

# **ESCUELA POLITÉCNICA NACIONAL**

## **FACULTAD DE INGENIERÍA EN SISTEMAS**

**UN GESTOR DE TAREAS BASADO EN KANBAN PARA LA  
PLANIFICACIÓN OPERATIVA: DESARROLLO DEL FRONT-END Y  
BACK-END DE UNA APLICACIÓN WEB BAJO EL ENFOQUE ÁGIL**

**DESARROLLO DEL FRONT-END**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO  
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SOFTWARE**

**DAVID SEBASTIÁN LEÓN OCHOA**

**david.leon@epn.edu.ec**

**DIRECTOR: CARLOS EFRAÍN IÑIGUEZ JARRÍN**

**carlos.iniguez@epn.edu.ec**

**DMQ, agosto 2024**

## **CERTIFICACIONES**

Yo, David Sebastián León Ochoa, declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

---

**DAVID SEBASTIÁN LEÓN OCHOA**

Certifico que el presente trabajo de integración curricular fue desarrollado por David Sebastián León Ochoa, bajo mi supervisión.

---

**CARLOS EFRAÍN IÑIGUEZ JARRÍN**  
**DIRECTOR**

## **DECLARACIÓN DE AUTORÍA**

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

DAVID SEBASTIÁN LEÓN OCHOA

CARLOS EFRAÍN IÑIGUEZ JARRÍN

## **DEDICATORIA**

A mi madre, quien ha sido mi fuente de inspiración y el pilar más fuerte en mi vida. Mamá, desde pequeño me mostraste la importancia del esfuerzo y la dedicación. Tus sacrificios, trabajando incansablemente para que yo tuviera la oportunidad de estudiar, no pasan desapercibidos. Cada logro mío es un reflejo de tu amor y tu lucha. Gracias por cada abrazo, cada palabra de aliento y por nunca dejarme caer.

A mí mismo, por las innumerables noches sin dormir, las largas horas de trabajo y por no darme por vencido, a pesar de los desafíos que surgieron en el camino. A veces, las jornadas parecían interminables, pero el deseo de superarme y alcanzar mis metas me mantuvo firme. Hoy, miro atrás y me siento orgulloso de cada pequeño paso que di. Me dedico este logro como un recordatorio de mi fuerza y mi perseverancia. Gracias, David, por nunca rendirte.

## **AGRADECIMIENTO**

En primer lugar, quiero expresar mi profunda gratitud a Dios, por darme salud, fortaleza y sabiduría para alcanzar este logro tan significativo en mi vida.

A mi tutor, Carlos Íñiguez, le extiendo mi más sincero agradecimiento por su invaluable tutela y orientación a lo largo de este proceso. Su disposición para compartir sus vastos conocimientos y su constante apoyo fueron esenciales para mi crecimiento académico y profesional. Sus consejos y críticas constructivas me impulsaron a mejorar y a dar lo mejor de mí.

A mi madre y su pareja, Robin, quiero agradecerles desde el fondo de mi corazón por su apoyo incondicional y por siempre confiar en mí. Gracias por darme la oportunidad de estudiar y por brindarme el respaldo necesario para seguir adelante, incluso en los momentos más difíciles.

A mi padre y a mi hermano, por sus sabios consejos y por animarme a no abandonar la universidad. Sus palabras de aliento me recordaron que la vida es dura, pero que con esfuerzo y dedicación se puede lograr cualquier cosa.

A mi pareja, Luz, cuyo amor y apoyo me dieron la fuerza para seguir adelante. Gracias por creer en mí incluso cuando yo no lo hacía, por tus palabras de aliento y por estar siempre ahí para levantarme.

A mis amigos, Byron, Cristian y Elian, por su amistad y por los momentos compartidos. Gracias por hacer que mi experiencia universitaria haya sido inolvidable.

No puedo pasar por alto a mis fieles compañeros de cuatro patas, mi perro, Popeye, que en paz descanse, y mi gato, Simba. Gracias por acompañarme en tantas noches de desvelo y por estar siempre a mi lado cuando más lo necesitaba. Su compañía y amor incondicional han sido un consuelo y una motivación constante.

A todos ustedes, gracias por ser parte de este viaje y por contribuir de tantas maneras al logro de esta meta. Este triunfo es tanto mío como de ustedes.

# ÍNDICE DE CONTENIDO

<b>1</b>	<b>DESCRIPCIÓN DEL COMPONENTE DESARROLLADO .....</b>	<b>1</b>
1.1	Objetivo general.....	2
1.2	Objetivos específicos .....	2
1.3	Alcance.....	2
1.4	Marco teórico.....	3
1.4.1	Planificación operativa.....	3
1.4.2	Kanban .....	5
1.4.3	DCU (Diseño Centrado en el Usuario).....	7
1.4.4	Atomic Design .....	9
1.4.5	Next.js .....	11
1.4.6	Figma .....	15
1.4.7	Scrum .....	15
<b>2</b>	<b>METODOLOGÍA.....</b>	<b>19</b>
2.1	Entender y especificar el contexto de uso .....	19
2.2	Especificar los requisitos del usuario y de la organización .....	24
2.3	Producir soluciones de diseño .....	31
2.3.1	Mapa de navegación.....	31
2.3.2	Prototipos .....	36
2.4	Evaluar diseños contra los requisitos .....	40
2.5	Implementación .....	42
<b>3</b>	<b>RESULTADOS, CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>45</b>
3.1	Resultados.....	45
3.2	Conclusiones .....	49
3.3	Recomendaciones .....	50
<b>4</b>	<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>52</b>
<b>5</b>	<b>ANEXOS .....</b>	<b>55</b>

## RESUMEN

Este trabajo de integración curricular se enfoca en el desarrollo del Front-End de un gestor de tareas basado en la metodología Kanban para la planificación operativa empresarial. El objetivo principal es crear una aplicación web intuitiva y efectiva que permita a los usuarios gestionar proyectos y tareas de manera eficiente. Para ello, se ha seguido un enfoque de Diseño Centrado en el Usuario (DCU) y la metodología de diseño modular Atomic Design (AD).

El desarrollo del proyecto se dividió en varias etapas clave: entendimiento del contexto de uso, especificación de requisitos, producción de soluciones de diseño, evaluación de diseños contra los requisitos e implementación. En la primera etapa, se realizaron conversaciones con el Product Owner (PO) para comprender las necesidades de los usuarios. Estas conversaciones permitieron definir los perfiles Persona en la segunda etapa. En la tercera etapa, se desarrollaron mapas de navegación y prototipos de alta fidelidad utilizando la herramienta Figma, basados en los principios de DCU y AD. En la cuarta etapa, estos prototipos fueron evaluados mediante pruebas de usabilidad, lo que proporcionó retroalimentación valiosa para mejorar la navegación, accesibilidad y funcionalidad de la interfaz. En la quinta etapa, la implementación del Front-End se realizó utilizando tecnologías modernas como Next.js, TypeScript y Tailwind CSS, lo que permitió crear una aplicación web robusta y escalable.

Los resultados obtenidos muestran que las mejoras implementadas a lo largo del proyecto han contribuido significativamente al desarrollo del gestor de tareas.

**PALABRAS CLAVE:** Kanban, Diseño Centrado en el Usuario, Atomic Design, Front-End, Next.js, Gestor de Tareas, Planificación Operativa.

## **ABSTRACT**

This curricular integration work focuses on the development of the Front-End of a task manager based on the Kanban methodology for business operational planning. The main goal is to create an intuitive and effective web application that allows users to manage projects and tasks efficiently. To achieve this, a User-Centered Design (UCD) approach and the modular design methodology Atomic Design (AD) were followed.

The project development was divided into several key stages: understanding the context of use, specifying requirements, producing design solutions, evaluating designs against requirements, and implementation. In the first stage, discussions were held with the Product Owner (PO) to understand the users' needs. These conversations enabled the definition of Persona profiles in the second stage. In the third stage, navigation maps and high-fidelity prototypes were developed using the Figma tool, based on UCD and AD principles. In the fourth stage, these prototypes were evaluated through usability tests, which provided valuable feedback to improve the navigation, accessibility, and functionality of the interface. In the fifth stage, the Front-End implementation was carried out using modern technologies such as Next.js, TypeScript, and Tailwind CSS, creating a robust and scalable web application.

The results obtained show that the improvements implemented throughout the project have significantly contributed to the development of the task manager.

**KEYWORDS:** Kanban, User-Centered Design, Atomic Design, Front-End, Next.js, Task Manager, Operational Planning.

# 1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

El componente que se aborda en este proyecto consiste en el desarrollo del Front-End de un gestor de tareas basado en la metodología Kanban para la planificación operativa empresarial. El Front-End de la aplicación web será la interfaz a través de la cual los usuarios interactúen con el sistema gestor de tareas.

Un gestor de tareas es una herramienta diseñada para ayudar en la organización y seguimiento de las actividades dentro de un proyecto [1]. Permite asignar tareas a los miembros del equipo, establecer prioridades y fechas de entrega, así como registrar el progreso de cada actividad [2]. La planificación operativa, por su parte, es el proceso de formular y describir planes y actividades específicas que deben llevarse a cabo en las distintas áreas de una institución, con el objetivo de alcanzar metas a corto plazo [3].

Kanban es una metodología de gestión de proyectos que se basa en el mecanismo de programación “Just-in-Time” de Toyota [4]. Esta metodología se basa en limitar el trabajo en curso, lo que permite comenzar con algo nuevo solo cuando un bloque de trabajo anterior ha sido completado o avanzado [5]. Su propósito en la gestión operativa es permitir que los equipos se concentren en entregar valor a sus clientes, minimizando el trabajo en progreso y adaptándose diariamente a nuevas prioridades y requisitos con poco costo hundido [4], [6]. Los beneficiarios son los equipos que emplean Kanban para gestionar su trabajo, ya que ayuda a visualizar el flujo de trabajo, resaltar cuellos de botella tan pronto como ocurren, prevenir el trabajo prematuro marcado como completo y mantener claras las expectativas de calidad en cada paso. Esto mejora la eficiencia, la productividad y la calidad del trabajo entregado a los clientes [4], [7].

La necesidad de crear un sistema de tareas propio, en lugar de utilizar otros sistemas como Asana, Trello o Jira, se deriva de la especificidad de los requerimientos operativos de la organización y la búsqueda de una mayor adaptabilidad. Muchos sistemas existentes, como los mencionados anteriormente, no consideran la especificidad de los requerimientos operativos de cada organización, ya que están diseñados para un uso generalizado y pueden no ajustarse perfectamente a procesos únicos y personalizados.

Un sistema propio está diseñado para alinearse de manera precisa con los procesos específicos, objetivos y cultura de la organización. Un sistema desarrollado internamente no solo permite una personalización completa, sino que también facilita ajustes continuos

y rápidos en respuesta a cualquier cambio en los procesos o necesidades de la organización.

La importancia de la interfaz de usuario radica en su capacidad para ser intuitiva y fácil de usar para evitar que el usuario tenga que pensar en cómo interactuar con la aplicación o sitio web. Una interfaz bien diseñada debe ser evidente, obvia y autoexplicativa, permitiendo a los usuarios navegar a través de ella con facilidad, lo cual mejora significativamente la experiencia general del usuario y su satisfacción al usar la aplicación o sitio web [8].

Dado su enfoque en la interfaz de usuario, este componente será desarrollado siguiendo metodologías centradas en el usuario y ágiles para lograr una interfaz intuitiva, atractiva y funcional que permita a los usuarios visualizar, crear, editar y gestionar sus tareas de manera eficiente.

## **1.1 Objetivo general**

Diseñar e implementar el Front-End de un gestor de tareas basado en Kanban siguiendo un enfoque centrado en el usuario y los principios del desarrollo ágil.

## **1.2 Objetivos específicos**

1. Analizar los requisitos de usuario
2. Diseñar prototipos de interfaz de usuario
3. Evaluar los prototipos
4. Implementar los prototipos

## **1.3 Alcance**

El desarrollo del componente tiene como alcance el diseño y desarrollo de las interfaces de usuario del sistema gestor de tareas, aplicando dos aristas principales:

- **Metodología de Diseño Centrado en el Usuario (DCU).**- Las interfaces de usuario serán desarrolladas siguiendo las fases de la metodología DCU: análisis de contexto, especificación de requisitos, generación de soluciones y evaluación. En este sentido, el proceso del DCU permitirá definir las interfaces de usuario desde sus requisitos hasta su evaluación e implementación.

- **Marco de trabajo SCRUM.-** El marco de trabajo ágil SCRUM permitirá gestionar el proyecto de manera eficaz y rápida. La aplicación de los artefactos y ceremonias de SCRUM permitirá aligerar el proceso de desarrollo de las interfaces de usuario, realizando entregas eficientes y de valor al usuario.

## **1.4 Marco teórico**

En este apartado se abordarán los conceptos fundamentales que sustentan el desarrollo con enfoque ágil del front-end de un sistema gestor de tareas basado en Kanban para la planificación operativa.

### **1.4.1 Planificación operativa**

La planificación operativa es un proceso fundamental dentro de la gestión administrativa de cualquier institución/empresa que se encarga de definir las actividades y recursos necesarios para alcanzar los objetivos a corto plazo [3]. Este proceso implica la programación detallada de tareas, la asignación de responsabilidades y la elaboración de presupuestos que aseguren la utilización adecuada de los recursos disponibles. La planificación operativa debe ser flexible y adaptativa, permitiendo ajustes en función de las circunstancias cambiantes, lo cual es crucial para mantener la relevancia y efectividad de las acciones planificadas.

#### **1.4.1.1 Gestión de tareas**

La gestión de tareas es un proceso que abarca la supervisión y control de todas las etapas de una tarea dentro de un proyecto, desde su inicio hasta su finalización, adaptándose a cambios en tiempo real para asegurar el éxito en términos de presupuesto, tiempo, alcance y recursos asignados [8]. La gestión efectiva de tareas requiere la habilidad de anticipar y mitigar problemas potenciales, lo que contribuye a mantener el proyecto en curso y alineado con sus objetivos estratégicos. La gestión de tareas influye directamente en la planificación operativa, ya que permite identificar y asignar los recursos necesarios, establecer prioridades claras y ajustar el cronograma de actividades en función del progreso y los obstáculos encontrados.

#### **1.4.1.2 Gestor de tareas**

Un gestor de tareas es una herramienta esencial para la organización y seguimiento de proyectos, permitiendo asignar tareas, establecer plazos y monitorear, optimizando la

carga de trabajo de cada miembro del equipo, mejorando la productividad y reduciendo errores como la duplicación de esfuerzos, el incumplimiento de plazos y la falta de seguimiento adecuado [2]. Contar con un gestor de tareas adecuado es fundamental para el éxito de cualquier proyecto, asegurando que todas las actividades se realicen de manera organizada y puntual.

Existen diversas herramientas implementadas para facilitar la administración de tareas. Estos gestores de tareas se adaptan a diferentes necesidades y estilos de trabajo, ofreciendo funcionalidades específicas y seguimiento del progreso. A continuación, se presentan algunos de los gestores de tareas más populares:

- **Asana:** herramienta de gestión de proyectos y tareas que ofrece una variedad de vistas, incluyendo listas, tableros Kanban, calendarios y cronogramas [9]. Es útil para equipos que necesitan colaborar en proyectos y realizar un seguimiento detallado del progreso de las tareas.
- **Trello:** herramienta de gestión de proyectos basada en tableros Kanban [10]. Permite la supervisión de tareas mediante la adición de archivos, listas de verificación (checklists) y automatizaciones personalizables según las necesidades específicas del equipo [10]. Trello es conocido por su interfaz intuitiva y su flexibilidad para adaptarse a diferentes tipos de proyectos.
- **Jira:** herramienta de gestión de proyectos desarrollada por Atlassian, especialmente popular entre equipos de desarrollo de software. Permite a los equipos planificar, supervisar, publicar y ofrecer soporte de software de manera segura [11]. Compatible con metodologías ágiles como Scrum y Kanban, Jira permite gestionar proyectos desde la planificación hasta la ejecución, facilitando la coordinación y comunicación entre los equipos [11].

Aunque las herramientas mencionadas son efectivas y ampliamente utilizadas, presentan ciertas limitaciones que las hacen menos adecuadas para cumplir con los objetivos específicos de este proyecto:

- **Limitada personalización:** Trello, Asana y Jira están diseñados para un uso generalizado y, aunque permiten ciertas personalizaciones, no siempre se ajustan a los requisitos específicos y únicos de cada organización. En este proyecto, necesitamos una herramienta que se ajuste a los procesos operativos y culturales

específicos de la organización. Por ejemplo, nuestro sistema incluye diferentes perfiles: Gestor Empresarial (GE), Gestor de Área (GA), Líder de Área (LA) y Técnico (TEC). El GE puede crear proyectos y dividirlos en tareas, asignando estas tareas a los GA. El GA puede ver las tareas que le han sido asignadas y subdividir las en tareas más específicas, asignándolas a su vez a los LA. Similar al GA, el LA puede asignar tareas más específicas a los TEC. Por último, el TEC es el único que puede mover las tareas de una columna a otra, siendo el detonante para que todas las tareas superiores se actualicen automáticamente. Esta estructura jerárquica y funcional de perfiles y responsabilidades es única y específica a las necesidades operativas y culturales de nuestra organización, y no se ajusta completamente a las capacidades de personalización ofrecidas por Trello, Asana o Jira.

- **Limitaciones en la adaptabilidad a procesos únicos:** Trello y Asana están diseñados para ofrecer soluciones estándar que pueden no considerar la complejidad o especificidad de ciertos flujos de trabajo. Este proyecto requiere una herramienta que puede ser ajustada de manera continua para adaptarse a cambios en los procesos y necesidades organizacionales. Por ejemplo, como se explicó en el párrafo anterior, el flujo de trabajo involucra varios niveles de gestión y seguimiento de tareas. Este flujo de trabajo requiere una flexibilidad que Trello y Asana ofrecen de manera limitada. La solución propuesta en este documento permite ajustes continuos en las asignaciones y visualizaciones de tareas para adaptarse a cambios dinámicos en los procesos organizacionales, lo cual es fundamental para mantener la eficiencia operativa.
- **Costos y dependencia de terceros:** Utilizar herramientas como Jira implica una dependencia de servicios externos que pueden generar costos significativos a largo plazo, además de limitaciones en el control total sobre el sistema. En cambio, desarrollar una solución propia garantiza independencia y control total sobre el sistema y sus modificaciones futuras.

#### 1.4.2 Kanban

Kanban es una metodología de gestión del flujo de trabajo ideada por Taiichi Ohno en la década de 1940 en Japón, se desarrolló para mejorar el proceso de producción de Toyota incorporando elementos de la producción ajustada [12]. La producción ajustada, también

conocida como lean manufacturing, es una metodología de gestión enfocada en maximizar el valor del proceso productivo mediante la eliminación de desperdicios, la mejora continua y la optimización de recursos, con el objetivo de entregar productos de alta calidad [13].

El enfoque de Kanban consiste en limitar el trabajo en progreso (WIP, *work in progress*, en inglés), lo que facilita la identificación y resolución de cuellos de botella en tiempo real, mejorando así la productividad y la eficiencia del equipo [14].

