicador por cada par de imágenes
391       .join("");

```

Nota. Código que renderiza los emprendimientos acorde a las categorías. Tomado de autoría propia.

Implementación de Carrusel de Imágenes

Para cada producto, se ha implementado un carrusel de imágenes que permite a los usuarios visualizar múltiples fotos asociadas con un producto de forma atractiva y organizada. La lógica del carrusel está contenida dentro de la función `renderProducts`, donde se crean elementos HTML dinámicamente para cada imagen adicional del producto. El carrusel está optimizado para

mostrar pares de imágenes por diapositiva, y los controles de navegación se ajustan automáticamente según la cantidad de imágenes.

Figura 11

Código carrusel de ítems

```
393     const carouselItems = product.additionalImages
394       .map((image, index) => {
395         const nextImage = product.additionalImages[(index + 1) % product.additionalImages.length];
396         const isEven = index % 2 === 0;
397         return isEven ? `
398           <div class="carousel-item ${index === 0 ? "active" : ""}">
399             <div class="d-flex">
400               
401               
402             </div>
403           </div>
404         ` : '';
405       })
406       .filter((_, index) => index % 2 === 0)
407       .join("");
```

Nota. Código en JavaScript del carrusel de imágenes. Tomado de autoría propia.

Funcionalidad de Pantalla Completa para Imágenes

Se añadió una función llamada `openFullscreen(imgElement)` que permite a los usuarios expandir cualquier imagen del producto a pantalla completa al hacer clic. Esta función mejora la usabilidad del sitio, proporcionando una mejor visualización de los detalles del producto. Es compatible con varios navegadores y utiliza métodos específicos como `requestFullscreen`, `mozRequestFullScreen`, `webkitRequestFullscreen` y `msRequestFullscreen` para garantizar la compatibilidad cruzada.

Figura 12

Full screen código principal

```
361 // FULL SCREEN DE LAS IMAGENES
362 function openFullscreen(imgElement) {
363     if (imgElement.requestFullscreen) {
364         imgElement.requestFullscreen();
365     } else if (imgElement.mozRequestFullScreen) { // Firefox
366         imgElement.mozRequestFullScreen();
367     } else if (imgElement.webkitRequestFullscreen) { // Chrome
368         imgElement.webkitRequestFullscreen();
369     } else if (imgElement.msRequestFullscreen) { // IE/Edge
370         imgElement.msRequestFullscreen();
371     }
372 }
```

Nota. Código del full screen en las imágenes publicitarias. Tomado de autoría propia.

Sistema de Categorías Para Los Emprendimientos

La implementación de este nuevo sistema de categorías impulsó la creación de nuevas ventanas HTML, expandiendo la estructura de la Vitrina Virtual de cuatro a trece ventanas. A pesar de esta expansión, se logró una reducción considerable del código, gracias a la refactorización y popularización de componentes. Cada ventana fue diseñada para ser más específica y cumplir con objetivos claros, lo que no solo optimizó la navegabilidad, sino que también simplificó el mantenimiento futuro del software.

Figura 13

Código de categorías de la vitrina virtual web

```
<div class="tab-content" id="myTabContent">
  <div class="tab-pane fade show active" id="home" role="tabpanel" aria-labelledby="home-tab">
    <div class="row fondoindex">
      <div class="col-12">
        <a href="alimentos.html">
          
        </a>
        <a href="moda.html">
          
        </a>
        <a href="calzado.html">
          
        </a>
        <a href="tecnologias.html">
          
        </a>
        <a href="belleza.html">
          
        </a>
      </div>
    </div>
  </div>
</div>
```

Nota. Código con las imágenes y referencias de cada categoría de productos. Tomado de autoría propia.

Script de Formulario de Registro


En la segunda versión de la vitrina virtual se agregó la opción de registro de emprendimientos de forma directa, ahora los emprendedores solo deben llenar un registro y subir la información de sus emprendimientos, agilizando y facilitando el proceso de registro de nuevos emprendimientos a la página. Estos registros se realizan por medio de Google Forms, los datos son cargados a una hoja de cálculo y allí son gestionados por un script que genera de forma automática el código con el nuevo emprendimiento ya debidamente diligenciado. Posterior a esto se debe cargar las imágenes subidas por el interesado y adjuntarlas al código final, es necesario que el programador o encargado verifique el contenido de cada emprendimiento antes de subirlo a la página y hacer que este sea visible al público.

Figura 14

Formulario de registro de emprendimientos

Formulario De Registro Del Emprendimiento

Por favor, completa los detalles siguientes para registrar su emprendimiento en nuestra base de datos.

stiven4111@gmail.com [Cambiar de cuenta](#) 

El nombre, el correo y la foto asociados a tu cuenta de Google se registrarán cuando subas archivos y envíes este formulario

* Indica que la pregunta es obligatoria

Nombre del Emprendimiento: *

Tu respuesta

Descripción Corta Del Emprendimiento: *

Respuesta corta de no mas de 15 palabras

Tu respuesta

Categoría *

Solo puedes seleccionar una categoria

- alimentos
- moda
- calzado
- tecnologias
- belleza

Nota. Formulario de registro de emprendimientos en Google Forms. Tomado de autoría propia.

Figura 15

Script automático de código de emprendimientos

