

Proyecto:



# ANiMATi

## LIBRERÍA

App Mobile y Página Web

---

Cohorte 2023

**Integrantes:**

- Burrut, Yasmin
- De Nieto, Gabriel
- Ferace, Luciana
- Galarza, Brian
- Palacio, Lautaro
- Palomeque, Dalila
- Rochetti, Uriel
- Samsam, Emir
- Velazquez, Noemí

## Ficha del documento

Fecha	Revisión	Autor	Referencia de Cambio
15/04/2025	1.0.0	Gabriel De Nieto	Se comienza la construcción del documento.
18/04/2025	1.0.1	Dalila Palomeque	Agregado de la Historias de Usuario y tareas.
22/04/2025	1.0.2	Luciana Ferace	Correcciones en el documento.
23/04/2025	1.0.3	Lautaro Palacio	Correcciones en el documento.
25/04/2025	1.0.4	Burrut Yasmin Velazquez Noemí Rochetti Uriel	Correcciones en la redacción de las TK y US.
12/06/2025	1.0.5	Gabriel De Nieto	Agregado de US y Tk, correcciones.

# **Índice**

## **Contenido**

Introducción.....	1
Propósito.....	1
Alcance.....	1
Personal involucrado.....	2
Definiciones, acrónimos y abreviaturas.....	4
Referencias.....	4
Resumen.....	5
Descripción general.....	5
Perspectiva del producto.....	5
Características de los usuarios.....	6
Restricciones.....	6
Requisitos específicos.....	6
Product Backlog:.....	7
US:.....	7
TK:.....	9
Sprints.....	14
Sprint 0:.....	14
Sprint 1:.....	17
Sprint 2:.....	21
Anexo I.....	24

## Introducción

Somos un grupo de estudiantes del ISPC, cohorte 2023, y estamos desarrollando el proyecto **Tienda de Compra “Animati”**. Tiene como objetivo ofrecer una experiencia de compra sencilla y accesible para nuestros usuarios, permitiéndoles adquirir productos personalizados de alta calidad.

## Propósito

El propósito de “**Animati**” es brindar a los usuarios una plataforma digital eficiente para la compra de productos personalizados. Queremos mejorar la experiencia de compra online ofreciendo una variedad de artículos creativos y únicos, que satisfagan las necesidades y gustos específicos de nuestros clientes.

## Alcance

El alcance del proyecto incluye el desarrollo de una aplicación móvil y una página web que permita a los usuarios:

- Registrarse e iniciar sesión.
- Navegar y buscar productos personalizados.
- Añadir productos a un carrito de compras.
- Realizar el proceso de pago de manera segura.
- Acceder a su historial de compras.
- Gestionar su perfil de usuario.

La aplicación estará disponible para dispositivos Android y contará con una interfaz intuitiva, diseñada para ofrecer una experiencia de usuario óptima.

## Personal involucrado

<b>Nombre</b>	Luciana Ferace
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Facilitadora Team Develop y Team Testing
<b>Información de contacto</b>	feraceluciana@gmail.com

<b>Nombre</b>	Yasmin Burrut
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Team Develop
<b>Información de contacto</b>	nawsper@gmail.com

<b>Nombre</b>	Emir Samsam
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Team Develop
<b>Información de contacto</b>	ea.samsam@gmail.com

<b>Nombre</b>	Dalila Palomeque
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Team Develop y Team Testing
<b>Información de contacto</b>	dalilapalomeque@mi.unc.edu.ar

<b>Nombre</b>	Lautaro Palacio
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Team Develop
<b>Información de contacto</b>	pala797@gmail.com

<b>Nombre</b>	Noemí Andrea Velazquez
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Team Develop
<b>Información de contacto</b>	velazqueznoemiandrea@gmail.com

<b>Nombre</b>	Gabriel De Nieto
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Scrum Master, Team Develop y Team Testing
<b>Información de contacto</b>	gabrielnietodejesus@gmail.com

<b>Nombre</b>	Uriel Rochetti
<b>Rol</b>	Developer
<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Team Develop
<b>Información de contacto</b>	urielrochetti@gmail.com

<b>Nombre</b>	Brian Galarza
<b>Rol</b>	Developer

<b>Categoría Profesional</b>	Estudiante
<b>Responsabilidad</b>	Team Develop
<b>Información de contacto</b>	