Según E. Brechner en "Agile Project Management with Kanban" [4], Kanban permite una visualización completa del trabajo mediante un tablero que lleva el mismo nombre de la metodología. El tablero Kanban está dividido en columnas que representan diferentes estados del trabajo, tales como 'Por hacer', 'En progreso' y 'Terminado', como se muestra en la **Figura 1.1**. Cada tarea o ítem de trabajo se desplaza de manera secuencial de izquierda a derecha a través del tablero a medida que progresa. Este enfoque visual facilita a los equipos ver el estado actual del trabajo, identificar cuellos de botella, y asegurar que no exista más tareas en progreso de las que el equipo puede manejar.



**Figura 1.1.** Tablero Kanban [15].

La flexibilidad y adaptabilidad de Kanban convierten a esta metodología en una de las más idóneas para entornos que experimentan cambios frecuentes en los requisitos del proyecto. La flexibilidad de Kanban radica en su capacidad para gestionar diferentes tipos de tareas

y proyectos simultáneamente, permitiendo la integración de nuevas tareas y la modificación de las existentes sin irrumpir el flujo de trabajo. Por otro lado, la adaptabilidad de Kanban proviene de la capacidad de Kanban para ajustar de manera continua las prioridades y los flujos de trabajo sin necesidad de esperar a un ciclo de desarrollo específico. A diferencia de otras metodologías, como Scrum y Waterfall, que requieren ciclos de desarrollo fijos y predefinidos, Kanban permite ajustes continuos sin esperar al final de un Sprint, lo que la hace ideal para proyectos que necesitan adaptarse rápidamente a nuevas condiciones o problemas emergentes [14].

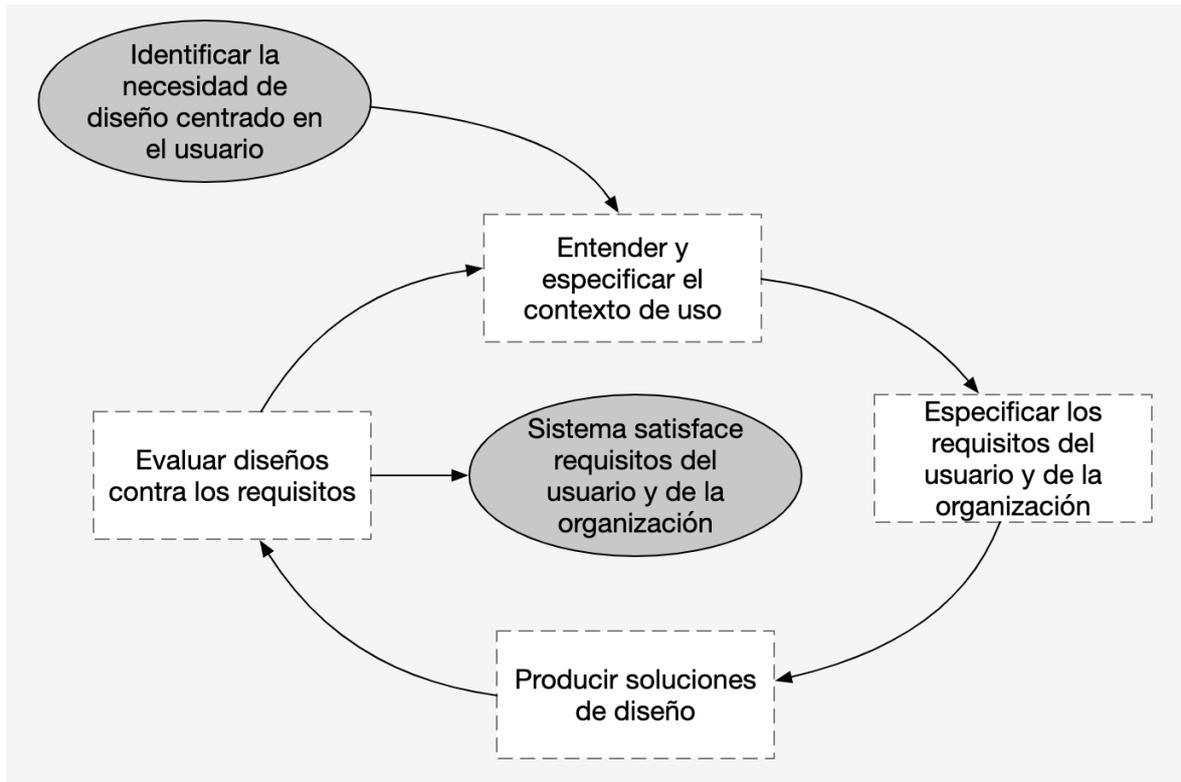
Kanban promueve la mejora continua a través del Kaizen, una filosofía japonesa que se centra en la mejora continua y que alienta a los equipos a identificar y implementar mejoras de manera proactiva. Esto no solo incrementa la eficiencia, sino que también mejora la calidad del trabajo entregado, haciendo que cualquier problema de calidad sea inmediatamente visible y sujeto a rápida corrección [14].

En el proyecto bajo estudio, Kanban es aplicada como una metodología esencial en la planificación y ejecución operativa. La implementación de Kanban mejora la productividad del equipo, asegurando que los procesos se mantengan ágiles y alineados con los objetivos de la organización. En efecto, Kanban permite i) una visualización clara del flujo de trabajo, ii) identificar oportunamente cuellos de botella, iii) mantener un enfoque en la entrega continua de valor al equipo, iv) adaptarse rápidamente a los cambios y necesidades emergentes.

### **1.4.3 DCU (Diseño Centrado en el Usuario)**

El Diseño Centrado en el Usuario (DCU) es un enfoque de diseño de productos que coloca a los usuarios finales en el centro del proceso de creación, adaptando los productos a sus necesidades, deseos y limitaciones [16]. El DCU se enfoca en los usuarios, sus necesidades y contextos específicos, asegurando que los productos desarrollados sean eficaces, eficientes y satisfactorios de usar [17]. Un diseño exitoso debe involucrar a los usuarios desde las primeras etapas del desarrollo para garantizar que sus necesidades sean realmente comprendidas y atendidas.

La **Figura 1.2** muestra el ciclo de vida del DCU según la norma ISO 9241-210. Este ciclo representa un enfoque iterativo y sistemático para asegurar que el diseño de un sistema cumpla con las necesidades y expectativas de los usuarios finales y de la organización.



**Figura 1.2.** Ciclo de Vida del DCU [17].

Las cuatro etapas del proceso del DCU se describen a continuación:

- **Entender y especificar el contexto de uso:** Es crucial entender quiénes son los usuarios, qué tareas realizarán, y en qué contextos operarán. Esto implica realizar investigaciones de usuarios, entrevistas, y observaciones para obtener una visión clara y precisa de sus necesidades y limitaciones [17].
- **Especificar los requisitos del usuario y de la organización:** La información obtenida en la etapa anterior arroja gran cantidad de información relevante de los usuarios y el contexto de uso, no obstante, esta información no está ordenada. En esta fase se procede a sintetizar la información reconociendo e identificando claramente los requisitos del usuario y de la organización. Los requisitos del usuario se centran en las necesidades y expectativas de los usuarios finales, mientras que los requisitos de la organización pueden incluir objetivos comerciales, restricciones presupuestarias y lineamientos técnicos [17]. Este equilibrio asegura que el diseño final sea viable tanto para los usuarios como para la organización. Durante esta etapa, se crean los perfiles Persona, que son representaciones detalladas de los

tipos de usuarios que potencialmente usarán el producto a diseñar. Estos perfiles reflejan las características demográficas, las necesidades, las preferencias, los comportamientos y los objetivos de grupos de usuarios significativos [16].

- **Producir soluciones de diseño:** En esta etapa, se generan soluciones de diseño basadas en los requisitos especificados en la etapa anterior. El proceso suele ser iterativo, involucrando la creación de prototipos que se refinan progresivamente a través de la retroalimentación continua de los usuarios [17]. Utilizando los perfiles Persona, los diseñadores desarrollan prototipos de baja o alta fidelidad que materializan las soluciones de diseño. El prototipado facilita la exploración de diferentes enfoques en la solución de problemas específicos de usabilidad y funcionalidad antes de la producción final, asegurando que el producto final esté alineado con las necesidades y expectativas de los usuarios [16].
- **Evaluar diseños contra los requisitos:** La evaluación continua de las soluciones de diseño con usuarios reales es fundamental para asegurar que el producto cumple con sus expectativas y necesidades. Las pruebas de usabilidad y las evaluaciones heurísticas son métodos comunes para identificar problemas y oportunidades de mejora a lo largo del ciclo de vida del proyecto [17]. Los prototipos juegan un papel crucial en esta etapa, ya que permiten a los usuarios interactuar con versiones tempranas del producto y proporcionar retroalimentación valiosa. Evaluar los diseños en esta etapa permite identificar y corregir problemas antes de que se conviertan en costosos errores en fases posteriores del desarrollo. Según la ISO 9241-210 [17], esta evaluación debe ser continua y adaptativa.

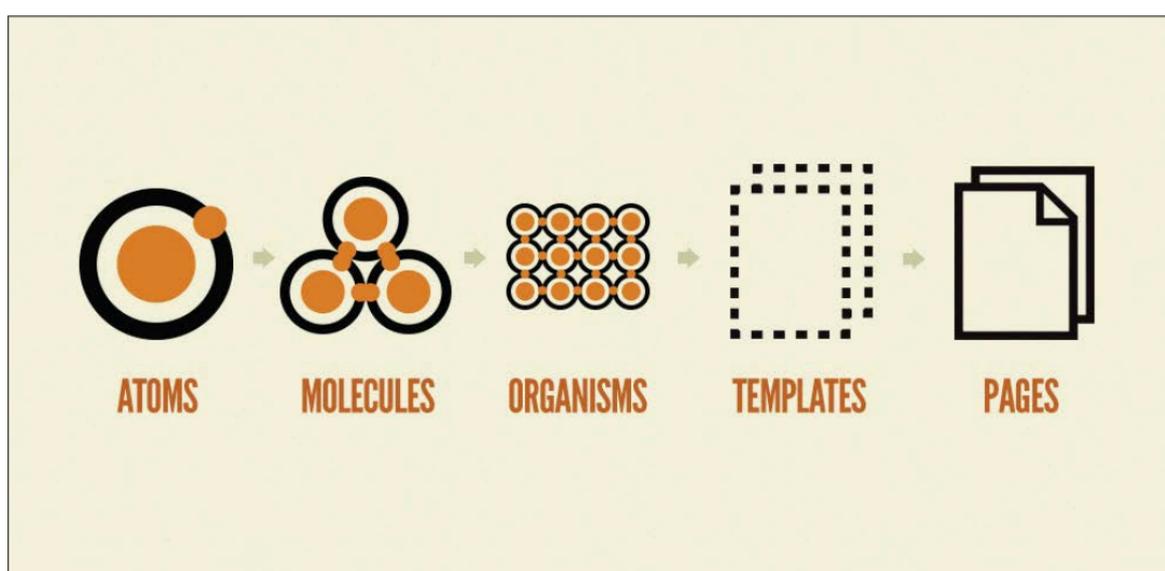
Finalmente, se verifica que el sistema desarrollado cumple con los requisitos tanto de los usuarios como de la organización. Esto implica validar que todas las necesidades y expectativas de los usuarios han sido satisfechas y que el sistema es funcional y eficiente dentro del contexto organizacional. Asegurarse de que el sistema satisface estos requisitos garantiza la efectividad y la aceptación del producto final [17].

#### **1.4.4 Atomic Design**

Atomic Design (AD) es una metodología de diseño de interfaces de usuario de aplicaciones web y móviles con un enfoque *modular*, es decir, la interfaz de usuario es considerada como un *todo* que está formado por componentes o elementos más simples, sus *partes*. El

conjunto de elementos, se considera un sistema de diseño consistente y escalable [18]. Un sistema de diseño se refiere a un conjunto de patrones y componentes reutilizables que guían y estandarizan el desarrollo de interfaces de usuario, asegurando coherencia visual y funcional a lo largo de un proyecto [19]. AD se basa en la idea de construir sistemas de diseño a partir de componentes pequeños y reutilizables, de manera similar cómo los átomos forman moléculas y estas a su vez forman organismos más complejos.

AD se basa en cinco niveles jerárquicos: átomos, moléculas, organismos, plantillas y páginas, como se muestra en la **Figura 1.3**. Estos niveles permiten una estructura clara y organizada para la construcción de interfaces de usuario complejas.



**Figura 1.3.** Niveles del Atomic Design [18].

- **Átomos:** Los átomos representan los elementos más básicos de una interfaz de usuario, como etiquetas, botones y entradas de formulario [18]. Cada átomo tiene propiedades específicas que determinan su comportamiento y estilo en el sistema de diseño.
- **Moléculas:** Las moléculas son grupos de átomos que trabajan juntos como una unidad funcional, como un componente de formulario de búsqueda que combina una etiqueta de formulario, un campo de entrada y un botón [18]. Esta combinación de átomos crea componentes reutilizables y consistentes dentro del sistema de diseño.

- **Organismos:** Los organismos son componentes más complejos formados por la combinación de múltiples moléculas y/o átomos, como un encabezado de página que contiene un logotipo, una barra de navegación y un formulario de búsqueda [18]. Diseñar organismos eficaces puede mejorar significativamente la usabilidad y coherencia de una interfaz de usuario.
- **Plantillas:** Las plantillas son estructuras de página que organizan organismos y moléculas en un diseño coherente, mostrando la disposición de los componentes dentro de una página sin el contenido final [18]. Las plantillas proporcionan una visión de la estructura subyacente del contenido y ayudan a definir la arquitectura de la información de la página.
- **Páginas:** Las páginas son instancias concretas de plantillas con contenido real que muestran cómo los componentes se integran para formar la interfaz final que verá el usuario, permitiendo probar y validar la efectividad del diseño bajo condiciones reales de uso [18]. El uso de páginas proporciona una referencia valiosa para el desarrollo y mantenimiento continuo del sistema de diseño.

Utilizar AD en este proyecto permite crear una interfaz de usuario con una filosofía modular, consistente y escalable. AD facilita la reutilización de componentes, lo que acelera el desarrollo y mantiene la coherencia visual y funcional de toda la aplicación. Sin AD, el diseño y desarrollo de la interfaz de usuario podría volverse desorganizado y el mantenimiento de la interfaz de usuario sería difícil, ya que los componentes no estarían estructurados de manera lógica y reutilizable. Además, AD ayuda a enfrentar cambios y actualizaciones de manera ágil, lo que es crucial en un entorno en constante evolución.

#### 1.4.5 Next.js

Next.js es un marco de trabajo basado en React que abstrae y configura automáticamente herramientas necesarias como el agrupamiento y la compilación, permitiendo a los desarrolladores centrarse en la construcción de aplicaciones web [20]. Next.js permite utilizar componentes React para crear interfaces de usuario y proporciona funciones y optimizaciones adicionales.

Al elegir Next.js sobre React sin un framework adicional, se obtiene una serie de ventajas clave que mejoran significativamente la experiencia de desarrollo y el rendimiento de la aplicación:

- **Hot reloading:** Esta característica permite que los cambios en el código se reflejen instantáneamente en la aplicación sin necesidad de recargar la página, lo que acelera el desarrollo y mejora la productividad del desarrollador [21].
- **Page based-routing:** Next.js proporciona un sistema de enrutamiento basado en archivos, donde cada archivo en la carpeta 'pages' se convierte automáticamente en una ruta [21]. Esto simplifica enormemente la gestión de rutas en comparación con configurar rutas manualmente en una aplicación React básica.
- **Automatic code splitting:** Next.js realiza la división automática del código, cargando solo el necesario para la página actual. Esto reduce significativamente el tamaño del paquete inicial y mejora los tiempos de carga [21]. Sin un framework como Next.js se necesitaría configurar manualmente la división del código usando herramientas como Webpack.
- **Page prefetching:** Next.js soporta el prefetching de páginas, lo que significa que las páginas vinculadas se cargan en segundo plano antes de que el usuario haga clic en el enlace [21]. Esto hace que la navegación sea mucho más rápida y fluida.
- **Hot module replacement (HMR):** Next.js incluye HMR de serie, lo que permite ver los cambios en el código en tiempo real sin necesidad de recargar la página [21]. Esto acelera el proceso de desarrollo y mejora la productividad del desarrollador.
- **Server-side rendering (SSR):** Next.js permite el renderizado del lado del servidor, lo que mejora el tiempo de carga inicial de la página y facilita la indexación por motores de búsqueda [21]. En contraste, una aplicación React sin SSR se renderiza completamente en el lado del cliente, lo que puede resultar en tiempos de carga inicial más lentos y dificultades para el SEO. El SEO (Search Engine Optimization) se refiere a las prácticas y técnicas utilizadas para mejorar la visibilidad y el ranking de un sitio web en los resultados de los motores de búsqueda [22].

Next.js ofrece una variedad de características que facilitan el desarrollo de aplicaciones web robustas y eficientes. La **Tabla 1.1** detalla las principales características de Next.js.

**Tabla 1.1.** Principales Características de Next.js [20].

Característica	Descripción
Enrutamiento	Un enrutador basado en un sistema de archivos creado sobre componentes de servidor que admite diseños, enrutamiento anidado, estados de carga, manejo de errores y más.
Representación	Renderizado del lado del cliente y del lado del servidor con componentes de cliente y servidor. Optimizado aún más con renderizado estático y dinámico en el servidor con Next.js. Streaming en tiempos de ejecución Edge y Node.js.
Obtención de datos	Obtención de datos simplificada con async/await en componentes del servidor y una fetch API extendida para memorización de solicitudes, almacenamiento en caché y revalidación de datos.
Estilo	Compatibilidad con sus métodos de estilo preferidos, incluidos módulos CSS, Tailwind CSS y CSS-in-JS
Optimizaciones	Optimizaciones de imágenes, fuentes y secuencias de comandos para mejorar los elementos básicos de la web y la experiencia del usuario de su aplicación.
TypeScript	Soporte mejorado para TypeScript, con una mejor verificación de tipos y una compilación más eficiente, así como un complemento y verificador de tipos personalizados de TypeScript.

#### 1.4.5.1 TypeScript

TypeScript es un lenguaje de programación que extiende JavaScript para facilitar el desarrollo de aplicaciones. Este lenguaje agrega una sintaxis adicional a JavaScript para una mejor integración con el editor, permitiendo detectar errores tempranamente, y convierte el código en JavaScript para su ejecución en cualquier entorno donde JavaScript funcione, ya sea en un navegador, en Node.js o Deno [23].

TypeScript ofrece inferencia de tipos y compresión JavaScript, proporcionando herramientas avanzadas que mejoran la gestión, seguridad y mantenibilidad de código en proyectos grandes sin añadir código adicional [23]. TypeScript no solo es compatible con todo el código JavaScript existente, sino que también introduce un sistema modular, clases, interfaces y un sistema de tipos graduales, facilitando una transición fluida para los

programadores de JavaScript sin requerir una reescritura o anotaciones significativas [24]. Además, TypeScript facilita la detección temprana de errores a través de su sistema de tipos, mejorando así la calidad del código y la productividad del desarrollador.

