



Tecnológico de Monterrey

Evidencia 1. Documento de la solución del ret

Alejandra Estefanía Rico González	A01749850
Sandra Paulina Herrera Rebollo	A01798452
Ian Alexei Martínez Armendáriz	A017532888
Fernanda Ponce Maciel	A01799293
Lorena Abigail Solís de los Santos	A01746602

Integración de seguridad informática, redes y sistemas de software

Grupo 401

Equipo: 05

Fecha de entrega:

Domingo 20 de octubre de 2024

Etapa 1. Documentación SRS.....	2
Etapa 2. Diseño.....	31
Etapa 3. Desarrollo.....	51
URL del repositorio del Proyecto.....	52
Etapa 4. Pruebas.....	80
Aportaciones individuales.....	132
Referencias.....	132

Etapa 1. Documentación SRS

Problemática que se va a resolver.

Pobreza menstrual

La pobreza menstrual es un fenómeno que afecta a millones de mujeres y niñas en todo México, especialmente en comunidades vulnerables. Esta problemática abarca la falta de acceso a productos menstruales seguros, información adecuada sobre la menstruación y las condiciones higiénicas necesarias para gestionar el ciclo menstrual de manera digna y saludable. Ya que, la falta de recursos puede llevar a consecuencias graves, como infecciones, absentismo escolar y estigmatización social.

Solución Propuesta

“Todas brillamos” busca abordar la pobreza menstrual a través de una plataforma digital que permite a las personas acceder a información confiable y actualizada sobre la menstruación. La app proporcionará recursos educativos que explican el proceso natural del ciclo menstrual, desmitificando tabúes y ofreciendo orientación sobre las mejores prácticas de higiene menstrual. Además, la plataforma incluirá una sección de preguntas frecuentes, respondidas con información confiable, actualizaciones sobre nuevas tecnologías y productos menstruales, como alternativas ecológicas, productos reutilizables, y las razones de su uso. Esto empodera a las mujeres y las jóvenes para tomar decisiones informadas sobre su salud y bienestar.

Apoyo a comunidades en desarrollo

Muchas mujeres en comunidades en desarrollo enfrentan barreras significativas para acceder a los productos y servicios básicos relacionados con la menstruación. La falta de acceso a toallas sanitarias, educación adecuada y apoyo comunitario agrava la pobreza menstrual y perpetúa la desigualdad de género. Además, las limitaciones económicas y sociales en estas comunidades hacen que las mujeres y niñas sean más vulnerables a la exclusión y marginación.

Solución Propuesta

“Todas brillamos”, no sólo busca educar por medio de esta aplicación móvil, sino también generar un impacto directo en estas comunidades. Por cada toalla sanitaria que se compre a través del app, se destinará un porcentaje de las ganancias para apoyar programas que proporcionen productos menstruales gratuitos a mujeres y niñas en comunidades en desarrollo. Este enfoque no solo asegura que más mujeres y niñas tengan acceso a productos menstruales, sino que también contribuye a la mejora de sus condiciones de vida al apoyar el desarrollo comunitario.

Además, se fomentará la creación de espacios dentro del app donde se resuelvan dudas y temas preguntados por los mismos usuarios, resolviendo dudas frecuentes y temas relevantes. Así podemos asegurar que nuestros usuarios entiendan la importancia de la higiene menstrual, cómo utilizar los productos de manera efectiva, y más.

Requerimientos funcionales.

Crear un nuevo usuario:

- El usuario debe poder crear una cuenta utilizando su cuenta de correo.
- El sistema debe registrar información llenada por el usuario como Nombre, teléfono, contraseña, correo, fecha de nacimiento, localidad, dirección de envío y CURP.

Inicio de sesión:

- El usuario deberá colocar únicamente su teléfono y su contraseña para acceder a su cuenta desde el app.
- En caso de que olvide su contraseña, habrá una opción de mandar enlace al correo de la cuenta para poder cambiara la contraseña.

Gestión del perfil:

- El usuario debe poder cambiar su contraseña desde la aplicación.

Consultas de información:

- El usuario debe poder acceder a artículos educativos sobre la menstruación y salud sexual.

Gestión del Ciclo menstrual:

- La aplicación debe proporcionar predicciones del próximo ciclo basado en los datos ingresados por el usuario.
- El usuario debe poder recibir recordatorios relacionados con su ciclo menstrual, como próximos períodos o recordatorios de tomar medicamentos.

Tienda de Productos:

- El usuario debe poder ver una lista de productos menstruales disponibles para la compra, incluyendo descripciones, precios y disponibilidad.
- El usuario debe poder seleccionar productos y añadirlos a un carrito de compras.
- La aplicación debe permitir al usuario completar la compra utilizando Stripe como método de pago.

Patrocinadores:

- El usuario debe poder acceder a la información de cuenta de banco para poder depositar el monto que quiere dar hacia la fundación.

Gestión de contenido (Administrador):

- El administrador debe poder agregar, editar o eliminar artículos, videos, y recursos educativos en la plataforma.
- El administrador debe poder gestionar la lista de productos menstruales, actualizando la disponibilidad y precios.

Sopporte y atención al usuario:

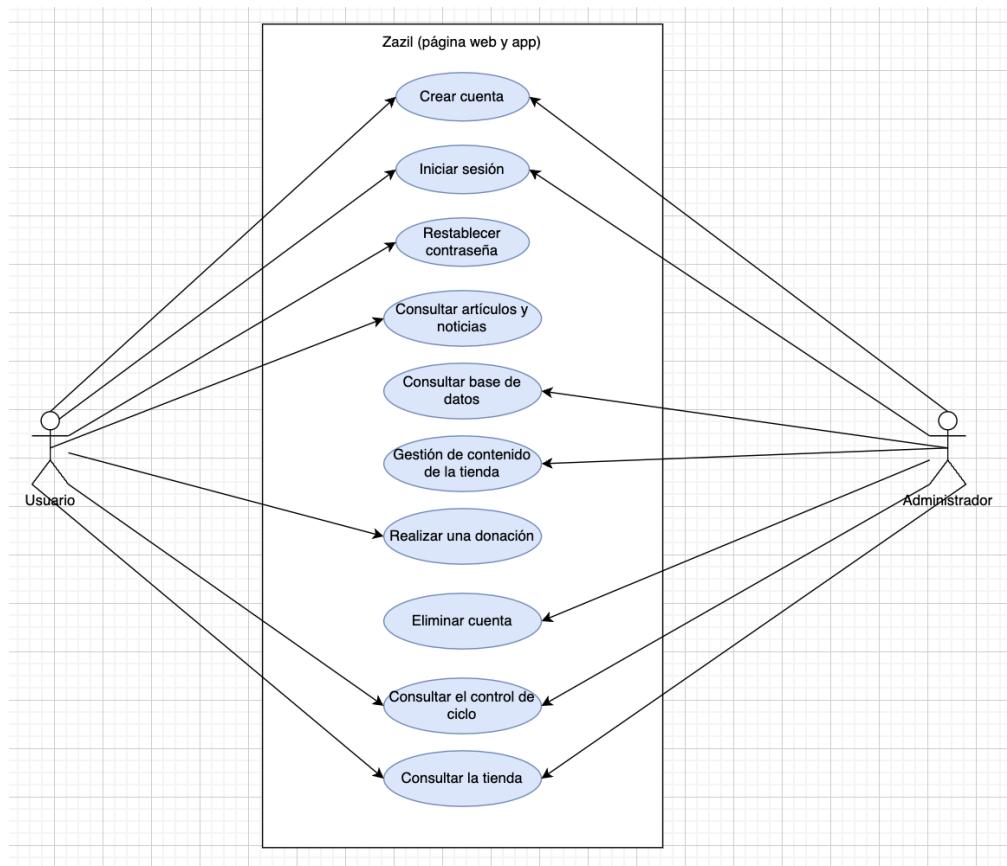
- El usuario podrá ponerse en contacto con Zazil a través de sus redes sociales y surge algún problema técnico.

Requerimientos no funcionales.

- **Plataforma:**
 - Para el desarrollo de la aplicación se utilizó la plataforma Android studio.
 - Para la página web, se utilizó **Visual Studio** para su desarrollo y gestionada con **Create React App**.
- **Firebase Authentication:** Autenticación segura basada en tokens para gestionar sesiones de usuario y administrador.
- **bcrypt:** Encriptación de contraseñas para garantizar la seguridad de las credenciales de los usuarios.
- **Selección de hardware seguro (servidor, proveedor de nube):** El hardware que se usará se compone de las computadoras personales de los integrantes del equipo.
- **HTTPS (TLS/SSL):** Comunicación encriptada para proteger la integridad y confidencialidad de los datos.
- **Stripe:** Procesamiento seguro de pagos en línea.
- **Selección de software seguro .**
 - Android studio como entorno de desarrollo de la aplicación.
 - Visual Studio como entorno de desarrollo de la página web.
 - Firestore y Realtime Database para la base de datos.
 - Firebase como servidor web.

Modelo de Casos de uso (Diagrama y Casos de uso extendidos) e Historias de usuario.

- Actores:
- **Usuario:** Cualquier persona que utilice la app para obtener información sobre la menstruación o para comprar productos menstruales.
- **Administrador:** Persona encargada de gestionar el contenido de la app, los productos, y las donaciones.



Casos de Uso Extendidos

1. Crear una cuenta

Caso de uso: Crear una cuenta
Actores: Usuario y Administrador del sistema
Objetivo: Crear una cuenta en la aplicación móvil para un usuario nuevo.
Descripción general: Un usuario quiere crear una cuenta en esta aplicación, el sistema se encargará de recopilar sus datos y validar que su información sea válida y correcta.

Precondiciones: El usuario ha iniciado el proceso de crear una cuenta y cuenta con toda la información requerida.

Referencia cruzada: RF1, RNF1, RNF2

Curso típico del evento

<i>Acción del actor</i>	<i>Respuesta del sistema</i>
1. El usuario da clic en el botón de crear cuenta.	
2. Ingresa la información que la aplicación le solicita.	3. La aplicación registra la información del usuario.
	4. La aplicación valida que la información sea correcta.
	5. La aplicación crea y registra la cuenta.
6. El usuario tiene acceso a la aplicación	

Alternativas de curso:

Si el usuario decide no terminar el proceso de crear la cuenta.

4.1 La aplicación no registra la información del usuario y el proceso se cancela.

Si el usuario no puede verificar su información como correo electrónico.

3.1 La aplicación no puede completar el proceso de crear cuenta, el usuario deberá intentarlo nuevamente; si vuelve a suceder deberá mandar mensaje a la asociación.

2. Iniciar sesión

Caso de uso: Iniciar sesión

Actores: Usuario y Administrador del sistema.

Objetivo: Inicio de sesión de un usuario ya registrado.

Descripción general: Procedimiento a seguir del usuario, ya sea una persona menstruante, no menstruante o un administrador del sistema, al intentar iniciar sesión en la aplicación.

Precondiciones: El usuario ya tiene un teléfono y contraseña válidos registrados en la aplicación.

Referencia cruzada: RF2, RNF3, RNF4

Curso típico del evento

Acción del actor	Respuesta del sistema
1. El usuario ingresa su nombre de usuario y contraseña. 3. El usuario puede acceder a su cuenta en la aplicación.	2. La aplicación verifica que la información sea correcta.

Alternativas del curso:

Si el usuario no tiene una cuenta

1-3 El usuario deberá crear una cuenta desde cero.

Si el usuario ingresa los datos erróneos, se bloqueará su cuenta.

1-2. El usuario tendrá tres intentos para poder iniciar sesión, si los datos son erróneos o no coinciden con ningún usuario registrado, la aplicación bloqueará a ese usuario y deberá pedir ayuda al administrador para poder tener acceso nuevamente.

Si el usuario no recuerda su contraseña, puede solicitar restablecerla.

1.1 El usuario podrá pasar por un método de verificación para autenticar que es el propietario de la cuenta y seguirá un proceso para restablecer su contraseña.

3. Restablecer Contraseña

Caso de uso: Restablecer Contraseña

Actores: Usuario.

Objetivo: Restablecer la contraseña en caso de olvido.

Descripción general: Si un usuario no recuerda su contraseña, puede solicitar un enlace para cambiarla. El sistema envía un correo de verificación y permite al usuario ingresar una nueva contraseña.

Precondiciones: El usuario ya ha creado una cuenta.

Referencia cruzada: RF2, RNF3	
Curso típico del evento	
<i>Acción del actor</i>	<i>Respuesta del sistema</i>
1. El usuario selecciona "¿Olvidaste tu contraseña?". 3. El usuario establece una nueva contraseña.	2. El servidor manda un correo indicando las instrucciones para crear nueva contraseña.. 4. El sistema guarda los nuevos datos.
Alternativas del curso: Si el correo no coincide con una cuenta registrada: 1-2 El sistema muestra un error y solicita intentar nuevamente. Si el enlace de recuperación expira: 2-3 Se debe solicitar un nuevo enlace.	

4. Consultar Artículos y Noticias

Caso de uso: Consultar Artículos y Noticias	
Actores: Usuario.	
Objetivo: Permitir que los usuarios accedan a artículos y noticias sobre salud.	
Descripción general: El usuario puede acceder a la sección "Hablemos de..." para leer publicaciones sobre diversos temas relacionados a la cultura del ciclo menstrual.	
Precondiciones: El usuario ha iniciado sesión en la aplicación. El usuario está en la sección de "Hablemos de...".	
Referencia cruzada: RF4, RNF1	
Curso típico del evento	
<i>Acción del actor</i>	<i>Respuesta del sistema</i>
1. El usuario accede a "Hablemos de...".	2. La aplicación muestra los artículos más recientes.

- | | |
|--------------------------------------|--|
| 3. El usuario revisa la información. | |
|--------------------------------------|--|

Alternativas del curso:

Si no hay nuevas publicaciones
 2.1 El sistema muestra las publicaciones más recientes disponibles.

5. Consultar base de datos

Caso de uso: Consultar base de datos.						
Actores: Administrador del sistema.						
Objetivo: Consultar los datos de registro de los usuarios, los productos de la tienda virtual y las ventas realizadas.						
Descripción general: El administrador accede a la base de datos para gestionar o analizar la información almacenada.						
Precondiciones: El administrador ha accedido a la base de datos en su dispositivo.						
Referencia cruzada: RF8, RNF1, RNF5						
Curso típico del evento						
<table border="1"> <thead> <tr> <th><i>Acción del actor</i></th> <th><i>Respuesta del sistema</i></th> </tr> </thead> <tbody> <tr> <td>1. El administrador abre la base de datos desde el servidor.</td> <td>2. El servidor muestra los datos de la base de datos.</td> </tr> <tr> <td>3. El administrador realiza consultas o modificaciones según sea necesario.</td> <td>4. El sistema ejecuta las consultas o cambios y muestra los resultados</td> </tr> </tbody> </table>	<i>Acción del actor</i>	<i>Respuesta del sistema</i>	1. El administrador abre la base de datos desde el servidor.	2. El servidor muestra los datos de la base de datos.	3. El administrador realiza consultas o modificaciones según sea necesario.	4. El sistema ejecuta las consultas o cambios y muestra los resultados
<i>Acción del actor</i>	<i>Respuesta del sistema</i>					
1. El administrador abre la base de datos desde el servidor.	2. El servidor muestra los datos de la base de datos.					
3. El administrador realiza consultas o modificaciones según sea necesario.	4. El sistema ejecuta las consultas o cambios y muestra los resultados					
Alternativas del curso:						
<p>Si hay un error en la consulta 3-4 El sistema muestra un mensaje de error y sugiere revisar la sintaxis o los parámetros de consulta.</p>						

6. Gestión de Contenido de la Tienda.

Caso de uso: Gestión de Contenido de la Tienda	
Actores: Administrador del sistema.	
Objetivo: Administrar los productos de la tienda, poder eliminar, agregar y modificar.	
Descripción general: El administrador puede gestionar los productos y agregar, modificar o eliminar contenido de los productos desde la plataforma web.	
Precondiciones: El administrador ingresa a la página web y tiene la información a agregar y/o modificar.	
Referencia cruzada: RF8, RNF1, RNF6	
Curso típico del evento	
<i>Acción del actor</i> 1. El administrador accede al panel de gestión en la página web. 2. El administrador modifica los productos o agrega uno nuevo. 3. El administrador da click en “Agregar/Guardar”.	<i>Respuesta del sistema</i> 4. La página web guarda los cambios. 5. La aplicación se actualiza con la nueva información.
Alternativas del curso: Si la conexión con la base de datos falla 4-5 El administrador no puede realizar cambios hasta que se restablezca la conexión.	

7. Realizar una Donación.

Caso de uso: Realizar una Donación.
--

Actores: Usuario.								
Objetivo: Realizar una donación para apoyar comunidades vulnerables pero recibiendo información por parte de la fundación sobre cómo ayudó esa aportación.								
Descripción general: El administrador realiza publicaciones sobre el ciclo menstrual y anuncios sobre los nuevos productos que llegarán pronto o ya están disponibles en la tienda virtual.								
Precondiciones: El usuario ha iniciado sesión en la aplicación y revisa la sección de “Patrocinadores”.								
Referencia cruzada: RF7, RNF1								
Curso típico del evento								
<table border="1"> <thead> <tr> <th><i>Acción del actor</i></th> <th><i>Respuesta del sistema</i></th> </tr> </thead> <tbody> <tr> <td>1. El usuario está en la sección de “Patrocinadores”.</td> <td></td></tr> <tr> <td>2. El usuario elige un monto para donar.</td> <td></td></tr> <tr> <td>3. El usuario realiza la donación al depositar en la cuenta de banco.</td> <td>4. El sistema registra la donación.</td></tr> </tbody> </table>	<i>Acción del actor</i>	<i>Respuesta del sistema</i>	1. El usuario está en la sección de “Patrocinadores”.		2. El usuario elige un monto para donar.		3. El usuario realiza la donación al depositar en la cuenta de banco.	4. El sistema registra la donación.
<i>Acción del actor</i>	<i>Respuesta del sistema</i>							
1. El usuario está en la sección de “Patrocinadores”.								
2. El usuario elige un monto para donar.								
3. El usuario realiza la donación al depositar en la cuenta de banco.	4. El sistema registra la donación.							
Alternativas del curso: Si el sistema no registra la donación 3-4 El usuario se deberá poner en contacto con el administrador de Zazil.								

8. Eliminar cuenta

Caso de uso: Eliminar cuenta
Actores: Administrador del sistema.
Objetivo: El Administrador puede eliminar una cuenta por falta de uso, comportamientos inapropiados y ofensivos.
Descripción general: Cuando el Administrador determina que un usuario no ha estado activo durante un tiempo considerable y/o ha cometido un acto inapropiado, la cuenta quedará eliminada y le informará al usuario.

Precondiciones: El Administrador inicia sesión; revisa la actividad del usuario.	
Referencia cruzada: RF8, RNF1, RNF5	
Curso típico del evento	
<i>Acción del actor</i>	<i>Respuesta del sistema</i>
<p>1. Después de haber analizado la cuenta del usuario, se determina si se debe eliminar o no.</p> <p>2. El Administrador se dirige a la función de “Eliminar cuenta”.</p> <p>3. Selecciona la cuenta que va a eliminar e informa el porqué.</p> <p>5. El Administrador confirma que la cuenta ha sido eliminada.</p>	<p>4. La aplicación registra la solicitud de eliminar la cuenta.</p> <p>6. La aplicación elimina la cuenta.</p> <p>7. La aplicación informa al usuario que su cuenta ha sido eliminada y las razones por las cuales se eliminó.</p>
Alternativas del curso:	
<p>Si se desea eliminar una cuenta que previamente ha sido eliminada.</p> <p>3-4. El administrador recibe notificaciones de la solicitud, se revisará en la base de datos algún registro sobre esa cuenta y no se podrá volver a eliminar a menos que se haya dado de alta nuevamente.</p>	

9. Consultar el control del ciclo

Caso de uso: Consultar el control del ciclo
Actores: Usuario y Administrador del sistema.
Objetivo: Los usuarios pueden registrar la duración de su ciclo menstrual, los síntomas e indicar el tipo de flujo; el administrador se asegurará de que esa información se esté guardando en la aplicación y base de datos.
Descripción general: La función del “Control del ciclo” ayuda a los usuarios a tener un control sobre su ciclo menstrual.

Precondiciones: El usuario y administrador han iniciado sesión en la aplicación.	
Referencia cruzada: RF5, RNF1	
Curso típico del evento	
<i>Acción del actor</i>	<i>Respuesta del sistema</i>
<p>1. El usuario inicia sesión en la aplicación.</p> <p>2. El usuario accede a la pestaña “Control del ciclo”.</p> <p>5. El usuario podrá registrar la información correspondiente a su ciclo actual.</p> <p>7. El administrador asegurará que esa información se guarde en la base de datos para darle continuidad al ciclo del usuario.</p>	<p>3. La aplicación muestra el calendario con la fecha reciente para registrar su nuevo ciclo.</p> <p>4. El usuario navega por el calendario para ver los ciclos anteriores y registrar uno nuevo.</p> <p>6. El sistema guardará la información en una base de datos.</p>
Alternativas del curso:	
<p>Si el usuario no puede visualizar sus ciclos anteriores.</p> <p>4. 1 El usuario deberá notificar al administrador para que revise la base de datos del sistema y se asegure que los datos se guardaron pero no se están reflejando o no se guardaron los datos y notificar al usuario.</p> <p>Si la aplicación no guarda los datos del ciclo.</p> <p>5.1 El administrador deberá revisar el algoritmo de la aplicación y notificar al usuario sobre los posibles problemas técnicos.</p>	

10. Consultar la tienda

Caso de uso: Consultar la tienda virtual
Actores: Usuario y Administrador del sistema.
Objetivo: Los usuarios pueden consultar y comprar los productos existentes en la tienda; el

administrador podrá gestionar los productos y lo relacionado con el proceso de compra.

Descripción general: La tienda virtual es un espacio donde los usuarios pueden consultar y adquirir los productos proporcionados por la empresa de *Zazil*; el administrador podrá actualizar información como la cantidad de productos, imágenes, precios, descuentos, etc..

Precondiciones: El usuario ha iniciado sesión en la aplicación y se dirige a la sección de tienda virtual. El administrador ha iniciado sesión, revisa la información con la base de datos y ajusta la información necesaria respecto a los productos.

Referencia cruzada: RF6, RNF1, RNF2, RNF5

Curso típico del evento

<i>Acción del actor</i>	<i>Respuesta del sistema</i>
1. El usuario inicia sesión en la aplicación.	
2. El usuario accede a la sección de “Tienda”.	3. La aplicación muestra los productos con su descripción y costo.
4. El usuario selecciona los productos que desea comprar, selecciona el método de pago y la cantidad de productos.	4. La aplicación procesa la información de compra y procesa el método de pago.
6. El administrador puede cambiar información del producto como descuentos.	5. La aplicación valida el pago, se realiza exitosamente la compra y se notifica al usuario y administrador. 7. La aplicación actualiza la información de los productos.

Alternativas del curso:

Si el método de pago no es aprobado.

4. 1 La aplicación notificará que el pago no fue procesado, se deberá intentar nuevamente o con un método diferente.

Si la aplicación no actualiza los datos de los productos.

6-7 El administrador deberá estar revisando que la información se haya guardado, sino realizar alguna notificación al usuario y aplicar cupones en los descuentos que no se apliquen automáticamente.

Alcance de la aplicación

Se va a desarrollar una aplicación para dispositivos android, la cual va a contar con una página de bienvenida con el logo, seguido de (si es la primera vez que se instala) un registro o inicio de sesión por medio de correo electrónico o google, si no entonces se muestra **el logo**, seguido de las siguientes secciones:

- **Página principal** donde se mostrarán las noticias sobre salud sexual, nuevos productos, descuentos, novedades.
- **Una tienda** donde se mostrarán los productos, precio y disponibilidad.
- Distintos **métodos de pago**: tarjeta o transferencia.
- Un **calendario menstrual**: en el cual se podrá realizar un seguimiento del ciclo del usuario y predicción del siguiente ciclo
- **Información sobre el tema**: donde se tenga la información básica
- **Perfil** donde se encontrarán los datos personales del usuario
- **Configuración** en caso de querer cambiar datos de usuario
- **Notificaciones** que dependen de si es la persona menstruante o su compañero

Junto con una página, la cual será el backend de la aplicación, contando con:

- **Vista de cliente**
 - juntar todos los links en un lugar
 - dar a conocer la asociación
 - su historia, misión, visión, objetivos
 - sus acciones
 - sus proyectos
 - integrantes del proyecto
 - formas de ponerse en contacto con la asociación
 - instrucciones para ayudar
 -
- **Vista de administrador**
 - los datos de la aplicación
 - almacenar los datos de la aplicación
 - administrar los datos

- impacto de la ayuda

Matriz de riesgos

Tener una matriz de riesgos en esta primera fase del proyecto fue fundamental porque nos permitió identificar y evaluar posibles obstáculos que podrían afectar el éxito del proyecto. Al anticipar riesgos relacionados con la definición de requerimientos, la organización del equipo, y el diseño del prototipo, pudimos desarrollar estrategias de mitigación desde el principio. Esto no solo nos está preparando mejor para enfrentar imprevistos, sino que también nos ayuda a tomar decisiones más informadas y a priorizar tareas críticas. En resumen, la matriz de riesgos es un herramienta clave para asegurar que el proyecto avance de manera controlada y eficiente.

	IDENTIFICACIÓN			ANÁLISIS		EVALUACIÓN	TRATAMIENTO	
	Sistemas	Proceso	Riesgo	Probabilidad de que ocurra	Severidad del daño	Jerarquización de impacto	Control	Encargado
1	Aplicación	Definición de la dirección y asignación de tareas	No desarrollar los componentes correctamente	2	3	3	Aumentar la comunicación con el equipo	Líder de equipo
2	Aplicación	Desarrollo completo de cada componente	Componentes incompletos debido a la falta de tiempo	1	2	2	Aumentar la confianza y fomentar la comunicación de cada integrante del equipo, por medio de reuniones periódicas para dar a conocer fallas y poder encontrar una solución	Todo el equipo
3	Aplicación	Integración de todos los componentes y recursos para un funcionamiento correcto del proyecto y el cumplimiento de las especificaciones con la organización socioformadora	Fallas al integrar el proyecto debido a que el desarrollo de los componentes no van dirigidos al mismo objetivo	2	3	3	Tener bien definida la dirección, alcances, objetivos del proyecto y de cada componente que lo constituye, asegurando que cada integrante del equipo lo entienda. De igual forma definir con exactitud cada tarea a realizar (evitando ambigüedades) y el objetivo y los límites que tendrá (para que no desarrollen ni más ni menos).	Todo el equipo
4								

Gantt/Cronograma del reto

Estamos utilizando esta herramienta para gestionar y distribuir el proyecto de manera eficiente y estructurada, lo que nos permite cumplir con los plazos establecidos y las expectativas de nuestros socio formadores. El uso del diagrama de Gantt nos proporciona una visión clara del progreso de cada tarea, lo que facilita la identificación de posibles bottle-necks y nos permite tomar decisiones proactivas para mantener el proyecto en marcha. Además, esta herramienta nos ayuda a dividir el trabajo de manera equitativa entre los miembros del equipo, asegurando que cada uno tenga una carga de trabajo manejable y pueda contribuir de manera efectiva al éxito del proyecto. Gracias a esta organización, no solo mantenemos un alto nivel de coordinación, sino

que también fomentamos un ambiente de trabajo motivador y colaborativo, donde todos estamos alineados hacia un objetivo común.

