



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Software

**Rediseño UX/UI de un sistema de gestión de proyectos
y tareas mediante Design Thinking y Atomic Design
para mejorar la gestión del equipo de
desarrollo y la racionalización de recursos de la
empresa LatamReady (2023)**

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el Título Profesional de Ingeniero de Software

AUTOR

Andrés Leonardo SATORNICIO MEDINA

ASESOR

José César PIEDRA ISUSQUI

Lima, Perú

2025



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Satornicio, A. (2025). *Rediseño UX/UI de un sistema de gestión de proyectos y tareas mediante Design Thinking y Atomic Design para mejorar la gestión del equipo de desarrollo y la racionalización de recursos de la empresa LatamReady (2023)*. [Trabajo de Suficiencia Profesional de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Software]. Repositorio institucional Cybertesis UNMSM.

Metadatos complementarios

Datos de autor	
Nombres y apellidos	Andrés Leonardo Satornicio Medina
Tipo de documento de identidad	DNI
Número de documento de identidad	70003980
URL de ORCID	https://orcid.org/0009-0002-9093-6648
Datos de asesor	
Nombres y apellidos	José Cesar Piedra Isusqui
Tipo de documento de identidad	DNI
Número de documento de identidad	25628915
URL de ORCID	https://orcid.org/0000-0003-3549-172X
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Cesar Augusto Angulo Calderón
Tipo de documento	DNI
Número de documento de identidad	32907109
Miembro del jurado 1	
Nombres y apellidos	Armando David Espinoza Robles
Tipo de documento	DNI
Número de documento de identidad	08633326
Datos de investigación	
Línea de investigación	No aplica
Grupo de investigación	No aplica
Agencia de financiamiento	Sin financiamiento

Ubicación geográfica de la investigación	País: Perú Departamento: Lima
	Provincia: Lima Distrito: Cercado de Lima Jr. Carlos Amezaga No. 375 Universidad Nacional Mayor de San Marcos Latitud: -12.0564232 Longitud: -77.0843327
Año o rango de años en que se realizó la investigación	2024
URL de disciplinas OCDE	2.02.04 -- Ingeniería de sistemas y comunicaciones https://purl.org/pe-repo/ocde/ford#2.02.04



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
Escuela Profesional de Ingeniería de Software

Acta Virtual de Sustentación
del Trabajo de Suficiencia Profesional

Siendo las 18:00 horas del día 10 de enero del año 2025, se reunieron virtualmente los docentes designados como Miembros de Jurado del Trabajo de Suficiencia Profesional, presidido por el Mg. Angulo Calderón César Augusto (Presidente), Lic. Espinoza Robles Armando David (Miembro) y el Mg. Piedra Isusqui José César (Miembro Asesor), usando la plataforma Meet (<https://meet.google.com/jtg-ystx-bsz>), para la sustentación virtual del Trabajo de Suficiencia Profesional intitulado: **“REDISEÑO UX/UI DE UN SISTEMA DE GESTIÓN DE PROYECTOS Y TAREAS MEDIANTE DESIGN THINKING Y ATOMIC DESIGN PARA MEJORAR LA GESTIÓN DEL EQUIPO DE DESARROLLO Y LA RACIONALIZACIÓN DE RECURSOS DE LA EMPRESA LATAMREADY (2023)”**, por el Bachiller **Satornicio Medina Andrés Leonardo**; para obtener el Título Profesional de Ingeniero de Software.

Acto seguido de la exposición del Trabajo de Suficiencia Profesional, el Presidente invitó al Bachiller a dar las respuestas a las preguntas establecidas por los miembros del Jurado.

El Bachiller en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez a las observaciones y preguntas formuladas por los señores miembros del Jurado.

Finalmente habiendo efectuado la calificación correspondiente por los miembros del Jurado, el Bachiller obtuvo la nota de **17 (DIECISIETE)**.