Utilizamos TypeScript en este proyecto por algunas razones. En primer lugar, su sistema de tipos permite identificar y corregir errores antes de ejecutar el código, lo cual es fundamental para mantener la robustez del sistema. También, su integración con herramientas de desarrollo modernas mejora significativamente la experiencia del desarrollador, haciendo que el desarrollo sea más fluido y menos propenso a errores. Si no utilizáramos TypeScript, estaríamos más expuestos a errores que podrían detectarse solo en tiempo de ejecución, lo que incrementaría el tiempo de depuración y el riesgo de introducir bugs en el sistema.

#### **1.4.5.2 Tailwind CSS**

Tailwind CSS es un marco de hojas de estilo (CSS) que permite a los desarrolladores crear diseños rápidamente mediante el uso de clases predefinidas directamente en el HTML [25]. Este marco facilita la construcción de cualquier diseño imaginable, proporcionando un conjunto de bloques de construcción que cubren las propiedades CSS más importantes y que pueden ser fácilmente extendidos para adaptarse a necesidades específicas [26].

Una de las características más destacadas de Tailwind CSS es su método de funcionamiento. Escanea todos los archivos HTML, componentes JavaScript y cualquier otra plantilla en busca de nombres de clases, generando los estilos correspondientes y escribiéndolos en su archivo CSS estático [27]. Este proceso elimina la necesidad de un tiempo de ejecución, haciendo que Tailwind sea rápido, flexible y confiable.

Desde la versión 2.0, Tailwind CSS es compatible con las versiones estables más recientes de Chrome, Firefox, Edge y Safari, aunque no ofrece soporte para ninguna versión de Internet Explorer [26]. Esta compatibilidad con los navegadores modernos asegura que los desarrolladores puedan crear diseños modernos y responsivos sin necesidad de preocuparse por problemas de compatibilidad.

En consecuencia, Tailwind CSS es utilizado en este proyecto para aplicar, de manera rápida, aspectos visuales a los elementos HTML que conforman la interfaz de usuario.

### **1.4.6 Figma**

Figma es una herramienta de diseño que permite crear prototipos interactivos de interfaces de usuario. Esta herramienta se destaca por su accesibilidad, ya que ofrece un plan inicial gratuito que permite a los usuarios utilizar la mayoría de sus características sin incurrir en costos. Además, Figma es una herramienta basada en la nube, lo que significa que puede ser utilizada en cualquier sistema operativo: Windows, Mac, Linux y ChromeOS, tanto a través de una aplicación de escritorio como directamente en un navegador web [28].

Una de las principales ventajas de Figma es su capacidad para permitir que varios diseñadores y miembros del equipo trabajen simultáneamente en tiempo real. Esta funcionalidad mejora significativamente el flujo de trabajo colaborativo, facilitando una cooperación más fluida y productiva entre los miembros del equipo. Según los principios fundamentales de Figma, “la colaboración es difícil. Nosotros la hacemos más fácil” [28].

Figma también se caracteriza por integrar una suite completa de herramientas de diseño en una sola plataforma. Esto incluye desde la lluvia de ideas y la creación de wireframes hasta la creación de prototipos y el intercambio de recursos. Figma no solo se centra en el diseño, sino que también genera código CSS, iOS y Android para que los desarrolladores lo utilicen, lo que facilita la transición del diseño al desarrollo [28].

En el contexto de este proyecto, Figma es utilizado como herramienta soporte para todo el proceso de diseño de interfaz de usuario, desde la creación de wireframes para definir la estructura básica de la interfaz, hasta la creación de prototipos interactivos que simulan el comportamiento real de las interfaces de usuario del sistema gestor de tareas basado en Kanban. Con Figma se diseñan las interfaces de usuario del sistema, asegurando que todos los elementos visuales y funcionales estén bien definidos antes de pasar a la etapa de implementación.

### **1.4.7 Scrum**

Scrum es un marco de trabajo ágil diseñado para desarrollar, entregar y mantener productos complejos de alta calidad. Este marco de trabajo se organiza a través de eventos periódicos, roles claramente definidos y artefactos específicos todo bajo los principios del empirismo que son transparencia, inspección y adaptación [29].

- **Transparencia:** La transparencia implica que todos los aspectos del proceso y del trabajo en curso sean visibles para todos los involucrados, permitiendo que las decisiones clave se basen en el estado visible de los tres artefactos formales: el Product Backlog, el Sprint Backlog y el Incremento [29]. La falta de transparencia puede conducir a decisiones equivocadas, lo que disminuye el valor del producto y aumenta los riesgos asociados al proyecto.
- **Inspección** La inspección en Scrum implica revisar regularmente los artefactos y el progreso hacia los objetivos acordados para identificar cualquier desviación, facilitada a través de sus cinco eventos: el Sprint, la Sprint Planning, el Daily Scrum, el Sprint Review y el Sprint Retrospective [29]. La inspección es fundamental para la adaptación; sin la capacidad de inspeccionar adecuadamente, cualquier ajuste o cambio en el proceso sería inútil.
- **Adaptación:** La adaptación es el proceso de ajustar aspectos del proyecto o del proceso de trabajo cuando se detectan desviaciones significativas o productos inaceptables [29]. Esto debe hacerse lo más pronto posible para evitar que las desviaciones crezcan y afecten negativamente el resultado final. Se espera que los equipos de Scrum adapten sus métodos y procesos en el momento en que descubren nueva información a través de la inspección.

#### 1.4.7.1 Eventos de Scrum

Respecto a los eventos, Scrum los define como ciclos temporales diseñados para crear regularidad y limitar la necesidad de reuniones no programadas. Scrum incluye cinco eventos: 1) el *Sprint*, que es un tiempo fijo durante el cual se crea un producto funcional y potencialmente desplegable 2) la *Sprint Planning*, que determina qué se va a realizar, 3) el *Scrum Daily*, que tiene como objetivo mejorar la comunicación y la eliminación de obstáculos 4) la *Sprint Review*, que demuestra los resultados obtenidos y 5) el *Sprint Retrospective*, que busca mejorar el proceso [29]. Cada uno de estos eventos sirve para asegurar que se mantenga el enfoque en los objetivos del ciclo.

#### 1.4.7.2 Artefactos de Scrum

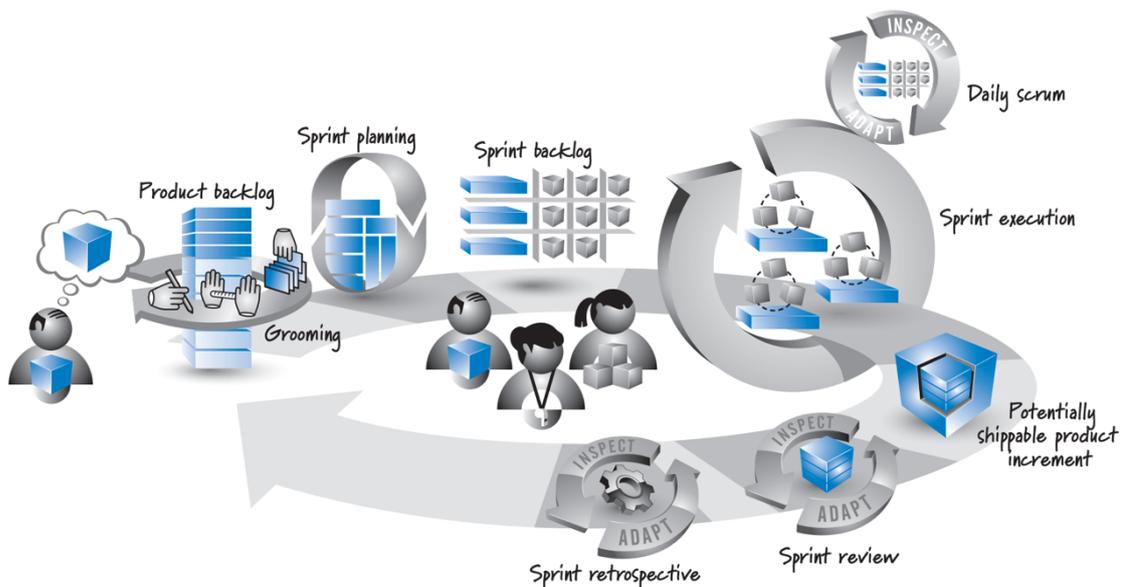
Respecto a los artefactos, Scrum define tres principales: 1) el *Product Backlog*, que contiene los requisitos del producto, 2) el *Incremento*, que es la suma de todos los elementos del Product Backlog completados durante los Sprints, y 3) el *Sprint Backlog*, que

contiene los elementos seleccionados para el Sprint junto con un plan para entregar el Incremento [29]. Cada artefacto mencionado tiene un compromiso específico asociado: la Visión del Producto para el Product Backlog, el Objetivo del Sprint para el Sprint Backlog y la Definición de Terminado para el Incremento [29]. La Visión del Producto es una descripción clara y concisa del propósito del producto [30]. La Definición de Terminado es un concepto fundamental en Scrum que describe las condiciones que un Incremento debe cumplir para considerarse completo y listo para ser entregado [31].

### 1.4.7.3 Roles de Scrum

Respecto a los roles, Scrum distingue tres principales: 1) el *Product Owner*, quien maximiza el valor del producto trabajando con el Product Backlog; 2) el *Scrum Master*, que asegura que el equipo siga la teoría, prácticas y reglas del marco de trabajo, y 3) los *Desarrolladores*, responsables de crear el Incremento del producto cada Sprint [29].

La **Figura 1.4** ilustra como se integran la mayoría de las actividades y artefactos de Scrum.



**Figura 1.4.** Marco de Trabajo de Scrum [32].

Se va a describir la imagen paso a paso, comenzando desde la parte izquierda y avanzando en sentido horario alrededor de la flecha principal en bucle, que representa el Sprint.

El diagrama inicia con el Product Owner, quien tiene una visión de lo que desea crear, representado por un cubo. Este cubo puede ser extenso, por lo que es necesario

desglosarlo en una serie de características a través de una actividad llamada Grooming. Estas características se recopilan en una lista priorizada conocida como el Product backlog

El siguiente paso en el proceso es el Sprint Planning, ubicada justo a la derecha del Product Backlog. Durante el Sprint Planning, el equipo de desarrollo selecciona un subconjunto de características del Product Backlog que creen que pueden completar durante el próximo Sprint. Una vez completado el Sprint Planning, pasamos al Sprint Execution. Un componente clave de este paso es el Scrum Daily, donde los miembros del equipo discuten brevemente su progreso, planes para el día y cualquier obstáculo que estén enfrentando.

Hacia el final del Sprint, se realiza un Sprint Review. En esta reunión, el equipo demuestra el Incremento del producto potencialmente entregable que han desarrollado durante el Sprint. Después del Sprint Review, el equipo participa en el Sprint Retrospective. Esta es una oportunidad para que el equipo reflexione sobre su desempeño durante el Sprint, discutiendo qué fue bien, qué se puede mejorar y cómo pueden mejorar en sus procesos para el próximo Sprint.

## 2 METODOLOGÍA

En esta sección se detalla el desarrollo llevado a cabo durante este proyecto, el cual se ha estructurado mediante el enfoque de Diseño Centrado en el Usuario (DCU). El desarrollo se presenta a través de las cuatro etapas fundamentales del DCU: 1) Entender y especificar el contexto de uso, 2) Especificar los requisitos del usuario y de la organización, 3) Producir soluciones de diseño, y 4) Evaluar diseños contra los requisitos. Adicionalmente, se incluirá una sección dedicada a la implementación del sistema.

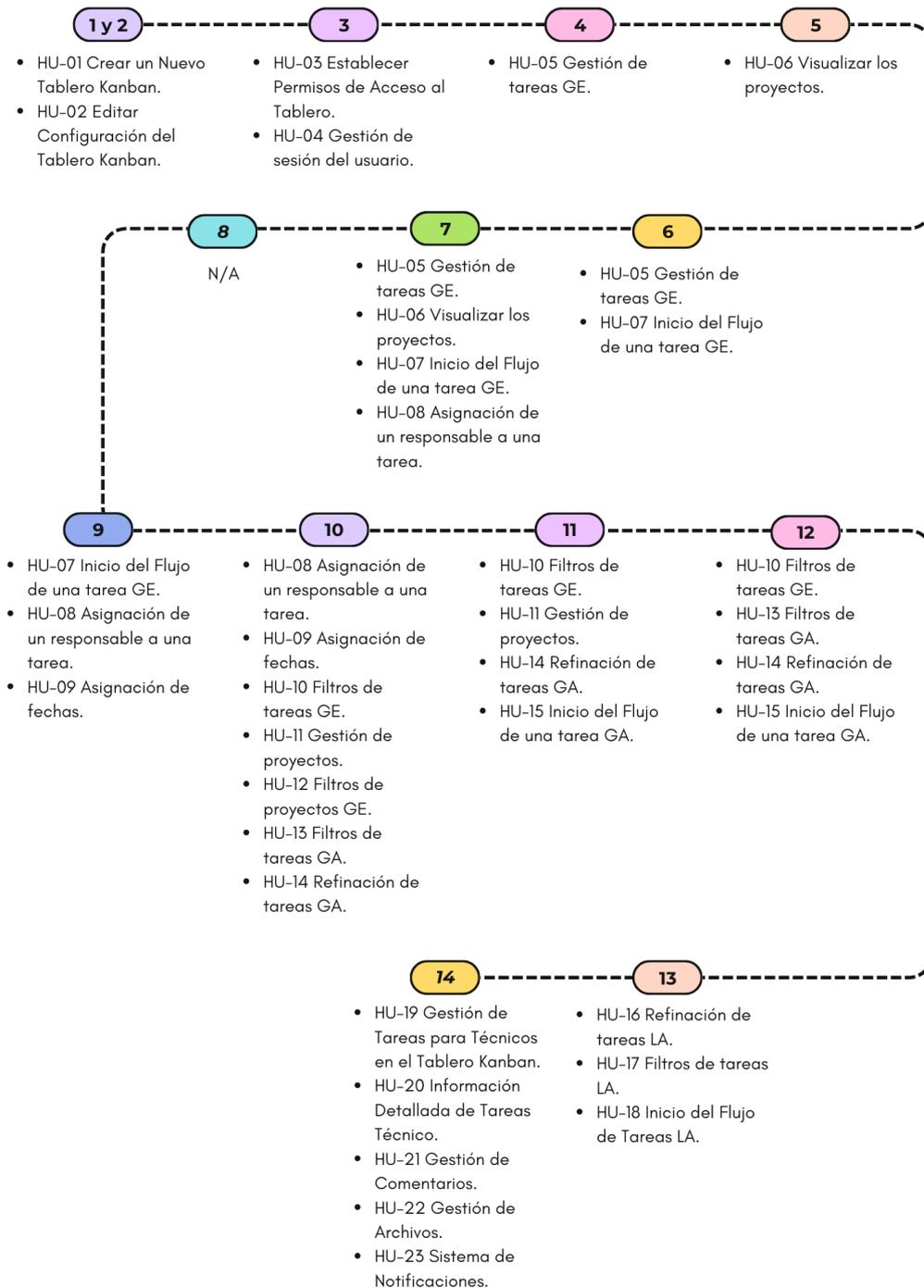
### 2.1 Entender y especificar el contexto de uso

En esta etapa, utilizamos la técnica de la entrevista para profundizar en la comprensión de las historias de usuario (HU). En cada Sprint, se realizaron conversaciones directas y recurrentes con el Product Owner (PO) para discutir, analizar y entender detalladamente las HU.

Las discusiones ayudaron a entender mejor las necesidades y expectativas de los usuarios y proporcionaron una visión clara de las funcionalidades de la aplicación. Por ejemplo, una de las consideraciones clave que emergió de las conversaciones fue la identificación de 5 perfiles Persona en el dominio, cada uno con responsabilidades específicas que influyen en su interacción con el sistema.

- Gestor Empresarial (GE),
- Gestor de Área (GA),
- Líder de Área (LA),
- Técnico,
- Administrador (del sistema).

A continuación, se presenta una síntesis de la toma de decisiones realizada en cada Sprint, en función de las conversaciones con el PO. La tabla del **ANEXO I** detalla las HU asociadas a cada Sprint, centrándose exclusivamente en las HU de negocio que aportan funcionalidades de valor directo al cliente. El lector puede contrastar la síntesis con la tabla del anexo. En la **Figura 2.1** se muestra una línea de tiempo de las HU por Sprint.



**Figura 2.1.** Línea de Tiempo de las Historias de Usuario por Sprint

**Sprint 1 y 2.-** Se llevó a cabo una conversación directa con el PO sobre la HU-01 y HU-02 para comprender mejor la necesidad de los usuarios relacionada a crear nuevos tableros Kanban y editar su configuración. En la HU-01, se discutió la importancia de permitir a los usuarios proporcionar un nombre y descripción al tablero, así como configurar la cantidad

de columnas iniciales mediante un formulario. En el Sprint 5, debido a la HU-06, esta idea fue modificada y se eliminó la opción de configurar la cantidad de columnas iniciales. Esto se acordó durante la conversación con el PO, quién estableció que cada proyecto tendría una cantidad de columnas predefinida. Respecto a la HU-02, la conversación con el PO destacó la necesidad de permitir, a los usuarios, editar la configuración de un tablero Kanban existente, incluyendo la adición, modificación o eliminación de columnas. Sin embargo, en el Sprint 5, la HU fue descartada y se decidió no implementar la gestión para las columnas.

**Sprint 3.-** Se llevó a cabo una conversación crucial con el PO centrada en la HU-03 y en la HU-04. En la HU-03 se identificaron los 5 perfiles que la aplicación debía incluir y que se ha mencionado al inicio de esta sección, no obstante, a medida que avanzaba el proyecto, se tomó la decisión de no implementar el perfil del Administrador ya que las funcionalidades requeridas por los otros perfiles cubrían la mayoría de las necesidades operativas y estratégicas de la aplicación. En la HU-04, se discutió la importancia de implementar una interfaz de inicio y cierre de sesión para que los usuarios puedan acceder y salir de sus cuentas de manera segura. Para el inicio de sesión, se requerirá un formulario con los campos de usuario (correo del usuario), contraseña y perfil. El cierre de sesión se realizará mediante un botón en la cuenta del usuario.

**Sprint 4.-** Se mantuvo una discusión con el PO sobre la HU-05 donde se enfatizó la necesidad de que el GE disponga de un Backlog para visualizar, crear, editar y eliminar tareas. La creación y edición de tareas se realizarían mediante un formulario que incluiría campos para el nombre y la descripción de la tarea.

**Sprint 5.-** La discusión sobre la HU-06 destacó la importancia de que el GE pueda acceder a un listado de proyectos para conocer todos los proyectos de la institución; una necesidad importante ya que esclareció que la aplicación incluirá una lista de proyectos, y que dentro de cada proyecto se gestionarían las tareas correspondientes. En este contexto, la HU-01 fue modificada determinando que el nombre y la descripción del proyecto se mostrarán en una ventana de información al ingresar al proyecto. Además, se planteó que, a futuro, esta información podría ser editada a través de un formulario de edición del proyecto. Asimismo, se decidió eliminar la HU-02 debido a que cada proyecto contaría con un tablero Kanban de tres columnas predeterminadas y que representarían los estados de la tarea: Por Hacer, En Progreso y Terminado.