Tr Actividad	Prio	Encargado	Estatus	Fecha de inicio	Fecha de entrega	Tr Entregables	Tr Notas
Definir las características principales de la página	P3	Equipo	Completed	8/16/2024	8/19/2024	Documento con características y componentes de la página	Tener las características y componentes necesarios para comenzar el boceto de la página
Definir las características principales de la aplicación	P3	Equipo	Completed	8/16/2024	8/19/2024	Documento con características y componentes de la aplicación	Tener las características y componentes necesarios para comenzar el boceto de la aplicación
Diseño de la aplicación	P3	Equipo	Completed	8/17/2024	8/19/2024	Documento con imágenes del diseño	Tener el diseño de la aplicación para la detección de componentes necesarios a implementar
Diseño de la página	P3	Equipo	Completed	8/17/2024	8/19/2024	Documento con imágenes del diseño	Tener el diseño de la página para la detección de componentes necesarios a implementar
Prototipo no funcional de la aplicación	P2	Ian Alexei Martí...	Completed	8/18/2024	8/20/2024	Presentación con partes de la aplicación	Fin de tener correcciones de la asociación
Prototipo no funcional de la página	P2	Lorena Abigail ...	Completed	8/18/2024	8/20/2024	Presentación con partes de la página	Fin de tener correcciones de la asociación
Correcciones de la junta	P3	Lorena Abigail ...	Completed	8/20/2024	8/21/2024	Corrección de las observaciones de la asociación en las presentaciones de la página y de la aplicación	Tener el diseño base ya aceptado
Documento ReqSoft	P3	Equipo	In progress	8/16/2024	8/23/2024	Documento completo con correcciones	Para comenzar la documentación del proyecto
Propuesta de base de datos	P2	Alejandra Estef...	Not started	8/21/2024	8/23/2024	Diagrama propuesta de base de datos	Conexión página-aplicación
Finalización del curso de node.js	P3	Alejandra Estef...	In progress	8/19/2024	8/27/2024	Curso finalizado	Notas
Elección de mascota de la aplicación	P1	Equipo	In progress	8/22/2024	8/23/2024	Imagen de la mascota de la aplicación	Notas
Presentación de la página al socio formador	P3	Lorena Abigail ...	Completed	8/19/2024	8/21/2024	Comentario del socioformador aceptando el prototipo	Notas
Página de presentación (app)	P3	Lore y Alex	Not started	8/24/2024	8/26/2024	Página donde se presente la asociación	Notas
Login (app)	P2	Lore y Alex	Not started	8/24/2024	8/26/2024	Página donde los usuarios se registren	Notas
Página de inicio (app)	P2	Lore y Alex	Not started	8/25/2024	8/28/2024	Página donde ya se muestre la página que elegimos como inicio	Notas
Vista de administrador página	P3	Sandy y Fer	Not started	8/22/2024	8/26/2024	Vista del administrador en la página	Para mostrarlo a la asociación
Splash screen con el logo de zazil	P3	Lore y Alex	Not started	8/23/2024	8/24/2024	La vista de la splash screen cuando el usuario abre la app	Notas

Column C	August 2024													
	Mo 16	Tu 17	Sa 18	Sun 19	Mon 20	Tue 21	Wed 22	Thu 23	Fri 24	Sat 25	Sun 26	Mon 27	Tue 28	
Equipo	<div style="background-color: #90EE90; border: 1px solid black; padding: 2px;">Definir las características principales de la página Aug 16 - Aug 19</div> <div style="background-color: #90EE90; border: 1px solid black; padding: 2px;">Definir las características principales de la aplicación Aug 16 - Aug 19</div> <div style="background-color: #90EE90; border: 1px solid black; padding: 2px;">Diseño de la aplicación Aug 17 - Aug 19</div> <div style="background-color: #90EE90; border: 1px solid black; padding: 2px;">Diseño de la página Aug 17 - Aug 19</div> <div style="background-color: #FFFFCC; border: 1px solid black; padding: 2px;">Documento ReqSoft Aug 16 - Aug 23</div> <div style="background-color: #FFFFCC; border: 1px solid black; padding: 2px;">Elección de mascota de la aplicación Aug 22 - Aug 23</div>													
Ian Alexei Martínez...														
Lorena Abigail Solís de los Santos														
Alejandra Estefanía Rico González														
Lore y Alex														
Sandy y Fer														

Aportaciones en Primera Etapa

- Equipo: Como equipo, realizamos una lluvia de ideas en la que establecimos la visión general de nuestra aplicación y las ideas principales detrás del proyecto. En ese proceso, cada miembro del grupo aplicó su experiencia y contribuyó con ideas para el diseño y las funciones de la aplicación.

Estamos investigando sobre los requisitos que debemos cumplir para hacer que la aplicación sea legal y segura en términos de recopilación de datos sensibles de los usuarios. También estamos investigando y elaborando el contenido que necesitamos para la aplicación de manera que pueda ser entregado de manera ética a los usuarios finales.

El desarrollo del prototipo fue también una de las cosas críticas de este paso, mediante el cual pudimos discutir el real potencial de la aplicación y hacer recomendaciones de mejora en base a las sugerencias e ideas que se extrajeron durante el proceso de trabajo colaborativo. La etapa nos permitió revisar algunas cosas para poder hacer una implementación más detallada de lo que vamos a hacer en las siguientes fases de desarrollo.

Hemos sido capaces, a través de un esfuerzo personal y de equipo en áreas clave, de poner en su lugar la estructura para la aplicación: sólida, práctica y enfocada en el proyecto.

- Fernanda: En equipo hicimos una lluvia de ideas en la cual yo aporté ideas acerca del diseño y contenido de la aplicación, mi contribución individual al proyecto se centró en la definición y el análisis de los requerimientos no funcionales. Consideré la importancia de asegurar que la plataforma cumpla con las normativas legales y de seguridad, especialmente en el manejo de datos sensibles relacionados con la salud de los usuarios. También investigué las mejores prácticas para seleccionar hardware y software seguros que garanticen la

integridad y la disponibilidad de la aplicación, protegiendo la información personal y de salud de los usuarios.

- Sandra: De forma individual, yo realicé algunas anotaciones durante la presentación del socio formador sobre algunos de los requerimientos de la aplicación, en conjunto realizamos una lluvia de ideas donde aporté sugerencias de acuerdo a algunas aplicaciones ya hechas y después de haber revisado las redes sociales del socio formador.
- Estefanía: Durante esta primera fase del proyecto, como equipo, contribuimos al proceso de brainstorming para proponer nuestra idea de prototipo, la cual desarrollaremos tanto individualmente como en colaboración. En mi trabajo individual, me dediqué a recopilar información clave, tal como lo indicó el socio formador, para asegurar su adecuada difusión a través de este proyecto. Además, trabajé en las propuestas sobre cómo incorporar esta información en la aplicación móvil. También desarrollé los casos de uso de la aplicación para optimizar su funcionalidad y alinearla con los objetivos del proyecto.
- Lorena: De forma individual recopilé las tareas que como equipo realizaríamos para implementarlo en el diagrama de gantt, también después de discutirlo como equipo realicé la matriz de riesgos de nuestro proyecto. También realicé el prototipo de la página web y se la presenté a la asociación mediante un video explicándole cada componente, parte y función. En equipo valoramos los que pidió la asociación, buscamos e investigamos en sus redes sociales y concluimos que requerían de una página web. También como equipo por medio de la lluvia de ideas buscamos una forma de cumplir con los requerimientos de la asociación, pero a la vez buscar una forma de hacerla diferente al resto.

- Alexei: De forma individual realicé el prototipo de nuestra aplicación donde mostramos la mayoría de las funciones que tendrá la aplicación junto con detalles a mejorar y algunos detalles que se eliminarán gracias a la lluvia de idea que realizamos después del prototipo.

¿Hubo autonomía, apertura intelectual y carácter propositivo de manera individual durante la actividad?

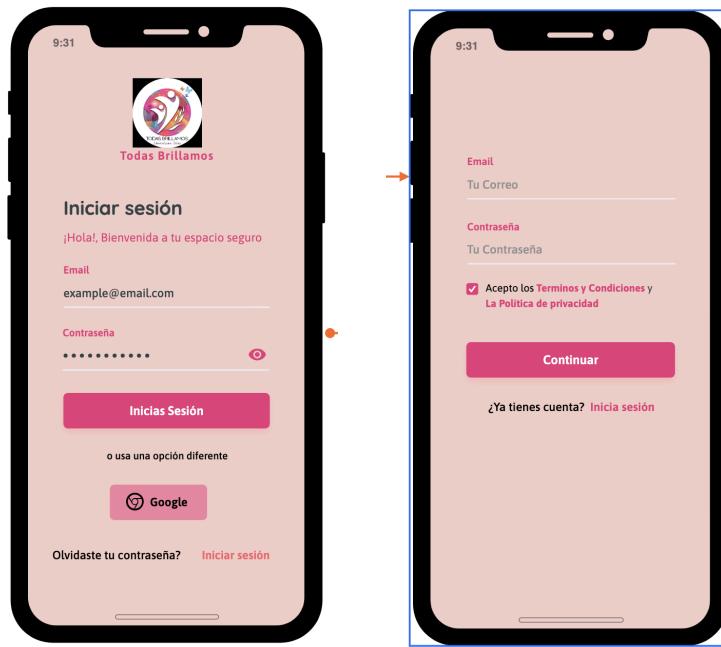
Sí, durante esta primera fase del proyecto, que incluyó la definición de requerimientos, la organización del equipo, la creación de diagramas de caso de uso y el diseño del prototipo, hubo una notable autonomía, apertura intelectual y carácter propositivo por parte de cada miembro. A pesar de ser un trabajo colaborativo, cada uno de nosotros asumió responsabilidades individuales que fueron esenciales para el desarrollo de esta etapa.

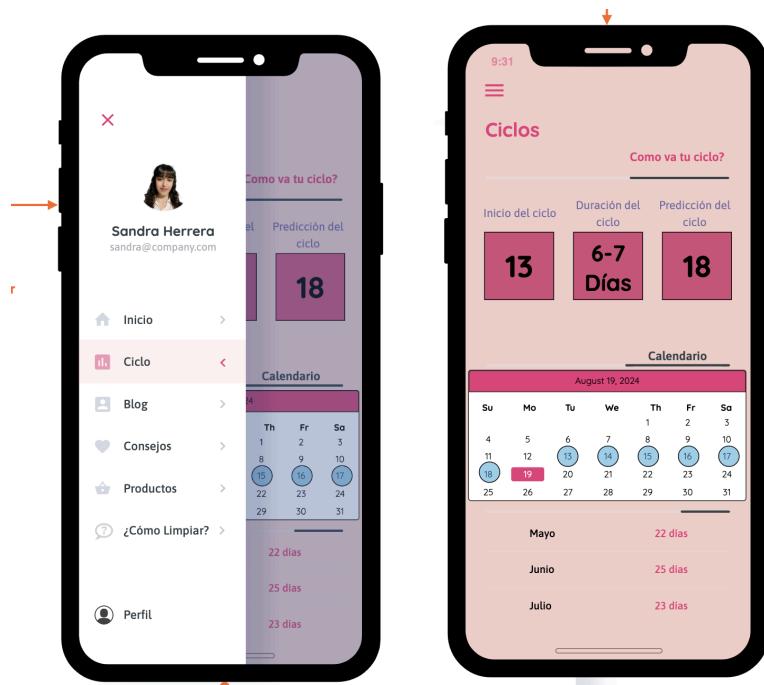
La autonomía se manifestó en la forma en que cada miembro se encargó de investigar y definir aspectos clave del proyecto de manera independiente, lo que nos permitió avanzar de manera eficiente en la definición de los requerimientos y en la estructuración de nuestras ideas. Esto no solo facilitó el trabajo en equipo, sino que también contribuyó a una mejor organización y claridad en los roles de cada uno.

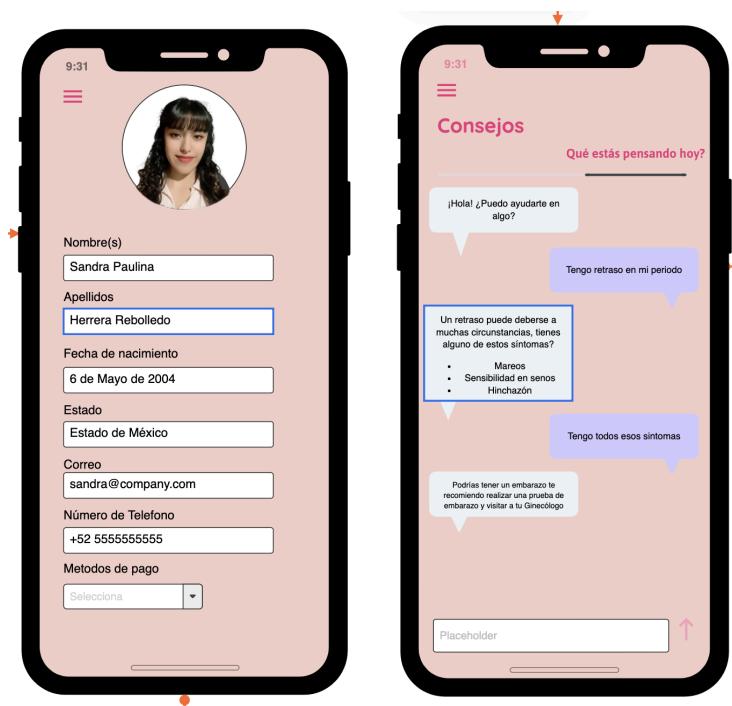
La apertura intelectual se vio reflejada en la disposición de cada integrante para considerar distintas perspectivas y enfoques, especialmente al momento de diseñar el diagrama de caso de usos y en la conceptualización del prototipo. Todos aportamos ideas que enriquecieron el proceso y nos ayudaron a tomar decisiones informadas y consensuadas.

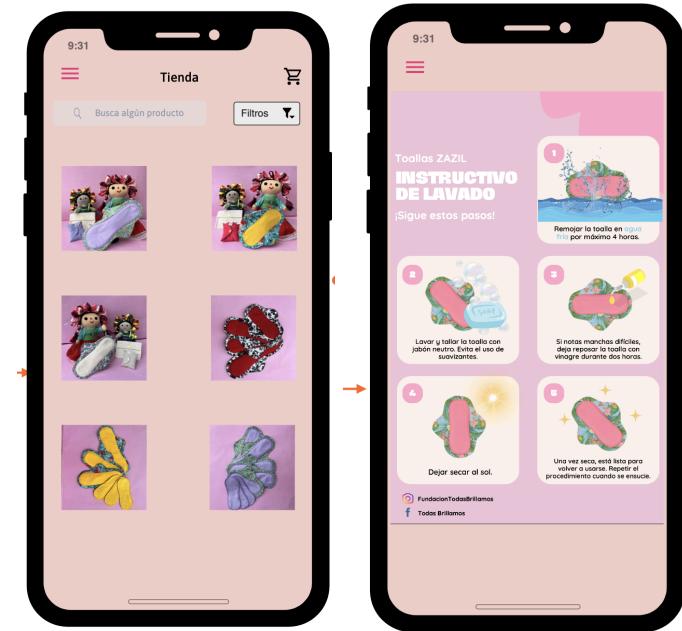
Finalmente, el carácter propositivo de cada uno fue clave en esta fase, ya que permitió que el proyecto avanzara con propuestas concretas y bien fundamentadas, tanto en el diseño del prototipo como en la organización del trabajo. Estas contribuciones individuales no solo fueron valoradas, sino que también jugaron un papel crucial en la definición de la dirección que tomará el proyecto en las próximas fases.

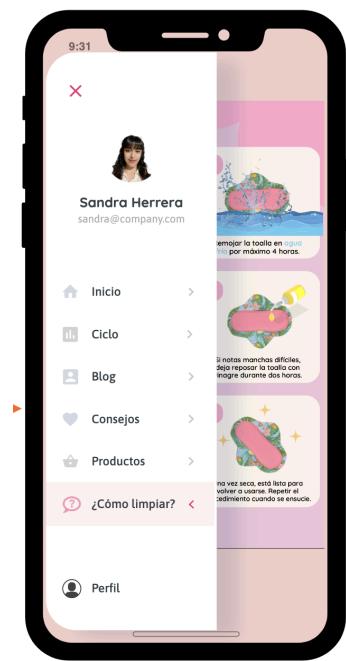
Prototipo del diseño de la Aplicación Móvil











Prototipo del diseño de la Página Web

https://a017466020.wixstudio.io/my-site-2 Conecta tu dominio

 Todas Brillamos

Historia Marca ZAZIL Actividades [Contactar](#) [Iniciar Sesión](#)

✨ Todas Brillamos ✨

¡Libertad para Todas!

Somos una organización dedicada a empoderar a mujeres y niñas mediante educación, salud y desarrollo personal. Proporcionamos productos de higiene menstrual y educación para promover la igualdad de género y la autonomía femenina.



Sobre Nosotros

"Somos la Fundación Todas Brillamos AC, una organización sin fines de lucro y Donataria Autorizada comprometida con el cambio positivo en la sociedad mexicana. Nuestro enfoque se centra en la igualdad de género, el empoderamiento de las mujeres y la erradicación de la pobreza menstrual."

Trabajamos incansablemente para promover la igualdad, empoderar a las mujeres y asegurar que todas tengan acceso a productos de higiene menstrual.

Únete a nosotros en esta importante causa y juntos hagamos brillar a todas las personas*.



[Conoce nuestra historia](#)

"LIBERTAD PARA TODAS"

Es más que un lema; es la esencia de nuestra misión. Creemos que todas las personas tienen un brillo único y merecen vivir con libertad y dignidad.

Aquí algunos conceptos importantes.

Menstruación

Desprendimiento y eliminación normal de sangre y del tejido que reviste el útero, que ocurre cada mes, el flujo menstrual del útero pasa por el cuello uterino hasta la vagina y sale del cuerpo.

Pobreza menstrual

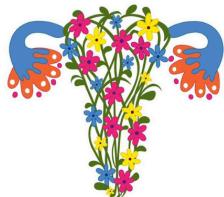
Se define como la incapacidad de permitirse productos para la higiene íntima femenina, es un problema grave, especialmente en los países en desarrollo.

Violencia menstrual

El estigma social y cultural sobre la menstruación como algo "sucio" o "impuro", es un factor más de la violencia de género hacia las personas menstruantes.

Sangrado consciente

Permite comprender los indicadores físicos y emocionales sentidos, y asociarlos a cada una de las cuatro fases del ciclo con el fin de realizar los cambios más adecuados para nosotras.



ZAZIL

Conoce nuestra marca **ZAZIL**

Zazil es una marca comprometida con el bienestar de las mujeres y el cuidado del medio ambiente. Su misión es proporcionar soluciones innovadoras y sostenibles para el periodo menstrual. ¿Cómo lo hacen? A través de la creación de toallas femeninas reutilizables.

[Leer más](#)

ACTIVIDADES

Septiembre 12, 2024



Aplicación para todos

Nuestra nueva aplicación piensa en todos, tanto en las personas menstruantes como en sus parejas. Algo innovador para incluir a todos en el tema de la menstruación y eliminar el tabú.

Junio 19, 2024



Zazil: Cambiemos el juego de la menstruación

Invitados a Cadena Política, conducido por Yoselin Velázquez, en facebook live.

Julio 12, 2024



Programa "Más por los demás"

Gracias a la entrega de cupones que [saludigina_mx](#) otorgó a la [fundaciontadasbrillamos](#), ahora es posible que las personas que necesiten realizar estudios de laboratorio para su salud lo hagan de forma gratuita.

Productos más populares

Zazil es una marca comprometida con el bienestar de las mujeres y el cuidado del medio ambiente.
¡Checa algunos de nuestros productos más solicitados!



Tualla Feminina Reutilizable
Regular Menstrua
\$150



Tualla Feminina Reutilizable
Nocturna Menstrua
\$180



12 Unidades
Períodos/Días
Reutilizables
\$40



Tualla Feminina Reutilizable
Nocturna Menstrua
\$180



Tualla Feminina Reutilizable
Menstrua
\$130



KIT ZAZIL 3
\$773.50 (envío)

NOTICIAS

Alimentos ricos en magnesio, como el chocolate oscuro y los frutos secos, pueden ayudar a reducir los calambres menstruales. Añádelos a tu dieta para aliviar el dolor

¿Relación entre la dieta y la menstruación?
Información que quizás podría ayudarte

Estudio realizado en Estados Unidos para medir los metales en los tampones demostró que contienen arsénico y plomo.

¿Metales pesados en tampones?
Información importante para tu salud

Cada año, millones de productos desechables de higiene menstrual terminan en vertederos y océanos, contaminando nuestro planeta. ¡Juntas podemos hacer una gran diferencia!

Toallas ¿800 años en desintegrarse?
Razones para optar por alternativas reutilizables

EVENTOS

Noviembre 12, 2023



Carrera con causa

Únete y participa en la carrera con causa que se llevará a cabo el 16 de diciembre en Ameyalco Querétaro.

Te esperamos con los brazos abiertos y con ganas de hacer un cambio. ❤️

Octubre 26, 2023



Festival turístico "Encuentros nacionales"

¡Qué emoción que **#Zazil** las toallas femeninas reutilizables, estará presente en el 1er festival turístico en Naucalpan de Juárez! No te puedes perder la oportunidad de conocer estas increíbles toallas que cuidan de ti y del medio ambiente. Así que guarda la fecha, el 27 y 28 de octubre, y ven al Parque Toreo. Estaremos ahí para mostrarte todas las ventajas de usar toallas femeninas reutilizables.

¡Te esperamos con los brazos abiertos!

Abrial 28, 2024



Festival cultural "Del vino y el queso"

Fundación **Todas Brillamos** AC, participando en el **Festival Cultural del VINO y el QUESO** en Naucalpan de Juárez. Los invitamos a disfrutar de este gran evento donde podrán aprovechar y encontrar a **Toallas Zazil**.



Contáctanos

Tel: +52 56 2808 3883
Mail: contaco@fundaciontodasbrillamos.org

Menú

[Historia](#)
[Marca ZAZIL](#)
[Actividades](#)

Política

[Política de Privacidad](#)

Redes sociales

[Facebook](#)
[Instagram](#)
[Tiktok](#)
[Youtube](#)

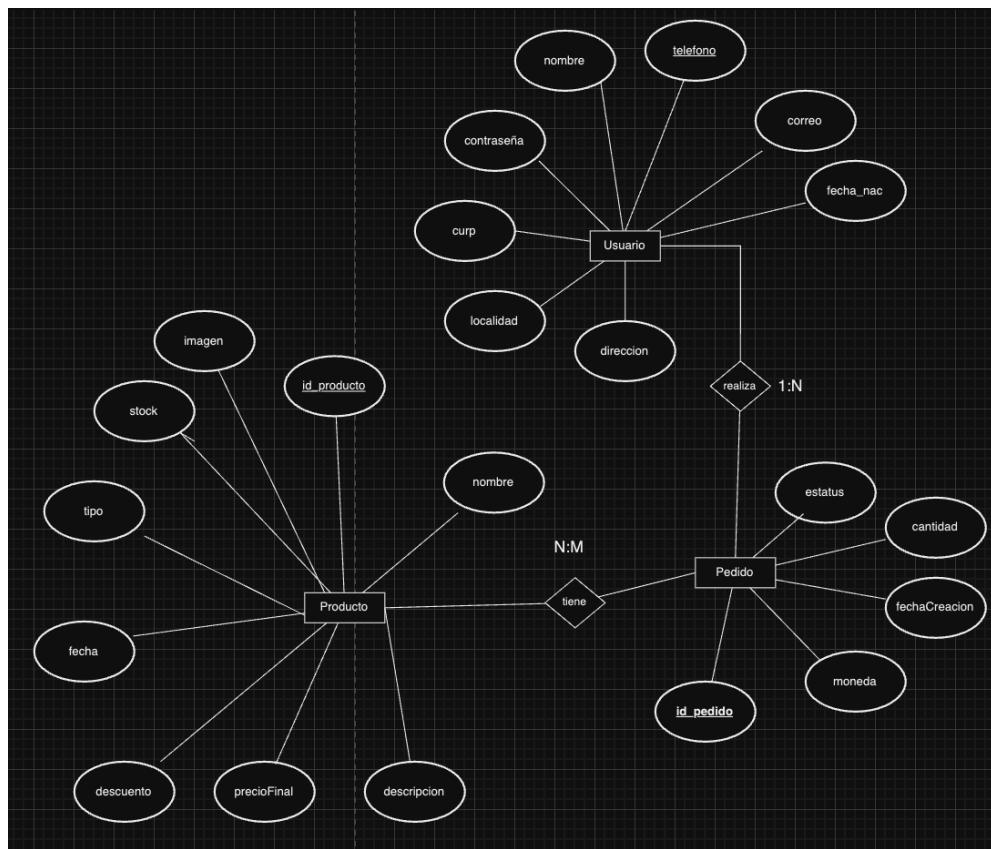
© 2035 por Nombre del negocio. Creado con [Wix Studio™](#)

Etapa 2. Diseño

Base de datos

Se desarrolló una base de datos en Firebase con el objetivo de almacenar y gestionar la información de los usuarios de manera eficiente, lo que permite que ‘TodasBrillamos’ pueda utilizar estos datos para alcanzar sus objetivos comerciales. Esta solución no solo facilita la organización de la información, sino que también habilita el análisis en profundidad de métricas clave, como la edad promedio de los usuarios, sus localidades, y otros datos esenciales que proporcionan una visión más clara sobre el perfil de los clientes. Cada campo incluido en la base de datos fue cuidadosamente discutido y evaluado en función de su relevancia para el análisis del negocio, garantizando que los datos recopilados apoyen decisiones estratégicas informadas y oportunas.

Diagrama Entidad - Relación Normalizado



Modelo Relacional

Usuario(correo, fecha_nac, nombre, contraseña, curp, localidad, telefono, direccion)

Producto(id_producto, nombre, descuento, precioFinal, tipo(ToallaPanti, ToallaTeen, ToallaNocturna, ToallaRegular, kits, pañales), descripción, fecha, stock, imagen)

Pedido(id_pedido, telefono, fecha_pago, estatus, moneda, cantidad, fechaCreacion)

Pedido_Producto(id_pedido, id_producto, cantidad)

Diccionario

Tabla Usuario

Nombre del atributo	Tipo de dato	Longitud	Descripción
<u>telefono</u>	int	25	Se trata de la llave primaria de la tabla usuario, que se va a requerir cada vez que el usuario haga inicio de sesión. (Llave primaria)
correo	varchar	255	Este dato se va a requerir cada vez que el usuario inicie una cuenta para posibles notificaciones.
contraseña	varchar	100	Contraseña que cada usuario asignará para su cuenta.
nombre	varchar	100	Nombre completo del usuario.
fecha_nac	date	-	Fecha de nacimiento del usuario.
curp	char	18	CURP del usuario (único para cada usuario).
localidad	varchar	100	El estado de México del usuario
direccion	varchar	100	Dirección completa del usuario, este dato

			es necesario para los envíos de la tienda online.
--	--	--	---

Tabla Pedido

Nombre del atributo	Tipo de dato	Longitud	Descripción
id_pedido	int	11	Identificador único del pedido (clave primaria).
fechaCreacion	date	-	Fecha en la que se realiza el pago del pedido.
estatus	varchar	50	El estatus en el que su pedido se encuentra
cantidad	decimal	10,2	Monto total del pedido.
moneda	varchar	255	Es el tipo de moneda con la que el usuario pago.
telefono	varchar	255	Telefono del usuario que realiza el pedido (clave foránea).

Tabla Producto

Nombre del atributo	Tipo de dato	Longitud	Descripción
id_producto	int	11	Identificador único del producto (clave primaria).
nombre	varchar	100	Nombre del producto.
precioFinal	decimal	10,2	Precio del producto en formato decimal, el calculo descontando el 'descuento' ya realizado.
tipo	enum	-	Tipo de producto (ToallaPanti , ToallaTeen , ToallaNocturna , ToallaRegular , kits , panales).
descripción	text	-	Descripción del producto.
stock	int	255	Cantidad del producto que se encuentra en el stock
fecha	timestamp		Fecha en la que se añadió o se actualizó el producto.
descuento	int	255	Se trata de la cantidad de descuento que tiene el producto
imagen	string	255	La imagen se registra como cadena de texto

			que contiene la URL de la imagen almacenada en Firebase Storage.
--	--	--	--

Tabla Pedido_Producto

Nombre del campo	Tipo de Dato	Longitud	Descripción
id_pedido	int	11	Identificador del pedido (clave foránea).
id_producto	int	11	Identificador del producto (clave foránea).
cantidad	int	4	Cantidad de unidades del producto en el pedido.