## Definiciones, acrónimos y abreviaturas

Nombre	Descripción
<b>Usuario</b>	Persona que usará el sistema para gestionar procesos
<b>SIS-I</b>	Sistema de Información Web para la Gestión de Procesos Administrativos y Académicos
<b>ERS</b>	Especificación de Requisitos Software
<b>RF</b>	Requerimiento Funcional
<b>RNF</b>	Requerimiento No Funcional
<b>FTP</b>	Protocolo de Transferencia de Archivos
<b>Moodle</b>	Aula Virtual
<b>US</b>	Historias de usuario
<b>TK</b>	Tareas a realizar

## Referencias

Título del Documento	Referencia
Standard IEEE 830 - 1998	IEEE

## **Resumen**

El proyecto **Tienda de Compra “Animati”** tiene como objetivo crear plataformas que faciliten la compra de productos personalizados, brindando una experiencia de usuario intuitiva y segura. Con un enfoque en la personalización y la facilidad de uso, Animati busca satisfacer las demandas del mercado de e-commerce ofreciendo una amplia gama de productos únicos. El proyecto está alineado con las tendencias actuales del comercio electrónico y se desarrollará cumpliendo con los requisitos establecidos y dentro del cronograma fijado. El equipo de desarrollo se compromete a entregar una solución completa y funcional que cumpla con las expectativas de los usuarios y las necesidades del mercado.

## **Descripción general**

“Animati” es una tienda de compras móvil que ofrece una amplia gama de productos personalizados, incluyendo separadores, sets de stickers, cubecrafts, cajas personalizadas, colgantes, blocks de notas, llaveros, y más. Los usuarios pueden explorar el catálogo, aplicar filtros de búsqueda, y realizar compras directamente desde la aplicación.

## **Perspectiva del producto**

“Animati” se enmarca dentro de la tendencia creciente del comercio electrónico y la personalización de productos. Su diseño y funcionalidad están orientados a facilitar la compra de artículos únicos y personalizados, brindando a los usuarios una plataforma moderna y confiable para realizar sus adquisiciones.

## Características de los usuarios

<b>Tipo de usuario</b>	<b>Administrador</b>
<b>Formación</b>	Manejo de herramientas informáticas
<b>Actividades</b>	Control y manejo del sistema en general

<b>Tipo de usuario</b>	<b>Visitante</b>
<b>Formación</b>	Manejo de entornos App
<b>Actividades</b>	Observa e indaga información sobre productos y servicios disponibles y compras que puede realizar en la tienda virtual de papelería.

## Restricciones

- La aplicación debe ser compatible con la versión estable actual de Android.
- La seguridad de los datos de los usuarios es prioritaria, por lo que se implementarán medidas de encriptación para proteger la información personal y las transacciones.
- El proyecto deberá completarse dentro del cronograma establecido por el ISPC.

## Requisitos específicos

- **Registro e inicio de sesión:** El usuario debe poder registrarse con un correo electrónico y contraseña, y poder iniciar sesión de manera segura.
- **Navegación de productos:** La aplicación debe permitir la visualización y búsqueda de productos con opciones de filtrado.
- **Carrito de compras:** El usuario debe poder añadir productos a su carrito, revisar los artículos seleccionados y proceder al pago.

- **Proceso de pago:** Se debe integrar una pasarela de pago segura para procesar las compras.
- **Historial de compras:** Los usuarios deben poder acceder a un historial de todas sus compras realizadas.
- **Gestión de perfil:** Los usuarios deben poder editar su perfil para mantener su información actualizada.

## **Product Backlog**

Se presenta un listado de la totalidad de actividades realizables por el usuario y tareas vinculadas al proceso de desarrollo.

### **User Stories:**

**#US1 / Crear el mapa del sitio:** Como desarrollador quiero que la aplicación web tenga un mapa del sitio creado y documentado para que lo tengamos de referencia a la hora de trabajar en la aplicación.

**#US2 / Revisar y actualizar IEEE830:** Como miembro del equipo de desarrollo quiero que se mantenga actualizado el documento IEEE830 para cumplir con los requisitos del proyecto y mantener la documentación adecuada

**#US3 / Restaurar contraseña en Web:** Como usuario olvidé mi contraseña quiero poder solicitar un link de recuperación para establecer una nueva contraseña y poder acceder nuevamente a mi cuenta.

**#US4 / Permitir edición de perfil de usuario en Mobile:** Como usuario registrado quiero actualizar mis datos personales desde mi perfil para mantener mi información actualizada.