**Sprint 6.-** Se discutió con el PO sobre la HU-07. Se identificó la necesidad de permitir que los GE inicien el flujo de trabajo de una tarea asignada a un subordinado y se concluyó que, para iniciar el flujo de una tarea, ésta debía estar previamente asignada a un responsable. Dada la complejidad de esta HU, se dividió la HU para simplificarla de lo cual emergió la HU-08, enfocada en la asignación de tareas a un responsable.

**Sprint 7.-** La conversación con el PO profundizó los detalles de la HU-07 y HU-08. Se descubrió que, al iniciar el flujo de una tarea asignada, ésta debería aparecer en el Backlog y en el tablero Kanban de la persona responsable. Se decidió que, al iniciar el flujo, la tarea cambiaría de estado, lo que motivó la creación de un nuevo estado para las tareas denominado “Nuevo”, que se aplicaría a una tarea, al ser creada. Sin embargo, este estado no se mostraría como una columna en el tablero Kanban, sino que se visualizaría únicamente en el listado de tareas del Backlog. Para la asignación de un responsable, se utilizaría un formulario; necesidad que se decidió incorporar al formulario de la HU-05.

**Sprint 8.-** En este Sprint no se discutió ninguna Historia de Usuario de negocio.

**Sprint 9.-** La conversación con el PO se centró en la HU-09. Se decidió que los Gestores Empresariales y de Área (GE y GA) puedan establecer fechas de inicio y fin a las tareas y, en el caso del GE, también a los proyectos. La necesidad, en el caso de las tareas, se incluyó en el formulario de la HU-05. En el caso de proyectos, se acordó incluir las fechas en un formulario para la creación y edición de proyectos, tema que se profundizará en el Sprint 10.

**Sprint 10.-** La discusión con el PO abordaron las HU: 10, 11, 12, 13 y 14. Las HU-10, HU-12 y HU-13 abordaron la necesidad, para el GE y GA, de filtrar tareas y proyectos, así como de ordenar por sus características, de forma ascendente y descendente. La HU-10 trató la capacidad del GE para filtrar tareas por nombre, responsable y estado. La HU-12 abordó la posibilidad de que el GE pueda filtrar los proyectos por nombre. La HU-13 abordó la necesidad del GA de filtrar las tareas por nombre, estado y proyecto. Por otro lado, en la HU-11 se destacó la importancia de que el GE pueda gestionar los proyectos de la empresa, incluyendo la creación, edición, modificación y eliminación de proyectos y que la edición y modificación de proyectos se realizaría mediante un formulario similar al de las tareas, que incluiría los campos: Nombre, Descripción, Fecha de Inicio y Fecha de Fin. En la conversación acerca de la HU-14, se discutió la necesidad de que los GA puedan refinar las tareas asignadas, descomponiéndolas en subtareas más específicas. Se resaltó la

importancia de permitir al GA ingresar a una tarea para crear y asignar subtareas a subordinados. Surgió la necesidad de crear una ventana para ver la información detallada de una tarea y un apartado para gestionar las subtareas. Esta funcionalidad de visualización detallada de las tareas se adaptaría y aplicaría a todos los perfiles, según sus necesidades específicas.

**Sprint 11.-** La conversación con el PO sobre la HU-15, discutió la necesidad de que los Líderes de Área (LA) puedan iniciar el flujo de una tarea. Esta HU, aunque similar a la HU-07, tiene enfoque específico en el LA.

**Sprint 12.-** Se revisaron y abordaron las Historias de Usuario previamente discutidas en los Sprints 10 y 11.

**Sprint 13.-** Se discutieron HU similares a las del Sprint 10, pero esta vez enfocadas en el flujo de trabajo del LA. En la HU-16, se habló sobre el refinamiento de tareas para el LA, que se gestionaría de manera similar al refinamiento de tareas de los GA. También se discutieron la HU-17 y la HU-18. La HU-17 es similar a la HU-13, pero enfocada en las tareas del LA. En cuanto a la HU-18, se revisaron los temas ya abordados en la HU-07 y HU-08.

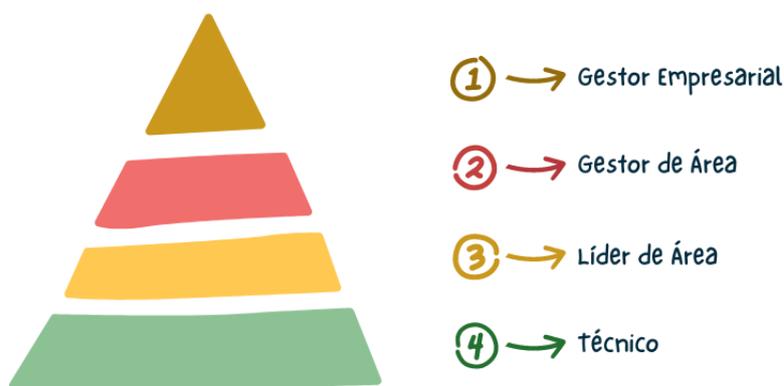
**Sprint 14.-** Se abordaron varias HU relacionadas con la gestión de tareas de los Técnicos. En la HU-19, se discutió la necesidad de que los Técnicos puedan arrastrar y soltar sus tareas en el tablero Kanban para gestionar el estado y el progreso de las tareas. Se destacaron puntos importantes, como que los Técnicos sean los únicos capaces de desplazar las tareas de una columna a otra, y que este progreso debería reflejarse en todas las tareas superiores mediante la visualización, en cada tarea, del estado y un porcentaje de completitud. Al mover una tarea de una columna a otra, el Técnico actualizará el porcentaje de completitud y el estado tanto de esa tarea como de todas las tareas superiores. También se discutió que al mover una tarea a la columna de "Terminado", ésta entrará en "revisión", una fase donde el superior inmediato confirmará la completitud de la tarea antes de cambiarla al estado "Aprobado". Si el superior encuentra que la tarea no cumple con los requisitos, podrá rechazarla y cambiar su estado a "Por Hacer". Es importante destacar que el estado "Aprobado" no se representará como una columna en el tablero Kanban. Este estado estará visible únicamente en i) la tabla del Backlog, ii) la información detallada de la tarea, y iii) el tablero Kanban, mediante un color específico. Además, se decidió que las tareas con estado "Aprobado" se mantengan en la columna de

"Terminado", pero con un color diferente a las tareas con el estado "Terminado". En la HU-20, se subrayó la importancia de permitir que los técnicos visualicen información detallada de cada tarea en el tablero, incluyendo el estado, proyecto, fechas y descripción, para obtener una perspectiva completa. Respecto a las HU 21 y 22, se trataron temas de gran valor para el negocio y que están relacionadas con el comunicar y evidenciar el trabajo realizado por los usuarios. La HU-21 se centró en la necesidad de que los usuarios puedan agregar, editar, eliminar y visualizar comentarios en las tareas del tablero Kanban. La HU-22 se centró en la importancia de que los usuarios puedan gestionar archivos adjuntos en las tareas, permitiendo agregar, editar, eliminar y visualizar documentos relacionados. Esto permitirá a un superior inmediato verificar las evidencias sobre la tarea completada para luego cambiar el estado de la tarea a "Aprobado". Se decidió que estas funcionalidades estarían disponibles dentro de la información detallada de una tarea, y por lo tanto, estarían accesibles para el superior inmediato. Además, se estableció que los comentarios en una tarea serían intercambiados únicamente entre el creador de la tarea y el responsable asignado. Por ejemplo, un GE podría comentar en una tarea con un GA, un GA podría comentar con un LA, y un LA podría comentar con un Técnico. Esta interacción se da de manera bidireccional en todos los casos. En cuanto al almacenamiento de archivos, se decidió que los archivos cargados a una tarea se almacenarán en todas las tareas superiores. Esto significa que, si un Técnico sube un archivo, este archivo también se mostrará en todas las tareas padres relacionadas. De igual manera, si un LA o un GA suben archivos, estos se mostrarán en las tareas superiores correspondientes. Por último, en la discusión con el PO acerca de la HU-23, se planteó la importancia de que los usuarios reciban notificaciones cuando se suba un archivo, se agregue un comentario o se cambie el estado de una tarea, asegurando que estén al tanto de las actualizaciones importantes y puedan supervisar adecuadamente el progreso del equipo.

## **2.2 Especificar los requisitos del usuario y de la organización**

En esta etapa se organiza y sintetiza la información obtenida en la etapa anterior, identificando claramente los requisitos tanto de los usuarios como de la organización. Para esto se ha aplicado la técnica "Persona" obteniendo cuatro perfiles que representan a los diferentes tipos de usuarios que interactuarán con el gestor de tareas. Estos perfiles incluyen características demográficas, descripciones personales, comportamientos,

necesidades, objetivos y frustraciones de los grupos de usuarios clave. La **Figura 2.2** muestra la jerarquía que hay entre los diferentes perfiles.



**Figura 2.2.** Jerarquía de perfiles

- a) Gestor Empresarial: Gestiona y supervisa todos los proyectos de la empresa. Tiene capacidad para crear tareas y asignarlas a sus subordinados, además de iniciar y supervisar el flujo de trabajo para asegurar que todo se ejecute correctamente. El detalle de este perfil se encuentra representado en la **Figura 2.3**.
- b) Gestor de Área: Gestiona las tareas asignadas por el inmediato superior. Puede desglosar estas tareas en subtareas más específicas y asignarlas a sus subordinados. Su responsabilidad principal es asegurar que estas subtareas se completen a tiempo. Este perfil se encuentra representado en la **Figura 2.4**.
- c) Líder de Área: Tiene los mismos objetivos y responsabilidades del Gestor de Área. En este sentido, el Líder de Área está también representado por el perfil de la **Figura 2.4**.
- d) Técnico: Responsable de ejecutar las tareas asignadas por su superior inmediato. Utiliza el tablero Kanban para actualizar el estado de las tareas, moviendo las tareas de una columna a otra, conforme se van completando. Este perfil se encuentra representado en la **Figura 2.5**.
- e) Administrador del Sistema: Gestiona y controla los permisos de acceso, asegurando que cada usuario tenga el nivel adecuado de acceso a los tableros Kanban y las tareas según su perfil. Este perfil se encuentra representado en la **Figura 2.6**.

**Roberto García**  
Gestor Empresarial

"La eficiencia es la clave del éxito."

**DATOS DEMOGRÁFICOS**

- Edad: 42 años
- Género: Masculino
- Ubicación: Quito, Ecuador
- Estado Civil: Casado

**TECNOLOGÍAS**

Desktop Laptop Mobile

### DESCRIPCIÓN PERSONAL

Roberto es un líder experimentado con más de 10 años de experiencia en la empresa. Es una persona enfocada y orientada a resultados. Suele ser pragmático y eficiente en la toma de decisiones. Le gusta trabajar en equipo y valora la eficacia en la ejecución de tareas. Es proactivo y busca constantemente maneras de mejorar los procesos para optimizar el rendimiento de su equipo.

### COMPORTAMIENTOS:

- Prioriza la eficiencia y la productividad en el trabajo.
- Tiende a ser organizado y metódico en la planificación y ejecución de tareas.
- Delega responsabilidades de manera clara, asignando tareas específicas a cada miembro del equipo.
- Fomenta la comunicación efectiva y la transparencia en el flujo de trabajo.

### NECESIDADES Y OBJETIVOS

- Acceder a la cuenta de forma rápida y segura.
- Distribuir las responsabilidades entre los miembros de la organización con el fin de lograr los objetivos empresariales.
- Asegurar que cada tarea asignada se complete a tiempo y con alta calidad.
- Informar regularmente sobre el progreso de las actividades al supervisor inmediato.

### FRUSTRACIONES

- Retrasos en la ejecución de tareas.
- Falta de comunicación o transparencia en el progreso de las tareas.
- Resistencia al cambio por parte del equipo.
- Obstáculos imprevistos que afectan la consecución de objetivos.

Figura 2.3. Perfil Persona del Gestor Empresarial

**María Rodríguez**  
Gestor de Área/Líder de Área

"El éxito de nuestro equipo depende de la colaboración y el esfuerzo conjunto."

**DATOS DEMOGRÁFICOS**

- Edad: 38 años
- Género: Femenino
- Ubicación: Quito, Ecuador
- Estado Civil: Casado

**TECNOLOGÍAS**

Desktop Laptop Mobile

### DESCRIPCIÓN PERSONAL

Marta es una líder empresarial integral con una amplia experiencia en la gestión de equipos jerárquicos. Su enfoque estratégico y orientación a resultados la destacan como una figura clave en la consecución de objetivos organizacionales. Es una persona motivada, equitativa y capaz de inspirar colaboración y excelencia.

### COMPORTAMIENTOS:

- Coordinación eficiente de tareas a través de la delegación y asignación de responsabilidades.
- Establecimiento de metas claras y medibles para garantizar la eficiencia operativa.
- Fomento de un ambiente de trabajo colaborativo y positivo, promoviendo la comunicación efectiva.
- Toma de decisiones rápidas y efectivas para superar obstáculos y alcanzar los objetivos establecidos.

### NECESIDADES Y OBJETIVOS

- Ingresar rápidamente y con seguridad a la cuenta.
- Entender y cumplir las tareas asignadas por mi superior.
- Delegar eficazmente el trabajo entre mis subordinados.
- Supervisar el progreso del trabajo asignado.
- Informar regularmente a mi superior sobre el avance del trabajo.

### FRUSTRACIONES

- Obstáculos que impidan el cumplimiento de los objetivos establecidos por la alta dirección.
- Falta de colaboración o comunicación efectiva dentro de los diferentes niveles jerárquicos.
- Ineficiencias operativas que afecten la productividad y calidad de la ejecución.

Figura 2.4. Perfil Persona del Gestor de Área/Líder de Área

**Sarah González**  
Técnico

"Cada tarea es una oportunidad para aprender y crecer."

**DATOS DEMOGRÁFICOS**

- Edad: 28 años
- Género: Femenino
- Ubicación: Quito, Ecuador
- Estado Civil: Soltera

**TECNOLOGÍAS**

Desktop Laptop Tablet Mobile

#### DESCRIPCIÓN PERSONAL

Sarah es una profesional joven y entusiasta que ha estado trabajando en la empresa durante los últimos 3 años. Es dedicada, comprometida y se esfuerza por realizar sus tareas con precisión. Le gusta aprender y busca constantemente maneras de mejorar sus habilidades. Se siente motivada por el logro personal y la contribución al éxito del equipo.

#### NECESIDADES Y OBJETIVOS

- Entrar a la cuenta de forma ágil y segura.
- Visualizar las tareas asignadas de manera clara y comprensible.
- Ejecutar las tareas asignadas y actualizar su progreso.
- Informar al inmediato superior sobre las labores ejecutadas.

#### COMPORTAMIENTOS:

- Cumple con las tareas asignadas de manera diligente y con atención a los detalles.
- Colabora con otros miembros del equipo para garantizar un flujo de trabajo eficiente.
- Es receptiva a la retroalimentación y busca mejorar continuamente.
- Se adapta fácilmente a cambios en las prioridades y enfoques.

#### FRUSTRACIONES

- Falta de claridad en las expectativas de las tareas asignadas.
- Obstáculos que impiden completar las tareas de manera oportuna.
- Falta de reconocimiento por el esfuerzo y el trabajo bien hecho.
- Falta de oportunidades para aprender y crecer profesionalmente.

Figura 2.5. Perfil Persona del Técnico

**Alejandro Martínez**  
Administrador del Sistema

"La tecnología es el corazón de nuestra operación."

**DATOS DEMOGRÁFICOS**

- Edad: 35 años
- Género: Masculino
- Ubicación: Quito, Ecuador
- Estado Civil: Casado

**TECNOLOGÍAS**

Desktop Laptop Mobile

#### DESCRIPCIÓN PERSONAL

Alejandro es un profesional en TI con una sólida experiencia en administración de sistemas. Es metódico, analítico y tiene una gran atención al detalle. Le gusta mantenerse actualizado sobre las últimas tendencias tecnológicas y aplicarlas de manera efectiva en su trabajo. Es un solucionador de problemas dedicado y disfruta de la responsabilidad de garantizar que los sistemas informáticos funcionen sin problemas.

#### NECESIDADES Y OBJETIVOS

- Proporcionar asistencia técnica a los usuarios con el fin de garantizar su acceso eficiente a las tareas que deben realizar.

#### COMPORTAMIENTOS:

- Administra y mantiene la infraestructura tecnológica de la empresa.
- Crea y gestiona cuentas de usuario, asigna permisos y perfiles de acceso.
- Implementa medidas de seguridad para proteger la integridad de los datos.
- Colabora con otros departamentos para asegurar la integración efectiva de sistemas.

#### FRUSTRACIONES

- Falta de recursos para implementar actualizaciones tecnológicas necesarias.
- Resistencia al cambio por parte de los usuarios.
- Problemas de seguridad que podrían comprometer la integridad de los datos.
- Falta de reconocimiento por el trabajo detrás de escena que realiza.

Figura 2.6. Perfil Persona del Administrador

Aunque el perfil Administrador fue definido en esta etapa del proyecto, éste no fue implementado en el aplicativo. Sin embargo, se incluye su definición para que pueda ser considerado en futuras implementaciones.

La importancia de los perfiles Persona radica en la capacidad para ir de lo “abstracto” a lo “concreto”. En efecto, del objetivo abstracto en cada perfil Persona, es posible derivar el conjunto de Historias de Usuario (HU) que se plantea desarrollar para cumplir de manera concreta dicho objetivo. En este contexto, podría decirse que el objetivo es el “qué” y las HU el “cómo”.

Considerando que se realizó un total de 14 Sprints, siguiendo el marco de trabajo SCRUM, y que en cada Sprint se abordó un conjunto de HU, es importante entonces, resumir la correspondencia de estas HU con los objetivos de los perfiles Persona definidos anteriormente. La **Tabla 2.1** presenta la relación “perfil Persona – historia de usuario” donde se indica el perfil Persona y las Historias de Usuario desarrolladas a lo largo del proyecto, para cumplir con cada objetivo de los usuarios.