A continuación el Presidente de Jurados el Mg. Angulo Calderón César Augusto, declara al Bachiller **Ingeniero de Software**.

Siendo las 18.55 horas, se levantó la sesión.

Presidente

Mg. Angulo Calderón César Augusto

Miembro

Mg. Espinoza Robles Armando David

Miembro Asesor

Mg. Piedra Isusqui José César



CERTIFICADO DE SIMILITUD

Yo, PIEDRA ISUSQUI JOSÉ CÉSAR en mi condición de asesor acreditado con la Resolución Decanal N° 001059-2024-D-FISI/UNMSM y Oficio N° 0023-PTTSP-FISI-2024 del Trabajo de Suficiencia Profesional cuyo título es Rediseño UX/UI de un sistema de gestión de proyectos y tareas mediante Design Thinking y Atomic Design para mejorar la gestión del equipo de desarrollo y la racionalización de recursos de la empresa LatamReady (2023), presentado por el bachiller SATORNICIO MEDINA ANDRÉS LEONARDO para optar el título de Ingeniero de Software CERTIFICO que se ha cumplido con lo establecido en la Directiva de Originalidad y de Similitud de Trabajos Académicos, de Investigación y Producción Intelectual. Según la revisión, análisis y evaluación mediante la plataforma PIDS, el documento evaluado cuenta con el 10% de similitud, nivel **PERMITIDO** para continuar con los trámites correspondientes y para su **publicación en el repositorio institucional**.

Se emite el presente certificado en cumplimiento de lo establecido en las normas vigentes, como uno de los requisitos para la obtención del título.

Lima, 8 de enero de 2025

Firma del Asesor

Nombres y Apellidos del asesor: PIEDRA ISUSQUI JOSÉ CÉSAR

DNI 25628915

FICHA CATALOGRÁFICA

REDISEÑO UX/UI DE UN SISTEMA DE GESTIÓN DE PROYECTOS Y TAREAS
MEDIANTE DESIGN THINKING Y ATOMIC DESIGN PARA MEJORAR LA
GESTIÓN DEL EQUIPO DE DESARROLLO Y LA RACIONALIZACIÓN DE
RECURSOS DE LA EMPRESA LATAMREADY (2023)

AUTOR: SATORNICIO MEDINA, ANDRES LEONARDO

ASESOR: PIEDRA ISUSQUI JOSÉ CESAR

TÍTULO PROFESIONAL: Título profesional de Ingeniero de Software

ÁREA/PROGRAMA/LÍNEA DE INVESTIGACIÓN:

Ingeniería / Industria, Innovación e Infraestructura / Ingeniería de Software

PREGRADO: Universidad Nacional Mayor de San Marcos – Facultad de Ingeniería de
Sistemas e informática – Escuela Profesional de Ingeniería de Software

Formato 21 X 29.7 cm

Páginas: xii,77

DEDICATORIA

A mis padres. Quienes en todo momento me
brindaron su apoyo incondicional y dedicación,
tanto en mi desarrollo personal como profesional.

AGRADECIMIENTOS

A mi asesor por su valioso apoyo y orientación a lo largo de todo el desarrollo de este trabajo de suficiencia profesional.

A la Universidad Nacional Mayor de San Marcos por brindarme las herramientas para convertirme en un profesional de éxito.

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA DE SOFTWARE

Rediseño UX/UI de un sistema de gestión de proyectos y tareas mediante Design Thinking y Atomic Design para mejorar la gestión del equipo de desarrollo y la racionalización de recursos de la empresa LatamReady (2023)

Autor: Satornicio Medina, Andrés Leonardo

Asesor: Piedra Isusqui, José Cesar

Título: Trabajo de Suficiencia Profesional para optar el Título de Ingeniero de Software

Fecha: Diciembre 2024

RESUMEN

El presente trabajo de suficiencia profesional se centra en el rediseño UX/UI de un sistema de gestión de proyectos y tareas llamado Smartready, destinado a mejorar el flujo de trabajo del equipo de desarrollo de la empresa LatamReady. La motivación principal para este rediseño radica en la carencia de un software de gestión que satisfaga las necesidades específicas del equipo. El objetivo principal de este proyecto es rediseñar Smartready para convertirlo en un sistema que mejore la gestión de tareas y proyectos, alineándose con los flujos de trabajo establecidos del equipo y que, además, sea amigable, fácil de usar y cumpla con los requerimientos de sus usuarios.