Consideraciones Generales del diseño de la Base de Datos

El diseño de la base de datos para la aplicación zazil ha sido optimizado para funcionar eficientemente en un entorno de aplicación móvil, proporcionando un equilibrio entre la capacidad de almacenamiento y el rendimiento del sistema. A continuación, se explican las razones detrás de las elecciones de tipo de dato y longitud para cada atributo en el modelo relacional:

1. *Uso de VARCHAR(255) para correos electrónicos y otros atributos de texto:*

VARCHAR(255) es una longitud estándar muy aceptada para almacenar correos y otros atributos de texto en aplicaciones. Hemos decidido que esta longitud es suficiente para la mayoría de los casos de uso y permite flexibilidad para futuros ajustes sin requerir modificaciones en la estructura de la base de datos.

Impacto en el rendimiento: Utilizar VARCHAR nos permite que la base de datos solo ocupe espacio en función de la longitud real de los datos, optimizando el almacenamiento.

2. *Uso de ENUM para el tipo de productos:*

Hemos usado ENUM para definir el tipo de producto (ToallaPanti, ToallaTeen, ToallaNocturna, ToallaRegular, kits, pañales) con el objetivo de garantizar la integridad de los datos y limitar los valores a opciones predefinidas. Esto evita errores al insertar datos y facilita la validación dentro de la aplicación.

Impacto en el rendimiento: ENUM mejora el rendimiento en las consultas al ocupar menos espacio, entre otras cosas.

3. *Uso de INT(11) para claves primarias y otros identificadores:*

INT(11) es una elección estándar para claves primarias en MySQL, proporcionando un rango suficiente para la mayoría de las aplicaciones sin incurrir en excesivo consumo de almacenamiento. El uso de AUTO_INCREMENT permite la gestión automática de identificadores únicos.

Impacto en el rendimiento: INT es un tipo de dato eficiente para operaciones de comparación y búsqueda, lo cual es crucial para mantener la rapidez en aplicaciones móviles.

4. *Uso de DECIMAL(10, 2) para precios y montos:*

DECIMAL(10, 2) los hemos utilizado para manejar valores monetarios, permitiendo hasta 8 dígitos enteros y 2 decimales. Este formato asegura que los cálculos financieros sean precisos, evitando problemas de redondeo que pueden ocurrir con otros tipos de datos numéricos.

Impacto en el rendimiento: Este tipo de dato es apropiado para aplicaciones que requieren

precisión en transacciones monetarias, como en el caso de precios de productos y totales de pedidos.

Optimización para Aplicación Móvil

El diseño de la base de datos ha sido pensado para optimizar la capacidad de respuesta de la aplicación móvil. Al utilizar tipos de datos adecuados y eficientemente ajustados, se minimiza la carga en las operaciones de inserción, actualización y consulta, asegurando así una experiencia de usuario fluida y un buen rendimiento de la base de datos.

Firebase evidencia de Base de Datos

Tabla de Usuarios

users
▼ — 5541840736
└—— birthdate: "17/10/1974"
└—— curp: "\$2a\$10\$Xo3WZVLXYecpdIjYUPW7wOleUGNJF8eLSw8o873B27uQt1kL0eA4G"
└—— direction: "Calle Hidalgo, Colonia El Parque, Municipio de Cancún, CP 77500"
└—— email: "yeniarmdz@gmail.com"
└—— location: "Quintana Roo"
└—— name: "Yeni"
└—— password: "\$2a\$10\$238.uVxvPs/cHT8H1jQk5.eElb7SUcQNykLno/1pUPHsnHqfA8eLG"
└—— phone: "5541840736"

Tabla de Productos

<p>+ Agregar documento</p> <p>1QX1ecnsmKbAZQnVdnrf ></p> <p>24wtdjUQpogebs0Y4ZcG</p> <p>3cmfP5Ka0drF50tbqchF</p> <p>5BnrwzVLpTzsrtQFPqri</p> <p>5xhzgEwmCcm8Is04HX0v</p> <p>8K2p2XAyQ2lpDvGRD7z1</p> <p>818EKhD952xNPcz7xcd</p> <p>9haiUH2UUTJECbFGRjd3</p> <p>9i6n1KAc9ySmhY2hFFMy</p> <p>A7AyZBX03BDqutX53S2m</p> <p>Areli 5 piezas</p> <p>Aynii</p> <p>DzH62rCij8oCydKb40Ln</p>	<p>+ Iniciar colección</p> <p>+ Agregar campo</p> <p>category: "Kits"</p> <p>description: "Lavable y reutilizable, fácil de limpiar y secar. Incluye: 2 toallas regulares 2 toallas nocturnas 1 toalla teen"</p> <p>discountAmount: "650.00"</p> <p>discountPercent: "0"</p> <p>img: "https://firebasestorage.googleapis.com/v0/b/zazil- daa11.appspot.com/o/1728521443584KitsArelip.png? alt=media&token=a56b27b9-aa56-4c78-ab2a- 03695548200d"</p> <p>price: "650"</p> <p>product: "Areli 5 piezas"</p> <p>stock: "5"</p> <p>timeStamp: 9 de octubre de 2024, 6:51:04 p.m. UTC-6</p>
--	--

Tabla de Pedidos

<p>+ Agregar documento</p> <p>0pyug0DQmt65kfKlfnO ></p> <p>1JMBLB9tG8Jd9VnH1ZQu</p> <p>1QKumkiQzKgzRDQcWL3L</p> <p>1WpAGW5X5V0D8MCXMC0A</p> <p>1fyvVGGccjpzS23q237V</p> <p>2Mie2Hszsf655xJGTWeu</p> <p>367Nqe2lyzdRZf47tkL9</p> <p>3B6uB5EvMzC9wnaqKXRq</p>	<p>+ Iniciar colección</p> <p>+ Agregar campo</p> <p>amount: 1000</p> <p>createdAt: 17 de octubre de 2024, 8:47:25 a.m. UTC-6</p> <p>currency: "mxn"</p> <p>id: "0pyug0DQmt65kfKlfnO"</p> <p>product: "Areli"</p> <p>status: "Pendiente"</p>
---	--

Diagrama de la arquitectura del sistema.

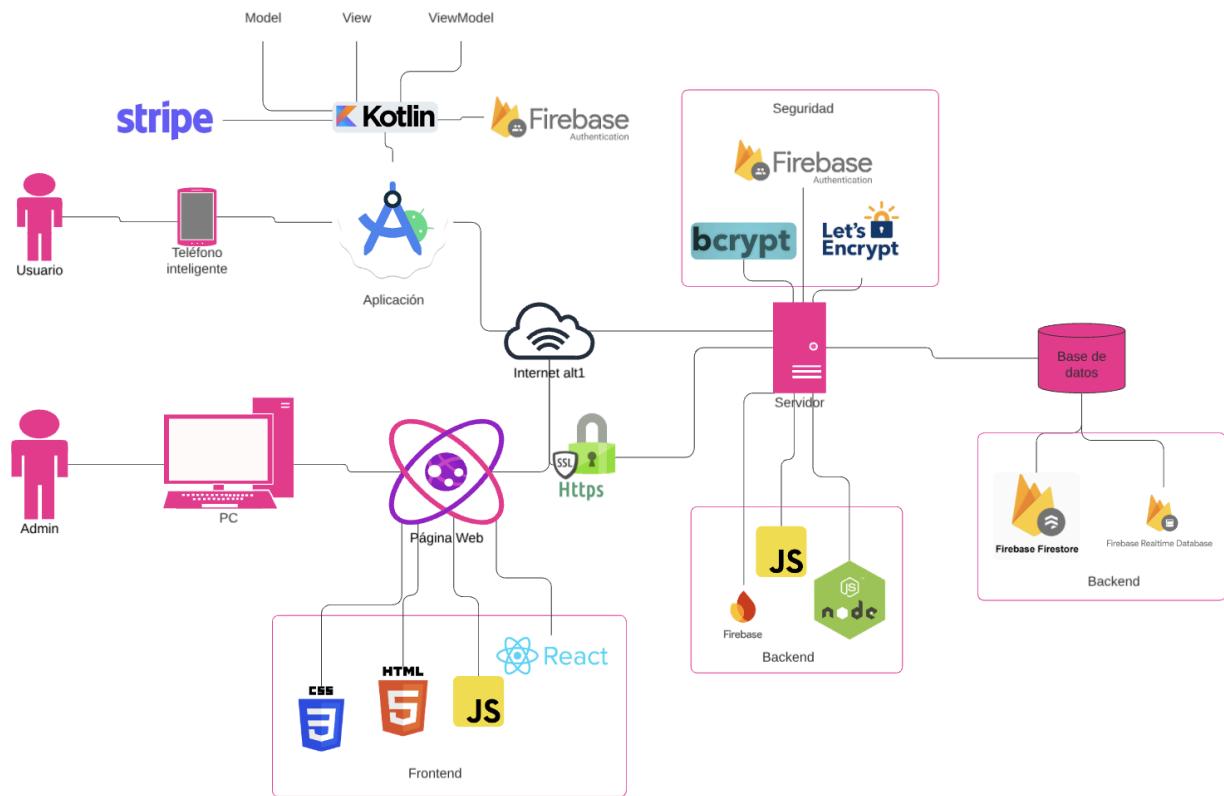
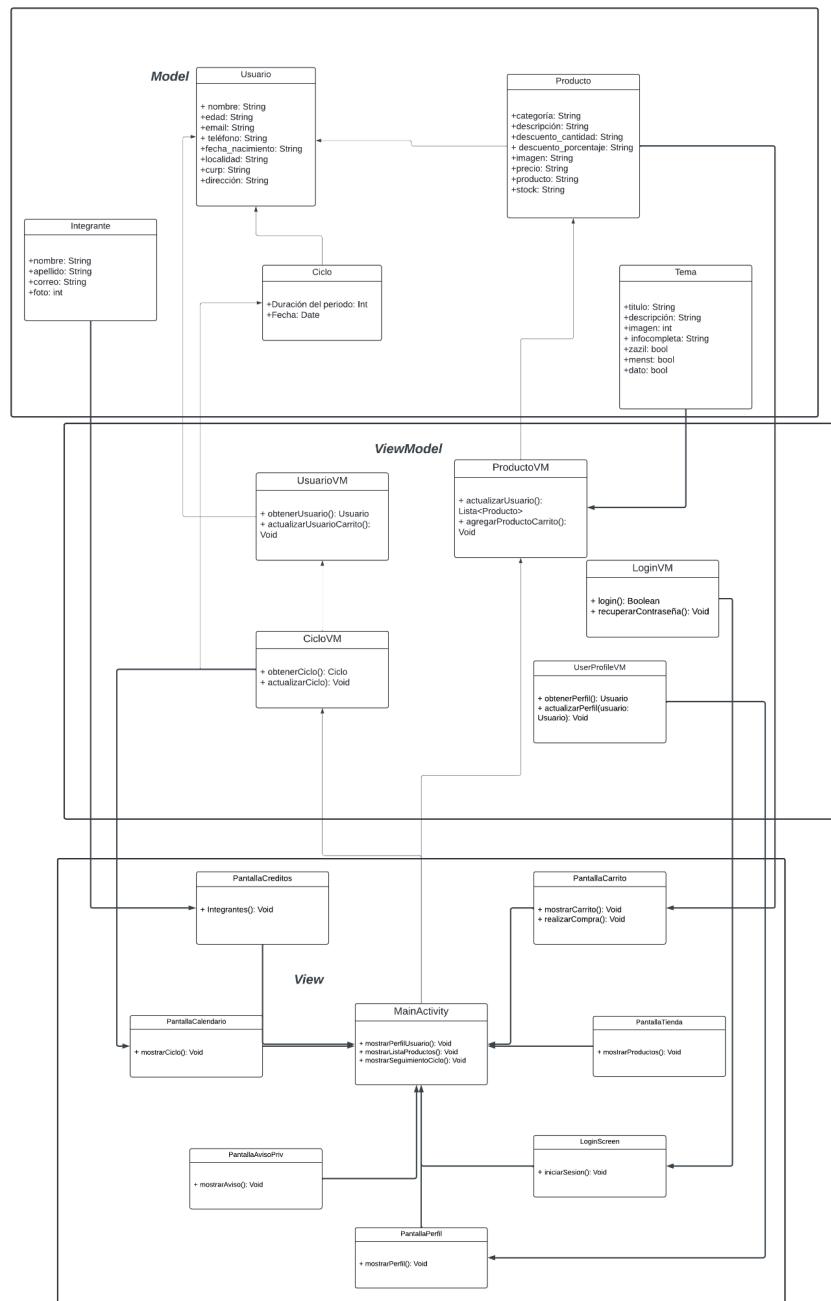
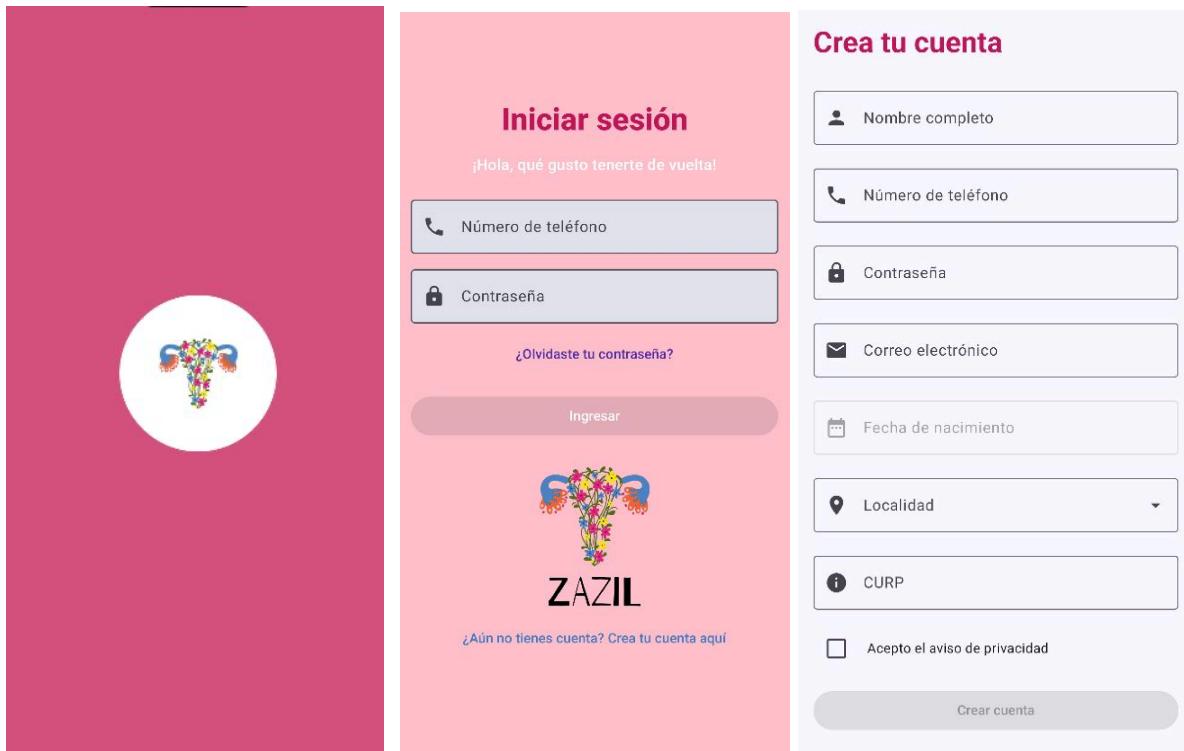


Diagrama de clases de la app (MVVM).



Diseño completo de las interfaces de usuario.

Aplicación móvil



CONÓCENOS

Promueve:

- Erradicar la pobreza menstrual
- Eliminar el uso de desechables
- Impulsar el autoempleo

Al diseñar toallas con materiales de alta calidad, hipoalergénicos y absorbentes, garantizan una experiencia cómoda y segura durante el período menstrual. Pero lo más importante es que son reutilizables, lo que significa que ayudan a reducir la generación de residuos y contribuyen a la conservación del medio ambiente.



Misión:

HABLEMOS DE...

¿Cómo lavar mis toallas ZAZIL?

¿Sabes cómo lavar tu toalla Zazil? ¡ES MUY FÁCIL! Aquí te mostramos los pasos para lavar tu toalla y reutilizarla.

[Leer más...](#)

Beneficios de usar toallas ZAZIL

Conoce los beneficios de las Toallas Zazil! ¡Transforma tu cuidado menstrual y siente la diferencial!

[Leer más...](#)

Sabías que...

Las toallas sanitarias también contaminan

Decemos disminuir la contaminación de los mares. Toallas reutilizables ayudan al medio ambiente.

[Leer más...](#)

Las estaciones pueden afectar el ciclo menstrual

¿Conoces aquellos beneficios del verano? Empecemos a amar cada estación [Lee más...](#)

[Regresar](#)

Las estaciones pueden afectar el ciclo menstrual

¿Conoces aquellos beneficios del verano? Empecemos a amar cada estación del año.

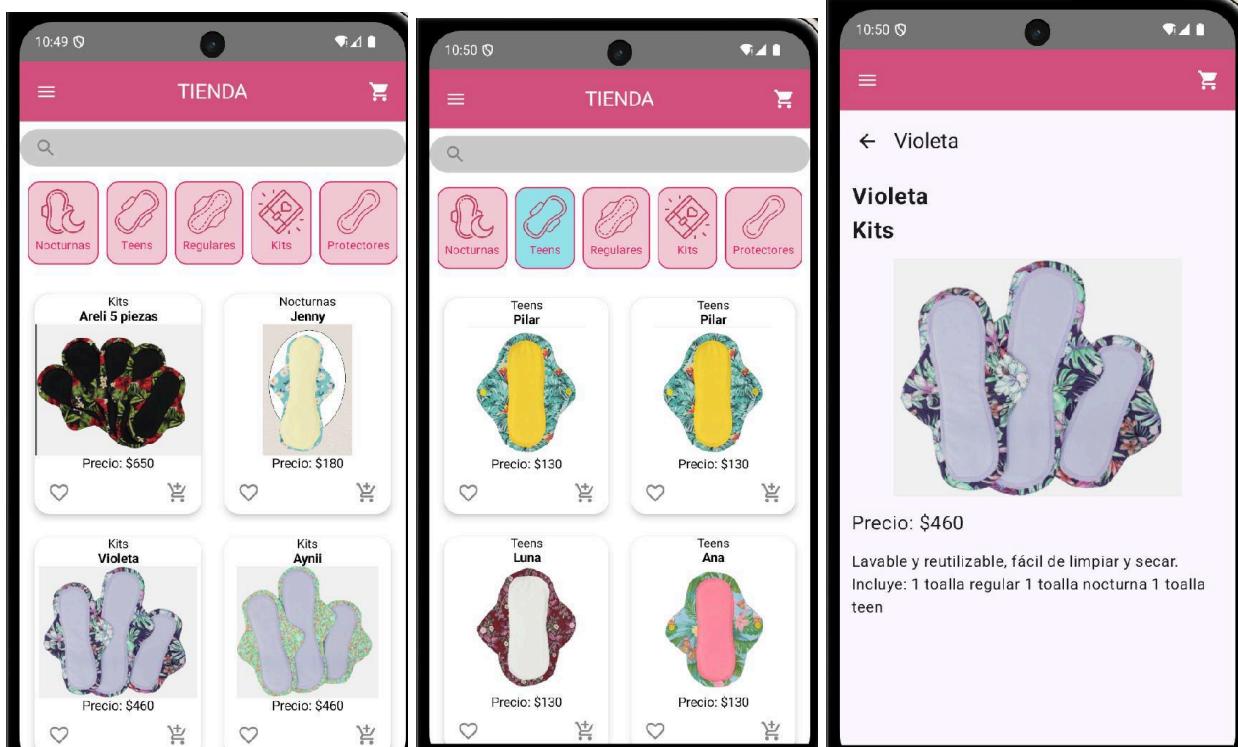
¿Sabías qué?

LAS ESTACIONES PUEDEN AFECTAR AL FLUJO MENSTRUAL

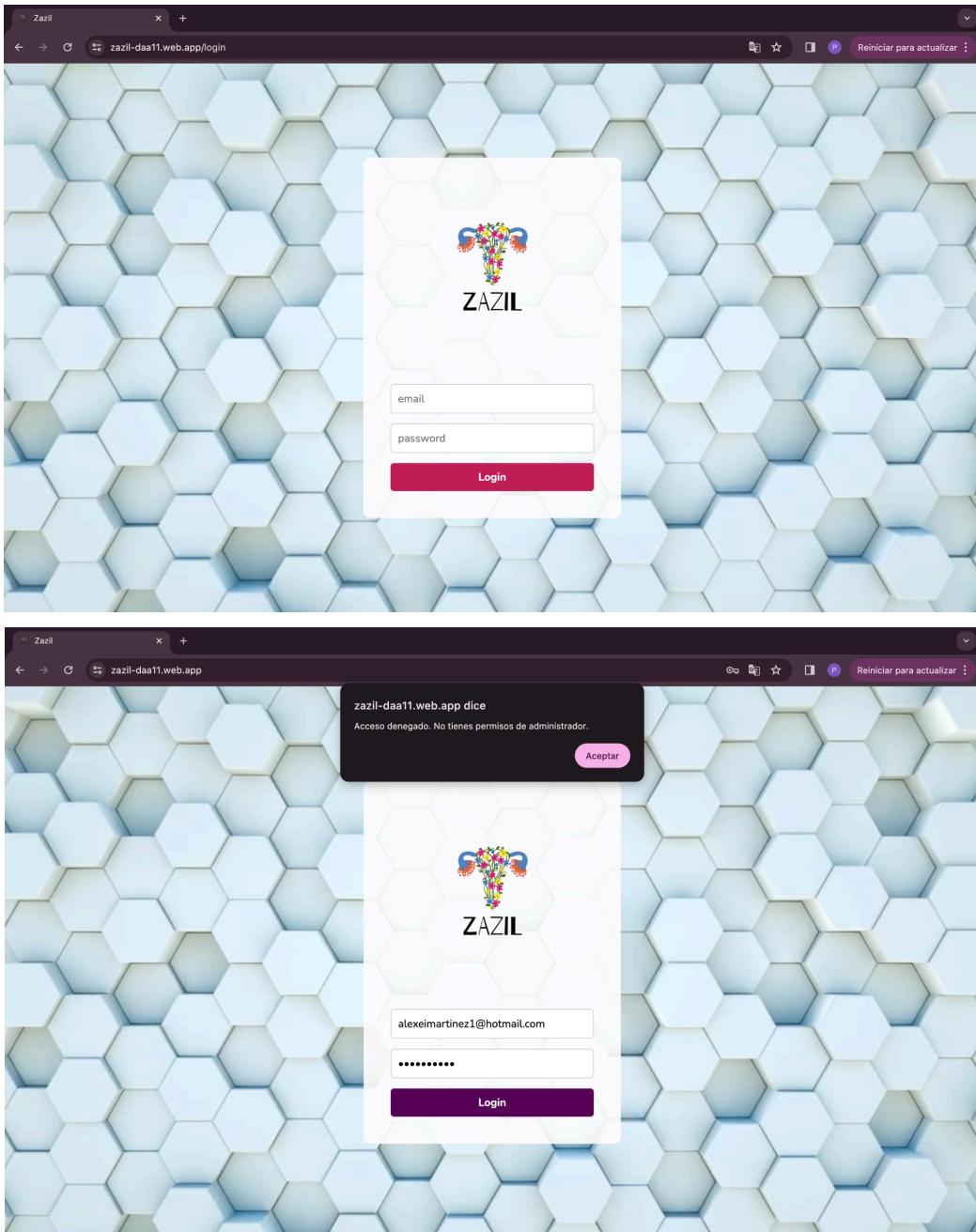


Hay estudios que señalan que los rayos del sol influyen en la menstruación

En concreto, si antes de la ovulación de una mujer (justo antes de que le venza la regla) hace



Página web



ZAZIL

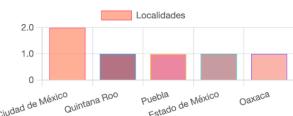
Panel de administración

Pantalla principal

- Dashboard**
- Contenido**
 - Usuarios**
 - Productos**
 - Pedidos**
 - Logout**

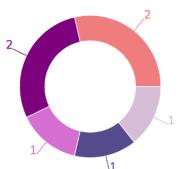
Usuarios: 14 | [Ver usuarios](#)
Productos en stock: 56 | [Ver detalles](#)
Pedidos realizados: 156 | [Ver ordenes](#)

Localidades



Datos basados en las localidades de los usuarios.

Toallas más vendidas



Categoría	Cantidad
Luna	2
Ana	1
Arelí	1
Astrid	1
Violeta	1

Últimos pedidos

ID Pedido	Fecha	Precio	Tipo de moneda	Estatus
NbERSWxJgb4EEukD1xdh	20/10/2024	242000	mxn	Pendiente

ZAZIL

Panel de administración

Historial de Usuarios

<input type="checkbox"/>	ID	User	Contraseña	Teléfono	Correo	Fecha de Nacimiento	CURP
<input type="checkbox"/>	5541...	Yeni	\$2a\$10\$238.uVxpPs/cHT8H1j...	5541840736	yeniarmdz@gmail.com	17/10/1974	\$2a\$10\$Xe3WZVLXyec...
<input type="checkbox"/>	5555...	Allan	\$2a\$10\$W2xYPrd0KkV6DS...	5555867046	allan123@gmail.com	16/08/1983	\$2a\$10\$5Anmf1ETSc8F...
<input type="checkbox"/>	5563...	Ian Alexei	\$2a\$10\$ipFO12M6mBCN8dy...	5563190498	alexeimartinez1@hotmail.com	13/11/2002	\$2a\$10\$9k/x72TMtA4C...
<input type="checkbox"/>	5563...	Papu pro	\$2a\$10\$qHLOQiGTTnxdhVn...	5563728193	papupro@gmail.com	16/10/1984	\$2a\$10\$tvYfoPHHPDjs...
<input type="checkbox"/>	5567...	Tigrito	\$2a\$10\$Rb79Laaj09UBYjdSS...	5567325548	tigri123@gmail.com	21/07/1998	\$2a\$10\$L4dgld/mVnw...
<input type="checkbox"/>	5572...	Lorena Solis	\$2a\$10\$nw05.Q2voGYT3LUz...	5572247242	lorena_solis@hotmail.com	12/06/2002	\$2a\$10\$AFQnR6kOD4T...
<input type="checkbox"/>	5572...	pepe perez	\$2a\$10\$puHOketOdPw3sWG...	5572247243	pepeperez@hotmail.com	14/10/1982	\$2a\$10\$6XzTz0WiO5T...
<input type="checkbox"/>	5578...	Bluey	\$2a\$10\$elempFPoxdj5UVuRr...	5578402168	bluey@gmail.com	07/01/1991	\$2a\$10\$yKqaR6zlCqB0l...
<input type="checkbox"/>	5585...	Peter Parker	\$2a\$10\$6Gx.Vl9VKjm3ZVfgSr...	5585412574	spidermam123@gmail.com	25/11/1992	\$2a\$10\$7E5DSYkDLrNs...

1-10 of 14 < >

Zazil

Panel de administración

=ZAZIL

Pantalla principal.