**Tabla 2.1.** Objetivos y Historias de Usuario Asociadas de los Perfiles Persona

Perfil	Objetivo	Historia de Usuario
Gestor Empresarial	Acceder a la cuenta de forma rápida y segura.	HU-04 Gestión de sesión del usuario
		HU-05 Gestión de tareas GE
	Distribuir las responsabilidades entre los miembros de la organización con el fin de lograr los objetivos empresariales.	HU-07 Inicio del Flujo de una tarea GE
		HU-08 Asignación de un responsable a una tarea
		HU-11 Gestión de proyectos
	Asegurar que cada tarea asignada se complete a tiempo y con alta calidad.	HU-09 Asignación de fechas
		HU-10 Filtros de tareas GE
		HU-23 Sistema de Notificaciones
	Informar regularmente sobre el progreso de las actividades al supervisor inmediato.	HU-06 Visualizar los proyectos
		HU-12 Filtros de proyectos GE

Gestor de Área/Líder de Área	Ingresar rápidamente y con seguridad a la cuenta.	HU-04 Gestión de sesión del usuario
	Entender y cumplir las tareas asignadas por mi superior.	HU-07 Inicio del Flujo de una tarea GE
		HU-14 Refinación de tareas GA
		HU-16 Refinación de tareas LA
	Delegar eficazmente el trabajo entre mis subordinados.	HU-08 Asignación de un responsable a una tarea
		HU-14 Refinación de tareas GA
		HU-15 Inicio del Flujo de una tarea GA
		HU-16 Refinación de tareas LA
		HU-18 Inicio del Flujo de Tareas LA
	Supervisar el progreso del trabajo asignado.	HU-13 Filtros de tareas GA
		HU-17 Filtros de tareas LA
		HU-23 Sistema de Notificaciones
Informar regularmente a mi superior sobre el avance del trabajo.	HU-21 Gestión de Comentarios	
	HU-23 Sistema de Notificaciones	
Técnico	Entrar a la cuenta de forma ágil y segura.	HU-04 Gestión de sesión del usuario
	Visualizar las tareas asignadas de manera clara y comprensible.	HU-18 Inicio del Flujo de Tareas LA
		HU-20 Información Detallada de Tareas Técnico

	Ejecutar las tareas asignadas y actualizar su progreso.	HU-19 Gestión de Tareas para Técnicos en el Tablero Kanban
		HU-22 Gestión de Archivos Técnico
	Informar al inmediato superior sobre las labores ejecutadas.	HU-21 Gestión de Comentarios Técnico
		HU-23 Sistema de Notificaciones

De la **Tabla 2.1**, y tomando como ejemplo a la persona "Roberto" (el *Gestor Empresarial*), se puede inferir que para del objetivo:

- **"Acceder a la cuenta de forma rápida y segura"** se deriva en la HU-04, que asegura un inicio de sesión eficiente y seguro para Roberto.
- **"Distribuir las responsabilidades entre los miembros de la organización para lograr los objetivos empresariales"**, deriva en las HU:
  - HU-05 que permite a Roberto generar tareas que luego las puede distribuir entre sus subordinados.
  - HU-07 que facilita a Roberto iniciar del flujo de una tarea, permitiendo distribuir la tarea a los responsables asignados.
  - HU-08 que permite a Roberto asignar responsables a cada tarea, asegurando una distribución equitativa del trabajo dentro del equipo.
  - HU-11 que proporciona a Roberto las herramientas necesarias para gestionar los proyectos de la empresa, asegurando su ejecución para cumplir con los objetivos empresariales.
- **"Asegurar que cada tarea asignada se complete a tiempo y con alta calidad"**, deriva en las HU:
  - HU-09, que permite a Roberto asignar fechas a las tareas con el fin de especificar su cumplimiento en plazos establecidos;

- HU-10, que permite a Roberto especificar filtros para enfocarse en un grupo específico de tareas y verificar su progreso.
- HU-23, que implementa un sistema de notificaciones para mantener a Roberto al día con el progreso de las tareas, garantizando así la calidad de las mismas.
- **"Informar regularmente sobre el progreso de las actividades al supervisor inmediato"** se derivan las HU:
  - HU-06, que permite a Roberto visualizar todos los proyectos.
  - HU-12, que facilita a Roberto aplicar filtros específicos para verificar el avance de los proyectos. Así, Roberto puede informar a su supervisor inmediato, el progreso de las actividades.

## **2.3 Producir soluciones de diseño**

En esta etapa, se desarrollaron soluciones de diseño (prototipos de interfaz de usuario) basadas en las Historias de Usuario (HU) identificadas en la etapa anterior. Por su puesto, el desarrollo de las soluciones de diseño, y su evolución, fue tratado en cada Sprint, considerando el conjunto de HU del Sprint.

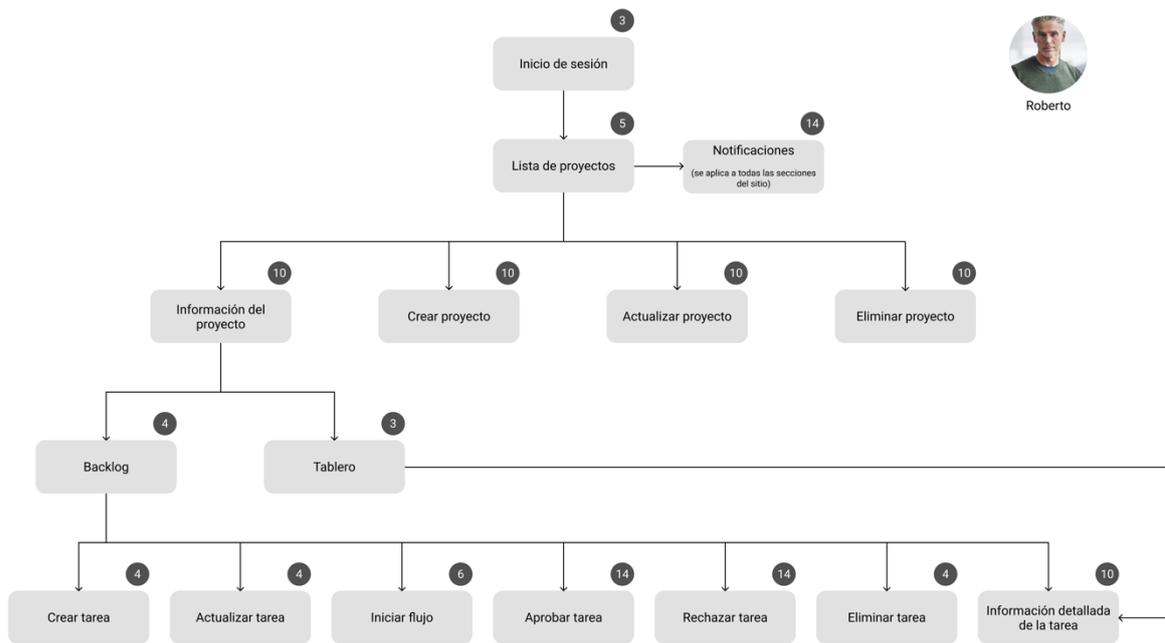
En general, el desarrollo de las soluciones de diseño incluyó la creación de un mapa de navegación, que fue evolucionando tras cada Sprint, y el diseño de prototipos interactivos, ajustados a las HU y propias de cada Sprint. Mientras el mapa de navegación detalla la estructura y flujo de la aplicación, los prototipos permiten visualizar y evaluar la funcionalidad y usabilidad de las interfaces de usuario del gestor de tareas. En consecuencia, las soluciones de diseño se enfocan en cumplir con las necesidades y expectativas de los distintos perfiles Persona: Gestor Empresarial, Gestor de Área, Líder de Área y Técnico.

### **2.3.1 Mapa de navegación**

A excepción del Administrador, por cada perfil Persona se desarrolló un mapa de navegación. A continuación, por cada perfil Persona se presenta el mapa de navegación final, el resultante de la evolución tras los diferentes Sprint. En el mapa, cada página contiene un número que hace referencia al número de Sprint en el que la página fue creada

o actualizada. Esto permite explicar visualmente cómo fue evolucionando el mapa hasta llegar al mapa final.

La **Figura 2.7** muestra el mapa de navegación final correspondiente a Roberto.



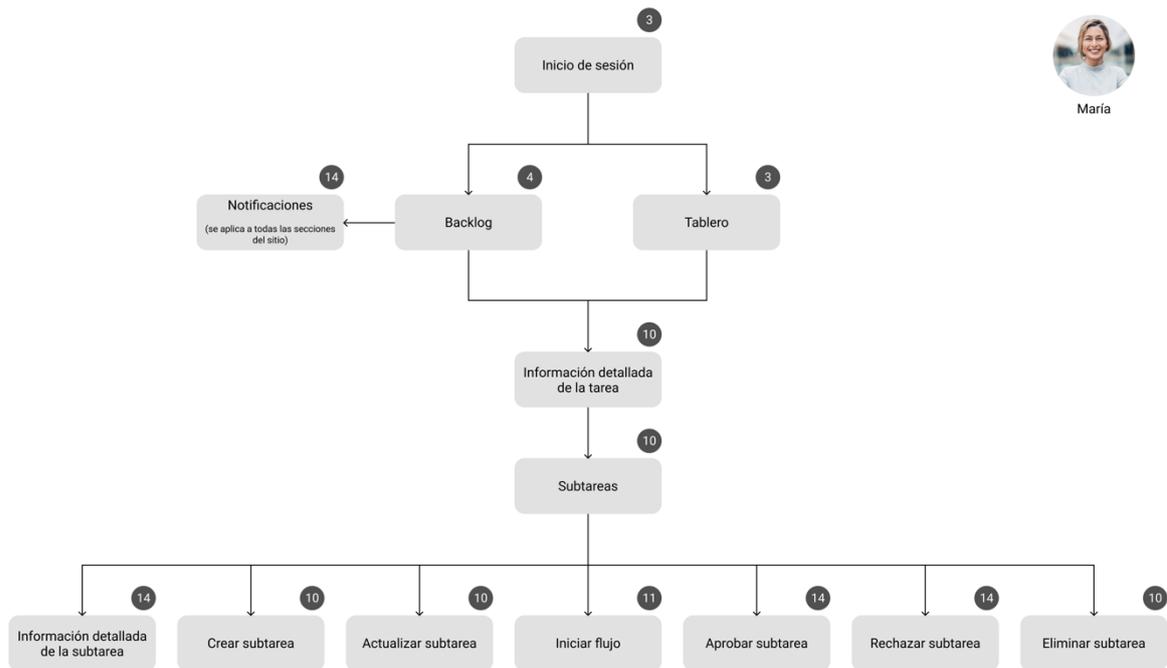
**Figura 2.7.** Mapa de Navegación del Gestor Empresarial

**Contexto general:** Roberto accederá a la aplicación mediante la página “inicio de sesión”. A continuación, se le presentará la “lista de proyectos” de la empresa. Desde esta lista, Roberto podrá acceder a diferentes páginas para visualizar, crear, actualizar o eliminar un proyecto. Al ingresar a “Información del proyecto”, Roberto podrá navegar al “Backlog” y “tablero Kanban” de un proyecto específico. Desde el Backlog, será capaz de crear, editar, iniciar el flujo, aprobar, rechazar, eliminar o ver la información detallada de una tarea del Backlog. Roberto puede acceder a la “información detallada de una tarea” tanto desde el Backlog como desde el tablero Kanban. Finalmente, la página de “notificaciones”, estará accesible desde cualquier parte del sitio, una vez que el usuario haya iniciado sesión.

**Evolución:** Los primeros indicios del mapa fueron en el Sprint 3. En este Sprint únicamente se tuvo las páginas “inicio de sesión” y “tablero Kanban”. Mas tarde, en el Sprint 4 y siguiendo la HU-05, al mapa se agregaron cuatro páginas: “Backlog”, “Crear Tarea”, “Actualizar Tarea” y “Eliminar tarea”. En ese punto de tiempo, se consideró que después de iniciar sesión, Roberto tendría acceso tanto al Backlog como al tablero Kanban. En el

Sprint 5 y considerando la HU-6, se agregaron las páginas “lista de proyectos”. El flujo logrado hasta entonces cambia para que esta nueva página se presente una vez que Roberto inicie sesión. Desde la página de lista de proyectos, se puede acceder al Backlog y al tablero Kanban. En el Sprint 6, surgió la necesidad de iniciar el flujo de una tarea (HU-07) lo que dio lugar a la creación de la página “inicio del flujo” de una tarea, accesible desde la página del Backlog. En el Sprint 10, para cumplir con las HU-11 y HU-14, se agregaron las páginas para, en el caso de la HU-1, “crear”, “editar” y “eliminar” un proyecto (accesibles desde la página de lista de proyectos), e “información detallada de una tarea”, en el caso de la HU-14. Aunque la HU-14 originalmente centraba el refinamiento de tareas por parte del Gestor de Área, durante la conversación con el PO en la etapa 1 del DCU, se acordó que Roberto también incluiría la capacidad de visualizar la información detallada de una tarea. Esta página es accesible tanto desde el Backlog como desde el tablero Kanban. En el Sprint 14 se realizó la última modificación del mapa de navegación y que está relacionada a la HU-19 que aborda la gestión de tareas del Técnico y HU-23, relacionada con el sistema de notificaciones. En el caso de la HU-19, se identificó la necesidad de que, cuando un técnico cambie una tarea al estado “Terminado”, su superior inmediato pueda verificar la correcta ejecución y tenga la opción de aprobar o rechazar una tarea. Esta funcionalidad se extendió a todos los perfiles, permitiendo que Roberto pueda aprobar o rechazar tareas completadas por sus Gestores de Área. Roberto accedería a esta funcionalidad desde su Backlog. Para llevar a cabo la HU-23, se creó la página “Notificaciones”, accesible desde cualquier parte del sitio una vez que Roberto haya iniciado sesión.

A continuación, se presenta el mapa de navegación final de María, representado en la **Figura 2.8**.



**Figura 2.8.** Mapa de Navegación del Gestor de Área/Líder de Área

**Contexto general:** Inicialmente, María tendrá acceso a una página dedicada al “inicio de sesión”. Una vez iniciada la sesión, podrá acceder a su “Backlog” y a su “tablero Kanban”. Desde estas dos páginas, podrá visualizar la “información detallada de una tarea”. Dentro de la página de información detallada de la tarea, podrá ver las subtareas asociadas. Desde la página de “subtareas”, María tendrá varias opciones: acceder a la “información detallada de una subtarea”, así como crear, actualizar, iniciar el flujo, aprobar, rechazar y eliminar subtareas. Finalmente, tendrá acceso a la página de “notificaciones”, la cual será accesible desde cualquier parte del sitio una vez que María haya iniciado sesión.

**Evolución:** En el Sprint 3, con la HU-04, se trató la gestión de sesión del usuario, por lo que, se crearon dos páginas: Inicio de sesión y el tablero Kanban. Después de iniciar sesión, María tendría acceso únicamente a su tablero Kanban en este Sprint. Durante el Sprint 4, surgió la HU-05, que se refiere a la gestión de tareas del Gestor Empresarial. En la conversación con el PO realizada en la etapa 1 del DCU, se discutió la posibilidad de disponer de un Backlog, por lo que este también fue implementado en el perfil de María. Una vez que María inicie sesión, tendrá acceso al Backlog. En el Sprint 10, para cumplir con los requerimientos de la HU-14, que trata sobre la refinación de tareas de María, se crearon varias páginas adicionales. En la conversación con el PO durante la etapa 1 del

DCU, se discutió la importancia de tener una página para visualizar la información detallada de la tarea. Por esta razón, se agregó dicha página al mapa de navegación. Desde la página de información detallada de la tarea, se puede acceder a la página de las subtareas, donde María puede crear, actualizar y eliminar subtareas. En el Sprint 11, la HU-15 abordó la necesidad de iniciar el flujo de una subtask, por lo que se creó una página específica para este propósito, accesible desde la página de subtareas. En el Sprint 14, se realizaron las últimas modificaciones al mapa de navegación de María. Con la HU-19, que trata sobre la gestión de tareas para un Técnico, se discutió con el PO la posibilidad de aprobar y rechazar tareas, lo que resultó en la creación de dos nuevas páginas. La HU-21 y la HU-22, centradas en la gestión de archivos y comentarios respectivamente, destacaron la necesidad de que María pudiera visualizar la información detallada de una subtask y realizar dichas acciones dentro de esta página. Finalmente, la HU-23, relacionada con el sistema de notificaciones, permitió la creación de una página para visualizar las notificaciones del usuario, accesible desde cualquier parte del sitio una vez que María haya iniciado sesión.

Por último, se revisa el mapa de navegación final de Sarah, cuya representación se encuentra en la **Figura 2.9**.



**Figura 2.9.** Mapa de Navegación del Técnico

**Contexto general:** Inicialmente, Sarah tendrá acceso a la página de “inicio de sesión”. Después de iniciar sesión, se le presentará su “tablero Kanban”, desde el cual podrá acceder a la “información detallada de una tarea”. Además, tendrá acceso a la página de “notificaciones”, la cual será accesible desde cualquier parte del sitio una vez que Sarah haya iniciado sesión.

**Evolución:** En el Sprint 3, se habla acerca de la HU-04, la cual hace referencia a la gestión de sesión del usuario, por lo que, para cumplir con los requisitos de esta HU, se creó una página de inicio de sesión para Sarah. Luego de que Sarah ingresara sus credenciales, podría navegar a la página del tablero Kanban. En el Sprint 14, con la HU-20 y la HU-23, se realizó la última modificación del mapa de navegación de Sarah. La HU-20 se refiere a la visualización de la información detallada de una tarea, a la cual se accedería desde el tablero Kanban. Por otro lado, la HU-23 hace referencia al sistema de notificaciones, que sería accesible desde cualquier página del sistema una vez iniciada la sesión.

### **2.3.2 Prototipos**

Se realizó la creación de prototipos como parte esencial de la fase de preparación del proyecto. Para crear los prototipos se utilizó Figma. Los perfiles Persona desarrollados, los mapas de navegación y la interacción con el PO fueron cruciales para garantizar que todos los requisitos y expectativas fueran claramente comprendidos y reflejados en el diseño. Los prototipos permitieron iterar rápidamente sobre el diseño, incorporando feedback y ajustando detalles antes de avanzar a la fase de desarrollo.

A continuación, se presentan los prototipos desarrollados, enfocándose únicamente en las páginas más relevantes del sistema. El lector puede revisar el **ANEXO II** para visualizar los prototipos en detalle.

En la **Figura 2.10** se muestra la interfaz de inicio de sesión que utilizan todos los perfiles.



Figura 2.10. Prototipo de Inicio de Sesión

Roberto es el único usuario que dispone de una lista de proyectos. El prototipo de esta interfaz se encuentra en la **Figura 2.11**.