```
1 function updateJSONTemplate() {
2   var sourceSheet = SpreadsheetApp.getActiveSpreadsheet().getSheetByName("Respuestas de formulario 1");
3   var lastRow = sourceSheet.getLastRow();
4
5   if (lastRow > 1) {
6     var dataRange = sourceSheet.getRange(2, 1, lastRow - 1, 16); // Incluyendo la columna de "Procesado"
7     var data = dataRange.getValues();
8
9     var outputSheet = SpreadsheetApp.getActiveSpreadsheet().getSheetByName("JSON Output");
10    if (!outputSheet) {
11      outputSheet = SpreadsheetApp.getActiveSpreadsheet().insertSheet();
12      outputSheet.setName("JSON Output");
13    }
14    // No limpiar la hoja de salida para mantener los datos existentes
15
16    data.forEach(function(row, index) {
17      if (row[15] !== "Si") { // Verificar si ya ha sido procesado
18        var json = buildJson(row, index);
19        var formattedJson = formatCustomJSON(json);
20        // Insertar en la primera fila para que los registros más nuevos aparezcan primero
21        outputSheet.insertRowBefore(1).getRange(1, 1).setValue(formattedJson);
22        sourceSheet.getRange(index + 2, 16).setValue("Si"); // Marcar como procesado
23      }
24    });
25  } else {
26    Logger.log("No hay datos para procesar.");
27  }
28 }
```

Nota. Script que genera el código de cada emprendimiento subido por medio del formulario. Tomado de autoría propia.

Fase De Desarrollo Del Del Código Del Aplicativo Móvil

En esta fase, se detalla el desarrollo y la estructura del código de la aplicación móvil "Vitrina Virtual". Esta aplicación tiene como objetivo principal ofrecer una experiencia de navegación web fluida dentro de la aplicación, asegurando una correcta visualización de contenidos y manejo de conectividad. A continuación, se explican en detalle los componentes principales del código, su funcionalidad, y cómo cada parte contribuye a la operación general del aplicativo, desde la configuración inicial del entorno y manejo de la conectividad, hasta la implementación de características específicas del WebView y la gestión del ciclo de vida de la aplicación. Esta descripción proporciona una visión clara de cómo se ha diseñado y desarrollado la aplicación para cumplir con sus objetivos funcionales.

Importaciones y Configuración Inicial

El código importa varias clases necesarias para la funcionalidad de la aplicación, como gestionar la conectividad a Internet, mostrar páginas web dentro de la aplicación utilizando WebView, manejar vistas y configuraciones de cookies, y trabajar con alertas y actividades.

Figura 16

Importe de librerías Android Studio

```
1 package com.vitrina.appvitrina;
2
3 import android.annotation.SuppressLint;
4 import android.content.BroadcastReceiver;
5 import android.content.Intent;
6 import android.net.Uri;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.webkit.CookieManager;
10 import android.webkit.WebResourceRequest;
11 import android.webkit.WebSettings;
12 import android.webkit.WebView;
13 import android.webkit.WebViewClient;
14 import androidx.appcompat.app.AlertDialog;
15 import android.content.Context;
16 import android.net.ConnectivityManager;
17 import android.net.NetworkInfo;
18 import android.content.IntentFilter;
19 import android.widget.Toast;
20 import androidx.appcompat.app.AppCompatActivity;
21 import android.webkit.WebChromeClient;
22 import android.widget.FrameLayout;
23 import java.util.Arrays;
24 import java.util.List;
```

Nota. Librerías exportadas para la app en Android Studio. Tomado de autoría propia.

Clase Principal MainActivity

La clase MainActivity extiende AppCompatActivity, que permite usar funciones modernas de Android con compatibilidad hacia atrás. Dentro de esta clase, se definen varias variables de instancia.

WebView myWebView: Se utiliza para mostrar contenido web dentro de la aplicación. Permite cargar una página web directamente en la interfaz de la aplicación.

NetworkChangeReceiver networkChangeReceiver: Es un receptor de difusión que escucha los cambios en la conectividad de red del dispositivo.

FrameLayout fullScreenContainer: Contenedor que se utiliza para mostrar contenido en pantalla completa, como videos.

View customView y WebChromeClient.CustomViewCallback customViewCallback:
Variables que gestionan vistas personalizadas cuando se utiliza el modo de pantalla completa en el WebView.

Figura 17

Clase MainActivity Android Studio