- Dashboard
- Contenido
- Usuarios
- Productos
- Pedidos
- Logout

Historial de Pedidos

ID	Fecha	Precio	Tipo de moneda	Estatus	Action
Opyu...	17/10/2024	1000	mxn	Pendiente	<button>Borrar</button>
1JMB...	17/10/2024	83000	mxn	Pendiente	<button>Borrar</button>
1QKU...	17/10/2024	83000	mxn	Pendiente	<button>Borrar</button>
1Wp...	17/10/2024	83000	mxn	Aprobado	<button>Borrar</button>
1fyvV...	20/10/2024	219000	mxn	Pendiente	<button>Borrar</button>
2Mie2...	17/10/2024	83000	mxn	Aprobado	<button>Borrar</button>
367N...	17/10/2024	83000	mxn	Pendiente	<button>Borrar</button>
3B6u...	17/10/2024	83000	mxn	Pendiente	<button>Borrar</button>
3GqC...	17/10/2024	83000	mxn	Pendiente	<button>Borrar</button>

1-10 of 156 < >

Zazil

Panel de administración

=ZAZIL

Agregar Nuevo Producto

ID	Producto	Descripción	Precio	Categoría	Nuevo	Descuento	Porcentaje ...	Precio con D
1QXle...	Areli 5 piezas	Lavable y reutilizable, fácil de limpiar y secar. Incl...	650	Kits	No	No	0	650.00
24wt...	Jenny	Toalla reutilizable. Ideal para flujo abundante. He...	180	Nocturnas	No	No	0	180.00
3cmf...	Violeta	Lavable y reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00
5Bnr...	Aynii	Lavable y reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00
5xhzg...	Ana	Lavable y reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00
8K2p...	Areli	Lavable y reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00
Bl8EK...	Maite	Toalla reutilizable ideal para flujo ligero. Hecha a ...	40	Pantiprotectores d...	No	No	0	40.00
9haiU...	Carol	Toalla reutilizable ideal para flujo ligero. Hecha a ...	40	Pantiprotectores d...	No	No	0	40.00
9i6n1...	Abi	Toalla reutilizable ideal para flujo ligero. Hecha a ...	40	Pantiprotectores d...	No	No	0	40.00

1-10 of 56 < >

Zazil

zazil-daa11.web.app/products

Reiniciar para actualizar

Panel de administración

Agregar Nuevo Producto

Nº	Precio	Categoría	Nuevo	Descuento	Porcentaje ...	Precio con Descuento	Stock	Action
reutilizable, fácil de limpiar y secar. Incl...	650	Kits	No	No	0	650.00	En stock: 5 unidades	Ver Borrar
tilizable. Ideal para flujo abundante. He...	180	Nocturnas	No	No	0	180.00	En stock: 1 unidades	Ver Borrar
reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00	En stock: 1 unidades	Ver Borrar
reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00	En stock: 1 unidades	Ver Borrar
reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00	En stock: 1 unidades	Ver Borrar
reutilizable, fácil de limpiar y secar. Incl...	460	Kits	No	No	0	460.00	En stock: 1 unidades	Ver Borrar
tilizable ideal para flujo ligero. Hecha a ...	40	Pantiprotectores d...	No	No	0	40.00	En stock: 1 unidades	Ver Borrar
tilizable ideal para flujo ligero. Hecha a ...	40	Pantiprotectores d...	No	No	0	40.00	En stock: 1 unidades	Ver Borrar
tilizable ideal para flujo ligero. Hecha a ...	40	Pantiprotectores d...	No	No	0	40.00	En stock: 1 unidades	Ver Borrar

1–10 of 56 < >

Zazil zazil-daa11.web.app/products/iQXlecnsmKbAZQnVdnrf Reiniciar para actualizar

Panel de administración

Actualizar Producto

Imagen: 

Nombre del Producto: Areli 5 piezas

Descripción del Producto: Lavable y reutilizable, fácil de limpiar y secar. Incluye: 2 t

Precio: 650

Porcentaje de Descuento: 0

Cantidad con Descuento: 650.00

Categoría del Producto: Kits

Cantidad de Stock: 5

Actualizar

Zazil zazil-daa11.web.app/products/new Reiniciar para actualizar

Panel de administración

Agregar Nuevo Producto

Imagen: 

Nombre del Producto: Nombre

Descripción del Producto: Descripción

Precio: 100

Categoría del Producto: Nuevo

Categoría: Categoría

Porcentaje de Descuento: 20

Descuento:

Precio con Descuento: 0

Cantidad de Stock: Número de unidades

Enviar

Integración de seguridad en sistemas computacionales.

En la actualidad, la seguridad en las aplicaciones web es un factor crítico para proteger la información sensible de los usuarios y garantizar la integridad del sistema. En esta sección del documento estaremos presentando una serie de mecanismos de seguridad que aseguren la transmisión de datos, la autenticación y la integridad de la información de la aplicación.

1. Encriptación de la comunicación entre el Cliente y el Servidor

Para proteger los datos transmitidos entre la aplicación móvil y el servidor hemos optado por implementar una encriptación de extremo a extremo utilizando el protocolo HTTPS con SSL/TLS. Inicialmente, emplearemos 'Let's Encrypt', una autoridad certificadora gratuita, para generar certificados SSL válidos.

Medidas Iniciales

- Let's Encrypt: Implementaremos el certificado SSL en nuestro servidor Node.js para garantizar que las conexiones estén encriptadas y seguras.
- HTTPS en Android Studio: Configuraremos la aplicación móvil para que todas las solicitudes de red se realicen únicamente a través de HTTPS, asegurando que la comunicación esté protegida.
- React y el Administrador Web: Todas las solicitudes de la página web del administrador, se harán también a través de HTTPS

La implementación de esta medida en el proyecto nos brindará confidencialidad y encriptación de los datos transmitidos, la verificación de la identidad del servidor(evitando ataques intermediarios), y la implementación rápida y gratuita para automatizar la renovación de los certificados.

2. Encriptación de la Base de datos

Aunque no se manejan datos financieros directamente, los datos personales y credenciales de los usuarios siguen siendo sensibles. Para garantizar su seguridad en la base de datos MySQL, se tomarán las siguientes medidas:

Medidas Iniciales:

- *Hashing de contraseñas con bcrypt:* Todas las contraseñas de los usuarios serán encriptadas utilizando bcrypt antes de ser almacenadas en la base de datos, asegurando que incluso en caso de acceso no autorizado, no se pueda recuperar esta información.

Propuesta a Futuro:

- *Encriptación de Datos Sensibles:* Aunque el sistema de pago es en tiendas OXXO, si en el futuro se incorporan otros métodos de pago, se propondrá la implementación de encriptación de campos sensibles (como información de pago) en la base de datos.

3. Propuesta de Seguridad a Futuro: Certificados SSL Pagados

En el futuro, si el proyecto escala y requiere un mayor nivel de confianza y robustez, proponemos la adquisición de certificados **SSL de pago**. Estos certificados proporcionarán validación adicional de la identidad de la organización, lo que aumentará la credibilidad y confianza de los usuarios.

Beneficios de Certificados SSL Pagados:

- Mayor tiempo de validez (hasta 2-3 años) sin necesidad de renovaciones frecuentes.
- Validación de la identidad de la organización (OV SSL) o validación extendida (EV SSL), que incluye el nombre de la empresa en la barra de direcciones del navegador, incrementando la confianza del usuario.
- Garantías adicionales que ofrecen cobertura en caso de incidentes de seguridad.

Etapa 3. Desarrollo

Introducción

Este documento tiene como finalidad presentar de manera detallada la infraestructura técnica que se ha configurado para el desarrollo del proyecto. A lo largo del proceso, se describirán los elementos clave, como la configuración de servidores, bases de datos y herramientas de desarrollo, que garantizan el correcto funcionamiento del sistema. Asimismo, se documentarán las reuniones realizadas con el cliente, los desarrolladores y el equipo de pruebas, las cuales han sido fundamentales para alinear expectativas y tomar decisiones técnicas cruciales.

Además, se analizarán posibles ataques informáticos que podrían comprometer la seguridad del proyecto, identificando las vulnerabilidades más relevantes. A partir de ello, se explicarán los métodos implementados para proteger el sistema, tales como la encriptación de datos y la implementación de protocolos de seguridad, con el objetivo de asegurar la integridad y protección de la información gestionada.

URL del repositorio del Proyecto

Github página web: <https://github.com/SandyHerrera04/Zazil>

Usuario admin de prueba:

-zazil@yahoo.com

-zazil123

Github aplicación móvil: <https://github.com/SandyHerrera04/ZazilApp>

Usuario de prueba:

-5563190498

-Alexei1312

Documentación de código.

Todo el código desarrollado para la aplicación móvil y el sitio web ha sido subido de manera organizada al repositorio de GitHub, asegurando un control de versiones eficiente y un acceso centralizado para todo el equipo de desarrollo. Cada archivo ha sido documentado, explicando su funcionalidad principal, así como las funciones más importantes dentro de cada uno. Esta documentación tiene como objetivo facilitar la comprensión del código tanto para los desarrolladores actuales como para futuros colaboradores que necesiten hacer mejoras o mantenimiento del sistema.

Aplicación Móvil

```
9
10    /**
11     * Clase LoginViewModel
12     *
13     * Este ViewModel gestiona la lógica del inicio de sesión de la aplicación.
14     * Utiliza coroutines y un repository para manejar la autenticación del usuario.
15     */
16
17 @kotlinx.coroutines.SealedClass
18 sealed class LoginState {
19     object Success : LoginState()
20     object Loading : LoginState()
21     class Error(val message: String) : LoginState()
22 }
23
24 /**
25  * ViewModel para manejar el proceso de inicio de sesión
26  * Se comunica con el UserRepository para realizar el proceso de autenticación.
27  */
28
29 class LoginViewModel : ViewModel() {
30     private val userRepository = UserRepository()
31     private val _loginState = MutableStateFlow<LoginState>(LoginState.Loading)
32     val loginState: StateFlow<LoginState> = _loginState
33
34     /**
35      * Función login
36      *
37      * Inicia el proceso de autenticación utilizando el UserRepository.
38      * Actualiza el estado de inicio de sesión según el resultado de la operación.
39      *
40      * @param phone Número de teléfono del usuario.
41      * @param password Contraseña del usuario.
42      */
43
44     fun login(phone: String, password: String) {
45         _loginState.value = LoginState.Loading
46
47         viewModelScope.launch { this: CoroutineScope
48             userRepository.loginUser(phone, password,
49                 onSuccess = {
50                     _loginState.value = LoginState.Success
51                 },
52                 onFailure = { error ->
```

```
 9  /**
10   * Clase SignUpViewModel
11   *
12   * Este ViewModel maneja la lógica del registro de nuevos usuarios.
13   * Se comunica con el UserRepository para enviar los datos del usuario y
14   * gestionar el flujo de estado de la operación.
15   */
16
17  @sealed class SignUpState {
18      object Idle : SignUpState()
19      object Success : SignUpState()
20      object Loading : SignUpState()
21      data class Error(val message: String) : SignUpState()
22  }
23
24  /**
25   * ViewModel para manejar el proceso de registro de usuarios.
26   * Utiliza coroutines para ejecutar el registro en segundo plano.
27   */
28
29 class SignUpViewModel : ViewModel() {
30
31     private val _signUpState = MutableStateFlow<SignUpState>(SignUpState.Idle)
32     val signUpState: StateFlow<SignUpState> = _signUpState
33
34     private val auth: FirebaseAuth = FirebaseAuth.getInstance()
35     private val database = FirebaseDatabase.getInstance().reference
36
37
38    /**
39     * Función signUp
40     *
41     * Inicia el proceso de registro de un nuevo usuario.
42     * Actualiza el estado del registro según el resultado de la operación.
43     *
44     * @param username Nombre del usuario.
45     * @param email Correo electrónico del usuario.
46     * @param password Contraseña del usuario.
47     */
48
49     fun registerUser(
```

```
1 package tec.iamr.myapplication.ciclo.viewmodel
2
3 import androidx.lifecycle.ViewModel
4 import java.time.LocalDate
5
6 /**
7 * Clase CalculadoraMenstrualViewModel
8 *
9 * Este ViewModel maneja los cálculos relacionados con el ciclo menstrual de la usuaria,
10 * incluyendo la próxima menstruación y los días fértiles.
11 */
12
13 class CalculadoraMenstrualViewModel : ViewModel() {
14
15
16 /**
17 * Función calcularProximaMenstruacion
18 *
19 * Calcula la fecha de la próxima menstruación sumando la duración del ciclo a la fecha
20 * de la última menstruación.
21 *
22 * @param fechaUltimaMenstruacion Fecha de la última menstruación (LocalDate).
23 * @param duracionCiclo Duración del ciclo menstrual en días (Int).
24 * @return La fecha estimada de la próxima menstruación (LocalDate).
25 */
26
27 fun calcularProximaMenstruacion(fechaUltimaMenstruacion: LocalDate, duracionCiclo: Int): LocalDate {
28     return fechaUltimaMenstruacion.plusDays(duracionCiclo.toLong())
29 }
30
31
32 fun calcularDiasFertiles(proximaMenstruacion: LocalDate, duracionCiclo: Int): Pair<LocalDate, LocalDate> {
33     val ovulacion :LocalDate = proximaMenstruacion.minusDays((duracionCiclo / 2).toLong())
34     return Pair(ovulacion.minusDays( daysToSubtract: 4), ovulacion.plusDays( daysToAdd: 4))
35 }
36 }
```

```
1 package tec.lass.zazil_app.viewmodel
2
3 import com.google.firebase.auth.FirebaseAuth
4
5 /**
6 * Clase PasswordRecoveryViewModel
7 *
8 * Este ViewModel maneja la lógica para la recuperación de contraseñas.
9 * Se comunica con el UserRepository para enviar solicitudes de recuperación
10 * y gestiona el estado del proceso.
11 */
12
13 class PasswordRecoveryViewModel {
14     private val auth: FirebaseAuth = FirebaseAuth.getInstance()
15
16     /**
17      * Función sendPasswordResetEmail
18      *
19      * Inicia el proceso de recuperación de contraseña de un usuario.
20      * Actualiza el estado del proceso según el resultado de la operación.
21      *
22      * @param email Correo electrónico del usuario.
23      */
24
25     fun sendPasswordResetEmail(
26         email: String,
27         onSuccess: () -> Unit,
28         onFailure: (Exception) -> Unit
29     ) {
30         auth.sendPasswordResetEmail(email)
31             .addOnCompleteListener { task ->
32                 if (task.isSuccessful) {
33                     onSuccess()
34                 } else {
35                     task.exception?.let { onFailure(it) }
36                 }
37             }
38     }
39 }
40
41
```

```
48  /**
49   * Composable MyApp
50   *
51   * Esta función define la estructura general de la aplicación utilizando Jetpack Compose.
52   * Dependiendo de la ruta de navegación actual, decide si mostrar la pantalla de inicio de sesión o el menú lateral.
53   *
54   * @param navController Controlador de navegación utilizado para manejar las rutas entre pantallas.
55   */
56
57 @RequiresApi(Build.VERSION_CODES.O)
58 @Composable
59 fun MyApp(navController: NavHostController) {
60     val drawerState : DrawerState = rememberDrawerState(DrawerValue.Closed)
61     val scope : CoroutineScope = rememberCoroutineScope()
62     val backStackEntry : NavBackStackEntry? = navController.currentBackStackEntryAsState().value
63     val currentRoute : String? = backStackEntry?.destination?.route // Obtener la ruta actual
64
65     val productosState : SnapshotStateList<Producto> = remember { mutableStateListOf(*productos.toTypedArray()) }
66
67     if (currentRoute == "login" || currentRoute == "signup") {
68         Scaffold(
69             modifier = Modifier.fillMaxSize(),
70             content = { padding ->
71                 Column(
72                     modifier = Modifier
73                         .fillMaxSize()
74                         .padding(padding)
75                 ) { this: ColumnScope
76                     NavigationComponent(navController = navController, productosState = productosState)
77                 }
78             }
79         )
80     } else {
81         ModalNavigationDrawer(
82             drawerState = drawerState,
83             drawerContent = {
84                 // Contenido del menú lateral
85                 ModalDrawerSheet { this: ColumnScope
86                     DrawerHeader(
87                         currentRoute = currentRoute,
88                         onHeaderClick = {
89                             if (currentRoute != "perfil"){


```

```
162 }
163
164
165 /**
166 * Composable NavigationComponent
167 *
168 * Esta función define la navegación entre las diferentes pantallas de la aplicación utilizando NavHost.
169 *
170 * @param navController Controlador de navegación utilizado para cambiar de pantalla.
171 * @param productosState Lista mutable que contiene los productos de la tienda.
172 */
173
174 @RequiresApi(Build.VERSION_CODES.O)
175 @Composable
176 fun NavigationComponent(navController: NavController, productosState: MutableList<Producto>) {
    NavHost(navController = navController, startDestination = "login") { this: NavGraphBuilder
        composable(route: "login") { this: AnimatedContentScope | it: NavBackStackEntry |
            LoginScreen(navController = navController, viewModel = LoginViewModel())
        }
        composable(route: "signup") { this: AnimatedContentScope | it: NavBackStackEntry |
            SignUpScreen(navController = navController, viewModel = SignUpViewModel())
        }

        composable(route: "password_recovery") { this: AnimatedContentScope | it: NavBackStackEntry |
            PasswordRecoveryScreen(
                navController = navController,
                viewModel = PasswordRecoveryViewModel()
            )
        }
        composable(route: "inicio") { this: AnimatedContentScope | it: NavBackStackEntry |
            PantallaInicio(navController = navController)
            //Column(
            //    modifier = Modifier.fillMaxSize().padding(16.dp)) {
            //        Text("Este es el contenido principal de la app")
            //    }
        }
    }
}
```

```
1 package tec.lass.zazil_app.model
2
3 > import ...
4
5
6 /**
7 * Clase UserRepository
8 *
9 * Este repositorio maneja la lógica para registrar y autenticar usuarios en la base de datos de Firebase.
10 * Utiliza Firebase Realtime Database para almacenar y recuperar información del usuario.
11 */
12
13 class UserRepository {
14     private val database = FirebaseDatabase.getInstance().reference.child("users")
15
16     /**
17      * Función privada hashPassword
18      *
19      * Genera un hash seguro de la contraseña del usuario utilizando BCrypt.
20      *
21      * @param password Contraseña en texto claro.
22      * @return Cadena con la contraseña cifrada.
23      */
24
25     private fun hashPassword(password: String): String {
26         return BCrypt.hashpw(password, BCrypt.gensalt())
27     }
28
29     /**
30      * Función privada hashCurp
31      *
32      * Genera un hash seguro del CURP del usuario utilizando BCrypt.
33      *
34      * @param curp CURP en texto claro.
35      * @return Cadena con el CURP cifrado.
36      */
37
38     private fun hashCurp(curp: String): String {
39         return BCrypt.hashpw(curp, BCrypt.gensalt())
40     }
```

```
13 class UserRepository {
14     /**
15      * Función signUpUser
16      *
17      * Registra un nuevo usuario en la base de datos de Firebase. Verifica si el usuario ya existe antes de almacenar sus datos. Las contraseñas y el CURP son cifrados antes de almacenarse.
18      *
19      * @param name Nombre del usuario.
20      * @param phone Número de teléfono del usuario.
21      * @param password Contraseña del usuario.
22      * @param email Correo electrónico del usuario.
23      * @param birthdate Fecha de nacimiento del usuario.
24      * @param location Ubicación del usuario.
25      * @param curp CURP del usuario.
26      * @param onSuccess Callback que se ejecuta si el registro es exitoso.
27      * @param onFailure Callback que se ejecuta si ocurre un error, pasando el mensaje de error.
28      */
29
30     fun signUpUser(
31         name: String,
32         phone: String,
33         password: String,
34         email: String,
35         birthdate: String,
36         location: String,
37         curp: String,
38         onSuccess: () -> Unit,
39         onFailure: (String) -> Unit
40     ) {
41         val userRef : DatabaseReference = database.child(phone)
42
43         userRef.get().addOnSuccessListener { dataSnapshot ->
44             if (dataSnapshot.exists()) {
45                 onFailure("El usuario ya existe")
46             } else {
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77 }
```

```
13  class UserRepository {
102     /**
103      * Función loginUser
104      *
105      * Inicia sesión verificando si el número de teléfono existe y si la contraseña proporcionada coincide con
106      * la contraseña cifrada almacenada en la base de datos.
107      *
108      * @param phone Número de teléfono del usuario.
109      * @param password Contraseña del usuario en texto claro.
110      * @param onSuccess Callback que se ejecuta si el inicio de sesión es exitoso.
111      * @param onFailure Callback que se ejecuta si ocurre un error, pasando el mensaje de error.
112      */
113      fun loginUser(
114          phone: String,
115          password: String,
116          onSuccess: () -> Unit,
117          onFailure: (String) -> Unit
118      ) {
119          val userRef : DatabaseReference = database.child(phone)
120
121          userRef.get().addOnSuccessListener { dataSnapshot ->
122              if (dataSnapshot.exists()) {
123                  val storedHashedPassword : String = dataSnapshot.child("password").value as String
124
125                  // Comprobar si la contraseña es correcta
126                  if (BCrypt.checkpw(password, storedHashedPassword)) {
127                      onSuccess()
128                  } else {
129                      onFailure("Contraseña incorrecta")
130                  }
131              } else {
132                  onFailure("El usuario no existe")
133              }
134          }.addOnFailureListener { exception ->
135              onFailure(exception.message ?: "Error al verificar usuario")
136          }
137      }
138  }
```

Página Web

```
s App.js > ...

import Home from "./pages/home/Home";
import Login from "./pages/login/Login";
import List from "./pages/list/List";
import Single from "./pages/single/Single";
import New from "./pages/new/New";
import ProductDetails from "./pages/productDetails/ProductDetails";
import { BrowserRouter, Routes, Route, Navigate, Outlet } from "react-router-dom";
import { productInputs, userInputs } from "./formSource";
import "./style/dark.scss";
import { useContext } from "react";
import { DarkModeContext } from "./context/darkModeContext";
import { AuthContext } from "./context/AuthContext";

// UID del administrador autorizado
const ADMIN_UID = "dQEMV4CfeKQGzHtfyONy7fLDtN82"; // Reemplaza con el UID del administrador

function App() {
  const { darkMode } = useContext(DarkModeContext);
  const { currentUser } = useContext(AuthContext);

  // Verificación de autenticación y permisos del administrador
  const RequireAuth = () => {
    if (!currentUser) {
      return <Navigate to="/login" replace />; // Redirige al login si no está autenticado
    }
    if (currentUser.uid !== ADMIN_UID) {
      alert("Acceso denegado. No tienes permisos de administrador.");
      return <Navigate to="/login" replace />; // Redirige al login si no es administrador
    }
    return <Outlet />; // Permite el acceso si es administrador
  };

  return (
    <div className={darkMode ? "app_dark" : "app"}>
```

```
src > JS firebase.js > ...
1 import { initializeApp } from "firebase/app";
2 import { getAuth } from "firebase/auth";
3 import { getFirestore } from "firebase/firestore";
4 import { getStorage } from "firebase/storage";
5
6 // Verificar la clave de entorno
7 console.log("Firebase API Key:", process.env.REACT_APP_FIREBASE_KEY);
8
9 // Import the functions you need from the SDKs you need
10 //import { initializeApp } from "firebase/app";
11 // TODO: Add SDKs for Firebase products that you want to use
12 // https://firebase.google.com/docs/web/setup#available-libraries
13
14 // Your web app's Firebase configuration
15 const firebaseConfig = {
16   apiKey: process.env.REACT_APP_FIREBASE_KEY,
17   authDomain: "zazil-daa11.firebaseioapp.com",
18   databaseURL: "https://zazil-daa11-default-rtbd.firebaseio.com",
19   projectId: "zazil-daa11",
20   storageBucket: "zazil-daa11.appspot.com",
21   messagingSenderId: "107009042079",
22   appId: "1:107009042079:web:21a6cf88e62d716317e4aa"
23 };
24
25 // Inicialización de Firebase con manejo de errores
26 let app;
27 try {
28   app = initializeApp(firebaseConfig);
29   console.log("Firebase initialized:", app);
30 } catch (error) {
31   console.error("Error initializing Firebase:", error);
32 }
33
34 // Initialize Firebase
35 //const app = initializeApp(firebaseConfig);
```

```
src > pages > productDetails > ProductDetails.jsx > ProductDetails

218 import "./productDetails.scss";
219 import Sidebar from "../../components/sidebar/Sidebar";
220 import Navbar from "../../components/navbar/Navbar";
221 import DriveFolderUploadOutlinedIcon from "@mui/icons-material/DriveFolderUploadOutlined";
222 import { useEffect, useState } from "react";
223 import { doc, getDoc, updateDoc } from "firebase/firestore";
224 import { db, storage } from "../../firebase";
225 import { ref, uploadBytesResumable, getDownloadURL } from "firebase/storage";
226 import { useNavigate, useParams } from "react-router-dom";
227
228 const ProductDetails = () => {
229   const { productId } = useParams();
230   const [file, setFile] = useState(null);
231   const [data, setData] = useState({
232     price: "",
233     discountPercent: "",
234     discountAmount: ""
235   });
236   const [per, setPerc] = useState(null);
237   const navigate = useNavigate();
238
239   // Obtener los datos del producto desde Firebase
240   useEffect(() => {
241     const fetchProduct = async () => {
242       const docRef = doc(db, "products", productId);
243       const docSnap = await getDoc(docRef);
244       if (docSnap.exists()) {
245         setData(docSnap.data());
246       } else {
247         console.log("Producto no encontrado");
248       }
249     };
250     fetchProduct();
251   }, [productId]);
```

```
pages > login > Login.jsx > ...  
  
// Importaciones necesarias de React y otros módulos  
import { useContext, useState } from "react";  
import "./login.scss";  
import { signInWithEmailAndPassword } from "firebase/auth";  
import { auth } from "../../firebase";  
import { useNavigate } from "react-router-dom";  
import { AuthContext } from "../../context/AuthContext";  
import zazilLogo from "./zazil.png"; // Importación del logo de Zazil  
  
// Componente de Login  
const Login = () => {  
  const [error, setError] = useState(false); // Estado para manejar errores de login  
  const [email, setEmail] = useState(""); // Estado para manejar el email  
  const [password, setPassword] = useState(""); // Estado para manejar la contraseña  
  
  const navigate = useNavigate(); // Hook para la navegación  
  const { dispatch } = useContext(AuthContext); // Obtiene el dispatch del contexto de autenticación  
  
  // Maneja el evento de login  
  const handleLogin = (e) => {  
    e.preventDefault(); // Previene el comportamiento por defecto del formulario  
  
    // Autentica al usuario con Firebase  
    signInWithEmailAndPassword(auth, email, password)  
      .then((userCredential) => {  
        const user = userCredential.user; // Obtiene el usuario autenticado  
        dispatch({ type: "LOGIN", payload: user }); // Despacha la acción de login  
        navigate("/"); // Navega a la página principal  
      })  
      .catch((error) => {  
        setError(true); // Establece el estado de error si la autenticación falla  
      });  
  };  
  
  return (  
    <div className="login">  
      {/* Video de fondo */}  
      <video autoPlay loop muted className="video-bg">  
        <source src={require("./login.mp4")} type="video/mp4" />  
      </video>  
      <div className="login-content">
```

```
functions > js index.js > ...
19 // ...
20
21 const functions = require('firebase-functions');
22 const stripe = require('stripe')('sk_test_51QAc0XFxGNIrLy0AErZ13zuFKjSz2I07acT5IT6y4jwFey9oWozHfjpRu0wZHgIvqto5e3NY8nVA');
23 const admin = require('firebase-admin');
24 const { user } = require('firebase-functions/v1/auth');
25 admin.initializeApp();
26
27 const db = admin.firestore();
28
29 exports.createPaymentIntent = functions.https.onCall(async (data, context) => {
30   const amount = data.amount;
31   const currency = data.currency;
32
33   try {
34     // Crear el PaymentIntent
35     const paymentIntent = await stripe.paymentIntents.create({
36       amount,
37       currency,
38     });
39
40     // Guardar la orden en Firestore después de crear el PaymentIntent
41     const orderRef = db.collection('orders').doc(); // Genera un nuevo ID de orden
42     await orderRef.set({
43       id: orderRef.id,
44       paymentMethod: 'Stripe',
45       amount,
46       currency,
47       status: 'Pendiente',
48       createdAt: admin.firestore.FieldValue.serverTimestamp(),
49     });
50
51     return { clientSecret: paymentIntent.client_secret };
52   } catch (error) {
53     throw new functions.https.HttpsError('unknown', error.message, error);
54   }
55});
```

Desarrollo de componentes de software

Aplicación Móvil (Android Studio con MVVM)

La aplicación móvil de Zazil permite la gestión y visualización de funciones relacionadas con la salud menstrual y compras en línea, con una interfaz intuitiva y funciones completas.