**#US5 / Hacer deploy del backend en Railway:** Como desarrollador quiero desplegar el backend en Railway para que el equipo pueda trabajar sobre un backend disponible.

**#US6 / Implementar pantalla de pago y vinculación con MercadoPago:** Como usuario de la aplicación quiero realizar pagos a través de una pantalla integrada con Mercado Pago para completar mis compras de forma segura.

**#US7 / Implementar botón de arrepentimiento en Mobile:** Como usuario, quiero poder acceder a un botón de arrepentimiento luego de realizar una compra para solicitar la cancelación y devolución, cumpliendo con las regulaciones vigentes.

**#US8 / Modificar la conexión en mobile para que se conecte a la Api de Django:** Como usuario, quiero poder acceder a mi cuenta y realizar compras y/o visualizar productos, ya sea desde la app móvil, así como su versión web.

**#US9 / Confirmar visualmente al usuario la solicitud de arrepentimiento enviada:** Como usuario, quiero recibir una confirmación visual clara luego de enviar una solicitud de arrepentimiento, para saber que el proceso se realizó correctamente.

**#US10 / Visualización de historial de compras:** Como usuario quiero ser capaz de ver el historial de compras para tener un mejor control de las mismas y ver transacciones anteriores.

**#US11 / Adaptar el front de Web:** Como usuario quiero que tanto la aplicación como la web sean iguales para que parezcan un mismo producto y no sean confusos a la hora de usarlos.

**#US12 / Promocionar/Informar app mobile desde el sitio web:** Como desarrollador quiero que la aplicación web notifique al usuario la existencia de la app para sumar descargas y potenciar dicho canal de venta.

**#US13 / Cambiar contraseña en Web:** Como usuario quiero poder modificar mi contraseña personal desde la web para evitar problemas de seguridad y tener mayor control de mi cuenta.

**#US14 / Página de Inicio:** Como usuario quiero ingreso rápido intuitivo a la tienda para efectivizar la compra rápida.

**#US15 / Mobile front:** Como quiero que se readapten las llamadas a la api desde el front que la app vuelva a funcionar correctamente luego de las modificaciones realizadas en el sprint 1 de practicas profesionalizantes.

**#US16 / Mobile front:** Como usuario, quiero que el sitio se mantenga apegada a los estandares de accesibilidad establecidos por WCAG 2.2 AA asegurando una experiencia inclusiva para todos los usuarios.

**#US17 / Mobile cierre de sesion:** Como usuario, quiero poder acceder a un botón de cierre de sesión para impedir el ingreso a mi cuenta mientras yo no utilice la aplicación o desee ingresar otra persona desde mi dispositivo.

**#US18 / Mobile admin:** Como administrador, quiero acceder a un panel de gestión de productos para poder crear, editar y eliminar productos del sistema, de modo que pueda mantener actualizado el catálogo ofrecido a los usuarios.

**#US19 / Acceso :** Como usuario, queiro poder ingreasar a la web desde la app con un clik .

## Tasks:

### #TK 01

- TK01.1: Repasar los circuitos de la página web
- TK01.2: Realizar el mapa del sitio
- TK01.3: Subir el mapa del sitio al repositorio

### #TK 02

- TK02.1: Actualizar el documento

### #TK 03

- TK03.1: Crear formulario para solicitar restauración de contraseña
- TK03.2: Implementar lógica de envío de correo para restauración
- TK03.3: Crear pantalla para ingresar nueva contraseña

### #TK 04

- TK04.1: Crear campos editables en pantalla de perfil
- TK04.2: Validar formato de los datos ingresados
- TK04.3: Implementar lógica de guardado de cambios
- TK04.4: Confirmar cambios exitosos mediante mensaje

### #TK 05

- TK05.1: Realizar deploy del backend en Railway

### #TK 06

- TK06.1: Crear pantalla de pago en Mobile y Web
- TK06.2: Implementar integración con API de MercadoPago
- TK06.3: Confirmar pago exitoso y guardar resultado en la base de datos
- TK06.4: Mostrar mensajes de éxito o error al usuario tras el pago

### #TK 07

- TK07.1: Crear botón de arrepentimiento en la pantalla de historial de compras
- TK07.2: Implementar lógica de envío de solicitud de arrepentimiento al backend
- TK07.3: Crear endpoint en el backend para recibir solicitudes de arrepentimiento.
- TK07.4: Mostrar mensaje de confirmación tras enviar la solicitud