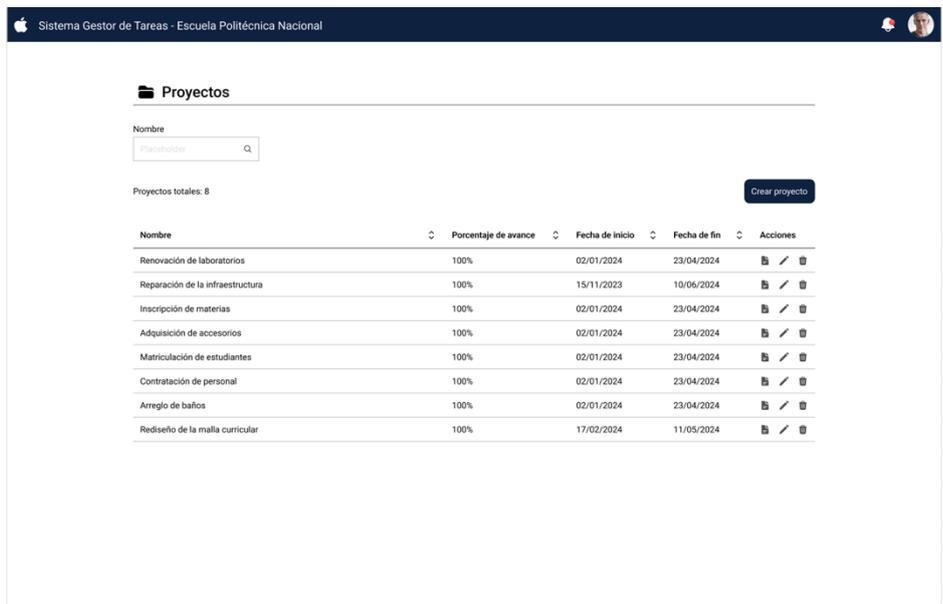
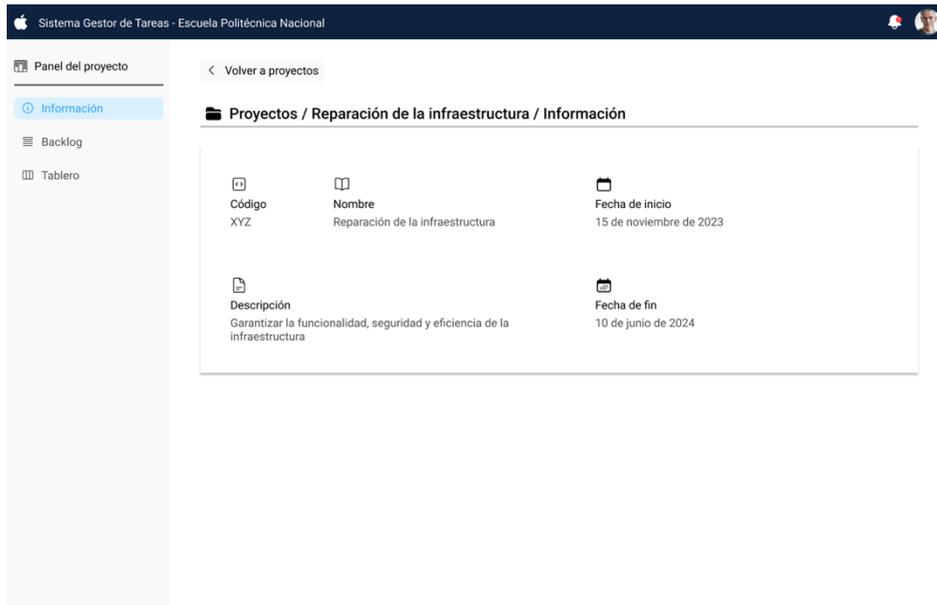


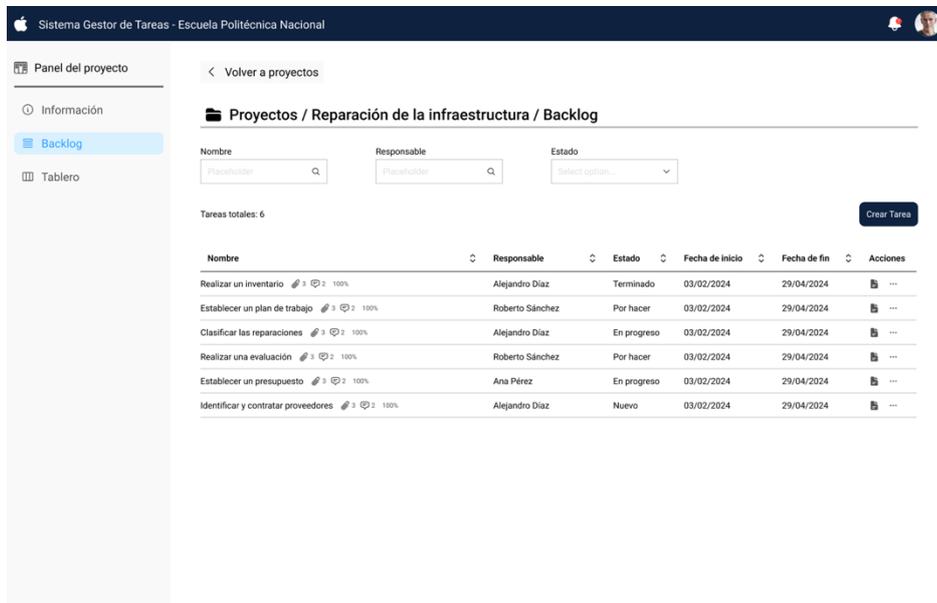
Figura 2.11. Prototipo de la Lista de Proyectos

Roberto también es el único que puede ver la información detallada de un proyecto. Esta interfaz se muestra en la **Figura 2.12**.



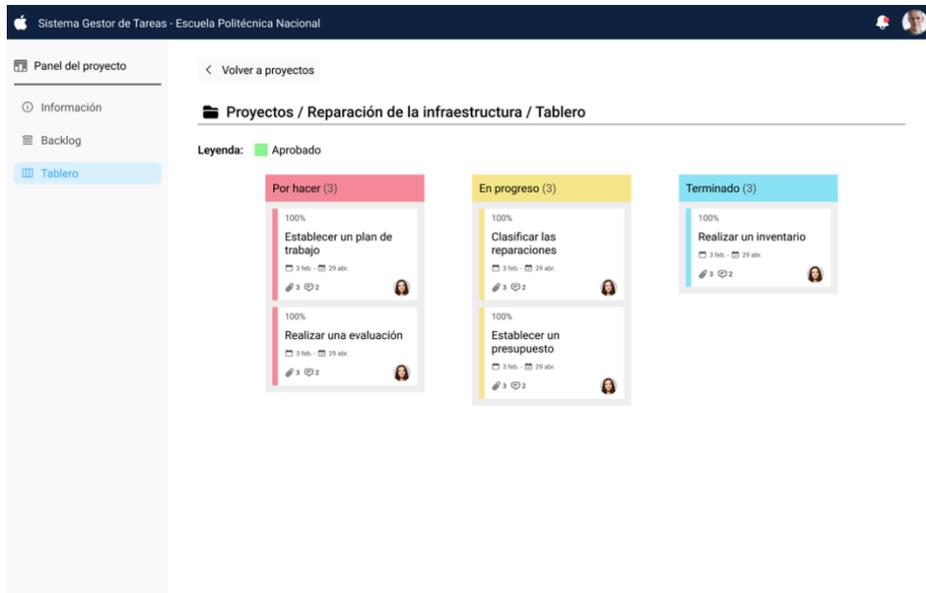
**Figura 2.12.** Prototipo de Información Detallada de un Proyecto

La interfaz Backlog, utilizada tanto por Roberto como por María, se presenta en la **Figura 2.13**. La versión de María es muy similar, con la excepción de que ella no puede gestionar las tareas.



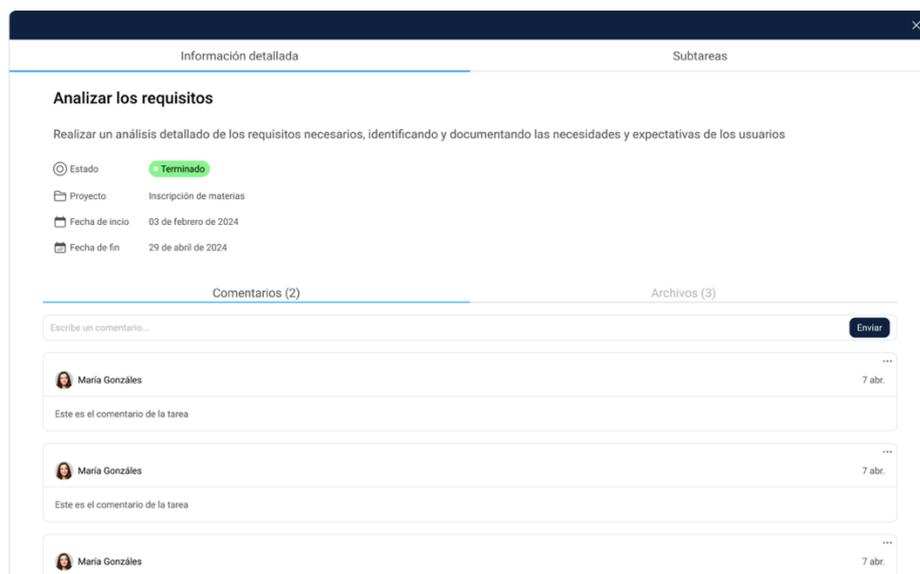
**Figura 2.13.** Prototipo del Backlog

A continuación, se muestra la interfaz “tablero Kanban”; la más importante del sistema. Todos los perfiles comparten una interfaz similar del tablero. Como ejemplo, en la **Figura 2.14** se presenta el tablero de Roberto.



**Figura 2.14.** Prototipo del Tablero Kanban

La **Figura 2.15** presenta la interfaz “información detallada de una tarea”. En este caso, se muestra el prototipo desarrollado para María. Todos los perfiles tienen una interfaz similar, con la excepción de que solo María tendrá acceso a las subtareas desde esta página.



**Figura 2.15.** Prototipo de la Información Detallada de una Tarea

Finalmente, la **Figura 2.16** presenta la página de las subtareas, exclusiva para el uso de María.

Nombre	Responsable	Estado	Fecha de inicio	Fecha de fin	Acciones
Analizar prerrequisitos  3  2 ✓ 2/4 100%	Pedro Enriquez	En progreso	02/01/2024	08/04/2024	...
Detallar la relación entre los requisitos  3  2 ✓ 2/4 100%	Cristian Medina	Terminado	02/01/2024	08/04/2024	...
Programar reuniones iniciales  3  2 ✓ 2/4 100%	Cristian Medina	Por hacer	02/01/2024	08/04/2024	...
Analizar procesos actuales  3  2 ✓ 2/4 100%	Oscar Tapia	Nuevo	02/01/2024	08/04/2024	...

**Figura 2.16.** Prototipo de las Subtareas

## 2.4 Evaluar diseños contra los requisitos

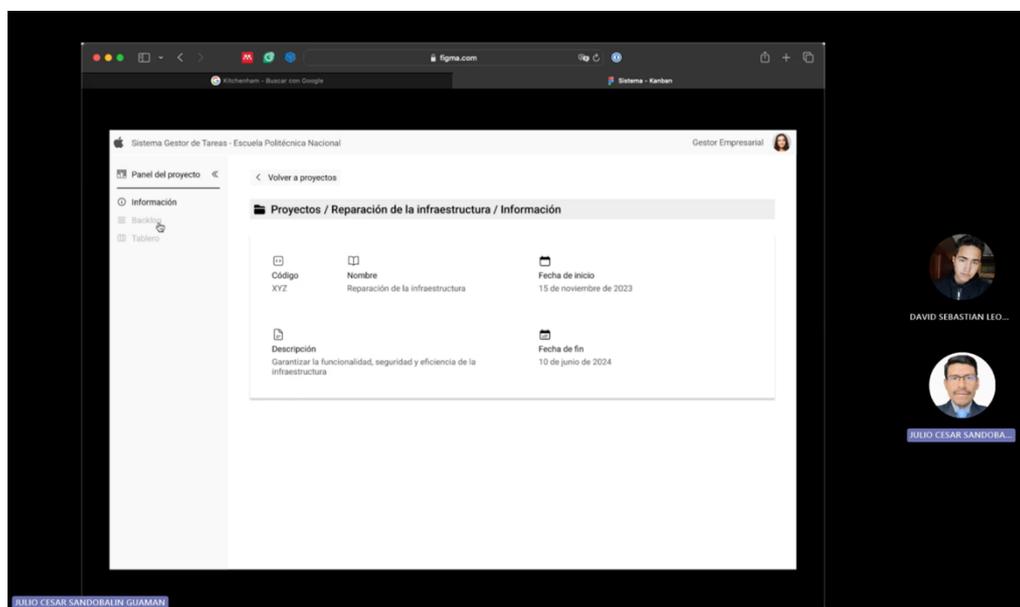
La etapa de evaluación de diseños contra los requisitos es crucial en el proceso de Diseño Centrado en el Usuario (DCU), ya que permite verificar que las soluciones de diseño desarrolladas cumplan con las necesidades y expectativas de los usuarios y la organización. Durante esta etapa, se llevaron a cabo pruebas de usabilidad con el PO que arrojaron retroalimentación valiosa para validar y refinar los diseños propuestos. A continuación, se describe el protocolo seguido para evaluar los diseños producidos en la etapa 3 del DCU.

- 1) Preparación: Se elaboró un guion detallado para las pruebas, incluyendo instrucciones para el PO y las tareas específicas a ejecutar. En el **ANEXO III** se presenta un ejemplo del guion utilizado durante las pruebas.
- 2) Ejecución de las pruebas: La mayoría de estas pruebas se llevaron a cabo virtualmente a través de la plataforma de Microsoft Teams. Además, se realizó una prueba de manera presencial, siguiendo el mismo guion y estructura que en las pruebas virtuales.

- 3) Desarrollo de la sesión: Se siguió el guion elaborado en la fase de la preparación. Además, se observaron y anotaron las acciones y comentarios del PO durante la realización de las tareas.
- 4) Recopilación de retroalimentación: La retroalimentación proporcionada por el PO sobre los prototipos fue recopilada de manera ordenada. Los detalles de la retroalimentación obtenida se encuentran en la sección **3.1 Resultados**.
- 5) Análisis y ajustes: Se analizó la retroalimentación obtenida durante las pruebas y se realizaron los cambios necesarios en los prototipos basados en los comentarios del PO. Los cambios más relevantes realizados a los prototipos se encuentran detallados en la sección **3.1 Resultados**.

La importancia del protocolo es garantizar que las pruebas de usabilidad sean consistentes y estructuradas, permitiendo recoger información valiosa para mejorar los prototipos en base a la experiencia y comentarios del PO.

En el **ANEXO IV** se encuentran los enlaces a todas las pruebas realizadas. Aunque no todas las pruebas fueron grabadas debido a limitaciones técnicas, en cada prueba se recopilaban observaciones y comentarios detallados. La **Figura 2.17** muestra una imagen tomada de una de las pruebas realizadas con el PO.

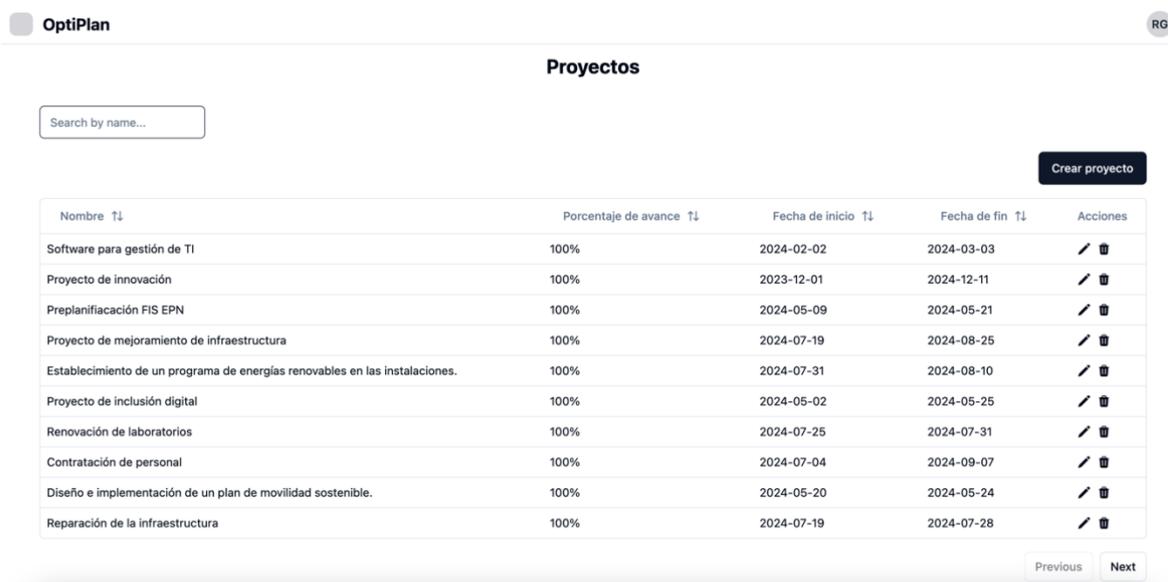


**Figura 2.17.** Prueba de Usabilidad con el PO

## 2.5 Implementación

La etapa de implementación abarca las actividades relacionadas con la construcción y desarrollo del sistema, basado en los diseños probados durante la fase de evaluación. En esta sección, se detallan los pasos y metodologías utilizados para transformar los prototipos en un sistema funcional y operativo. Para el desarrollo, se utilizaron Next.js, TypeScript y Tailwind, lo que permitió crear una aplicación robusta, tipada y con estilos modernos.

Es relevante mencionar que Next.js, al estar basado en React, también utiliza componentes en su funcionamiento. En este proyecto, se decidió estructurar el código siguiendo la metodología de Atomic Design (AD). Esta decisión permite una organización más clara, facilitando la reutilización de componentes y la escalabilidad del proyecto. La **Figura 2.18** muestra la página de lista de proyectos desarrollada en las tecnologías mencionadas y siguiendo la metodología AD.

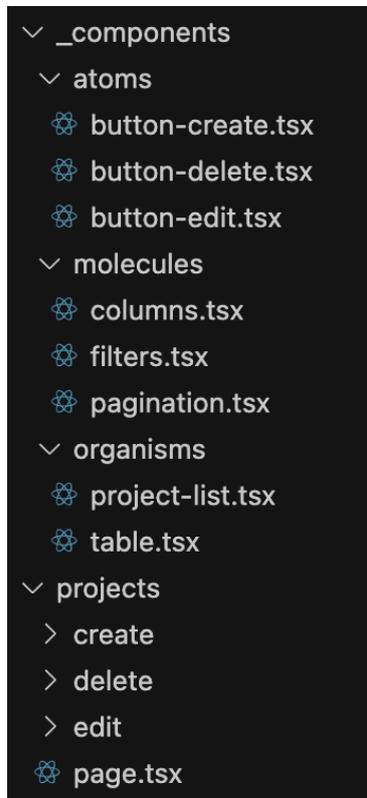


Nombre ↕	Porcentaje de avance ↕	Fecha de inicio ↕	Fecha de fin ↕	Acciones
Software para gestión de TI	100%	2024-02-02	2024-03-03	✎ 🗑
Proyecto de innovación	100%	2023-12-01	2024-12-11	✎ 🗑
Preplanificación FIS EPN	100%	2024-05-09	2024-05-21	✎ 🗑
Proyecto de mejoramiento de infraestructura	100%	2024-07-19	2024-08-25	✎ 🗑
Establecimiento de un programa de energías renovables en las instalaciones.	100%	2024-07-31	2024-08-10	✎ 🗑
Proyecto de inclusión digital	100%	2024-05-02	2024-05-25	✎ 🗑
Renovación de laboratorios	100%	2024-07-25	2024-07-31	✎ 🗑
Contratación de personal	100%	2024-07-04	2024-09-07	✎ 🗑
Diseño e implementación de un plan de movilidad sostenible.	100%	2024-05-20	2024-05-24	✎ 🗑
Reparación de la infraestructura	100%	2024-07-19	2024-07-28	✎ 🗑

**Figura 2.18.** Página de Lista de Proyectos

Para el desarrollo de la página de lista de proyectos, se consideraron y utilizaron varios componentes clave. La estructura de estos componentes se detalla en la **Figura 2.19**. Es importante mencionar que no todos los componentes que conforman esta página están ubicados en la estructura del código de la **Figura 2.19**. Esto se debe a que hay

componentes, como el header, que se utilizan en varias páginas del sistema. Por lo tanto, estos componentes se ubicaron en la raíz del proyecto para que puedan ser reutilizados.