Log in y Sign in: Los usuarios pueden registrarse y acceder a su cuenta de manera segura utilizando correo electrónico y contraseña. Se implementó autenticación mediante Firebase Authentication con validación de correo. *Ejemplo:* Si el usuario olvida su contraseña, recibe un correo para restablecerla.

Restablecer Contraseña: Proceso seguro que permite a los usuarios restablecer su contraseña a través de un enlace enviado por correo electrónico, garantizando accesibilidad y seguridad.

Inicio: Pantalla de bienvenida que muestra un menú principal con acceso a funciones clave como el calendario, tienda y área de contacto.

Calendario del Ciclo Menstrual: Permite registrar y visualizar las fechas importantes del ciclo menstrual (periodo, días fértiles y próxima menstruación). Usa corrutinas para cálculos en segundo plano sin bloquear la interfaz.

Hablemos de...: Espacio interactivo donde los usuarios pueden leer artículos sobre temas de salud y bienestar, además de interactuar con otros usuarios y la organización Zazil.

Conócenos: Sección informativa sobre la misión y visión de Zazil, así como sus iniciativas enfocadas en salud y bienestar.

Aviso de Privacidad: Muestra las políticas de privacidad, explicando cómo se almacenan y protegen los datos personales en Firebase.

Créditos: Reconoce al equipo de desarrollo con sus nombres y roles.

Patrocinadores: Espacio para que los usuarios lean la información sobre sus aportaciones y los datos sobre donde pueden depositar el dinero.

Tienda:

- **Lista de Productos:** Muestra productos disponibles, permitiendo navegar entre categorías y ofertas.
- **Carrito de Compras:** Los usuarios pueden añadir productos al carrito, modificar cantidades y eliminar artículos según sus necesidades.
- **Pago:** Integra Stripe para pagos seguros. Despues de cada transacción, los datos se registran en Firestore junto con el estado del pedido.

Back-End (Firebase, Node.js)

El backend gestiona los datos de usuarios, productos y pedidos mediante Firebase Realtime Database para usuarios y Firestore para productos y pedidos. Se utiliza Node.js para APIs personalizadas que conectan la aplicación y el sitio web.

Base de Datos:

- **Firebase Realtime Database:** Almacena la información de los usuarios.
- **Firestore:** Guarda los productos y pedidos con su historial.

Encriptación de Contraseñas: Las contraseñas de los usuarios se encriptan utilizando **bcrypt**, garantizando la protección de credenciales.

Validación de Usuarios: Autenticación basada en **tokens JWT** que verifica la identidad de los usuarios y administradores en cada sesión.

Página Web Administrativa (React con Create React App)

El sitio web administrativo permite gestionar productos, pedidos y usuarios. Asegura control total mediante un sistema de roles para administradores y usuarios.

Log in de Administrador: El acceso al panel de administración está restringido a usuarios con permisos, validando credenciales mediante **Firebase Authentication**.

Gráficas Dinámicas: Presenta estadísticas sobre pedidos y ventas en tiempo real mediante gráficos interactivos con **Chart.js** y **Recharts**.

Gestión de Productos:

- **Agregar, Eliminar y Modificar Productos:** El administrador puede gestionar el inventario en tiempo real desde la interfaz web. Ejemplo: El administrador puede añadir nuevos productos con campos como nombre, descripción, precio y stock disponible.
- **Control de Stock:** Se muestra la cantidad disponible para cada producto, indicando su disponibilidad en tiempo real.

Gestión de Pedidos: El administrador puede ver el historial de pedidos, actualizar su estado (pendiente, enviado, entregado) y gestionar devoluciones.

Rutas y Navegación (React Router DOM):

- **/login:** Página de inicio de sesión.
- **/dashboard:** Panel principal con gráficas y estadísticas.
- **/products:** Gestión de productos.
- **/orders:** Visualización y gestión de pedidos.
- **/users:** Administración de usuarios registrados.

Seguridad y Estilo

Seguridad:

- **Autenticación Firebase:** El sistema valida cada sesión mediante tokens de autenticación.
- **HTTPS (TLS/SSL):** Toda la comunicación entre la aplicación y los servidores se realiza bajo conexiones seguras.
- **Stripe:** Permite gestionar pagos en línea con seguridad.

Estilos:

- **Material-UI:** Utilizado para crear componentes visuales como tablas y formularios.
- **SASS:** Usado para personalizar estilos y definir temas.

Configuración de la infraestructura necesaria para desarrollar el proyecto.

Aplicación en Android Studio

La aplicación móvil ha sido diseñada utilizando Android Studio, con una estructura basada en el patrón MVVM (Model-view-ViewModel). Este patrón de diseño junto con la documentación del código, es esencial para asegurar una separación clara entre la lógica de negocio y la interfaz de usuario, lo que mejora la mantenibilidad y escalabilidad de la aplicación.

- **Model:** En esta sección hemos gestionado datos de la aplicación. Esta capa interactúa con fuentes de datos externas (como la base de datos), y proporciona los datos estructurados necesarios para la sección de 'View'.

- **View:** Esta capa es encargada de la representación visual. Igualmente, esta capa observa los cambios del ViewModel y actualiza su interfaz en consecuencia.
- **ViewModel:** Esta capa actúa como intermediaria entre el modelo y la vista. Se encarga de gestionar la lógica de presentación y asegura que la vista siempre tenga información actualizada de forma reactiva. Esta arquitectura nos ha ayudado a mejorar la independencia entre componentes, facilitando el testeo unitario y la integración de nuevas funcionalidades que hemos estado implementando.

Sitio web Administrativo de Zazil

El sitio web administrativo permite gestionar productos y pedidos, así como administrar los usuarios registrados. Ha sido desarrollado con **React utilizando Create React App (CRA)** en lugar de Next.js.

Estructura del Sitio Web:

- **Pages:**
Contiene las principales páginas del sitio, como:
 - **Pantalla de inicio de sesión:** Permite al administrador acceder de manera segura al sistema.
 - **Panel de administración:** Muestra métricas importantes y gráficas.
 - **Lista de productos y pedidos:** Permite gestionar productos y pedidos en tiempo real.
- **Components:**
Incluye componentes reutilizables como:
 - Formularios para agregar o editar productos.
 - Tablas dinámicas con información de productos, pedidos y usuarios.
 - Menús y botones, facilitando la navegación y las acciones del administrador.
- **Public:**
Almacena recursos estáticos como imágenes e íconos, accesibles sin procesamiento adicional del servidor.

Gestión de Dependencias:

El sitio web utiliza **npm** para la instalación y gestión de bibliotecas. Esto garantiza que las dependencias estén actualizadas e integradas correctamente en el proyecto.

Modularidad del Sitio Web:

- **Gestión de Productos:**

La funcionalidad está organizada en módulos como:

- **Formulario de productos:** Permite agregar, editar y eliminar productos.
- **Control de stock:** Monitorea la disponibilidad de productos, evitando ventas fuera de inventario.

- **Gestión de Pedidos:**

Los pedidos se manejan en un módulo separado, permitiendo modificar su estado (pendiente, enviado, entregado) sin afectar otras partes del sistema.

- **Autenticación y Seguridad:**

El inicio de sesión y la gestión de usuarios se centralizan en este módulo, utilizando **Firebase Authentication**. Esto garantiza que solo los usuarios autenticados tengan acceso al panel administrativo.

Base de Datos y Conexión Backend

La aplicación móvil y el sitio web dependen de una base de datos centralizada para gestionar usuarios, productos y pedidos.

- **Firebase Realtime Database:**

Almacena los datos de los usuarios y facilita su autenticación.

- **Firestore:**

Gestiona los productos y pedidos en tiempo real, permitiendo actualizaciones inmediatas.

- **Conexión API Segura:**

La comunicación entre la aplicación móvil y el sitio web se realiza mediante servicios API seguros desarrollados en **Node.js**, asegurando integridad y confidencialidad en la transferencia de datos.

Control de Versiones y Colaboración

El proyecto utiliza **Git** como sistema de control de versiones, permitiendo la colaboración eficiente entre los miembros del equipo. Cada cambio realizado en el código se registra en un repositorio compartido, facilitando la integración continua y el seguimiento del desarrollo.

Documentación de reuniones con el cliente, desarrolladores y equipo de pruebas.

Minutas con el cliente

Minuta de la reunión

Fecha y hora: 20 de agosto de 2024, 8:00 p. m.
Lugar: Sesión remota por Zoom
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

- Zazil

Virydiana Fernández

Agenda

- Validación de los requerimientos del software.
- Revisión del primer prototipo.

Aspectos positivos

- Requerimientos del software completos.
- Implementación de un calendario para el ciclo menstrual.

Cuestiones pendientes

- Revisar su catálogo y paleta de colores.
- Implementar aviso de privacidad.
- Identificar aspectos importantes de la lógica de la tienda.

Minuta de la reunión

Fecha y hora: 05 de septiembre de 2024, 4:00 p.m.
Lugar: Sesión remota
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

- Zazil

Virydiana Fernández

Agenda

- Revisión de la tienda.
- Requisitos para la página web.

Aspectos positivos

- Funcionalidad de la página web para recopilar datos de la app móvil.
- Interfaz intuitiva para el usuario.
- Servidor para el backend de la app y página web.

Cuestiones pendientes

- Implementar un stock de productos.
- Dedicar la página web solo para backend de la app.

Minuta de la reunión

Fecha y hora: 11 de septiembre de 2024, 10:00 a.m.
Lugar: Sesión remota
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

- Zazil

Virydiana Fernández

Agenda

- Revisión de la trazabilidad del producto.

Aspectos positivos

- Tener la lógica del funcionamiento de la tienda.

Cuestiones pendientes

- Agregar un campo para que los usuarios registren su dirección.
- Considerar la trazabilidad en la página web para el administrador.

Minuta de la reunión

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

- Zazil

Virydiana Fernández
Vanessa Rendón

Fecha y hora: 12 de septiembre de 2024, 10:00 a.m.
Lugar: Sesión remota
Solicitada por: Equipo 5

Agenda

- Revisión del prototipo de la app y página web.

Aspectos positivos

- La aplicación tiene las funcionalidades esperadas.
- El home es llamativo para el usuario.
- Información sobre el ciclo menstrual.
- Información sobre la fundación.

Cuestiones pendientes

- Colocar el logo de la fundación.
- Implementar sus redes sociales.
- Informar sobre el método de pago a utilizar.

Minuta de la reunión

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

- Zazil

Virydiana Fernández
Vanessa Rendón

Fecha y hora: 02 de octubre de 2024, 10:00 a.m.
Lugar: Sesión remota
Solicitada por: Equipo 5

Aspectos positivos

- Uso de método de pago Paypal.

Cuestiones pendientes

- Revisar la cuenta a utilizar.
- Hacer pruebas del método de pago.

Minutas con desarrolladores

Minuta de la reunión

Fecha y hora: 09 de agosto de 2024, 1:00 p. m.
Lugar: Sesión presencial, campus CEM
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

Agenda

- Revisión de requerimientos de la app.
- Boceto de un prototipo.

Aspectos positivos

- Lluvia de ideas con los aspectos importantes.
- Diseño de un boceto para la estructura de la app.
- Definición de los requerimientos de la app.

Cuestiones pendientes

- Buscar más información sobre la fundación.
- Investigar sobre aplicaciones similares.
- Tener una propuesta sobre el diseño de la app.

Minuta de la reunión

Fecha y hora: 16 de agosto de 2024, 11:00 a. m.
Lugar: Sesión presencial, campus CEM
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

Agenda

- Revisión de propuestas de diseño para la app.
- Implementación de herramientas para mejor organización.

Aspectos positivos

- Definición de requerimientos de la app.
- Empezar documento sobre los requerimientos.
- Definición de la página web.

Cuestiones pendientes

- Implementar diagrama de Gantt.
- Implementar matriz de riesgos.
- Tener una propuesta sobre el diseño de la app.

Minuta de la reunión

Fecha y hora: 20 de agosto de 2024, 9:00 p. m.
Lugar: Sesión en línea, zoom
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

Agenda

- Seguimiento de diagrama de Gantt y matriz de riesgos.
- Creación del primer prototipo para la app.

Aspectos positivos

- Tareas organizadas en el diagrama de gantt .
- Requerimientos de la app completos.

Cuestiones pendientes

- Ideas para el prototipo de una página web.

Minuta de la reunión

Fecha y hora: 23 de agosto de 2024, 8:00 p. m.
Lugar: Sesión en línea, zoom
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

Agenda

- Revisión sobre el prototipo de la página web.
- División de tareas sobre el documento SRS.

Aspectos positivos

- Tareas organizadas y colocadas en el diagrama de gantt .
- Prototipo de la página web muy avanzado.

Cuestiones pendientes

- Completar el documento SRS.

Minuta de la reunión

Fecha y hora: 28 de agosto de 2024, 4:00 p. m.
Lugar: Presencial, salón de clases
Solicitada por: Equipo 5

Asistentes de la reunión

- Equipo 5

Alejandra Estefanía Rico González
Sandra Paulina Herrera Rebollo
Ian Alexei Matrínez Armendáriz
Fernanda Ponce Maciel
Lorena Abigail Solís de los Santos

Agenda

- Revisión del documento SRS.
- Revisión de las mejoras al prototipo del app.

Aspectos positivos

- Casos de uso, requerimientos, diagramas, etc; completos.
- Prototipo de la tienda en la app e inicio de la app completos.

Cuestiones pendientes

- Diseño de la base de datos.
- Identificar y dividir elementos de la app para el MVM.

Identificación de ataques informáticos.

Ataques de Fuerza Bruta: Este tipo de ataque intenta descubrir contraseñas mediante prueba y error. Los marcos de seguridad como ISO/IEC 27001 recomiendan la implementación de controles que limiten los intentos de acceso fallidos, mientras que el GDPR impone medidas de seguridad para proteger los datos personales contra este tipo de ataques.

Inyección SQL: Un ataque común dirigido a bases de datos que permite a los atacantes acceder, modificar o eliminar información sin autorización. Las normativas de protección de datos, como el GDPR, exigen que las organizaciones protejan la integridad y seguridad de los datos personales.

Cross-Site Scripting (XSS): Los ataques XSS permiten a los atacantes injectar scripts maliciosos en sitios web legítimos, afectando la experiencia y seguridad de los usuarios. Las leyes de ciberseguridad, como la Ley Federal de Protección de Datos Personales en México, exigen medidas de protección contra estos ataques para garantizar la privacidad de los usuarios.

Ataques de Denegación de Servicio (DoS/DDoS): Estos ataques sobrecargan los sistemas, interrumpiendo el acceso legítimo a un servicio. La Convención de Budapest sobre ciberdelitos es un marco internacional que promueve la cooperación entre países para combatir este tipo de ataques y asegurar la infraestructura tecnológica.

Phishing: Este ataque busca engañar a los usuarios para que revelen información sensible, como contraseñas o datos de pago. Las regulaciones como el GDPR y la Ley Federal de Protección de Datos Personales obligan a las empresas a tomar medidas proactivas para educar a los usuarios y mitigar estos riesgos.

Competencia de Aprendizaje:

Este criterio exige que los desarrolladores sean capaces de identificar con claridad los diferentes tipos de ataques, comprendiendo tanto las implicaciones técnicas como las legales. Un buen entendimiento de los marcos de referencia internacionales, como ISO/IEC 27001, OWASP Top 10, y la Convención de Budapest, permite implementar medidas de protección efectivas y garantizar el cumplimiento con leyes como el GDPR y la Ley Federal de Protección de Datos Personales en México. Además, asegurar que las medidas preventivas respeten los derechos fundamentales de los usuarios, como el derecho a la privacidad, es esencial en la identificación y mitigación de ataques informáticos.

Al tener un conocimiento claro de estos ataques y cómo prevenirlos, se garantiza que la aplicación no solo cumpla con los requisitos legales, sino que también proteja a los usuarios de forma efectiva.

Métodos de protección de ataques informáticos.

Para prevenir el uso malicioso o indebido de la app Zazil, es crucial implementar una serie de medidas de seguridad que garanticen la protección de los datos y las transacciones. Una de las primeras barreras es la autenticación de usuarios mediante tokens, lo cual refuerza la seguridad de las cuentas y protege las transacciones de pago. Asimismo, es indispensable que los datos sensibles se encripten tanto durante su transmisión como en su almacenamiento, evitando accesos no autorizados. Para proteger la comunicación entre la aplicación y el servidor, es esencial utilizar cifrado HTTPS (TLS/SSL), lo que asegura que los datos en tránsito estén debidamente protegidos.

Por otra parte, implementar políticas estrictas de contraseñas seguras es fundamental para reducir la vulnerabilidad ante ataques de fuerza bruta. Esto implica requerir una longitud mínima y la combinación de caracteres especiales, números y letras. Adicionalmente, un proceso de revisión de código centrado en la seguridad permitirá detectar posibles vulnerabilidades desde las primeras etapas del desarrollo, minimizando así los riesgos potenciales en el futuro.

Como desarrolladores y estudiantes que participamos en la creación de la app Zazil, nuestra responsabilidad se centra en la fase de desarrollo y en la entrega del proyecto, asegurando que se implementen las medidas básicas de seguridad necesarias. Sin embargo, es importante señalar que ciertos aspectos, como el monitoreo continuo de la actividad de la aplicación o la realización de análisis de penetración, exceden nuestras competencias una vez que el proyecto se entrega a la empresa. Será responsabilidad de Zazil mantener estas medidas y llevar a cabo el monitoreo necesario para garantizar la seguridad a largo plazo del sistema.

Etapa 4. Pruebas

1. Introducción

Este documento ha sido una herramienta fundamental en la organización y registro de las pruebas realizadas a lo largo del desarrollo del proyecto. Su propósito principal es

mantener un control claro y preciso sobre las pruebas efectuadas, facilitando la identificación de mejoras y la planificación de las pruebas pendientes.

Gracias a este registro, hemos podido garantizar que cada funcionalidad del proyecto cumpla con los requisitos establecidos, tanto en términos de operación técnica como de experiencia de usuario. Además, este seguimiento nos ha permitido organizar las pruebas de forma sistemática, asegurando que se abordan aspectos clave como la funcionalidad, la seguridad de los datos (en especial en la encriptación de contraseñas), la conexión entre componentes, la responsividad en diferentes dispositivos, y la integración fluida de los métodos de pago.

Al completar este proceso de pruebas, no solo buscamos cumplir con los estándares del proyecto, sino también generar evidencias sólidas que demuestran que la solución es robusta, intuitiva y completamente funcional. Este enfoque nos permitirá presentar un producto final confiable, eficiente y preparado para su implementación real.

1.1. Propósito del Documento

El propósito de este documento es servir como una guía y referencia centralizada para la planificación, ejecución y seguimiento de todas las pruebas realizadas durante el desarrollo del proyecto. Este documento tiene como objetivo documentar de manera estructurada y detallada cada prueba, permitiendo un registro claro de los resultados obtenidos, así como las acciones correctivas necesarias para garantizar que cada componente cumpla con los requisitos de calidad, funcionalidad y seguridad.

Además, este documento proporciona un marco organizado para gestionar las pruebas futuras, asegurando que el proyecto pase por evaluaciones rigurosas en términos de funcionalidad, diseño, usabilidad, rendimiento y responsividad. A través de este control, buscamos ofrecer un producto que cumpla con los estándares establecidos y esté listo para su implementación final.

1.2. Alcance de las Pruebas

El alcance de las pruebas abarca todas las funcionalidades clave del proyecto, asegurando que cada componente funcione de manera adecuada y eficiente en diferentes entornos y condiciones. Esto incluye pruebas en diversas áreas como:

- **Funcionalidad:** Verificación de que todas las características implementadas operen según los requisitos definidos, incluyendo la interacción entre la aplicación y la base de datos, la correcta encriptación de datos sensibles, y la integración de métodos de pago.
- **Interfaz de usuario:** Evaluación del diseño y la navegabilidad entre pantallas, asegurando que la experiencia del usuario sea intuitiva y que los elementos visuales respondan correctamente en diferentes tamaños de pantalla y dispositivos.
- **Seguridad:** Revisión de la protección de los datos del usuario, especialmente en lo referente al manejo y encriptación de contraseñas y otra información sensible.
- **Responsividad:** Comprobación de que el sistema se ajuste y funcione correctamente en diferentes tipos de dispositivos (móviles, tablets, etc.) y resoluciones de pantalla.
- **Integración de componentes:** Validación de la conectividad y la comunicación fluida entre la aplicación, la base de datos, la página web, y los servicios externos como PayPal.
- **Rendimiento:** Pruebas de la capacidad de la aplicación y la web para manejar múltiples usuarios, cargas de datos, y transacciones de manera eficiente sin comprometer la velocidad o estabilidad del sistema.

1.3. Definiciones, Acrónimos y Abreviaturas

- DB: Base de Datos. Es el sistema que permite el almacenamiento y la gestión organizada de la información del proyecto.
- UI: Interfaz de Usuario (User Interface). Se refiere al diseño visual y la disposición de los elementos con los que el usuario interactúa dentro de la aplicación o página web.
- UX: Experiencia de Usuario (User Experience). Hace referencia a la forma en que los usuarios perciben y navegan el sistema, incluyendo su satisfacción y facilidad de uso.
- Kotlin: Lenguaje de programación utilizado para el desarrollo de la aplicación en Android.

- React: Biblioteca de JavaScript utilizada para el desarrollo de la interfaz de la aplicación web.
- PayPal: Plataforma de pagos en línea que se integra como método de pago en la aplicación.
- MySQL: Sistema de gestión de bases de datos relacional utilizado para almacenar y gestionar los datos del proyecto.
- HTTPS: Protocolo de Transferencia de Hipertexto Seguro (Hypertext Transfer Protocol Secure). Es un protocolo de comunicación en internet que asegura la transferencia de datos entre el cliente y el servidor de manera encriptada.
- API: Interfaz de Programación de Aplicaciones (Application Programming Interface). Se refiere a un conjunto de protocolos y herramientas que permiten que dos sistemas se comuniquen entre sí.
- CRUD: Operaciones básicas en la base de datos: Crear (Create), Leer (Read), Actualizar (Update), y Eliminar (Delete).
- SDK: Kit de Desarrollo de Software (Software Development Kit). Conjunto de herramientas que se utilizan para desarrollar aplicaciones.

2. Objetivos de las Pruebas

El principal objetivo de las pruebas es garantizar que el proyecto funcione de manera correcta, segura y eficiente antes de su implementación. Esto incluye validar la funcionalidad de las características desarrolladas, asegurar la protección de los datos sensibles, verificar la responsividad en diferentes dispositivos, comprobar la estabilidad y el rendimiento bajo diversas condiciones, y evaluar la correcta integración entre los componentes del sistema. Asimismo, las pruebas buscan identificar y corregir posibles errores para asegurar una experiencia de usuario óptima.

3. Alcance de las Pruebas

3.1. Funcionalidades a Probar

- Registro de usuario
- login de usuario
- cambiar contraseña
- Hablemos de...
- Calculadora menstrual
- Favoritos
- Filtros
- Usuario no existe
- Aviso de privacidad
- Créditos
- Inicio

3.2. Funcionalidades No a Probar

- Carrito
- Pago

4. Enfoque de las Pruebas

El enfoque general para la ejecución de pruebas en el proyecto Zazil está diseñado para garantizar la calidad y funcionalidad del producto. Se llevarán a cabo:

4.1 Tipos de Pruebas

1. **Pruebas Funcionales:** Se encargan de verificar que todas las características del sistema operen según lo esperado, incluyendo formularios y flujos de trabajo.
2. **Pruebas No Funcionales:**

- **Rendimiento:** Se encargan de evaluar la capacidad del sistema bajo cargas de trabajo variables.
 - **Seguridad:** Se encargan de identificar vulnerabilidades para proteger los datos de los usuarios.
 - **Usabilidad:** Se encargan de analizar la facilidad de uso de la aplicación y la página web.
3. **Pruebas de Compatibilidad:** Se encargan de analizar que la aplicación funcione en diversos dispositivos y sistemas operativos.

4.2 Técnicas de Pruebas

1. **Pruebas Manuales:** Se encargan de validar la funcionalidad de manera manual para cumplir con los requisitos.
2. **Pruebas Automatizadas:** Se encargan de implementar para mejorar la eficiencia y repetibilidad en futuras pruebas.
3. **Pruebas Exploratorias:** Se encargan de identificar problemas inesperados y evaluarán la experiencia del usuario de manera menos estructurada.

Este enfoque integral permite abordar todos los aspectos críticos del sistema, contribuyendo a un producto final confiable y de alta calidad.

5. Criterios de Aceptación

5.1. Criterios de Aprobación

Para que el proyecto Zazil sea considerado satisfactorio y cumpla con los estándares de calidad establecidos, deberá cumplir con los siguientes criterios de aprobación:

1. **Cumplimiento de Requisitos Funcionales:** Todas las funcionalidades especificadas deben estar implementadas y operar sin errores.
2. **Desempeño Aceptable:** La aplicación debe demostrar un buen rendimiento bajo diferentes cargas, con tiempos de respuesta rápidos y eficientes.

3. **Seguridad Adecuada:** No deben identificarse vulnerabilidades críticas durante las pruebas de seguridad, y se deben implementar las mejores prácticas de protección de datos.
4. **Experiencia de Usuario Positiva:** Las pruebas de usabilidad deben indicar que los usuarios pueden navegar por la aplicación y la página web de manera intuitiva y sin complicaciones.
5. **Compatibilidad:** La aplicación debe funcionar correctamente en todos los dispositivos y sistemas operativos especificados en el alcance de pruebas.

5.2. Criterios de Rechazo

Cualquier falla en los siguientes criterios resultará en el rechazo del producto hasta que se realicen las correcciones necesarias:

1. **Fallas en Funcionalidades Clave:** Si se encuentran errores críticos en las funciones principales que impiden su uso.
2. **Problemas de Desempeño:** Si la aplicación presenta tiempos de carga excesivos o no puede manejar la carga esperada de usuarios.
3. **Vulnerabilidades de Seguridad:** La identificación de vulnerabilidades críticas que puedan comprometer la seguridad de los datos de los usuarios.
4. **Experiencia de Usuario Deficiente:** Si las pruebas de usabilidad revelan que los usuarios enfrentan dificultades significativas al utilizar la aplicación.
5. **Incompatibilidad:** Si la aplicación no es funcional en los dispositivos y sistemas operativos especificados, afectando la accesibilidad para los usuarios.

6. Gestión de Pruebas

6.1. Roles y Responsabilidades

El equipo del proyecto Zazil está compuesto por cinco integrantes, cada uno con responsabilidades específicas en el proceso de pruebas:

- **Sandra, Fernanda y Estefanía:** Han estado realizando pruebas en el backend y la página web del administrador, enfocándose en aspectos de seguridad, funcionalidad y sincronización. Su trabajo es crucial para garantizar que el sistema sea seguro y que las interacciones entre componentes funcionen de manera adecuada.
- **Alexei y Lorena:** Se han encargado de las pruebas de funcionalidad, interfaz de usuario (UI) y experiencia de usuario (UX) en la aplicación desarrollada en Android Studio. Su objetivo es asegurar que la aplicación sea intuitiva y fácil de usar, proporcionando una experiencia positiva a los usuarios finales.