### #TK 08

- TK08.1: Agregar permisos de conexión a internet
- TK08.2: Implementar las librerías necesarias
- TK08.3: Diseñar el modelo de datos

- TK08.4: Realizar la petición HTTP

## #TK 09

- TK09.1: Implementar mensaje de éxito tras el envío correcto de la solicitud
- TK09.2: Implementar mensaje de error en caso de fallo en la solicitud

## #TK 10

- TK10.1 : Crear interfaz de usuario para el historial
- TK10.2: Conectar el endpoint al frontend
- TK10.3: Mostrar datos en la interfaz

## #TK 11

- TK11.1: Login
- TK11.2: Registro
- TK11.3: Inicio
- TK11.4: Producto
- TK11.5: Carrito de Compras
- TK11.6: Página sobre nosotros
- TK11.7 Contacto
- TK11.8: Perfil de usuario
- TK11.9: Recuperar contraseña
- TK11.10: Cambio de contraseña

## #TK 12

- TK12.1: Crear ventana emergente de notificación en landing web
- TK12.2: Redirección a Play Store desde ventana emergente

## #TK 13

- TK13.1: Incluir botón cambio contraseña en vista perfil
- TK13.2: Generar formulario para cambio contraseña
- TK13.3: Validar formato de nueva contraseña generada
- TK13.4: Implementar mensaje de cambio contraseña exitoso

## #TK 14

- TK14.2: Incluir acceso a tienda en página de inicio
- TK14.3: Incluir lista de productos destacados en pagina de inicio
- 

## #TK 15

- TK15.1: Analizar que endpoints están incompletos.
- TK15.2: Modificar la url con la que se llaman a los endpoints.

## #TK 16

- TK16.1:Revisar accesibilidad de funcionalidades preexistentes.
- TK16.2:Revisar estándares de accesibilidad de funcionalidades nuevas.

## #TK 17

- TK17.1:Crear botón de cierre de sesión en pantalla de datos de perfil.
- TK17.2:Implementar lógica del pedido de cierre de sesión al backend.
- TK17.3:Redireccionar a la vista de “iniciar sesión” una vez se confirme el cierre.

## #TK 18

- TK18.1:Implementar el listado de productos con nombre, precio, imagen, categoria y stock.
- TK18.2:Añadir funcionalidad para crear un nuevo producto.
- TK18.3:Añadir funcionalidad para editar un producto existente.
- TK18.4:Añadir funcionalidad para eliminar un producto con confirmación previa.
- TK18.5:Mostrar mensajes de exito o error para cada acción de panel (crear, editar, eliminar).

## #TK 19

- TK19.1:Generación de botón acceso a Sitio Web.

## Sprints

<b>Nº de sprint</b>	00
<b>Sprint Backlog</b>	<ol style="list-style-type: none"><li>1. Definir los roles de los miembros del equipo Scrum (Scrum Master y Developer Team).</li><li>2. Documentar roles en la Wiki del proyecto.</li><li>3. Evaluar el contenido y distribución de la tienda para adaptarlo al nuevo e-commerce/carrito de compras.</li><li>4. Evaluar la App y sus requerimientos asociados.</li><li>5. Plantear Historias de Usuario con redacción y nomenclatura adecuadas (ej: "#US01 Como usuario quiero...").</li></ol>

	<ol style="list-style-type: none"> <li>6. Especificar los criterios de aceptación en cada Historia de Usuario.</li> <li>7. Utilizar plantilla de Historias de Usuario.</li> <li>8. Documentar el proyecto en base al estándar IEEE 830 (más requerimientos).</li> <li>9. Crear un proyecto estilo Kanban en GitHub.</li> <li>10. Definir la estructura de la Wiki en el repositorio de GitHub.</li> <li>11. Cargar nombre y apellido de los integrantes del equipo, y roles.</li> <li>12. Registrar ceremonias Scrum: Planning, Review y Retrospective (con plan de mejora).</li> <li>13. Crear y agregar Issues en GitHub como Historias de Usuario.</li> <li>14. Respetar la nomenclatura sugerida para Issues.</li> <li>15. Asignar y distribuir Issues a cada miembro del equipo.</li> <li>16. Crear Milestones en GitHub por Sprint con fechas de inicio y fin.</li> <li>17. Realizar commits y registrar historial en la Wiki.</li> <li>18. Usar tablero Kanban con las columnas: Product Backlog, To Do, In Process, Finished.</li> <li>19. Esquematizar el proyecto web (Mapa de sitio).</li> </ol>
<b>Responsabilidades</b>	<p><b>Descripción del Trabajo en Equipo:</b></p> <p>El equipo llevó a cabo las tareas de manera colaborativa y organizada. Se conformó el grupo y se definió la organización del repositorio en GitHub. Se abordaron las siguientes actividades clave:</p> <ul style="list-style-type: none"> <li>● <b>Roles y Wiki:</b> Los roles de Scrum Master y equipo de desarrollo fueron definidos y documentados en la Wiki del proyecto. Todos participaron para garantizar claridad en las responsabilidades.</li> <li>● <b>Evaluación de la tienda y la App:</b> Se analizaron los contenidos y estructuras existentes, con el objetivo de alinearlos con el nuevo enfoque de e-commerce y carrito de compras.</li> </ul>