El rediseño de este sistema se llevó a cabo utilizando la metodología Design Thinking, apoyada por los principios de la metodología Atomic Design para la construcción de los componentes. Todo esto dentro de un marco de gestión ágil de proyectos utilizando Scrum.

Esto permitió no solo cumplir con las expectativas del equipo, sino también enfocarse en las necesidades del usuario, evaluar e iterar distintas soluciones y, finalmente, construir un sistema de calidad.

Como resultado, se obtuvo una mejora significativa en el diseño visual y en la usabilidad del sistema, así como una optimización de los procesos internos, lo que resultó en una mejora en la gestión del equipo y racionalización de recursos.

Palabras claves: Diseño UX, Diseño UI, Gestión de Proyectos, Gestión de tareas, Design Thinking, Atomic Design.

MAJOR NATIONAL UNIVERSITY OF SAN MARCOS
FACULTY OF SYSTEM AND INFORMATICS ENGINEERING
PROFESSIONAL SCHOOL OF SOFTWARE ENGINEERING

**UX/UI redesign of a project and task management system using Design Thinking
and Atomic Design to improve development team management and resource
rationalization of LatamReady (2023)**

Author: Satornicio Medina, Andrés Leonardo

Advisor: Piedra Isusqui, José Cesar

**Title: Professional Sufficiency Work to opt for the Professional Title of Software
Engineer**

Date: December 2024

ABSTRACT

This professional sufficiency work focuses on the UX/UI redesign of a project and task management system called Smartready, aimed at improving the workflow of the development team of the company LatamReady. The main motivation for this redesign lies in the lack of a management software that meets the specific needs of the team. The main objective of this project is to redesign Smartready to turn it into a system that improves task and project management, aligning it with the established workflows of the team and that, in addition, is friendly, easy to use and meets the requirements of its users.

The redesign of this system was carried out using the Design Thinking methodology, supported by the principles of the Atomic Design methodology for the construction of the components. All this within an agile project management framework using Scrum. This

allowed not only to meet the team's expectations, but also to focus on the user's needs, evaluate and iterate different solutions and, finally, build a quality system.

As a result, there was a significant improvement in the visual design and usability of the system, as well as an optimization of internal processes, resulting in improved team management and rationalization of resources.

Keywords: UX Design, UI Design, Project Management, Task Management, Design Thinking, Atomic Design.

ÍNDICE GENERAL

RESUMEN	v
ABSTRACT	vii
INDICE DE TABLAS	xi
INDICE DE FIGURAS	xii
INTRODUCCIÓN	1
CAPÍTULO I - TRAYECTORIA PROFESIONAL	3
1.1. Presentación Profesional.....	3
1.2. Educación y Certificaciones	3
1.3. Experiencia Laboral.....	4
CAPITULO II - CONTEXTO EN EL QUE SE DEARROLLÓ LA EXPERIENCIA.....	8
2.1. Empresa – Actividad que realiza	8
2.2. Misión de la empresa	8
2.3. Visión de la empresa	9
2.4. Organización de la empresa	9
2.5. Área, cargo y funciones desempeñadas	11
2.6. Experiencia profesional realizada en la organización.....	11
CAPITULO III – ACTIVIDADES DESARROLLADAS.....	13
3.1. Situación problemática	13