```
28 ▶ <> public class MainActivity extends AppCompatActivity {  
    11 usages  
29     private WebView myWebView; // Variable de instancia para almacenar la referencia del WebView.  
    5 usages  
30     private NetworkChangeReceiver networkChangeReceiver;  
    7 usages  
31     private FrameLayout fullScreenContainer;  
    7 usages  
32     private View customView;  
    2 usages  
33     private WebChromeClient.CustomViewCallback customViewCallback;
```

Nota. Clase publica principal de la app conocida como MainActivity. Tomado de autoría propia.

Método isConnected()

Este método verifica si hay una conexión a Internet activa. Utiliza ConnectivityManager para obtener el estado actual de la red. Si hay una red activa y está conectada o en proceso de conexión, devuelve true; de lo contrario, devuelve false.

Figura 18

Boolean de conexión isConnected

```
public boolean isConnected() {
    ConnectivityManager connectivityManager = (ConnectivityManager) getSystemService(Context.CONNECTIVITY_SERVICE);
    if (connectivityManager != null) {
        NetworkInfo activeNetwork = connectivityManager.getActiveNetworkInfo();
        return activeNetwork != null && activeNetwork.isConnectedOrConnecting();
    }
    return false;
}
```

Nota. Código que revisa cuando el dispositivo móvil está o no conectado a internet. Tomado de autoría propia.

Método onCreate(Bundle savedInstanceState)

Este método es parte del ciclo de vida de la actividad y se llama cuando la actividad se está creando. Aquí se configuran varios aspectos importantes de la interfaz de usuario y la funcionalidad que establece la vista principal de la actividad desde un archivo de diseño XML.

Se configura la pantalla para mostrar contenido en pantalla completa y ocultar la barra de navegación. Se inicializa el WebView y se llama a `setupWebView()` para configurarlo.

Se verifica la conectividad a Internet. Si no hay conexión, se muestra un cuadro de diálogo de alerta informando al usuario y ofreciendo opciones para reintentar o salir de la aplicación.

Figura 19

Método onCreate principal

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    // Configurar la vista para mostrar contenido en pantalla completa y ocultar la barra de navegación
    View decorView = getWindow().getDecorView();
    int uiOptions = View.SYSTEM_UI_FLAG_FULLSCREEN | View.SYSTEM_UI_FLAG_HIDE_NAVIGATION | View.SYSTEM_UI_FLAG_IMMERSIVE_STICKY;
    decorView.setSystemUiVisibility(uiOptions);

    myWebView = findViewById(R.id.index);
    fullScreenContainer = findViewById(R.id.fullscreen_container); // Asegúrate de que esto está correctamente inicializado
    setupWebView();

    if (!isConnected()) {...} else {
        // Inicia cualquier proceso que requiera conectividad
        setupWebView();
    }
}
```

Nota. Código que alberga diferentes funcionalidades como conexión, ocultar barra de navegación y el cargue de enlaces externos. Tomado de autoría propia.

Métodos onResume() y onPause()

Estos métodos gestionan el registro y la anulación del registro del NetworkChangeReceiver:

onResume(): Se llama cuando la actividad se reanuda. Aquí se registra el receptor para escuchar cambios en la conectividad.

onPause(): Se llama cuando la actividad está a punto de pasar a segundo plano. Aquí se anula el registro del receptor para evitar fugas de memoria.

Figura 20

Metodo onResume de segundo plano

```
@Override
protected void onResume() {
    super.onResume();
    if (networkChangeReceiver == null) {
        networkChangeReceiver = new NetworkChangeReceiver();
    }
    IntentFilter filter = new IntentFilter(ConnectivityManager.CONNECTIVITY_ACTION);
    registerReceiver(networkChangeReceiver, filter);
}
```

Nota. Código que guarda en cache la app para cuando se cierre y abre cargue más rápido.
Tomado de autoría propia.

Método setupWebView()

Configura las opciones y funcionalidades del WebView: Configuración del WebSettings, ajusta varias configuraciones del WebView como habilitar JavaScript, soporte de almacenamiento DOM, zoom y más.

Gestión de Cookies: Configura el CookieManager para aceptar cookies y cookies de terceros.