	<ul style="list-style-type: none"> <li>• <b>Historias de Usuario:</b> Se redactaron las Historias de Usuario respetando una nomenclatura adecuada (#US) y se detallaron los criterios de aceptación, utilizando la plantilla proporcionada.</li> <li>• <b>Documentación IEEE 830:</b> El equipo colaboró en la elaboración del documento IEEE 830, asegurando la inclusión de todos los requisitos necesarios para el cliente.</li> <li>• <b>Kanban:</b> Se creó un tablero Kanban en GitHub, con las columnas necesarias para reflejar el flujo de trabajo. Todos los miembros colaboraron en su mantenimiento.</li> <li>• <b>Wiki:</b> Se estructuró la Wiki incluyendo datos de los integrantes, ceremonias Scrum y documentos clave del proyecto.</li> <li>• <b>Issues y Milestones:</b> Se crearon y asignaron Issues como Historias de Usuario. Se agregaron Milestones por Sprint, con sus fechas de inicio y cierre.</li> <li>• <b>Esquema web:</b> Se trabajó colaborativamente en la esquematización del proyecto web, incluyendo el mapa del sitio.</li> </ul>
<b>Calendario</b>	<b>Start Date:</b> 15/08/2024 <b>End Date:</b> 22/04/2025
<b>Inconvenientes:</b> Durante el Sprint 0 se estableció la dinámica del grupo y el proceso de trabajo. Se presentaron desafíos iniciales en la organización y distribución de roles, pero fueron superados con acuerdos y planificación conjunta.	

<b>Nº de sprint</b>	<b>01</b>
<b>Sprint Backlog</b>	1. Crear una SPA a partir de la maqueta (landing page, dashboard, galería, quienes somos y contacto). 2. Implementar sistema de rutas. 3. Formularios de login y registro con validaciones específicas.

	<ol style="list-style-type: none"> <li>4. Implementar data binding entre componentes .ts y .html.</li> <li>5. Crear servicios para proveer datos a los componentes.</li> <li>6. Formularios reactivos.</li> <li>7. Dashboard con carrito de compras.</li> <li>8. Conexión al backend para login/registro y datos de productos.</li> <li>9. Rutas protegidas por autenticación/autorización.</li> <li>10. Definición de API REST y endpoints.</li> <li>11. Actualizar DER, Modelo Relacional y scripts de DB.</li> <li>12. Crear mínimo un modelo de caso de uso y diagrama de clases.</li> <li>13. Crear APIs REST: CRUD productos, carrito, login y registro.</li> </ol>
<b>Responsabilidades</b>	<p><b>Descripción del Trabajo en Equipo:</b></p> <p>El equipo se centró en la implementación funcional del desarrollo web, asegurando mejoras con respecto al módulo anterior.</p> <p><b>Frontend SPA:</b></p> <p>Se generó la estructura general de la aplicación SPA en Angular.</p> <p><b>Ruteo y vistas:</b></p> <p>Se implementaron rutas para navegación entre landing page, galería, dashboard, contacto y quienes somos.</p> <p><b>Formularios y validaciones:</b></p> <p>Se desarrollaron formularios reactivos para login y registro con restricciones personalizadas y validaciones.</p> <p><b>Servicios y binding:</b></p>