**Figura 2.19.** Estructura del Código para la Página de Lista de Proyectos

Ahora, se explica cómo los distintos componentes mostrados en la **Figura 2.19**, conforman la página mostrada en la **Figura 2.18**. Se comienza desde los elementos más pequeños hasta los más complejos, siguiendo la metodología del Atomic Design.

**Átomos.** - Los átomos son los componentes más básicos y fundamentales. En este caso, los átomos están representados por los componentes 'button-create', 'button-delete' y 'button-edit', que son los iconos de crear, eliminar y actualizar un proyecto, respectivamente. Estos botones están compuestos por un icono y una leyenda que se muestra al pasar el cursor por encima del componente.

**Moléculas.** - Las moléculas son componentes más complejos formados por la combinación de átomos. En nuestra aplicación, estos componentes son 'columns', 'filters' y 'pagination'.

- **Columns:** Representa una columna de la tabla y está formada por el título de la columna, un icono para ordenar los datos de manera ascendente y descendente, y las filas correspondientes a esa columna.
- **Filters:** Permite buscar un proyecto por nombre, facilitando la navegación y búsqueda dentro de la tabla.
- **Pagination:** Incluye los botones de "Previous" y "Next" para navegar entre las páginas de la tabla.

**Organismos.** - Los organismos son componentes aún más complejos que agrupan múltiples moléculas. En este caso, los organismos incluyen 'project-list' y 'table'.

- **Table:** Este componente reúne toda la información de las columnas para presentarlas en una sola tabla cohesiva.
- **Project-list:** Agrupa el filtro de búsqueda, el botón de crear proyecto, toda la tabla y los componentes de paginación.

**Páginas.** - Finalmente, en la capa más alta del diseño, tenemos las páginas. El page de Proyectos no solo incorpora el organismo 'data-table', sino que también añade el título de "Proyectos". También es importante mencionar que el header está colocado en el layout que envuelve a todo el sistema.

La librería shadcn/ui facilitó la creación de componentes necesarios para varias funcionalidades. En el caso de la lista de proyectos presentada en la **Figura 2.18**, se utilizó esta librería para implementar los filtros, botón de "Crear Proyectos", las columnas, el ordenamiento, la tabla y la paginación.

Otra consideración importante en el desarrollo de la aplicación es la implementación del inicio de sesión de los usuarios. Para esta funcionalidad, se utilizó Next Auth, una biblioteca de autenticación flexible y fácil de integrar con Next.js.

En el tablero del Técnico, se implementó la funcionalidad de arrastrar y soltar tareas entre columnas, ya que este rol es el único autorizado para gestionar el flujo de trabajo de esta manera. Para lograr esto, se utilizó la librería dnd-kit.

### 3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

#### 3.1 Resultados

En esta sección, se presentan los resultados obtenidos a partir de la evaluación detallada en la etapa de “Evaluar diseños contra los requisitos”. Los resultados se explicarán en función de las mejoras implementadas a lo largo del tiempo, basadas en la retroalimentación recibida. El objetivo es mostrar cómo estas mejoras han contribuido al desarrollo del proyecto, destacando los avances significativos y los cambios realizados en los prototipos de la aplicación.

Durante el desarrollo del proyecto, se realizaron un total de 5 pruebas y se obtuvieron un total de 27 retroalimentaciones por parte del PO. Estas retroalimentaciones fueron cruciales para identificar áreas de mejora y ajustar el diseño y funcionalidad de los prototipos.

A partir de las retroalimentaciones, se realizaron 23 cambios significativos en los prototipos. Estos cambios abarcaron desde ajustes menores en la interfaz de usuario hasta la implementación de nuevas funcionalidades y mejoras en la usabilidad del sistema.

A continuación, la **Tabla 3.1** muestra de manera detallada un resumen de la retroalimentación recibida y los cambios implementados en los prototipos durante las diversas pruebas.

**Tabla 3.1.** Resumen de Retroalimentación y Cambios Realizados en los Prototipos

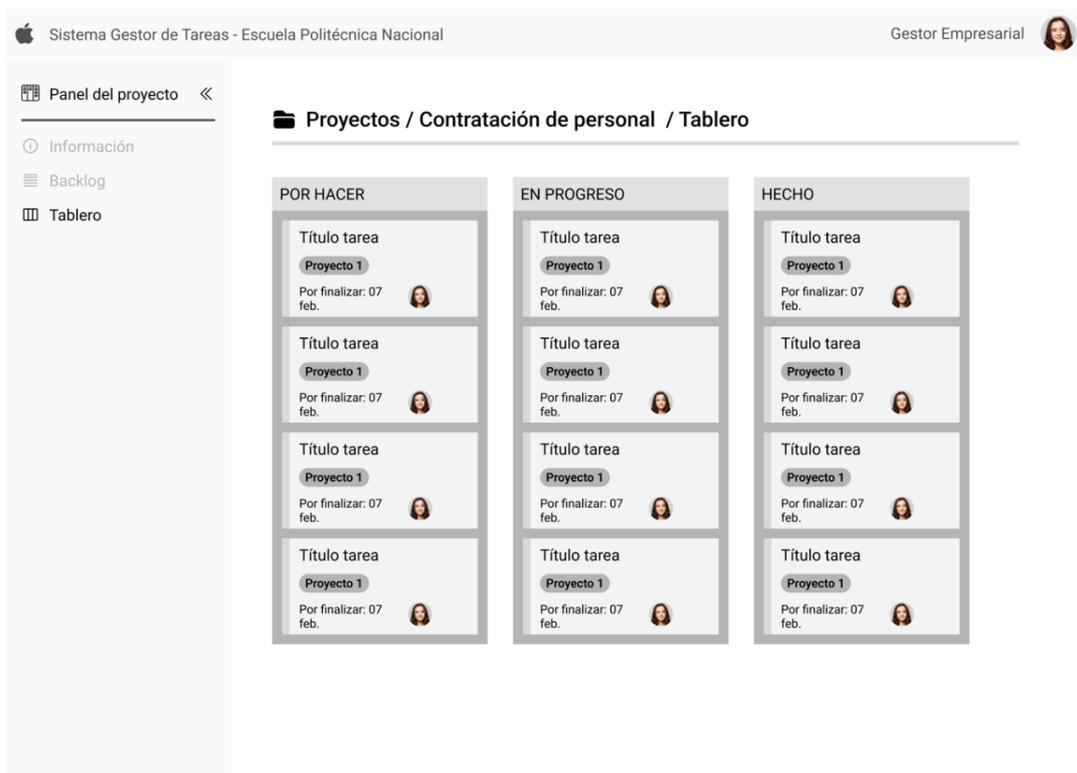
<b>Prueba</b>	<b>Retroalimentación</b>	<b>Cambios realizados</b>
Primera prueba	Añadir leyendas explicativas en los íconos de las acciones.	Se añadieron leyendas explicativas en los íconos.
	Confirmar la eliminación de proyectos y tareas con un mensaje de confirmación.	Mensaje de confirmación añadido para la eliminación.
	Añadir un efecto hover en las filas de la tabla para mejorar la visualización.	Efecto hover añadido en las filas de las tablas de la lista de proyectos y el Backlog.
	Mantener el ícono de iniciar flujo y al hacer clic mostrar un ícono de pausa.	Ícono de pausa añadido al iniciar flujo de una tarea.

	Renombrar la columna "Hecho" a "Terminado".	Columna renombrada a "Terminado".
	Incluir un botón para volver a la lista de proyectos.	Botón añadido para volver a la lista de proyectos.
	Añadir la opción de cerrar sesión al hacer clic en la foto de perfil.	Funcionalidad de cierre de sesión añadida a la foto de perfil.
	Mejorar la visibilidad del nombre del proyecto y la tarea padre en el formulario de crear y actualizar subtarea.	Reubicación del nombre del proyecto y la tarea padre en el formulario.
	Cambiar el término "Finaliza" en el tablero.	Texto cambiado a "Por finalizar".
	La primera opción de diseño es más limpia y minimalista.	N/A
Segunda prueba	Añadir un mensaje de confirmación al iniciar el flujo de una tarea.	Mensaje de confirmación añadido.
	Mostrar el nombre del responsable al pasar el cursor sobre su avatar.	Nombre del responsable mostrado al pasar el cursor sobre el avatar.
	Incluir un porcentaje de avance de las tareas tanto en el backlog como en el tablero.	Porcentaje de avance añadido en el backlog y en el tablero. También se añadió el porcentaje de avance la lista de proyectos.
	Añadir descripciones de las tareas y, además, añadir la opción para subir comentarios y archivos adjuntos a las tareas.	Ventana de información detallada de las tareas, opción para subir comentarios y archivos adjuntos añadidos.

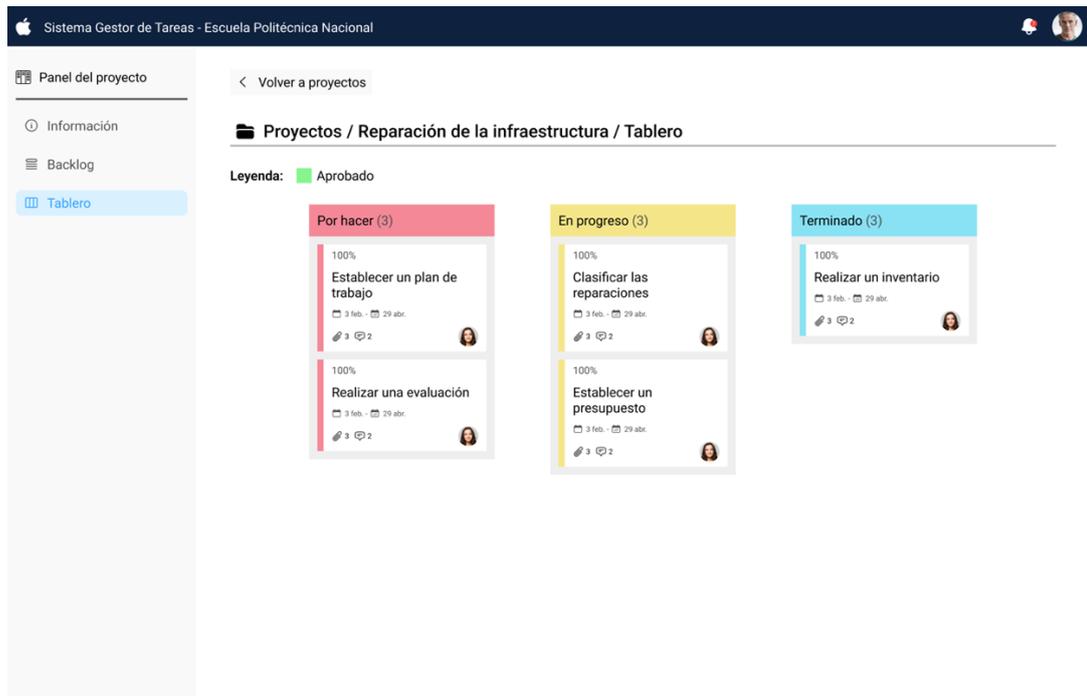
	Introducir un estado "Pendiente" al mover una tarea de "En progreso" a "Terminado".	Estado "Pendiente" añadido.
Tercera prueba	Añadir leyendas para archivos adjuntos, comentarios y lista de verificación en el backlog y el tablero.	Leyendas añadidas en el backlog y el tablero.
	Incluir un ícono de campana para las notificaciones.	Ícono de campana añadido para notificaciones.
	Mostrar el nombre del responsable en la información detallada de la tarea en lugar del avatar.	Nombre del responsable mostrado en lugar del avatar.
	Poner leyenda del ícono de hoja en el tablero, cambiarlo de ubicación o eliminarlo.	Ícono de hoja retirado.
	Reubicar la lista de verificación para que sea visible sin necesidad de desplazarse.	Lista de verificación reubicada alado de la información detallada de la tarea.
Cuarta prueba	Modificar el ingreso al sistema para que sea desde la columna de acciones mediante un ícono.	Ingreso al sistema modificado para usar un ícono en la columna de acciones.
	Mover las subtareas a una navegación entre pestañas para mejor visibilidad.	Subtareas movidas a una navegación entre pestañas.
	Eliminar la lista de verificación.	Lista de verificación eliminada.
Quinta prueba	Hacer más intuitiva la opción para agregar etiquetas en la vista de tareas.	N/A

	Añadir un botón para guardar cambios en la información detallada de la tarea.	N/A
	Reubicar la opción de ver elementos archivados para que sea más visible.	N/A
	Mostrar primero los comentarios y luego los archivos en la información detallada de la tarea.	Orden de visualización ajustado para mostrar primero los comentarios y luego los archivos.

Las retroalimentaciones recibidas durante las pruebas de los prototipos tuvieron un impacto significativo en la mejora y refinamiento del diseño y la funcionalidad de la aplicación. Para ejemplificar, se presenta uno de los cambios más relevantes realizados en los prototipos: con la retroalimentación obtenida, se pasó de lo que se muestra en la **Figura 3.1** a lo que se muestra en la **Figura 3.2**.



**Figura 3.1.** Prototipo del Tablero Kanban en la Primera Prueba



**Figura 3.2.** Prototipo Final Obtenido del Tablero Kanban

También es importante mencionar que, durante la revisión del Sprint, se tomaron cuatro decisiones significativas en el diseño de los prototipos, independientemente de las pruebas realizadas al PO:

- Eliminar la opción de cambiar la prioridad de los proyectos.
- Eliminar las etiquetas y elementos archivados.
- Añadir la funcionalidad de aprobación y rechazo de tareas.
- Incluir las subtareas en la información detallada de cada tarea. Esto, dado que no se pudo implementar el acordeón en el backlog del GALA, presentado en los prototipos iniciales.

## 3.2 Conclusiones

En esta sección se presentan las conclusiones del trabajo de integración curricular, en relación con los objetivos específicos del proyecto.

**En relación con el objetivo específico “1. Analizar los requisitos de usuario”**, se identificaron cuatro perfiles Persona clave: Gestor Empresarial, Gestor de Área/Líder de Área, Técnico y Administrador del Sistema. Cada perfil tiene funciones y necesidades específicas que fueron detalladas y documentadas en la sección 2.2. El perfil Persona del Administrador del Sistema no fue implementado en el aplicativo.

**En relación con el objetivo específico “2. Diseñar prototipos de interfaz de usuario”**, se crearon mapas de navegación detallados para cada perfil Persona. Utilizando la herramienta de Figma, se desarrollaron prototipos de alta fidelidad que incluyeron todas las funcionalidades necesarias para cada perfil Persona. Estos prototipos se basaron en principios de Diseño Centrado en el Usuario (DCU) y Atomic Design (AD), asegurando una interfaz de usuario intuitiva, efectiva y consistente.

**En relación con el objetivo específico “3. Evaluar los prototipos”**, los prototipos fueron evaluados mediante pruebas de usabilidad con el PO, lo que permitió identificar y corregir problemas de navegación, accesibilidad y funcionalidad. Además, la retroalimentación de los usuarios fue fundamental para realizar ajustes y mejoras en los prototipos, asegurando que la interfaz final cumpliera con las expectativas y necesidades de los usuarios.

**En relación con el objetivo específico “4. Implementar los prototipos”**, la implementación del Front-End se realizó utilizando tecnologías modernas como Next.js, TypeScript y Tailwind CSS, lo que permitió crear una aplicación web robusta, escalable y fácil de mantener. La estructura modular basada en Atomic Design (AD) facilitó la reutilización de componentes y aceleró el proceso de desarrollo, manteniendo la consistencia visual y funcional en toda la aplicación.

### **3.3 Recomendaciones**

En esta sección se detallan las recomendaciones basadas en el conocimiento y la experiencia acumulada a lo largo del desarrollo del proyecto.

- Comprender quiénes son los usuarios y cuáles son sus necesidades específicas es esencial para crear una interfaz intuitiva y efectiva. Esta recomendación es especialmente relevante en el contexto de Diseño Centrado en el Usuario (DCU), como en este proyecto, donde los perfiles Persona guían tanto el diseño como el desarrollo de la aplicación, asegurando que se satisfagan las expectativas y requerimientos del usuario final.

- Es fundamental definir con precisión las tecnologías que se emplearán en el desarrollo del proyecto desde las primeras etapas. Esto permite alinear al equipo, optimizar la planificación y evitar futuros contratiempos.
- Una estructura bien definida del código front-end desde el principio es esencial para facilitar el mantenimiento y la escalabilidad del proyecto. Se recomienda utilizar metodologías como Atomic Design, que permite organizar los componentes de manera modular y reutilizable, promoviendo la consistencia y eficiencia en el desarrollo. Asegurarse de seguir buenas prácticas en la organización y documentación del código también es crucial para la colaboración dentro del equipo y la sostenibilidad del proyecto a largo plazo.
- Es esencial tener un entendimiento profundo de la metodología que se utilizará en el desarrollo del proyecto. Este conocimiento permite una implementación efectiva de prácticas y procesos, asegurando que el equipo trabaje de manera organizada hacia el cumplimiento de los objetivos del proyecto.
- Mantener un ciclo continuo de retroalimentación y mejoras mediante pruebas de usabilidad periódicas. Para asegurar la calidad y repetibilidad de las pruebas, es esencial establecer un protocolo claro y detallado. Además, se recomienda grabar las sesiones de pruebas de usabilidad, proporcionando evidencia visual y auditiva que permita una revisión más exhaustiva. Estas grabaciones ayudan a identificar problemas y oportunidades de mejora que pueden no ser evidentes durante las pruebas en vivo.
- Una comunicación clara y transparente con los miembros del equipo es fundamental para asegurar que todos estén al tanto del progreso, los desafíos y las decisiones importantes. Esto facilita la colaboración efectiva, la resolución de problemas y el mantenimiento de un ambiente de trabajo cohesivo y productivo.
- No comprometerse a más trabajo del que se puede completar en un Sprint. Es vital mantener un enfoque realista y manejable para asegurar la calidad y el cumplimiento de los objetivos dentro del tiempo estipulado.