3.1.1. Problema general:	14
3.1.2. Problemas específicos:.....	14
3.2. Solución	15
3.2.1. Objetivos.....	15
3.2.2. Alcance.....	16
3.2.3. Etapas y Metodología	17
3.2.4. Fundamentos utilizados	22
3.2.5. Implementación de las áreas, procesos, sistemas y buenas practicas	27
3.3. Evaluación	67
3.3.1. Evaluación económica	67
3.3.2. Interpretación del VAN y TIR.....	68
CAPITULO IV - REFLEXIÓN CRITICA DE LA EXPERIENCIA.....	70
CAPITULO V – CONCLUSIONES Y RECOMENDACIONES	72
5.1. Conclusiones.....	72
5.2. Recomendaciones	72
5.3. Fuentes de información.....	73
5.4. Glosario.....	75

INDICE DE TABLAS

Tabla 1 Educación y certificaciones.....	3
Tabla 2 Experiencia Laboral	4
Tabla 3 Detalle de los sprints realizados durante el rediseño de Smartready	28
Tabla 4 Detalle de las tareas realizadas en el sprint 1	29
Tabla 5 Problemas encontrados de acuerdo con los principios heurísticos de Jakob Nielsen.....	30
Tabla 6 Detalle de las tareas realizadas en el sprint 2	37
Tabla 7 Detalle de las tareas realizadas en el sprint 3	46
Tabla 8 Detalle de las tareas realizadas en el sprint 4	49
Tabla 9 Detalle de las tareas realizadas en el sprint 5	52
Tabla 10 Detalle de las tareas realizadas en el sprint 6	58
Tabla 11 Detalle de las tareas realizadas en el sprint 7	62
Tabla 12 Detalle de las tareas realizadas en el sprint 8	64
Tabla 13 Costos por mes del personal a cargo del proyecto	67
Tabla 14 Costos por año de las herramientas utilizadas.....	68
Tabla 15 Cálculo del VAN	69

INDICE DE FIGURAS

Figura 1 Organigrama de la empresa LatamReady	9
Figura 2 Subáreas del área de Desarrollo	10
Figura 3 Análisis heurístico de Smartready	32
Figura 4 Ejemplo de wireframe de Smartready	35
Figura 5 Colores utilizados en el rediseño de Smartready.....	39
Figura 6 Íconos utilizados en el rediseño de Smartready	40
Figura 7 Átomos contruidos para el UI Kit.....	41
Figura 8 Detalle del componente de botón	42
Figura 9 Moléculas contruidas en el UI Kit	43
Figura 10 Detalle del componente de Tarjeta de notificaciones	44
Figura 11 Organismos contruidos en el UI Kit	45
Figura 12 Mockup del tablero de tareas	49
Figura 13 Mockups del drawer de detalle de tarea	52
Figura 14 Flujo de trabajo definido para la gestión de un issue en Smartready	55
Figura 15 Mockups de tablero de issue y modal de detalle de issue	57
Figura 16 Mockups para pruebas de jerarquía de la información.....	60
Figura 17 Mockup de carga masiva de tiempos	61
Figura 18 Mockups diseñados para el perfil del usuario	64
Figura 19 Mockup del timeline de tareas	66

INTRODUCCIÓN

En el mundo empresarial actual, la eficiencia y la productividad son factores clave para el éxito. Sin embargo, muchas empresas enfrentan desafíos significativos debido a la falta de herramientas tecnológicas adecuadas que permitan gestionar sus operaciones de manera eficiente. Un área crítica donde estos problemas pueden manifestarse es la gestión de proyectos y tareas. Cuando los equipos no cuentan con sistemas bien diseñados y optimizados para sus necesidades, los procesos se vuelven ineficientes, lo que incrementa costos y obstaculizan el correcto uso de los recursos.