Gestión de Vista en Pantalla Completa: Utiliza un WebChromeClient personalizado para manejar el modo de pantalla completa.

Control de Navegación: Implementa un WebViewClient personalizado para controlar cómo se cargan las URL. Si la URL corresponde a ciertos dominios externos, la carga se redirige a un navegador externo.

Figura 21

Código setupWebView configuraciones de la app

```
private void setupWebView() {  
  
    WebSettings webSettings = myWebView.getSettings();  
    webSettings.setLoadWithOverviewMode(true); // Ajusta la vista de carga para que la  
    webSettings.setUseWideViewPort(true); // Habilita el "viewport" amplio que permite  
    webSettings.setJavaScriptEnabled(true); // Habilita JavaScript para el WebView.  
    webSettings.setDomStorageEnabled(true); // Habilita el almacenamiento DOM, necesar  
    webSettings.setSupportZoom(true); // Habilita el soporte para zoom.  
    webSettings.setBuiltInZoomControls(true); // Habilita los controles de zoom integr  
    webSettings.setDisplayZoomControls(false); // Deshabilita la visualización de cont  
    webSettings.setTextZoom(110); // Ajusta el tamaño del texto en el WebView.  
    webSettings.setJavaScriptCanOpenWindowsAutomatically(false); // Deshabilita pop-up  
    // Configurar CookieManager para aceptar cookies y terceros cookies  
    CookieManager cookieManager = CookieManager.getInstance();  
    cookieManager.setAcceptCookie(true);  
    cookieManager.acceptThirdPartyCookies(myWebView);  
}
```

Nota. Código que carga las configuraciones principales de la app. Tomado de autoría propia.

Método onBackPressed()

Gestiona la acción del botón de retroceso del dispositivo: Si el WebView tiene historial de navegación, se retrocede en el historial. Si no hay historial de navegación, se muestra un cuadro de diálogo de confirmación preguntando al usuario si desea salir de la aplicación.

Figura 22

Código `onBackPressed`

```
public void onBackPressed() {
    if (myWebView != null && myWebView.canGoBack()) {
        myWebView.goBack(); // Permite regresar a la página anterior
    } else {
        // Mostrar diálogo de confirmación
        new AlertDialog.Builder(context: this)
            .setIcon(android.R.drawable.ic_dialog_alert)
            .setTitle("Salir de la aplicación")
            .setMessage("¿Quieres cerrar la APP?")
            .setPositiveButton(text: "Si", (dialog, which) -> {
                // Si decides seguir con la acción por defecto
                super.onBackPressed();
            })
            .setNegativeButton(text: "No", listener: null)
            .show();
    }
}
```

Nota. Código de advertencia al intentar salir de la app. Tomado de autoría propia.

Clase Interna `NetworkChangeReceiver`

Esta clase interna extiende `BroadcastReceiver` y se utiliza para escuchar los cambios en la conectividad de red del dispositivo.

Cuando cambia el estado de la conectividad, se muestra un `Toast` indicando si se ha perdido o recuperado la conexión a Internet.

Fase de Pruebas

Las pruebas son una etapa crucial en el desarrollo, donde se verifica y valida el funcionamiento, la usabilidad, la seguridad y la compatibilidad del sitio web. Estas pruebas se enfocan en los requisitos de la aplicación. Su principal objetivo es identificar y corregir posibles problemas antes de que el sitio web esté disponible para el público. A continuación, se presentan los resultados de las pruebas realizadas por los respectivos actores en el proyecto.

Figura 23

Prueba de rendimiento en GTmetrix



Nota. Prueba de rendimiento midiendo calidad de imágenes, texto, velocidad y bloqueos. Tomado de autoría propia.

Resultados

Enlace del sitio web: https://storage.googleapis.com/myvitrinavirtual_com/index.html