	<p>Se implementó la conexión entre componentes y servicios de datos.</p> <p><b>Conexión con backend:</b></p> <p>Se realizó conexión con la base de datos para recuperación de productos y autenticación de usuarios.</p> <p><b>Protección de rutas:</b></p> <p>Se agregaron rutas protegidas según autenticación.</p> <p><b>API y backend:</b></p> <p>Se definieron los endpoints REST, actualizó el modelo de datos y se programaron las APIs CRUD, carrito, login y registro.</p> <p><b>Documentación:</b></p> <p>Se actualizó el documento IEEE830, se realizaron commits, y se documentaron las ceremonias en Wiki con herramientas como Notion o Ideaboard.</p>
<b>Calendario</b>	<p><b>Start Date:</b> 23/04/25</p> <p><b>End Date:</b> 14/05/25</p>
<b>Inconvenientes:</b>	<p>No se reportaron mayores inconvenientes técnicos. Se continúa optimizando la comunicación en el equipo.</p>

<b>Nº de sprint</b>	<b>02</b>
<b>Sprint Backlog</b>	<ol style="list-style-type: none"> <li>1. Testing: incluir al menos 3 casos de prueba documentados para la aplicación web y otros 3 para la app móvil.</li> <li>2. Seguridad: especificar en la Wiki las medidas aplicadas en la aplicación web y móvil (validaciones, rutas protegidas, autenticación, etc.).</li> <li>3. Confirmaciones en GitHub por todos los miembros.</li> <li>4. Realizar merge a la rama principal (main) del código.</li> </ol>

	<p>5. Cierre de Issues y Milestones en GitHub.</p> <p>6. Actualización del Kanban.</p> <p>7. Documentar ceremonias de Planning, Review y Retrospective con herramientas sugeridas (Ideaboard, Notion, Mural).</p> <p>8. Completar documento IEEE830.</p> <p>9. Crear archivo README.md del proyecto actualizado.</p> <p>10. Aplicar principios de accesibilidad web/móvil.</p> <p>11. Verificar funcionamiento y compilación de la app.</p> <p>12. Asegurar navegabilidad completa entre activities, botones funcionales y navegación fluida.</p> <p>13. Crear una página de contacto.</p> <p>14. Aplicar una estética coherente (estilos, colores, imágenes).</p> <p>15. Implementar login funcional conectado a backend.</p> <p>16. CRUD funcional de productos/servicios y usuarios.</p> <p>17. Gestión de persistencia de datos.</p> <p>18. Carrito de compras funcional con integración de pasarela de pagos (Mercado Pago, Stripe o PayPal en modo sandbox).</p>
<b>Responsabilidades</b>	<p><b>Descripción del Trabajo en Equipo:</b></p> <p>Durante este sprint el equipo se centró en integrar y mejorar la app móvil utilizando Android Studio con Java y mantener consistencia con el desarrollo web.</p> <p><b>Testing:</b></p> <p>Se documentaron 3 casos de prueba para la web (login, registro, compra) y 3 para la app móvil (navegabilidad, autenticación y carga de productos), todos en la Wiki.</p> <p><b>Seguridad:</b></p> <p>En la Wiki se describieron las medidas aplicadas: autenticación JWT/Auth0, rutas protegidas, validaciones en formularios y control de accesos.</p>

	<p><b>App móvil:</b></p> <p>Se logró navegabilidad fluida entre las activities, incluyendo login, galería de productos, dashboard y contacto. Se implementó el botón de regreso y navegación principal.</p> <p><b>Accesibilidad y estética:</b></p> <p>Se incorporaron mejoras de accesibilidad (textos alternativos, navegación con teclado) y estilos visuales coherentes en ambas plataformas.</p> <p><b>Login y CRUD:</b></p> <p>Login conectado con backend funcional en web y app móvil. CRUD operativo para productos y usuarios.</p> <p><b>Carrito y pagos:</b></p> <p>Se integró pasarela de pagos sandbox en la app móvil y web (Mercado Pago).</p> <p><b>Documentación y organización:</b></p> <p>Kanban, Issues, milestones, y README.md actualizados. Se realizaron commits desde cuentas individuales. Se documentaron las ceremonias y se mantuvo actualizado el documento IEEE830.</p>
<b>Calendario</b>	<b>Start Date:</b> 15/05/25 <b>End Date:</b> 04/06/25
<b>Inconvenientes:</b>	

## **ANEXO I**

- ORGANIZACION: <https://github.com/orgs/AnimatiWA/repositories>
- FIGMA:  
<https://www.figma.com/design/39HJgo2ZMFNkbYRgmyhNZj/Animati-App>
- KANBAN <https://www.github.com/orgs/AnimatiWA/projects/4>