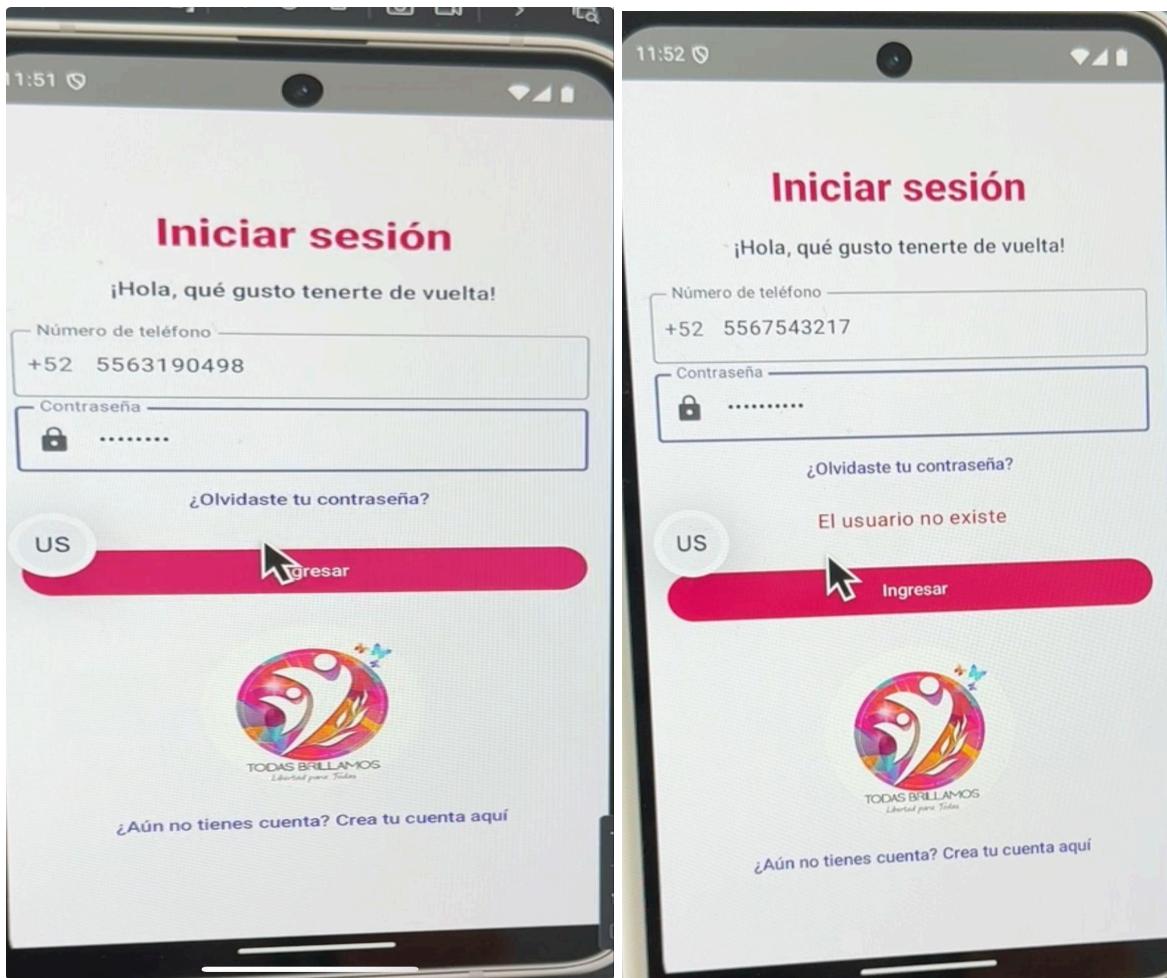
Cada miembro del equipo ha contribuido activamente al desarrollo y planeación de las pruebas, colaborando para identificar y resolver problemas y mejorar la calidad del proyecto en su conjunto.

6.2. Herramientas de Pruebas

Para la gestión de las pruebas realizadas durante el desarrollo del proyecto, utilizamos Trello como herramienta principal. Trello nos permite organizar y visualizar las diferentes tareas y pruebas de manera clara y estructurada, facilitando el seguimiento de su progreso. A través de tableros, listas y tarjetas, podemos asignar tareas, establecer prioridades, y registrar los resultados de cada prueba, lo que mejora nuestra capacidad para coordinar el trabajo en equipo y garantizar que todas las pruebas se lleven a cabo de manera efectiva.

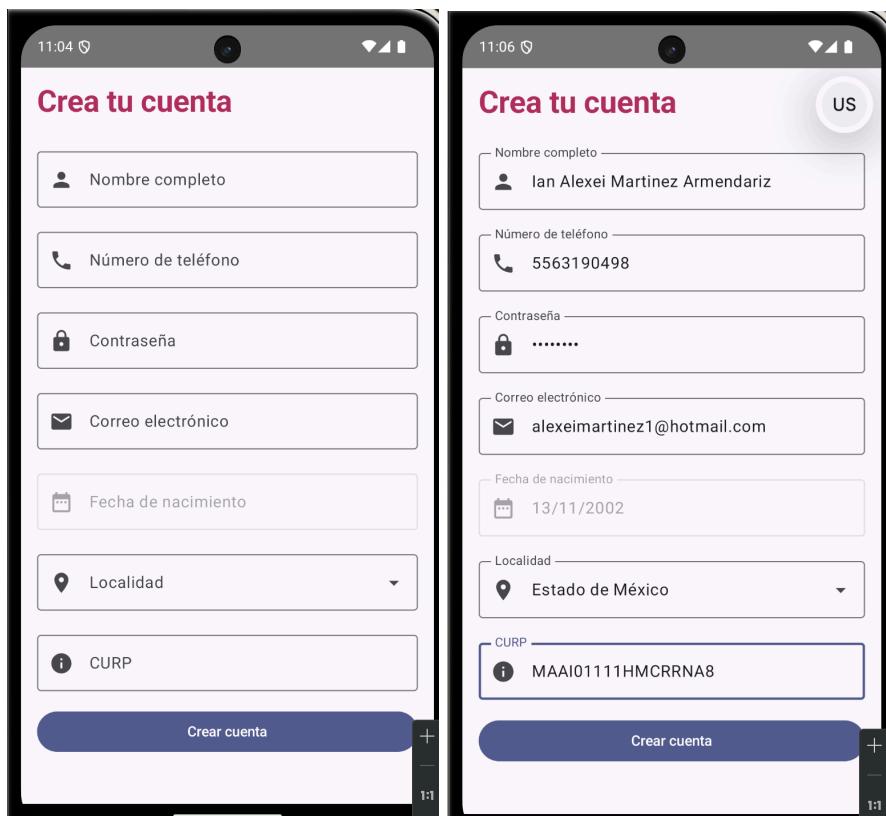
7. Entregables de las Pruebas

- Inicio de sesión y usuario no existente



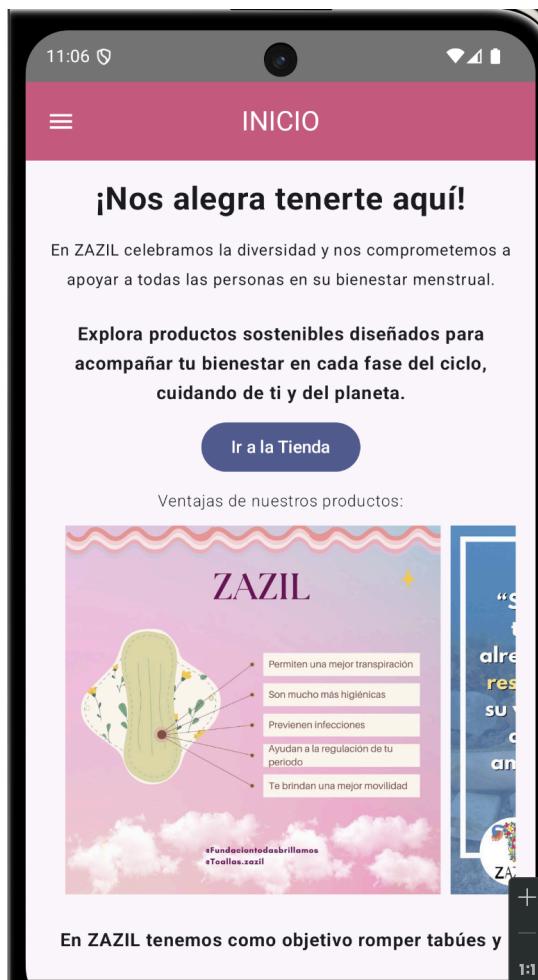
Al iniciar sesión se mandará directamente a la página de inicio, si el usuario no se ha registrado tendrá el mensaje de usuario no existe

- Crear cuenta de usuario



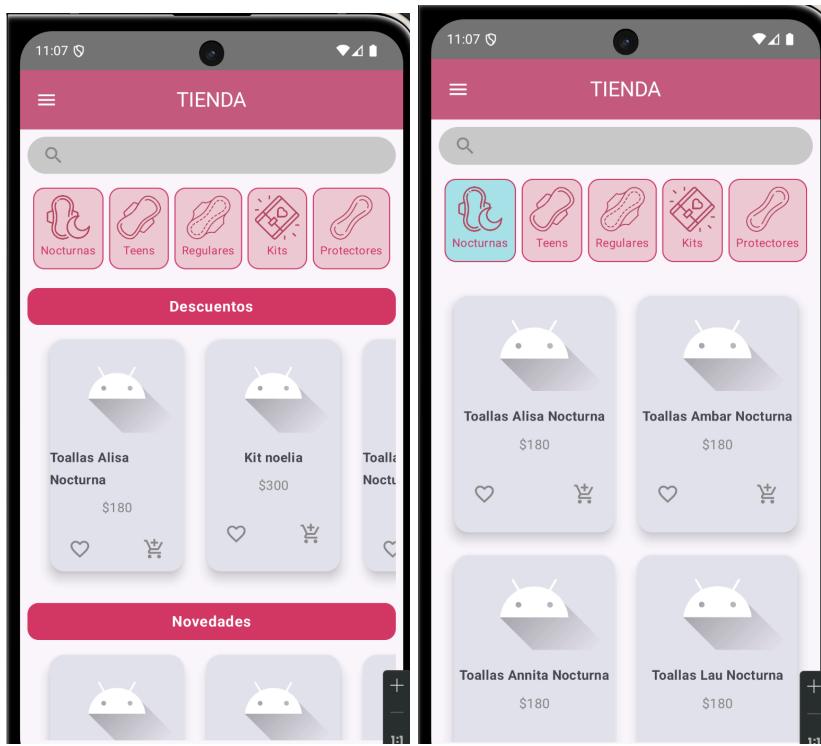
Se solicita la información correspondiente para crear un usuario, datos tanto básicos como los que la asociación requiere. La aplicación apoya al usuario para el ingreso de distintos datos: en la contraseña se oculta, en la fecha de nacimiento se proporciona un calendario, la localidad muestra opciones.

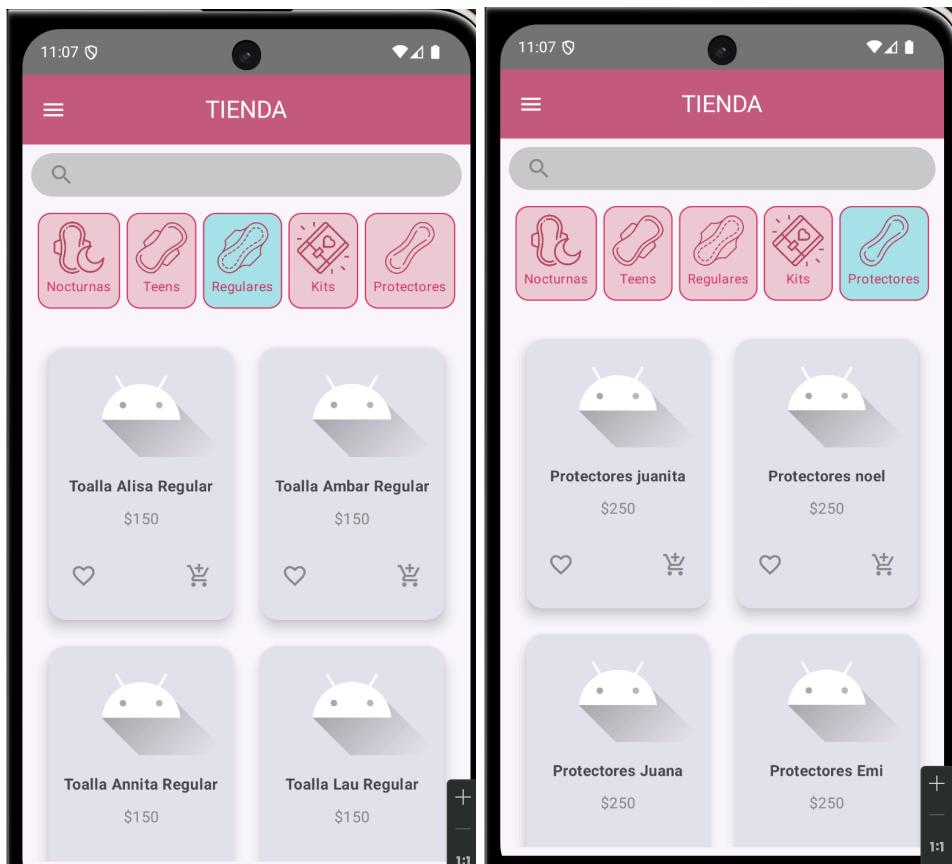
- Inicio



En la pantalla de inicio se muestran todas las secciones de forma llamativa para invitar al usuario a navegar en esas pantallas.

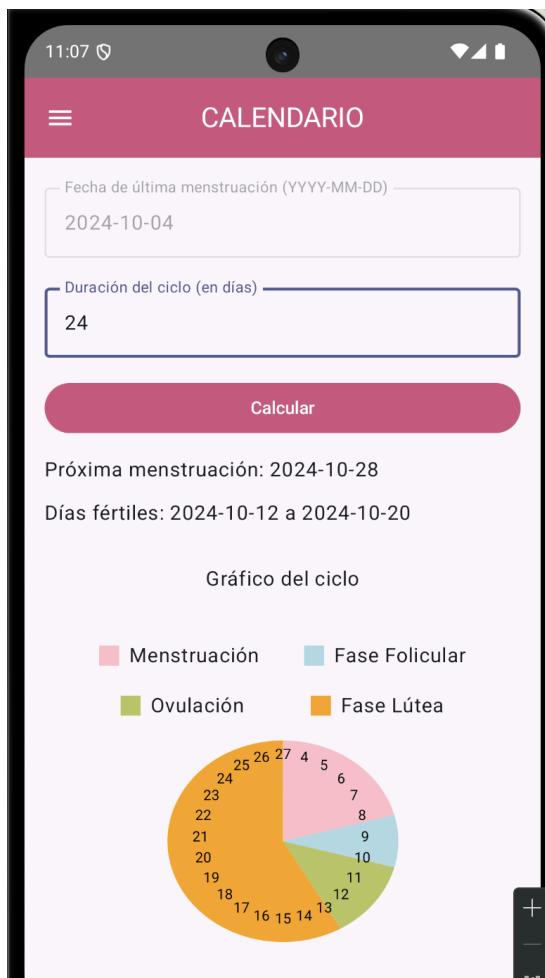
- Tienda y filtros





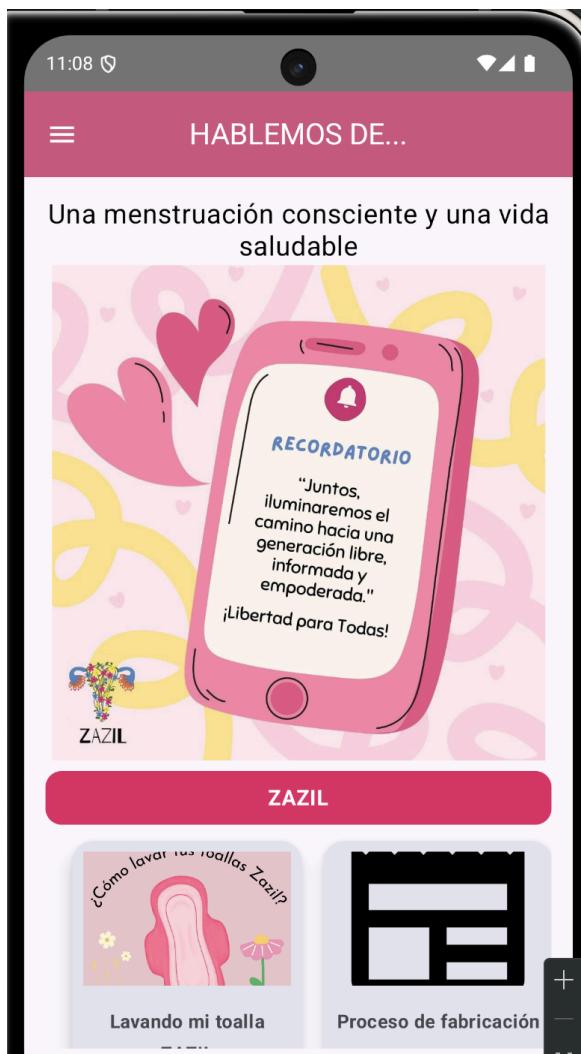
La sección de tiendas cuenta con el apartado de descuentos, novedades y todos los productos. Sin embargo para facilitarle la navegación al usuario, se incorporó la barra de búsqueda y una sección de filtros en donde al elegir uno, se cambia de color con el objetivo de recordarle al usuario su elección.

- Calendario menstrual



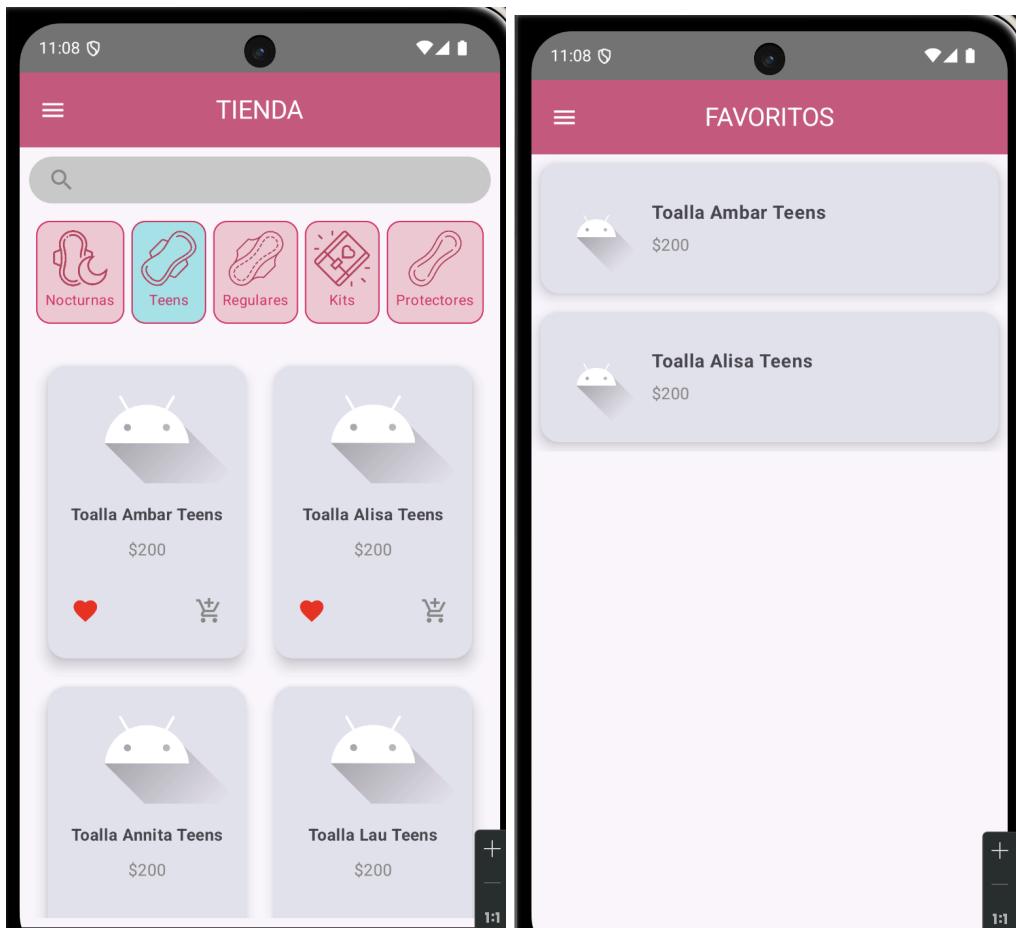
La pantalla de calendario menstrual apoya al usuario para calcular su siguiente periodo menstrual, en donde muestra las distintas etapas mostrando los días y se ajusta a la duración de cada persona.

- Hablemos de.....



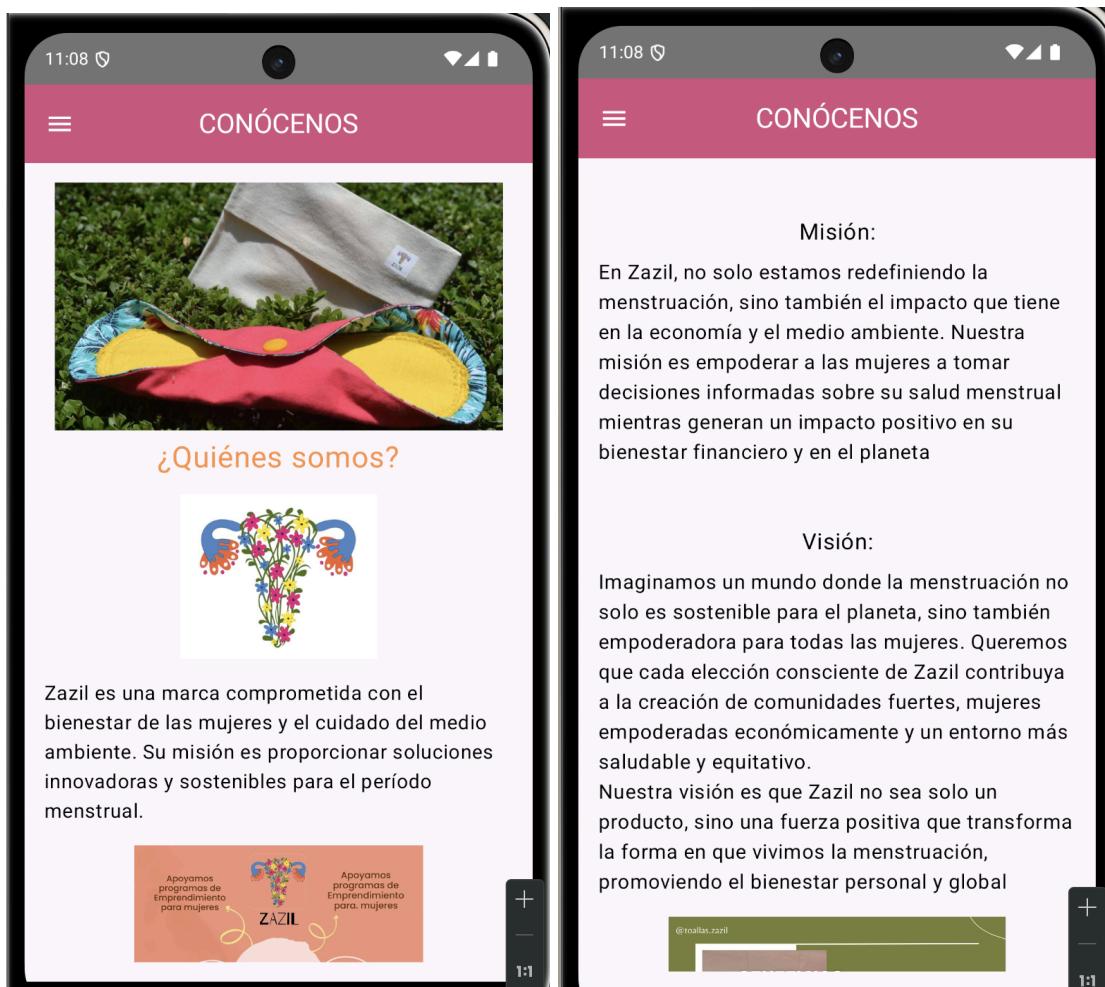
El apartado “Hablemos de...” tiene como objetivo informar y educar sobre distintos aspectos de salud menstrual, de zazil y datos curiosos, para romper el tabú.

- Favoritos



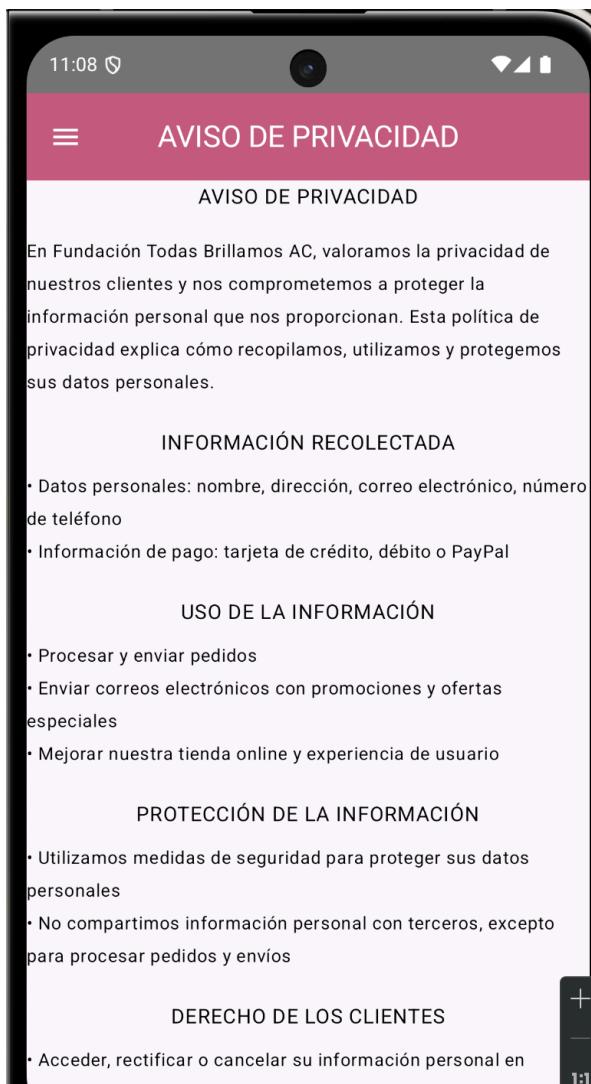
La pantalla de favoritos se llena dependiendo del usuario, el cual elegirá el producto de la tienda y al presionar el botón con forma de corazón se añadirá ese producto.

- Conocenos



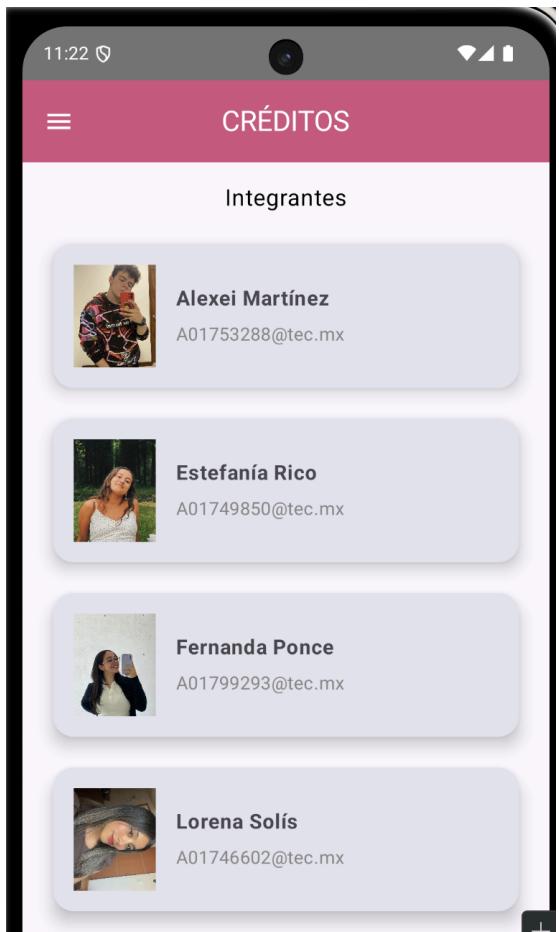
La pantalla de conócenos tiene como objetivo dar a conocer a la marca zazil y a la asociación todas brillamos.