Interacción del Usuario

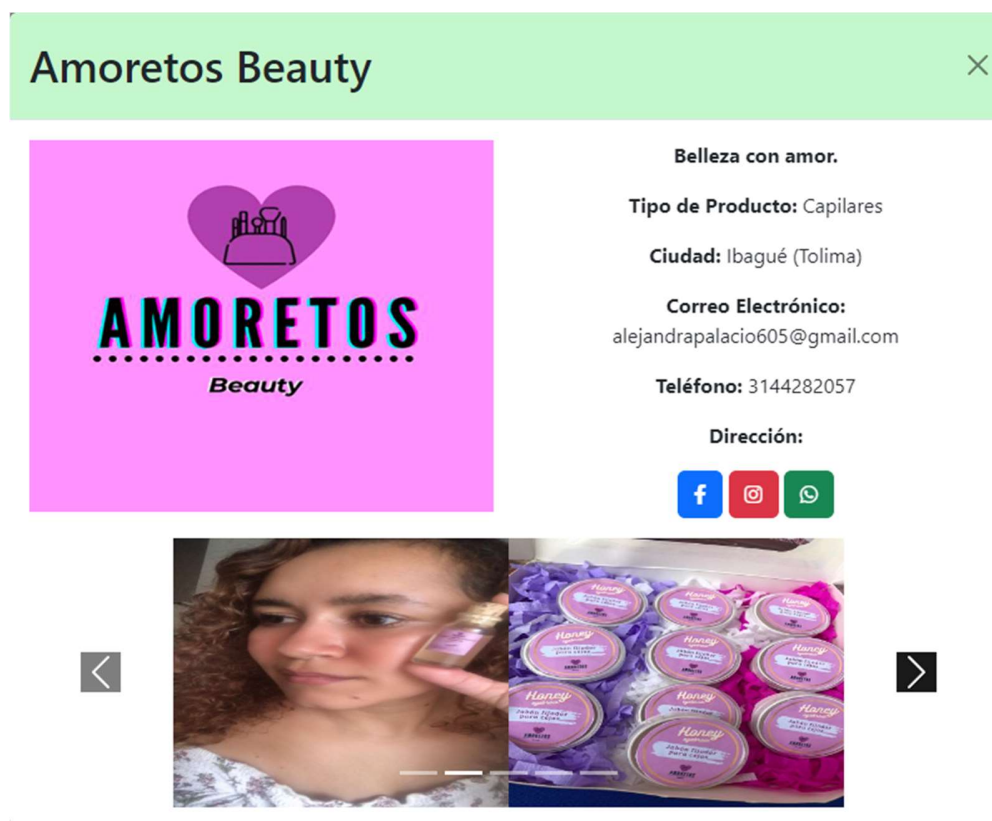
Los usuarios han mostrado un alto nivel de interacción con la aplicación, utilizando funcionalidades como la búsqueda y selección de productos en la vitrina virtual

Redes Sociales

El enlace a redes sociales ha permitido una mayor interacción entre la vitrina virtual y los usuarios.

Figura 24

Ventana ver más con redes sociales



Nota. Captura de emprendimiento Amoretos. Tomado de autoría propia.

Figura 25

Índice vista desde la app



Nota. Captura del índice desde la app vitrina virtual. Tomado de autoría propia.

Referencias Bibliográficas

- Enríquez, J. G., & Casas, S. I. (2014). Usabilidad en aplicaciones móviles. *Informes Científicos Técnicos - UNPA*, 5(2), 25–47. <https://doi.org/10.22305/ict-unpa.v5i2.71>
- Escobar-Reynel, J. L., Baena-Navarro, R., Giraldo-Tobón, B., Macea-Anaya, M., & Castaño-Rivera, S. (2021). Modelo de desarrollo para la construcción de aplicaciones móviles educativas. *TecnoLógicas*, 24(52), e2065. <https://doi.org/10.22430/22565337.2065>
- López Sanz, M., Sánchez Fúnquene, D. M., & Moreno Pérez, Á. (2015). Programación web en el entorno cliente. RA-MA Editorial. <https://elibro.net/en/lc/unapec/titulos/106486>
- Moreno Pérez, J. C. (2015). Programación. RA-MA Editorial. <https://elibro.net/en/lc/unapec/titulos/62476>
- Ollero Sánchez, C. (2015). Cuaderno del alumno: programación con lenguajes de guion en páginas web (UF1305). Confección y publicación de páginas Web (IFCD0110). Editorial CEP, S.L. <https://elibro.net/en/lc/unapec/titulos/50900>
- Salcedo González, Q., & Gallardo López, L. (2016). Modelo semiológico para diseñar y evaluar íconos de aplicaciones móviles (SM2Mobile). *Avances en Interacción Humano-Computadora*, (1), 18. <https://doi.org/10.47756/aihc.y1i1.4>
- Valarezo Loaiza, F., & Triviño, D. (2020). Scenarios to use React Native, native code or integrate both technologies. *ÑAWPAY Revista Técnica Tecnológica*, 2(1), 5. <https://doi.org/10.36500/nawpay.v2i1.95>

Velásquez, S. M., Monsalve Sossa, D. E., Zapata, M. E., Gómez Adasme, M. E., & Ríos, J. P.

(2019). Pruebas a aplicaciones móviles: avances y retos. **Lámpsakos**, (21), 39–50.

<https://doi.org/10.21501/21454086.2983>