## 4 REFERENCIAS BIBLIOGRÁFICAS

- [1] C. Liz Fraile, “Gestor de tareas: eTask.” 2015. Accessed: May 13, 2024. [Online]. Available: <https://hdl.handle.net/10016/25279>
- [2] “¿Por qué necesitas un gestor de tareas para tus proyectos? | Slack.” Accessed: May 15, 2024. [Online]. Available: <https://slack.com/intl/es-es/blog/productivity/gestor-de-tareas-por-que-necesitas-para-tus-proyectos>
- [3] X. M. Jimenez Nureña, “Planificación operativa y la gestión administrativa del instituto Fe y Alegría 57 de Trujillo,” *Repositorio Institucional - UCV*, 2021, Accessed: May 13, 2024. [Online]. Available: <https://repositorio.ucv.edu.pe/handle/20.500.12692/55545>
- [4] E. Brechner, *Agile Project Management with Kanban*. 2015. Accessed: May 08, 2024. [Online]. Available: [https://www.google.com.pe/books/edition/Agile\\_Project\\_Management\\_with\\_Kanban/eEzTBgAAQBAJ?hl=es-419&gbpv=0](https://www.google.com.pe/books/edition/Agile_Project_Management_with_Kanban/eEzTBgAAQBAJ?hl=es-419&gbpv=0)
- [5] H. Kniberg and M. Skarin, *Kanban and Scrum-making the most of both*, vol. 1. 2010.
- [6] L. Castellano Lendínez, “Kanban. Metodología para aumentar la eficiencia de los procesos,” *3c Tecnología: glosas de innovación aplicadas a la pyme*, ISSN-e 2254-4143, Vol. 8, Nº. 1, 2019, págs. 30-41, vol. 8, no. 1, pp. 30–41, 2019, doi: 10.17993/3ctecno/2019.
- [7] M. Darío, A. Serna, L. Felipe, C. Zapata, J. Andrés, and Z. Cortes, “Mejoramiento de procesos de manufactura utilizando Kanban,” *Revista Ingenierías Universidad de Medellín*, vol. 14, no. 27, pp. 221–233, 2015, Accessed: May 11, 2024. [Online]. Available: [http://www.scielo.org.co/scielo.php?script=sci\\_arttext&pid=S1692-33242015000200014&lng=en&nrm=iso&tIng=es](http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1692-33242015000200014&lng=en&nrm=iso&tIng=es)
- [8] S. Krug, *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*. in *Voices That Matter*. Pearson Education, 2013. [Online]. Available: <https://books.google.com.ec/books?id=QlduAgAAQBAJ>
- [9] “Software para gestión de proyectos • Asana.” Accessed: May 19, 2024. [Online]. Available: <https://asana.com/es/uses/project-management>
- [10] “Qué es Trello: descubre sus funciones, usos y todo lo que ofrece | Trello.” Accessed: May 19, 2024. [Online]. Available: <https://trello.com/es/tour>
- [11] “Bienvenido a Jira | Atlassian.” Accessed: May 19, 2024. [Online]. Available: <https://www.atlassian.com/es/software/jira/guides/getting-started/introduction#what-is-jira-software>
- [12] J. Martins, “¿Qué es la metodología Kanban y cómo funciona? [2024] • Asana.” Accessed: May 13, 2024. [Online]. Available: <https://asana.com/es/resources/what-is-kanban>

- [13] L. Ramírez, “¿Qué es el Lean Manufacturing o producción ajustada?,” Thinking for Innovation. Accessed: May 18, 2024. [Online]. Available: <https://www.iebschool.com/blog/que-es-lean-manufacturing-negocios-internacionales/>
- [14] M. Bermejo, “El Kanban,” *Barcelona, España: UOC*, vol. 8, 2011.
- [15] “What is Kanban? Learn About its Origins, Properties and Principles.” Accessed: May 14, 2024. [Online]. Available: <https://kanbanzone.com/resources/kanban/>
- [16] M. G. Domingo and E. M. Pera, “Diseño centrado en el usuario,” *Universitat Oberta de Catalunya*, pp. 9–12, 2010.
- [17] “ISO 9241-210:2019 - Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems.” Accessed: May 19, 2024. [Online]. Available: <https://www.iso.org/standard/77520.html>
- [18] B. Frost, *Atomic design*. Brad Frost Pittsburgh, 2016.
- [19] C. Frisoli, “Qué es un sistema de diseño y qué elementos lo componen.” Accessed: May 25, 2024. [Online]. Available: <https://blog.hubspot.es/website/sistema-de-diseno>
- [20] “Docs | Next.js.” Accessed: May 11, 2024. [Online]. Available: <https://nextjs.org/docs>
- [21] M. Thakkar, “Next.js,” *Building React Apps with Server-Side Rendering*, pp. 93–137, 2020, doi: 10.1007/978-1-4842-5869-9\_3.
- [22] N. Carroll, “SEARCH ENGINE OPTIMIZATION,” *CBU International Conference Proceedings*, vol. 3, pp. 506–510, Sep. 2015, doi: 10.12955/CBUP.V3.645.
- [23] “TypeScript: JavaScript With Syntax For Types.” Accessed: May 18, 2024. [Online]. Available: <https://www.typescriptlang.org/>
- [24] G. Bierman, M. Abadi, and M. Torgersen, “Understanding TypeScript,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8586 LNCS, pp. 257–281, 2014, doi: 10.1007/978-3-662-44202-9\_11.
- [25] “Tailwind CSS - Rapidly build modern websites without ever leaving your HTML.” Accessed: May 18, 2024. [Online]. Available: <https://tailwindcss.com/>
- [26] Ivaylo. Gerchev, “Tailwind CSS,” 2022.
- [27] “Installation - Tailwind CSS.” Accessed: May 18, 2024. [Online]. Available: <https://tailwindcss.com/docs/installation>
- [28] F. Staiano, “Designing and prototyping interfaces with Figma : learn essential UX/UI design principles by creating interactive prototypes for mobile, tablet, and desktop”.
- [29] K. Schwaber and J. Sutherland, “La guía de Scrum,” *Scrumguides. Org*, vol. 1, p. 21, 2013.

- [30] "What is a Product Vision | Scrum.org." Accessed: May 15, 2024. [Online]. Available: <https://www.scrum.org/resources/what-product-vision>
- [31] "What is a Definition of Done? | Scrum.org." Accessed: May 15, 2024. [Online]. Available: <https://www.scrum.org/resources/what-definition-done>
- [32] K. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. in Addison-Wesley signature series. Addison-Wesley, 2012. [Online]. Available: <https://books.google.com.ec/books?id=HkXX65VCZU4C>

## 5 ANEXOS

### ANEXO I. Detalle de Sprints e Historias de Usuario Asociadas

Sprint	Historia de Usuario (HU)	Descripción de la HU
1	HU-01 Crear un Nuevo Tablero Kanban	<b>Como</b> usuario, <b>quiero</b> poder crear un nuevo tablero Kanban desde cero, proporcionando un nombre, descripción y estableciendo configuraciones básicas como columnas iniciales <b>para</b> organizar y gestionar mejor mis tareas.
	HU-02 Editar Configuración del Tablero Kanban	<b>Como</b> usuario, <b>quiero</b> poder editar la configuración de un tablero existente, incluyendo la adición, modificación o eliminación de columnas, la actualización de la descripción y otras preferencias, <b>para</b> mantener mi tablero actualizado y alineado con los cambios en mis tareas.
2	HU-01 Crear un Nuevo Tablero Kanban	<b>Como</b> usuario, <b>quiero</b> poder crear un nuevo tablero Kanban desde cero, proporcionando un nombre, descripción y estableciendo configuraciones básicas como columnas iniciales <b>para</b> organizar y gestionar mejor mis tareas.
	HU-02 Editar Configuración del Tablero Kanban	<b>Como</b> usuario, <b>quiero</b> poder editar la configuración de un tablero existente, incluyendo la adición, modificación o eliminación de columnas, la actualización de la descripción y otras preferencias, <b>para</b> mantener mi tablero actualizado y alineado con los cambios en mis tareas.
3	HU-03 Establecer Permisos de Acceso al Tablero	<b>Como</b> administrador, <b>quiero</b> tener control sobre quién puede acceder y realizar cambios en los tableros Kanban <b>para</b> definir perfiles y permisos específicos.

	HU-04 Gestión de sesión del usuario	<b>Como</b> usuario, <b>quiero</b> tener una interfaz de inicio y cierre de sesión <b>para</b> acceder y salir de mi cuenta de manera eficiente.
4	HU-05 Gestión de tareas GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> tener la capacidad de visualizar, crear, editar y eliminar tareas <b>para</b> controlar eficazmente el flujo de trabajo y asegurar el cumplimiento de los objetivos del proyecto.
5	HU-06 Visualizar los proyectos	<b>Como</b> un Gestor Empresarial, <b>quiero</b> visualizar el listado de proyectos <b>para</b> conocer los proyectos de la institución.
6	HU-05 Gestión de tareas GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> tener la capacidad de visualizar, crear, editar y eliminar tareas <b>para</b> controlar eficazmente el flujo de trabajo y asegurar el cumplimiento de los objetivos del proyecto.
	HU-07 Inicio del Flujo de una tarea GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> poder iniciar el flujo de trabajo de una tarea asignada a un Gestor de Área <b>para</b> que este pueda visualizarla y comenzar a trabajar en ella.
7	HU-05 Gestión de tareas GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> tener la capacidad de visualizar, crear, editar y eliminar tareas <b>para</b> controlar eficazmente el flujo de trabajo y asegurar el cumplimiento de los objetivos del proyecto.
	HU-06 Visualizar los proyectos	<b>Como</b> un Gestor Empresarial, <b>quiero</b> visualizar el listado de proyectos <b>para</b> conocer los proyectos de la institución.
	HU-07 Inicio del Flujo de una tarea GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> poder iniciar el flujo de trabajo de una tarea asignada a un Gestor de Área <b>para</b> que este pueda visualizarla y comenzar a trabajar en ella.

	HU-08 Asignación de un responsable a una tarea	<b>Como</b> supervisor de un equipo, <b>quiero</b> poder asignar personal responsable a las tareas <b>para</b> asegurarme de que cada tarea tenga un encargado claro y así facilitar el seguimiento del progreso de las tareas asignadas a mis subordinados.
8	N/A	N/A
9	HU-07 Inicio del Flujo de una tarea GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> poder iniciar el flujo de trabajo de una tarea asignada a un Gestor de Área <b>para</b> que este pueda visualizarla y comenzar a trabajar en ella.
	HU-08 Asignación de un responsable a una tarea	<b>Como</b> supervisor de un equipo, <b>quiero</b> poder asignar personal responsable a las tareas <b>para</b> asegurarme de que cada tarea tenga un encargado claro y así facilitar el seguimiento del progreso de las tareas asignadas a mis subordinados.
	HU-09 Asignación de fechas	<b>Como</b> Gestor Empresarial (GE), Gestor de Área (GA), <b>quiero</b> asignar las fechas de inicio y de fin de una tarea, y de un proyecto en el caso del GE <b>para</b> optimizar la planificación y el seguimiento del cronograma del proyecto.
10	HU-08 Asignación de un responsable a una tarea	<b>Como</b> supervisor de un equipo, <b>quiero</b> poder asignar personal responsable a las tareas <b>para</b> asegurarme de que cada tarea tenga un encargado claro y así facilitar el seguimiento del progreso de las tareas asignadas a mis subordinados.
	HU-09 Asignación de fechas	<b>Como</b> Gestor Empresarial (GE), Gestor de Área (GA), <b>quiero</b> asignar las fechas de inicio y de fin de una tarea, y de un proyecto en el caso del GE <b>para</b> optimizar la planificación y el seguimiento del cronograma del proyecto.

	HU-10 Filtros de tareas GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> filtrar las tareas por nombre, responsable y estado, y ordenarlas ascendente y descendientemente según sus características <b>para</b> analizar las tareas de manera eficiente y organizada
	HU-11 Gestión de proyectos	<b>Como</b> Gestor Empresarial, <b>quiero</b> tener la capacidad de gestionar los proyectos de la empresa <b>para</b> asegurarme de que las actividades y objetivos empresariales se cumplan.
	HU-12 Filtros de proyectos GE	<b>Como</b> Gestor Empresarial, <b>quiero</b> buscar los proyectos por nombre; y ordenar ascendente y descendientemente sus características <b>para</b> analizar los proyectos de manera eficiente y organizada.
	HU-13 Filtros de tareas GA	<b>Como</b> Gestor de Área (GA), <b>quiero</b> tener la capacidad de filtrar las tareas por nombre, estado y proyecto; y ordenar las características de las tareas de forma ascendente y descendente <b>para</b> identificar rápidamente tareas específicas entre todas las tareas disponibles.
	HU-14 Refinación de tareas GA	<b>Como</b> Gestor de Área (GA), <b>quiero</b> refinar las tareas asignadas <b>para</b> descomponerlas en tareas más específicas y manejables.
11	HU-10 Filtros de tareas GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> filtrar las tareas por nombre, responsable y estado, y ordenarlas ascendente y descendientemente según sus características <b>para</b> analizar las tareas de manera eficiente y organizada
	HU-11 Gestión de proyectos	<b>Como</b> Gestor Empresarial, <b>quiero</b> tener la capacidad de gestionar los proyectos de la empresa <b>para</b> asegurarme de que las actividades y objetivos empresariales se cumplan.

	HU-14 Refinación de tareas GA	<b>Como</b> Gestor de Área (GA), <b>quiero</b> refinar las tareas asignadas <b>para</b> descomponerlas en tareas más específicas y manejables.
	HU-15 Inicio del Flujo de una tarea GA	<b>Como</b> Gestor de Área (LA), <b>quiero</b> poder iniciar el flujo de trabajo de una tarea asignada a un Líder de Área, <b>para</b> que mi subordinado pueda visualizarla y comenzar a ejecutarla.
12	HU-10 Filtros de tareas GE	<b>Como</b> Gestor Empresarial (GE), <b>quiero</b> filtrar las tareas por nombre, responsable y estado, y ordenarlas ascendente y descendente según sus características <b>para</b> analizar las tareas de manera eficiente y organizada
	HU-13 Filtros de tareas GA	<b>Como</b> Gestor de Área (GA), <b>quiero</b> tener la capacidad de filtrar las tareas por nombre, estado y proyecto; y ordenar las características de las tareas de forma ascendente y descendente <b>para</b> identificar rápidamente tareas específicas entre todas las tareas disponibles.
	HU-14 Refinación de tareas GA	<b>Como</b> Gestor de Área (GA), <b>quiero</b> refinar las tareas asignadas <b>para</b> descomponerlas en tareas más específicas y manejables.
	HU-15 Inicio del Flujo de una tarea GA	<b>Como</b> Gestor de Área (LA), <b>quiero</b> poder iniciar el flujo de trabajo de una tarea asignada a un Líder de Área, <b>para</b> que mi subordinado pueda visualizarla y comenzar a ejecutarla.
13	HU-16 Refinación de tareas LA	<b>Como</b> Líder de Área (LA), <b>quiero</b> descomponer las tareas asignadas <b>para</b> dividir las en subtareas más específicas y manejables.
	HU-17 Filtros de tareas LA	<b>Como</b> Líder de Área (LA), <b>quiero</b> tener la capacidad de filtrar las tareas por nombre, estado y proyecto; y ordenar las características de las tareas de forma ascendente y descendente <b>para</b>

		identificar rápidamente tareas específicas entre todas las tareas disponibles.
	HU-18 Inicio del Flujo de Tareas LA	<b>Como</b> Líder de Área (LA), <b>quiero</b> poder iniciar el flujo de trabajo de una tarea asignada a un Técnico, <b>para</b> que mi subordinado pueda visualizarla y comenzar a ejecutarla.
14	HU-19 Gestión de Tareas para Técnicos en el Tablero Kanban	<b>Como</b> Técnico, <b>quiero</b> poder arrastrar y soltar mis tareas en el tablero Kanban <b>para</b> moverlas a través de su flujo, de manera que pueda gestionar el estado y progreso de mis tareas.
	HU-20 Información Detallada de Tareas Técnico	<b>Como</b> Técnico, <b>quiero</b> visualizar la información detallada de una tarea en el tablero Kanban <b>para</b> obtener una perspectiva más completa de su estado, proyecto, fechas y descripción, de manera que pueda entender mejor su contexto y prioridad.
	HU-21 Gestión de Comentarios	<b>Como</b> usuario, <b>quiero</b> poder agregar, editar, eliminar y visualizar comentarios en las tareas del tablero <b>para</b> facilitar la comunicación y seguimiento dentro del equipo de trabajo.
	HU-22 Gestión de Archivos	<b>Como</b> usuario, <b>quiero</b> poder agregar, editar, eliminar y visualizar archivos adjuntos en las tareas del tablero Kanban <b>para</b> gestionar los documentos relacionados con las tareas.
	HU-23 Sistema de Notificaciones	<b>Como</b> usuario, <b>quiero</b> recibir notificaciones cuando se suba un archivo, se agregue un comentario o se cambie el estado de una tarea <b>para</b> estar al tanto de las actualizaciones importantes y poder supervisar adecuadamente el progreso de mis subordinados.

## ANEXO II. Prototipos del Sistema

Este anexo incluye el enlace que proporciona acceso a los prototipos del sistema.

Enlace: <https://www.figma.com/proto/AckndTIIvt0N2IsVch9XyY/Kanban?node-id=530-671&t=74EMZtHSyPxMZqeN-0&scaling=contain&content-scaling=fixed&page-id=248%3A22&starting-point-node-id=530%3A671>

## ANEXO III. Guion Ejemplificado para las Pruebas

Quiero que te sientas completamente libre de expresar tus ideas, comentarios o sugerencias relacionadas con el diseño y las funcionalidades de la aplicación. También te agradecería que mencionaras en voz alta cada paso que realizas, lo que ves en la pantalla y, si es necesario, lo que introducirías en cada campo para completar la actividad asignada.

- **Objetivo 1:** Acceder y navegar en el sistema
  - **Tarea 1:** Inicia sesión en el sistema como gestor empresarial.
- **Objetivo 2:** Gestionar proyectos
  - **Tarea 2:** Crea un proyecto llamado “Rediseño de la malla curricular” con la descripción “Reclutamiento y contratación de nuevo personal en la Facultad”, con fecha de inicio el 17 de febrero de 2024 y fecha de fin el 11 de mayo de 2024.
  - **Tarea 3:** Actualiza el proyecto “Adquisición de accesorios” cambiando el nombre a “Adquisición de equipos”.
  - **Tarea 4:** Elimina el proyecto llamado “Contratación de personal”.
- **Objetivo 3:** Visualizar detalles del proyecto
  - **Tarea 5:** Ingresa al proyecto “Reparación de la infraestructura” y observa la información del proyecto.
- **Objetivo 4:** Gestionar tareas en el backlog del proyecto

- **Tarea 6:** Dirígete al backlog del proyecto “Reparación de la infraestructura” y crea una nueva tarea llamada “Identificar y contratar proveedores,” asignando a “Alejandro Díaz” como responsable.
- **Tarea 7:** Actualiza la tarea “Establecer un plan de trabajo,” cambiando el responsable de “Roberto Sánchez” a “Ana Pérez”.
- **Tarea 8:** Elimina la tarea llamada “Realizar un inventario”.
- **Tarea 9:** Inicia el flujo de trabajo de la tarea “Identificar y contratar proveedores”.
- **Objetivo 5:** Revisar la información detallada de una tarea en el tablero del proyecto
  - **Tarea 10:** Dirígete al tablero del proyecto “Reparación de la infraestructura” e ingresa a la tarea “Establecer un presupuesto”.
- **Objetivo 6:** Finalizar la sesión
  - **Tarea 11:** Cierra la sesión de la cuenta.

#### **ANEXO IV. Pruebas de Usabilidad**

Primera prueba: <https://youtu.be/CowqyWlQi10>

Segunda prueba: <https://youtu.be/u9nAtudzmiw>

Tercera prueba: <https://youtu.be/7PB-FzZVDq0>

Cuarta prueba: sin grabación

Quinta prueba: [https://youtu.be/c8PbJy05\\_JA](https://youtu.be/c8PbJy05_JA)