- Aviso de privacidad



La pantalla contiene el aviso de privacidad y diversas políticas de la marca y asociación.

- Créditos



En la pantalla créditos se muestran a todos los integrantes del equipo.

8. Cronograma de Pruebas



Deadline sobre las pruebas en proceso: 11/10/24

Deadline sobre pruebas por hacer: 13/10/24

9. Recursos de Pruebas

9.1. Recursos Humanos

El equipo del proyecto Zazil está compuesto por cinco integrantes, quienes han asumido roles específicos en las pruebas:

1. **Sandra:** Participa activamente en las pruebas del backend y la página web del administrador, contribuyendo al análisis de seguridad y a la verificación de que las funcionalidades operen según lo especificado.
2. **Fernanda:** Se enfoca en la sincronización entre componentes y la funcionalidad del backend, realizando pruebas exhaustivas para garantizar una integración fluida en el sistema.
3. **Estefanía:** Responsable de las pruebas de seguridad y funcionalidad en el backend y la página web del administrador, asegurando que los sistemas sean robustos y estén protegidos contra vulnerabilidades.
4. **Alexei:** Encargado de las pruebas de funcionalidad y diseño de la interfaz de usuario (UI) en la aplicación Android, asegurándose de que la aplicación sea intuitiva y fácil de usar.
5. **Lorena:** Focalizada en las pruebas de experiencia de usuario (UX) y funcionalidad en la aplicación, trabajando para optimizar la interacción del usuario con la interfaz.

Cada miembro del equipo desempeña un papel esencial en el proceso de pruebas, colaborando para lograr un producto final de alta calidad.

9.2. Recursos de Hardware y Software

Para el desarrollo y las pruebas del proyecto Zazil, se han utilizado los siguientes recursos de hardware y software:

Recursos de Hardware

1. **Computadoras de desarrollo:** Computadoras que cumplen con los requisitos necesarios para el desarrollo y pruebas de software, incluyendo suficiente memoria RAM y capacidad de procesamiento.
2. **Dispositivos móviles:** Smartphones y tablets utilizados para realizar pruebas de funcionalidad y experiencia de usuario (UX) en la aplicación Android, asegurando su rendimiento en diferentes dispositivos.
3. **Servidores:** Servidores locales y/o en la nube que aloja la base de datos y permiten el acceso al backend y la página web del administrador.

Recursos de Software

1. **Sistemas Operativos:** Se utilizan sistemas operativos como Windows y Linux en las computadoras de desarrollo para facilitar la programación y pruebas de la aplicación.
2. **Kotlin:** Lenguaje de programación empleado para el desarrollo de la aplicación Android.
3. **React:** Biblioteca de JavaScript utilizada para la creación de la interfaz de usuario en la página web.
4. **Firestore y Realtime database:** Sistema de gestión de bases de datos relacional que se utiliza para almacenar y gestionar los datos del proyecto.
5. **Android Studio:** Entorno de desarrollo integrado (IDE) utilizado para desarrollar y probar la aplicación en Android.
6. **Trello:** Herramienta de gestión de proyectos que facilita la planificación, seguimiento y organización de las pruebas realizadas en el proyecto.
7. **Firebase:** Plataforma de despliegue en la nube que se utiliza como servidor para alojar el backend y la base de datos del proyecto, facilitando el acceso y la gestión de los recursos de manera eficiente.

Estos recursos son esenciales para el éxito del desarrollo y la implementación del proyecto Zazil, permitiendo realizar pruebas exhaustivas y garantizar la calidad del producto final.

10. Entorno de Pruebas

Las pruebas del proyecto Zazil se llevaron a cabo en un entorno configurado para garantizar la efectividad y la precisión de los resultados. A continuación, se describen los componentes del entorno de pruebas:

Configuraciones de Dispositivos Móviles

- **Dispositivos móviles:** Se realizaron pruebas en smartphones de diferentes marcas y modelos, asegurando la compatibilidad y la experiencia de usuario en diversas plataformas. Las versiones de los sistemas operativos Android serán actualizadas disponibles en el momento de las pruebas.

Conexiones de Red

- **Conexiones de red:** Las pruebas se llevarán a cabo en una red Wi-Fi estable, asegurando una conexión a Internet confiable para acceder a la base de datos y al servidor. También se evaluarán las condiciones de uso en redes móviles para verificar la funcionalidad de la aplicación en diferentes tipos de conexión.

Herramientas de Soporte

- **Hardware:** Se utilizaron 4 MacBook Pro (modelos 2021 y 2023) y una computadora Inspiron 14 DELL para el desarrollo y las pruebas del proyecto. Estos dispositivos están equipados con especificaciones adecuadas para manejar el desarrollo de software y realizar pruebas intensivas.
- **Software:** Se emplearon las versiones más actualizadas de herramientas como:
 - **MySQL:** Última versión estable disponible (actualmente, MySQL 8.0).
 - **React:** Versión más reciente en el momento de las pruebas (actualmente, React 18).
 - **Kotlin:** La versión más reciente (actualmente, Kotlin 1.7).
 - **Android Studio:** Versión más actual (actualmente, Android Studio Bumblebee).

Este entorno de pruebas está diseñado para proporcionar un marco robusto y versátil que permita identificar y resolver problemas de manera efectiva, asegurando la calidad y funcionalidad del producto final.

11. Gestión de Defectos

El proceso de gestión de defectos es fundamental para asegurar la calidad del producto final. Dentro de este proyecto, hemos establecido un flujo para la identificación, reporte, seguimiento y resolución de defectos, siguiendo mejores prácticas de la ingeniería de software. El proceso se lleva a cabo de la siguiente manera:

1. **Identificación de Defectos:** Durante las pruebas funcionales y no funcionales, los testers identifican cualquier comportamiento inesperado o problema que impida que la aplicación cumpla con los requisitos especificados. Estos defectos se documentan claramente, incluyendo información relevante como el contexto en el que se presentó el problema, pasos para reproducirlo, y la severidad del defecto.
2. **Reporte de Defectos:** Una vez identificado, cada defecto se reporta en nuestras minutos, y lo añadimos como prueba en Trello. El reporte incluye la prueba que se debe de hacer para poder abordarlo.
3. **Clasificación y Priorización:** Los defectos reportados son clasificados en función de su severidad e impacto en el sistema. Los defectos críticos, que pueden afectar la operación principal de la aplicación, reciben alta prioridad para su pronta resolución, mientras que los defectos menores se programan para corregirse en ciclos futuros.
4. **Asignación y Seguimiento:** Cada defecto/prueba es asignado a un miembro del equipo de desarrollo, quien es responsable de su análisis y resolución. Se lleva un seguimiento constante del estado del defecto, desde “para hacer” hasta “completo”.
5. **Resolución y Verificación:** Una vez que el desarrollador realiza la corrección del defecto, el equipo de pruebas procede a ejecutar las pruebas de verificación para confirmar que el problema ha sido resuelto sin afectar otras funcionalidades. Si el defecto persiste, se reabre y se somete nuevamente al ciclo de corrección.
6. **Documentación y Lecciones Aprendidas:** Cada defecto identificado y resuelto es documentado para su análisis posterior. Esto permite identificar patrones recurrentes y mejorar tanto los procesos de desarrollo como los de prueba, con el objetivo de reducir la recurrencia de defectos similares.

Este enfoque estructurado garantiza que los defectos se gestionen de manera eficiente, reduciendo al mínimo su impacto en la calidad del producto y asegurando una experiencia positiva para el usuario final. El objetivo final es entregar una aplicación robusta que permita la comercialización de las toallas menstruales, así como brindar funcionalidades adicionales como el blog y el calendario menstrual de manera confiable.

12. Evaluación de Riesgos y Contingencias

En el proceso de pruebas de este proyecto, se han identificado diversos riesgos que podrían impactar la calidad y el cronograma del desarrollo. A continuación, se describen los riesgos principales y las estrategias de contingencia diseñadas para mitigarlos:

1. **Riesgo: Retraso en la entrega de dispositivos de prueba**
 - **Contingencia:** Utilizar emuladores o simuladores para continuar con las pruebas mientras se resuelve el retraso en la entrega de los dispositivos físicos.
2. **Riesgo: Limitaciones de recursos humanos durante las pruebas**
 - **Contingencia:** Redistribuir tareas entre los miembros del equipo y considerar el uso de pruebas automatizadas para reducir la carga manual y acelerar el proceso.
3. **Riesgo: Fallos en la infraestructura o herramientas de prueba**
 - **Contingencia:** Tener disponible un entorno alternativo para pruebas y realizar respaldos frecuentes de los entornos y herramientas utilizadas, garantizando la rápida recuperación y continuidad del trabajo.
4. **Riesgo: Baja calidad del código entregado para pruebas**
 - **Contingencia:** Implementar pruebas unitarias y documentar el código, así como realizar revisiones de código periódicas para identificar y corregir problemas.
5. **Riesgo: Problemas de conectividad durante las pruebas en entornos remotos**

- **Contingencia:** Realizar pruebas fuera de línea cuando sea posible y almacenar los resultados para su sincronización posterior. También se recomienda tener acceso a redes alternativas, como conexiones móviles, para reducir el impacto.
6. **Riesgo: Escenario de prueba no representativo del entorno real**
- **Contingencia:** Realizar pruebas con usuarios potenciales, para garantizar una cobertura adecuada y representativa de las condiciones reales de los usuarios.

Estas estrategias buscan minimizar la probabilidad de ocurrencia y el impacto de los riesgos en el proceso de pruebas, asegurando la calidad del producto y cumpliendo con los tiempos previstos para el proyecto.

13. Criterios de Suspensión y Reanudación de las Pruebas

Suspensión: Las pruebas se suspenderán si se descubre un defecto crítico que imposibilite el avance del proceso de pruebas, o si existen problemas técnicos graves (como fallos en la infraestructura).

Reanudación: Las pruebas se reanudarán una vez que el defecto crítico haya sido corregido y verificado, y se confirme la estabilidad del entorno de pruebas.

14. Escenarios de Prueba y Casos de Prueba

14.1. Pruebas Unitarias

14.1.1. Escenarios de Prueba Unitarias

1. Escenario de Prueba Unitaria - login()

- **Título del Escenario:** Validación de la función login().
- **Descripción:** Verificar que la función login() maneja correctamente el proceso de autenticación y actualiza el estado según el resultado.
- **Precondiciones:** La función está configurada con UserRepository.
- **Pasos Generales:**
 1. Inicializar login() con un número de teléfono y contraseña válidos.
 2. Verificar que el estado cambia a Loading.
 3. Esperar el resultado del login.
- **Resultado Esperado:** La función cambia el estado a Success cuando las credenciales son correctas o a Error cuando son incorrectas.

2. Escenario de Prueba Unitaria - registerUser()

- **Título del Escenario:** Validación de la función registerUser().
- **Descripción:** Verificar que la función registerUser() maneja correctamente el proceso de registro de nuevos usuarios.
- **Precondiciones:** La función está conectada al UserRepository y Firebase.
- **Pasos Generales:**
 1. Inicializar el SignUpViewModel.
 2. Llamar a registerUser() con datos válidos de usuario.
 3. Observar que el estado cambia a Loading mientras se realiza el registro.
- **Resultado Esperado:** El estado cambia a Success tras el registro exitoso o a Error si falla.

3. Escenario de Prueba Unitaria - Clase SignUpState{}

- **Título del Escenario:** Validación del estado en SignUpState.
- **Descripción:** Verificar que el estado de SignUpState refleja correctamente el resultado del proceso de registro de usuario.
- **Precondiciones:** El UserRepository está inicializado y listo para el registro de usuarios.

- **Pasos Generales:**
 1. Inicializar SignUpState.
 2. Ingresar un nombre de usuario, correo y contraseña válidos.
 3. Llamar a signUp().
 4. Observar el estado de SignUpState.
- **Resultado Esperado:** El estado cambia a Success tras un registro exitoso o a Error en caso de fallo.

4. Escenario de Prueba Unitaria - calcularProximaMenstruacion()

- **Título del Escenario:** Validación de la función calcularProximaMenstruacion().
- **Descripción:** Verificar que la función calcularProximaMenstruacion() calcula correctamente la fecha de la próxima menstruación basada en la fecha de la última y la duración del ciclo.
- **Precondiciones:** Se tiene la fecha de la última menstruación y la duración del ciclo en días.
- **Pasos Generales:**
 1. Inicializar la función con una fecha de última menstruación válida y duración del ciclo.
 2. Llamar a calcularProximaMenstruacion().
- **Resultado Esperado:** La función retorna la fecha estimada de la próxima menstruación.

5. Escenario de Prueba Unitaria - calcularDiasFertiles()

- **Título del Escenario:** Validación de la función calcularDiasFertiles().
- **Descripción:** Verificar que la función calcularDiasFertiles() calcula correctamente los días fértiles en base a la próxima menstruación y la duración del ciclo.
- **Precondiciones:** Se tiene la fecha de la próxima menstruación y la duración del ciclo.
- **Pasos Generales:**

1. Inicializar la función con la fecha de la próxima menstruación y la duración del ciclo.
 2. Llamar a calcularDiasFertiles().
- **Resultado Esperado:** La función retorna el periodo de días fértiles como un par de fechas (inicio y fin).

6. Escenario de Prueba Unitaria - sendPasswordEmail()

- **Título del Escenario:** Validación de la función sendPasswordEmail().
- **Descripción:** Verificar que la función sendPasswordEmail() envía correctamente el correo de recuperación de contraseña.
- **Precondiciones:** El correo del usuario está registrado en el sistema.
- **Pasos Generales:**
 1. Inicializar la función con el correo electrónico del usuario.
 2. Llamar a sendPasswordEmail().
- **Resultado Esperado:** La función envía un correo de recuperación de contraseña al usuario.

7. Escenario de Prueba Unitaria - MyApp()

- **Título del Escenario:** Validación de la función MyApp() en Jetpack Compose.
- **Descripción:** Verificar que la función MyApp() define correctamente la estructura general de la aplicación y muestra las pantallas según la ruta actual.
- **Precondiciones:** El navController está inicializado y configurado con las rutas de navegación.
- **Pasos Generales:**
 1. Inicializar la función MyApp().
 2. Verificar que la pantalla de login se muestra correctamente cuando la ruta es login.
 3. Verificar que la pantalla de registro se muestra cuando la ruta es signup.

- **Resultado Esperado:** La función muestra correctamente las pantallas de login o registro según la ruta.

8. Escenario de Prueba Unitaria - NavigationComponent()

- **Título del Escenario:** Validación de la función NavigationComponent() en Jetpack Compose.
- **Descripción:** Verificar que la función NavigationComponent() maneja correctamente las rutas de navegación entre las diferentes pantallas de la aplicación.
- **Precondiciones:** El navController está configurado con las rutas correspondientes.
- **Pasos Generales:**
 1. Inicializar la función NavigationComponent().
 2. Navegar entre las pantallas de login, signup, y password recovery.
- **Resultado Esperado:** La función navega correctamente entre las pantallas según la ruta seleccionada.

9. Escenario de Prueba Unitaria - hashPassword()

- **Título del Escenario:** Validación de la función hashPassword() en UserRepository.
- **Descripción:** Verificar que la función hashPassword() genera correctamente un hash seguro de la contraseña del usuario.
- **Precondiciones:** La función bcrypt está configurada correctamente.
- **Pasos Generales:**
 1. Inicializar la función con una contraseña en texto claro.
 2. Llamar a hashPassword().
- **Resultado Esperado:** La función retorna un hash cifrado de la contraseña.

10. Escenario de Prueba Unitaria - hashCurp()

- **Título del Escenario:** Validación de la función hashCurp() en UserRepository.
- **Descripción:** Verificar que la función hashCurp() genera correctamente un hash seguro de la CURP del usuario.
- **Precondiciones:** La función bcrypt está configurada correctamente.
- **Pasos Generales:**
 1. Inicializar la función con una CURP en texto claro.
 2. Llamar a hashCurp().
- **Resultado Esperado:** La función retorna un hash cifrado de la CURP.

11. Escenario de Prueba Unitaria - signUpUser()

- **Título del Escenario:** Validación de la función signUpUser() en UserRepository.
- **Descripción:** Verificar que la función signUpUser() registra correctamente un nuevo usuario en Firebase.
- **Precondiciones:** El usuario no debe existir en la base de datos de Firebase.
- **Pasos Generales:**
 1. Inicializar la función con los datos de usuario (nombre, correo, contraseña, CURP).
 2. Llamar a signUpUser().
- **Resultado Esperado:** El usuario se registra correctamente o retorna un error si el usuario ya existe.

12. Escenario de Prueba Unitaria - loginUser()

- **Título del Escenario:** Validación de la función loginUser() en UserRepository.
- **Descripción:** Verificar que la función loginUser() maneja correctamente el inicio de sesión de un usuario, validando las credenciales proporcionadas.
- **Precondiciones:** El usuario está registrado en la base de datos de Firebase con contraseña cifrada.
- **Pasos Generales:**

1. Inicializar la función con el número de teléfono y la contraseña del usuario.
 2. Llamar a loginUser().
- **Resultado Esperado:** La función permite el inicio de sesión correcto o retorna un error si las credenciales son incorrectas.

13. Escenario de Prueba Unitaria - PU-13

- **Título del Escenario:** Verificar la integración entre el backend y el frontend de la tienda.
- **Descripción:** Validar que los productos agregados por el administrador en el backend se muestran correctamente en el frontend de la tienda.
- **Precondiciones:** El administrador ha agregado productos al sistema a través del backend.
- **Pasos Generales:**
 1. El administrador agrega productos al sistema desde el backend.
 2. El frontend de la tienda carga los datos de los productos.
 3. Verificar que los productos se muestran correctamente en la interfaz de la tienda.
- **Resultado Esperado:** Los productos agregados en el backend se visualizan correctamente en la tienda, incluyendo su información como nombre, precio y disponibilidad.

14. Escenario de Prueba Unitaria - PU-14

- **Título del Escenario:** Validar la integración entre el sistema de donaciones y el backend.
- **Descripción:** Verificar que las donaciones realizadas por los usuarios se registran correctamente en la base de datos y se reflejan en el perfil del usuario.
- **Precondiciones:** El usuario ha seleccionado realizar una donación.
- **Pasos Generales:**
 1. El usuario selecciona una cantidad para la donación.
 2. Se procesa la donación en el sistema y se guarda en la base de datos.
 3. Verificar que la donación se refleja correctamente en el perfil del usuario.

- **Resultado Esperado:** La donación se registra en la base de datos y se actualiza el perfil del usuario mostrando la cantidad donada.

15. Escenario de Prueba Unitaria - ProductDetails()

- **Título del Escenario:** Verificar que los productos se muestran correctamente en la página web.
- **Descripción:** Validar que los productos agregados por el administrador a través de la función ProductDetails() se muestran correctamente en la página web de la tienda.
- **Precondiciones:** El administrador ha agregado productos a la tienda mediante el backend.
- **Pasos Generales:**
 1. El administrador agrega productos a través del backend.
 2. Se llama a la función ProductDetails() para obtener los detalles de los productos.
 3. Verificar que los productos se visualizan correctamente en el frontend con toda la información relevante.
- **Resultado esperado:** Los productos se muestran correctamente en la tienda, con detalles como nombre, descripción, precio e imágenes.

16. Escenario de Prueba Unitaria - datatable()

- **Título del Escenario:** Poder visualizar los usuarios en la página web.
- **Descripción:** Verificar que los usuarios registrados en la aplicación pueden visualizarse correctamente en la página web a través de una tabla de datos.
- **Precondiciones:** Varios usuarios se han registrado en la aplicación.
- **Pasos Generales:**
 1. Los usuarios se registran en la aplicación.
 2. El administrador accede a la página web y carga los usuarios en una tabla de datos usando datatable().

3. Verificar que los usuarios se muestran correctamente en la tabla con su información detallada.
- **Resultado Esperado:** Los usuarios registrados se visualizan correctamente en la página web mediante una tabla de datos, mostrando información como nombre, correo y otros detalles.

17. Escenario de Prueba Unitaria - firebaseConfig{}

- **Título del Escenario:** Verificar la conexión con el servidor de Firebase.
- **Descripción:** Validar que la conexión con el servidor de Firebase funciona correctamente y que los datos de usuarios o donaciones se actualizan en tiempo real.
- **Precondiciones:** Se ha configurado firebaseConfig{} correctamente y varios usuarios se han registrado en la aplicación.
- **Pasos Generales:**
 1. Verificar la correcta configuración de firebaseConfig{} para conectar con Firebase.
 2. Realizar cambios en los datos de la aplicación (como usuarios o donaciones).
 3. Comprobar que los cambios se reflejan en la base de datos de Firebase.
- **Resultado Esperado:** La aplicación se conecta correctamente al servidor de Firebase y los cambios realizados se actualizan en la base de datos en tiempo real.

14.1.2. Casos de Prueba Unitarias

Plantilla de Caso de Prueba Unitaria:

ID	Descripción	Método/Función	Precondiciones	Entradas de Prueba	Resultado Esperado	Estado	Comentarios	Código de la Prueba
PU-01	Validar que la función login() valida que los valores de entrada sean correctos.	login()	Lista de usuarios guardados en la base de datos.	telefono = "+5512098960", password = "Miskey123"	Retorna "Success" y se puede ingresar a la app.	Aprobada	Necesario para el inicio de sesión.	<pre> fun login(phone: String, password: String, sessionViewModel: SessionViewModel) { _loginState.value = LoginState.Loading viewModelScope.launch { userRepository.login(phone, password) val user = userRepository.getLoggedInUser() if (user != null) { _loginState.value = LoginState.Success sessionViewModel.setUserName(user.name) } else { _loginState.value = LoginState.Error("User not found") } } } </pre>
PU-02	Verificar que la función registerUser() revisa que los datos ingresados sean válidos y se pueda registrar el usuario.	registerUser()	El usuario se va a registrar por primera vez.	nombre = "Lorena Solis", telefono = "+5572247242", email = "lorenas_solis@hotmail.com", password = "Miskey123", localidad = "Ciudad de México", curp = "SOLL020315MMCXXX00"	Verifica los datos y los guarda en la base de datos.	Aprobada	Necesario para el registro de un nuevo usuario.	<pre> registerUser(name = name, phone = phone, password = password, email = email, birthdate = birthdate, location = location, curp = curp, onSuccess = { _signUpState.value = SignUpState.Success }, onFailure = { errorMessage -> _signUpState.value = SignUpState.Error(errorMessage) }) </pre>
PU-03	Validar que se ha hecho un registro exitoso.	SignUpState()	El UserRepository está inicializado y listo para el registro de usuarios.	nombre = "Lorena Solis", telefono = "+5572247242", email = "lorenas_solis@hotmail.com", password = "Miskey123", localidad = "Ciudad de México", curp = "SOLL020315MMCXXX00"	Retorna el cálculo del próximo ciclo menstrual.	Aprobada	Necesario para el registro de un nuevo usuario.	<pre> sendEvent(SignUpState(object State : SignUpState { object Success : SignUpState() { object Loading : SignUpState() { fun calculateNextMenstruation(LocalDate durationCiclo: LocalDate, LocalDate durationCiclo: Int): LocalDate { return fechahUltimaMenstruacion.plusDays(durationCiclo.toLong()) } fun calculateDaysBeforeNextMenstruation(durationCiclo: Int): Pair<localdate, localdate=""> { val ovulation = proximaMenstruacion.minusDays((durationCiclo / 2).toLong()) return Pair(ovulation, minusDays(4)) } } } })) </localdate,></pre>
PU-04	Validar que la función retorna la fecha correcta de la próxima menstruación.	calcularProximaMenstruacion()	El usuario ha iniciado sesión y tiene la fecha de la última menstruación y la duración del ciclo en días.	fechahUltimaMenstruacion = LocalDate.of(2024, 10, 1), duracionCiclo = 28	Regresa la fecha del próximo ciclo.	Aprobada	Necesaria para la funcionalidad del calendario menstrual	<pre> calcularProximaMenstruacion(fechahUltimaMenstruacion: LocalDate, duracionCiclo: Int): LocalDate { return fechahUltimaMenstruacion.plusDays(duracionCiclo.toLong()) } fun calcularDiasAntesPróximaMenstruación(duracionCiclo: Int): Pair<localdate, localdate=""> { val ovulation = proximaMenstruacion.minusDays((duracionCiclo / 2).toLong()) return Pair(ovulation, minusDays(4)) } </localdate,></pre>
PU-05	Validar que la función calcula correctamente los días fértils.	calcularDiasFertiles()	El usuario ha resgitrado ciclos anteriores.	Fecha de inicio del último ciclo	Regresa un calendario indicando los días fértils del próximo ciclo.	Aprobada	Necesaria para la funcionalidad del calendario menstrual	<pre> calcularDiasFertiles(cicloActual: LocalDate, duracionCiclo: Int): LocalDate { val ovulation = proximaMenstruacion.minusDays((duracionCiclo / 2).toLong()) return Pair(ovulation, minusDays(4)) } </pre>

https://docs.google.com/spreadsheets/d/1v5jh6MLU8g2lxItC0_QM8yj6fxRHB_hyuCwoMssh9TM/edit?usp=sharing

14.1.3. Estrategia de Pruebas Unitarias

1. Cobertura de Código

La meta de cobertura de código será asegurar que al menos un **80%** del código relevante esté cubierto por las pruebas unitarias. Se generarán reportes de cobertura utilizando **Istanbul** para identificar las áreas del código que no están siendo cubiertas por las pruebas.

- **Cobertura Meta:**

- El **80% de cobertura de código** se refiere a líneas de código, ramas de decisión, y funciones. Esto incluye probar todos los flujos posibles en las funciones mencionadas.

- **Reporte de Cobertura:**

- Se generarán reportes de cobertura en formato HTML o LCOV, y se incluirán en el pipeline de CI/CD para que los desarrolladores puedan revisar la cobertura después de cada ejecución de prueba.

2. Estrategias de Prueba por Función

a) Función calculateTotal()

- **Framework:** Jest.
- **Automatización:** Se ejecutará automáticamente en el pipeline después de cada commit.
- **Cobertura:** Se cubrirán todas las combinaciones de productos y cantidades, probando casos de límite como listas vacías y precios cero.

b) Función validarEmail()

- **Framework:** Jest, con uso de mocks para simular correos inválidos y válidos.
- **Automatización:** Configurada para correr en cada push a la rama principal.
- **Cobertura:** Pruebas de todos los tipos de correos (válidos e inválidos) para asegurar que se cubran todos los casos posibles.

c) Función applyDiscount()

- **Framework:** Jest.
- **Automatización:** Integrada en el pipeline con GitHub Actions.
- **Cobertura:** Probar varios porcentajes de descuento, incluyendo casos como descuentos del 0% y 100%.

d) Función updateProfile()

- **Framework:** Jest con Sinon para simular el acceso a la base de datos.
- **Automatización:** Se ejecutará en cada merge.
- **Cobertura:** Probar que todos los campos del perfil se actualicen correctamente y que se manejen errores de validación.

e) Función calculateCyclePrediction()

- **Framework:** Jest.

- **Automatización:** Parte del pipeline en cada commit.
- **Cobertura:** Asegurar la predicción del ciclo con diferentes historiales menstruales, cubriendo casos de ciclos irregulares y regulares.

f) Función validatePassword()

- **Framework:** Jest.
- **Automatización:** Ejecutada en CI/CD en cada push.
- **Cobertura:** Probar contraseñas válidas e inválidas, cubriendo casos de límite como contraseñas vacías y caracteres especiales.

g) Función processPayment()

- **Framework:** Jest para simular la interacción con la API de pagos.
- **Automatización:** Integrada en CI/CD.
- **Cobertura:** Validar los diferentes flujos de pago (exitoso y fallido) con valores límites y casos erróneos.

h) Función getUserNotifications()

- **Framework:** Jest.
- **Automatización:** Se ejecuta como parte del pipeline después de cada commit.
- **Cobertura:** Probar que todas las notificaciones se generen correctamente en diferentes situaciones del sistema.

14.2. Pruebas de Integración

14.2.1. Escenarios de Prueba de Integración

Escenario 1:

- **Título del Escenario:** Registrar un nuevo usuario en la aplicación.
- **Descripción:** Validar que un usuario nuevo pueda registrarse en la aplicación utilizando Firebase Authentication y que sus datos se guarden correctamente en Firebase Realtime Database.
- **Precondiciones:** El usuario no debe estar registrado en Firebase.
- **Pasos Generales:**
 1. Ingresar los datos del usuario: nombre, teléfono, email, contraseña, CURP, localidad.
 2. Validar que el usuario no existe en Firebase Authentication.
 3. Registrar el usuario en Firebase Authentication.
 4. Guardar los datos del usuario en Firebase Realtime Database.
 5. Mostrar un mensaje de éxito de registro en la aplicación.
- **Resultado Esperado:** El usuario se registra exitosamente en Firebase Authentication y sus datos se guardan correctamente en Firebase Realtime Database.

Escenario 2:

- **Título del Escenario:** Inicio de sesión con credenciales válidas.
- **Descripción:** Verificar la interacción entre el frontend y Firebase al iniciar sesión con credenciales válidas.
- **Precondiciones:** El usuario debe estar registrado en Firebase con la contraseña cifrada.
- **Pasos Generales:**
 1. Ingresar el teléfono y la contraseña del usuario.
 2. Validar que el usuario exista en Firebase Authentication.
 3. Verificar la contraseña cifrada.
 4. Iniciar sesión si las credenciales son correctas.
 5. Mostrar la pantalla de inicio de la aplicación.
- **Resultado Esperado:** El usuario puede iniciar sesión correctamente o recibe un mensaje de error si las credenciales son inválidas.

Escenario 3:

- **Título del Escenario:** Mostrar productos agregados en la tienda online.

- **Descripción:** Validar que los productos agregados en Firebase Firestore se muestren correctamente en la tienda online del frontend.
- **Precondiciones:** El administrador ha agregado productos a Firebase Firestore.
- **Pasos Generales:**
 1. Agregar un producto en Firebase Firestore.
 2. Consultar los productos en la tienda online.
 3. Verificar si los productos se muestran en la interfaz.
- **Resultado Esperado:** Los productos se muestran correctamente en la tienda online del frontend.

Escenario 4:

- **Título del Escenario:** Donaciones conectadas a la base de datos.
- **Descripción:** Verificar que el sistema de donaciones se conecte correctamente a Firebase Realtime Database y refleje la información en el perfil del usuario.
- **Precondiciones:** El usuario ha seleccionado una cantidad para donar.
- **Pasos Generales:**
 1. Ingresar la cantidad de la donación.
 2. Guardar la donación en Firebase Realtime Database.
 3. Verificar que la donación se refleje en el perfil del usuario.
- **Resultado Esperado:** La donación se guarda correctamente en Firebase y se muestra en el perfil del usuario.

Escenario 5:

- **Título del Escenario:** Recuperación de contraseña.
- **Descripción:** Verificar la integración de la funcionalidad de recuperación de contraseña con Firebase Authentication.
- **Precondiciones:** El correo del usuario está registrado en el sistema.
- **Pasos Generales:**
 1. Ingresar el correo electrónico en la pantalla de recuperación.
 2. Enviar el correo de recuperación a través de Firebase.
 3. Verificar que el usuario recibe el correo y puede restablecer su contraseña.

- **Resultado Esperado:** El usuario recibe un correo de recuperación y puede restablecer su contraseña.

Escenario 6:

- **Título del Escenario:** Sincronización de productos entre frontend y Firebase.
- **Descripción:** Validar la sincronización del frontend y backend al actualizar productos en Firebase Firestore.
- **Precondiciones:** El administrador ha modificado los detalles del producto en el panel de administración.
- **Pasos Generales:**
 1. Modificar los detalles de un producto en el panel de administración.
 2. Actualizar los datos en Firebase Firestore.
 3. Verificar que los cambios se reflejen en la tienda online en tiempo real.
- **Resultado esperado:** Los productos se actualizan correctamente y los cambios se reflejan en la tienda en tiempo real.

Escenario 7:

- **Título del Escenario:** Actualización de estadísticas del usuario.
- **Descripción:** Verificar que las estadísticas del usuario se actualizan correctamente en Firebase Realtime Database y se reflejan en el frontend.
- **Precondiciones:** El usuario ha realizado acciones que afectan sus estadísticas (ej. compras, donaciones).
- **Pasos Generales:**
 1. Obtener las estadísticas actuales del usuario.
 2. Actualizar las estadísticas en Firebase Realtime Database.
 3. Verificar que los cambios se reflejen en el perfil del usuario en el frontend.
- **Resultado Esperado:** Las estadísticas del usuario se actualizan correctamente y se reflejan en su perfil.

Escenario 8:

- **Título del Escenario:** Subida de imagen de producto.

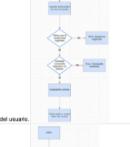
- **Descripción:** Validar que la imagen de un producto se suba correctamente a Firebase Storage y su URL se guarde en Firebase Firestore.
- **Precondiciones:** El administrador ha seleccionado una nueva imagen para el producto.
- **Pasos Generales:**
 1. Seleccionar una imagen para el producto.
 2. Subir la imagen a Firebase Storage.
 3. Guardar la URL de la imagen en Firebase Firestore.
 4. Mostrar la nueva imagen en la interfaz del producto.
- **Resultado Esperado:** La imagen se sube correctamente a Firebase Storage y la URL se actualiza en Firestore.

Escenario 9:

- **Título del Escenario:** Visualización de usuarios registrados en el administrador.
- **Descripción:** Verificar que los usuarios registrados se visualicen correctamente en la tabla del administrador en el frontend.
- **Precondiciones:** Varios usuarios han sido registrados en la aplicación.
- **Pasos Generales:**
 1. Obtener los datos de los usuarios registrados desde Firebase.
 2. Mostrar los usuarios en la tabla del administrador en el frontend.
 3. Verificar que los datos sean correctos y actualizados.
- **Resultado Esperado:** Los usuarios se muestran correctamente en la tabla del administrador.

14.2.2. Casos de Prueba de Integración

Plantilla de Caso de Prueba de Integración:

ID	Descripción	Componentes a Integrar	Precondiciones	Entradas de Prueba	Resultado Esperado	Estado	Comentarios	Diagramas de Flujo
PI-01	Validar la integración entre el frontend y Firebase al registrar un nuevo usuario.	registerUser(), Firebase Authentication, Firebase Realtime Database	Usuario no registrado en Firebase.	nombre = "Lorena Soto", teléfono = "3372247242", email = "lorencito.soto123@gmail.com", password = "Lorena123", localidad = "Coahuila", códigoPostal = "66000", id = "SOLICITUD19MNCXXXXX"	El usuario se registra exitosamente en Firebase Authentication y sus datos se guardan en Firebase Realtime Database.	Aprobada	Necesario para la creación de cuentas nuevas.	
PI-02	Verificar la interacción entre el frontend y Firebase al iniciar sesión con credenciales válidas.	loginUser(), Firebase Authentication, Realtime Database	Usuario registrado en Firebase y con contraseña válida.	teléfono = "3372247242", password = "Lorena123"	El usuario puede iniciar sesión correctamente o retorna un error si las credenciales son inválidas.	Aprobada	Importante para la validación de credenciales y seguridad del usuario.	
PI-03	Validar que los productos agregados en Firebase se muestren correctamente en la tienda online.	ProductDetail(), Firebase Firestore, Firebase Storage	El administrador ha agregado productos en Firebase.	Datos del producto en Firebase.	Los productos se muestran correctamente en la interfaz del frontend.	Aprobada	Crucial para la visibilidad de productos en la tienda.	
PI-04	Verificar que el sistema de donaciones se conecte correctamente a la base de datos de Firebase.	donate(), Firebase Realtime Database	El usuario ha seleccionado una cantidad para donar.	cantidadDonacion = 100	La donación se guarda en la base de datos de Firebase y se refleja en el perfil del usuario.	Pendiente	Esencial para el manejo de donaciones.	

<https://docs.google.com/spreadsheets/d/15qCYM8aA6cFzlTx64j7ULzb9fYrX3Ct6I9DpzMpYHLo/edit?usp=sharing>

14.2.3. Estrategia de Pruebas de Integración

1. Enfoque de las Pruebas de Integración:

- Las pruebas de integración se centrarán en verificar la correcta interacción entre los módulos y la integridad de los datos a lo largo de los flujos críticos de la aplicación, como la autenticación, gestión de perfil, procesamiento de pagos y notificaciones.
- Estas pruebas se realizarán siguiendo un enfoque **bottom-up**, donde las funciones unitarias ya probadas se combinarán progresivamente para asegurar que los módulos interactúen correctamente entre sí.

2. Herramientas y Frameworks Utilizados:

- **Jest:** Se utilizará este framework de pruebas para llevar a cabo las pruebas de integración en las funcionalidades de la aplicación, ya que permite la ejecución de pruebas tanto unitarias como de integración.
- **Supertest:** Para probar los endpoints de la API de la aplicación y asegurar que las solicitudes de los módulos se gestionen correctamente.
- **Postman/Newman:** Para probar manualmente las integraciones entre módulos a nivel de API, asegurando que los datos se transmiten correctamente entre los servicios.
- **Firebase Emulator Suite:** Para simular el entorno de Firebase y probar integraciones con la base de datos en un entorno seguro, antes de implementarlas en producción.

3. Automatización de Pruebas:

- Las pruebas de integración se integrarán en el pipeline de **CI/CD** (Integración Continua/Despliegue Continuo) mediante **GitHub Actions** o **Jenkins**.
- Cada vez que se realice un **commit** o una fusión en el repositorio, se ejecutarán automáticamente las pruebas de integración para validar que las nuevas características no rompen la interacción entre los módulos.
- Se configurarán **hooks de pre-commit** para ejecutar algunas pruebas clave a nivel local antes de enviar los cambios al repositorio.

4. Cobertura de Código:

- El objetivo es alcanzar una **cobertura de código** del 90% en las pruebas de integración, especialmente en las funcionalidades críticas como el proceso de registro, autenticación, ciclo menstrual, y procesamiento de pagos.
- Las herramientas como **Codecov** o **SonarQube** se utilizarán para generar informes de cobertura y asegurar que las integraciones están suficientemente cubiertas.
- Se priorizaron los flujos más utilizados por los usuarios, como la gestión del perfil y la tienda, asegurando que las interacciones entre el frontend y backend sean eficientes y robustas.

5. Componentes a Probar:

- **Módulo de Autenticación:** Verificar la correcta autenticación del usuario y su interacción con el perfil.
- **Módulo de Perfil de Usuario:** Verificar que los cambios en el perfil del usuario se reflejan tras iniciar sesión.
- **Módulo de Pagos:** Validar que el carrito de compras, la gestión de productos y los pagos interactúan de manera correcta.
- **Módulo de Notificaciones:** Asegurar que las notificaciones basadas en el calendario menstrual se envían correctamente.

6. Escenarios Críticos de Integración:

- **Autenticación y Gestión de Perfil:** Verificar que los datos del usuario se actualizan correctamente tras modificar su perfil.
- **Integración entre la Tienda y el Módulo de Pagos:** Asegurar que el sistema pueda procesar los pagos y actualizar el historial de compras.
- **Calendario Menstrual y Notificaciones:** Validar que el sistema envía las notificaciones de ciclo según la información registrada.
- **Contenido Dinámico en la Sección “Hablemos de...”:** Asegurar que los artículos agregados por el administrador se reflejan en la interfaz de los usuarios correctamente.

7. Planificación y Ejecución:

- Las pruebas de integración se ejecutarán en **ambientes de desarrollo y preproducción** antes de cada despliegue de producción.
- Se programarán **pruebas de regresión** automáticas cada vez que se agreguen nuevas características que impacten múltiples módulos.
- Los **testers** y **desarrolladores** trabajarán juntos para asegurar que los módulos recién integrados no rompan la funcionalidad existente.

14.3. Pruebas Funcionales y No Funcionales

14.3.1 Escenarios de Prueba

Escenario 1: Registro de Nuevo Usuario

- **Título del Escenario:** Registro de Nuevo Usuario.
- **Descripción:** Verificar que un nuevo usuario pueda registrarse exitosamente utilizando un correo electrónico válido y una contraseña segura.
- **Precondiciones:** El usuario no está registrado previamente.
- **Pasos Generales:**
 1. Abrir la aplicación.
 2. Navegar a la pantalla de registro.
 3. Ingresar un correo electrónico válido.
 4. Ingresar una contraseña segura.
 5. Confirmar la contraseña.
 6. Hacer clic en el botón “Registrar”.

- **Resultado Esperado:** El usuario recibe una confirmación de registro exitoso y puede iniciar sesión con sus credenciales.

Escenario 2: Inicio de Sesión con Credenciales Correctas

- **Título del Escenario:** Validar el Inicio de Sesión con Credenciales Correctas.
- **Descripción:** Verificar que un usuario registrado pueda iniciar sesión correctamente con sus credenciales.
- **Precondiciones:** El usuario ya está registrado.
- **Pasos Generales:**

1. Abrir la aplicación.
 2. Ingresar el correo electrónico registrado.
 3. Ingresar la contraseña correcta.
 4. Pulsar “Iniciar Sesión”.
- **Resultado Esperado:** El usuario accede al sistema exitosamente.

Escenario 3: Cambio de Contraseña

- **Título del Escenario:** Cambio de contraseña desde el perfil del usuario.
 - **Descripción:** Verificar que un usuario autenticado pueda cambiar su contraseña desde el perfil.
 - **Precondiciones:** El usuario está autenticado en la aplicación.
 - **Pasos Generales:**
 1. Navegar a la pantalla de perfil.
 2. Seleccionar la opción “Cambiar contraseña”.
 3. Ingresar una nueva contraseña válida.
 4. Confirmar la nueva contraseña.
 5. Guardar los cambios.
- **Resultado Esperado:** La contraseña es cambiada correctamente y el usuario puede iniciar sesión con la nueva contraseña.

Escenario 4: Consultar Artículos en la Sección “Hablemos de...”

- **Título del Escenario:** Consulta de Artículos en la Sección “Hablemos de...”.
- **Descripción:** Verificar que el usuario pueda consultar los artículos disponibles en la sección “Hablemos de...”.

- **Precondiciones:** El usuario está registrado y ha iniciado sesión.

- **Pasos Generales:**

1. Navegar a la sección “Hablemos de...”.

2. Consultar la lista de artículos.

- **Resultado esperado:** Los artículos se muestran correctamente.

Escenario 5: Cálculo del Ciclo Menstrual

- **Título del Escenario:** Verificación del cálculo del ciclo menstrual.

- **Descripción:** Verificar que el usuario pueda calcular su ciclo menstrual correctamente.

- **Precondiciones:** El usuario ha registrado al menos un ciclo anterior.

- **Pasos Generales:**

1. Abrir el calendario menstrual.

2. Revisar las predicciones del próximo ciclo.

- **Resultado Esperado:** El ciclo menstrual se predice correctamente basado en los datos anteriores.

Escenario 6: Agregar Producto al Carrito de Compras

- **Título del Escenario:** Verificar que el usuario pueda agregar productos al carrito.

- **Descripción:** Verificar que el usuario pueda agregar productos al carrito de compras en la tienda.

- **Precondiciones:** El usuario ha iniciado sesión y está en la tienda.

- **Pasos Generales:**

1. Seleccionar un producto.

2. Pulsar “Aregar al Carrito”.

- **Resultado Esperado:** El producto se agrega correctamente al carrito de compras del usuario.

Escenario 7: Hacer Donaciones al Completar una Compra

- **Título del Escenario:** Verificar que el usuario pueda hacer donaciones al completar una compra.
- **Descripción:** Verificar que el usuario pueda hacer una donación adicional al completar una compra.
- **Precondiciones:** El usuario ha añadido productos al carrito.
- **Pasos Generales:**
 1. Proceder al pago.
 2. Elegir la opción de donación.
 3. Completar el pago.

- **Resultado Esperado:** La donación es procesada correctamente junto con la compra.

Escenario 8: Visualización del Historial de Compras

- **Título del Escenario:** Verificar que el Usuario pueda ver su Historial de Compras.
 - **Descripción:** Verificar que el usuario pueda visualizar correctamente el historial de compras realizadas.
 - **Precondiciones:** El usuario ha realizado al menos una compra.
 - **Pasos Generales:**
 1. Navegar a la sección de perfil.
 2. Seleccionar “Historial de Compras”.
- **Resultado Esperado:** El historial de compras se muestra correctamente con la información de las transacciones anteriores.

Escenario 9: Verificar el Ciclo Menstrual Registrado

- **Título del Escenario:** Visualización del ciclo menstrual registrado.
- **Descripción:** Verificar que el usuario pueda ver su ciclo menstrual registrado anteriormente.
- **Precondiciones:** El usuario ha registrado un ciclo anterior.
- **Pasos Generales:**

1. Navegar a la sección de “Ciclo Menstrual”.
2. Revisar los ciclos anteriores.

- **Resultado Esperado:** El ciclo registrado se muestra correctamente.

Escenario 10: Agregar un Nuevo Producto a la Tienda (Administrador)

- **Título del Escenario:** Verificación de agregado de nuevo producto por el administrador.
- **Descripción:** Verificar que el administrador pueda agregar nuevos productos a la tienda con los detalles correctos.
- **Precondiciones:** El administrador ha iniciado sesión.
- **Pasos Generales:**
 1. Acceder al panel de administración.
 2. Agregar un nuevo producto con su nombre, precio, descripción y stock.
 3. Guardar el nuevo producto.
- **Resultado Esperado:** El producto es agregado correctamente a la tienda.

14.3.2. Casos de Prueba Funcionales y No Funcionales

Plantilla de Caso de Prueba:

ID del Caso de Prueba	Descripción	Precondiciones	Pasos	Datos de Entrada	Resultado Esperado	Estado	Comentarios
PT-01	Verificar el registro exitoso de un nuevo usuario con correo electrónico válido.	El usuario no está registrado.	1. Abrir la app. 2. Navegar a la pantalla de registro. 3. Ingresar datos de contraseña. 4. Confirmar el registro.	Correo válido, Contraseña segura	Registro exitoso y confirmación de usuario.	En progreso	Funcionalidad Crítica
PT-02	Validar inicio de sesión con credenciales correcta en la aplicación.	El usuario ya está registrado.	1. Ingresar credenciales. 2. Pulsar "Iniciar Sesión". 3. Ir al perfil.	Credenciales del usuario	Acceso exitoso al sistema	En progreso	Manejó de intentos fallidos perdiente.
PT-03	Verificar el cambio de contraseña desde el perfil del usuario.	El usuario está autenticado.	1. Seleccionar "Cambiar contraseña". 2. Ingresar nueva contraseña. 3. Guardar cambios.	Nueva contraseña válida	Cambio exitoso de contraseña.	No iniciado	Necesaria para seguridad del usuario.
PT-04	Validar que el usuario pueda consultar los artículos en la sección "Hablemos de...".	El usuario está registrado y ha iniciado sesión.	1. Navegar a la sección. 2. Revisar el listado de artículos.	Ninguna	Los artículos se muestran correctamente.	En progreso	
PT-05	Verificar que el usuario pueda calcular su ciclo menstrual correctamente.	El usuario ha registrado al menos un ciclo anterior.	1. Abrir el calendario menstrual. 2. Revisar el listado de ciclos.	Fecha del último ciclo	Predicción correcta del próximo ciclo.	No iniciado	Necesaria para la funcionalidad de la app.
PT-06	Verificar que el usuario puede agregar productos al carrito de compras.	El usuario ha iniciado sesión y está en la tienda.	1. Seleccionar un producto. 2. Pulsar "Añadir al carrito". 3. Proceder al pago.	Id del producto Cantidad de donación	Producto agregado al carrito. Donación procesada correctamente.	En progreso	Necesario para la funcionalidad de compras.
PT-07	Validar que el usuario pueda hacer donaciones adicionales al completar una compra.	El usuario ha añadido productos al carrito.	2. Elegir la cantidad de donación. 3. Completar el pago.			En progreso	Sin comentarios

<https://docs.google.com/spreadsheets/d/1l8El9KfpJuzs-oG7XydvU7KU0CNndadq2xBZIT67xkQ/edit?usp=sharing>

14.3.3. Estrategia de Pruebas de Integración

Para asegurarnos de que la aplicación funcione bien tanto en sus características principales como en otros detalles, hemos diseñado una estrategia de pruebas de integración que organiza los escenarios y casos de prueba de manera eficiente:

1. **Agrupación por Funcionalidad:** Hemos agrupado los escenarios y casos de prueba según las funcionalidades de la aplicación para mantener un enfoque ordenado y asegurarnos de que cada módulo se pruebe de manera independiente antes de realizar pruebas más complejas. Estas funcionalidades incluyen:
 - **Registro de Usuarios:** Pruebas para confirmar que los nuevos usuarios pueden registrarse sin problemas.
 - **Inicio de Sesión:** Pruebas para verificar que el inicio de sesión, la autenticación y la recuperación de contraseñas funcionen correctamente.
 - **Compra de Productos:** Pruebas que verifican todo el proceso de compra, desde la selección de productos hasta el pago.
 - **Navegación y Blog:** Pruebas para evaluar la facilidad de uso y la navegación entre las distintas secciones de la app, como el blog y el calendario menstrual.
 - **Administrador:** Pruebas de conexión en la base de datos para actualizar la página web dedicada a los administradores de la aplicación(tienda online).

2. **Prioridad de Pruebas:** Cada caso de prueba tiene una prioridad asignada según su importancia:

- **Alta:** Pruebas de funcionalidades críticas, como registro, inicio de sesión y pagos. Estas se realizan primero para asegurarnos de que todo lo esencial funcione bien.
- **Media:** Pruebas para funcionalidades importantes que, si fallan, no impedirían el uso general de la app, como el blog o la gestión de pedidos.
- **Baja:** Pruebas para características menos críticas, como la navegación general o elementos visuales que no afecten el funcionamiento de la aplicación.

Con este enfoque, podemos llevar a cabo las pruebas de manera organizada y eficiente asegurándonos de que primero funcionen los módulos más importantes antes de pasar a los detalles menos críticos.

15. Gestión de Cambios

Gestión de Cambios en los Requisitos de Prueba

Gestionar los cambios en los requisitos de prueba es clave para poder adaptarnos a nuevas necesidades sin perder la calidad del proyecto. Para hacerlo, seguimos estos pasos:

1. **Identificación del Cambio:** Los cambios pueden surgir porque alguien (por ejemplo, un cliente o un miembro del equipo) solicita algo diferente o porque descubrimos algo nuevo durante el desarrollo. Todos los cambios se registran con detalle, indicando por qué es necesario y qué impacto se espera.
2. **Evaluación del Impacto:** Cada cambio se analiza para ver cómo afectará tanto al proceso de pruebas como al desarrollo en general. Esto incluye revisar qué tan complejas serán las pruebas necesarias y los recursos que se requerirán para implementarlo.
3. **Aprobación del Cambio:** Despues de evaluarlo, el cambio se presenta a los responsables del proyecto (como los profesores o el equipo) para su aprobación. Solo si el cambio es aprobado se procede a implementarlo.

4. **Documentación:** Cada cambio aprobado se documenta de manera formal. Esta documentación incluye la descripción del cambio, su impacto, la fecha, y quién aprobó el cambio, para poder llevar un seguimiento claro del proyecto.

5. **Actualización de las Pruebas:** Una vez que el cambio está aprobado y documentado, actualizamos los planes de prueba y los casos de prueba para reflejar los nuevos requisitos.

De esta forma, podemos asegurarnos de que cualquier cambio se gestione de manera controlada, sin poner en riesgo el progreso del proyecto.

16. Aprobación del Plan de Pruebas

16.1. Autoridades de Aprobación

Las personas encargadas de aprobar las pruebas y producto somos durante la etapa de desarrollo:

- Estefanía Rico.
- Alexei Martínez.
- Lorena Abigaíl.
- Fernanda Ponce.
- Sandra Herrera.

Y durante la etapa de exposición:

- Encargada de Zazil.
- Roman Martínez.
- Enrique González.
- Marco González.

Aportaciones individuales

Integrante	Aportación Individual	Porcentaje de participación
Estefanía Rico.	Diagramas de bases de datos, base de datos desarrollo de back y front end	20%
Alexei Martínez.	Desarrollo de la aplicación, diseño del prototipo de la app y página web.	20%
Lorena Abigaíl.	Desarrollo de la aplicación, diseño del prototipo de la app y de la página web.	20%
Fernanda Ponce.	Desarrollo de página web, base de datos, seguridad para el inicio de sesión del admin.	20%
Sandra Herrera.	Desarrollo de la página web, implementación de la conexión entre servidor, página y app.	20%

Link al Tiktok:

<https://drive.google.com/file/d/1WtKLaWCtlvFVqIK71J2bKOgQJsqOACV2/view?usp=sharing>

Link al video del Funcionamiento:

<https://drive.google.com/file/d/1ivZBJU6BhquxRGZO2DlGP8Va0qHGiI2i/view?usp=sharing>

Referencias

Ley Federal de Protección de Datos Personales en Posesión de Particulares – Biblioteca Digital. (n.d.).

<https://idaip.org.mx/bibliotecadigital/product/ley-federal-de-proteccion-de-datos-personales-en-posesion-de-particulares/#:~:text=Posesi%C3%B3n%20de%20Particulares-,Ley%20Federal%20de%20Protecci%C3%B3n%20de%20Datos%20Personales%20en%20Posesi%C3%B3n%20de,autodeterminaci%C3%B3n%20informativa%20de%20las%20personas.>

Request a personalized demo of Secureframe. (n.d.). Secureframe.

https://secureframe.com/request-demo?qgad=693467736470&qgterm=iso27001%20certification&gad_source=1&gbraid=0AAAAACdQY7iJJB0sk6cXYYkB_eu-GLfbG&gclid=CjwKCAjw9eO3BhBNEiwAoc0-jekk6Wbsvl3SlrfwSXNBeuH79OcLK-J2Fokxj3mIi2hdLjCpRXzHBoC4BgQAvD_BwE

General Data Protection Regulation (GDPR) – legal text. (2024, 22 abril). General Data Protection Regulation (GDPR). <https://gdpr-info.eu/>

¿Qué es un certificado SSL? - Explicación del certificado SSL/TLS - AWS. (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/es/what-is/ssl-certificate/>

¿Qué son SSL, TLS y HTTPS? | DigiCert. (s. f.).

<https://www.digicert.com/es/what-is-ssl-tls-and-https>

REST API Tutorial. (2024). *What is an API?* Recuperado de <https://restfulapi.net/>

JetBrains. (2024). *Kotlin Documentation*. Recuperado de <https://kotlinlang.org/docs/>

PayPal. (2024). *Developer Documentation: PayPal API*. Recuperado de <https://developer.paypal.com/docs/>