Este trabajo de suficiencia profesional aborda estos problemas desde la perspectiva del diseño UX/UI, centrándose en el rediseño del sistema Smartready, utilizado por el equipo de desarrollo de LatamReady para la gestión de proyectos y tareas. Inicialmente, Smartready presentaba varias limitaciones en su experiencia de usuario y funcionalidades, lo que ralentizaba el flujo de trabajo de los empleados y los hacía depender de otras herramientas para completar sus tareas, incrementando los costos operativos de la empresa. Con esto en mente, el rediseño de Smartready se propuso abordar estas limitaciones y mejorar la experiencia del usuario.

El rediseño de Smartready, liderado desde el enfoque de UX/UI, tuvo como objetivo no solo mejorar la interfaz y la usabilidad del sistema, sino también mejorar los procesos internos de la empresa. A través de un proceso de diseño centrado en el usuario y la incorporación de nuevas funcionalidades, se logró aumentar el uso de la plataforma y reducir los costos operativos, mejorando la racionalización del equipo de desarrollo.

Para el correcto desarrollo de este proyecto se trabajó con la metodología Scrum, lo que permitió hacer entregables incrementales que aportaran valor. Adicionalmente, para establecer un diseño centrado en el usuario, se utilizó la metodología Design Thinking, que facilitó la identificación de las necesidades y problemas reales de los usuarios a través de la investigación y la experimentación. Asimismo, se aplicó la metodología Atomic Design para la creación y organización de los componentes del diseño, lo que permitió establecer una estructura escalable y consistente en la interfaz. Estos enfoques no solo permitieron mejorar la experiencia del usuario, sino que fueron clave para alcanzar los objetivos del equipo.

A continuación, se detallará la estructura del presente trabajo:

El CAPÍTULO I: Se describe mi trayectoria profesional y los conocimientos que he adquirido a lo largo de diversas experiencias laborales.

CAPÍTULO II: Se identifica el contexto en el que fue dada la experiencia y en el cual se desarrolló el proyecto.

CAPÍTULO III, Se explica el problema y la solución propuesta. También se mencionan las actividades desarrolladas para llegar a la solución.

El CAPÍTULO IV, se describe una reflexión crítica de la experiencia.

El CAPÍTULO V, se describen las conclusiones y recomendaciones del trabajo.

CAPÍTULO I - TRAYECTORIA PROFESIONAL

1.1. Presentación Profesional

El autor del presente trabajo tiene el grado de Bachiller en Ingeniería de Software. Actualmente, se desempeña como Diseñador UX/UI en la empresa ProntoPaga. Con más de 3 años de experiencia en el campo del diseño UX/UI y más de 2 años en el desarrollo de software.

1.2. Educación y Certificaciones

A continuación, En la Tabla 1 se detalla la educación y certificaciones del autor de este trabajo.

Tabla 1

Educación y certificaciones

Fecha	Descripción
2023	Digital House Curso de diseño UX
2022	Coder House Curso de diseño UX/UI
2022	Scrum Foundation Professional Certificate (SFPC)
2015-2020	Universidad Nacional Mayor de San Marcos Grado de Bachiller en la carrera de Ingeniería de Software

Nota. Fuente: Elaboración Propia

Adicional a ello cabe mencionar que el autor cuenta con un artículo científico llamado “Music Recommender System based on Sentiment Analysis Enhanced with Natural Language Processing Technics” publicado en el año 2023.

1.3.Experiencia Laboral

A continuación, en la Tabla 2 se detalla la información relacionada a la experiencia laboral del autor.

Tabla 2

Experiencia Laboral

ProntoPaga (abril 2024 - Presente)	
Cargo	Diseñador UX/UI
Funciones	- Análisis, propuesta y diseño de mejoras para el BackOffice de la empresa. - Diseño de una Landing Page que potencie el alcance de nuevos clientes y concrete ventas en la empresa. - Diseño de una plataforma Demo, para que los clientes potenciales naveguen por todos los métodos de pago con los que cuenta la empresa. - Rediseño de todo el flujo de pago que realiza un usuario mediante los distintos métodos de pago habilitados según el país en el que se encuentre. - Creación de librerías y componentes que facilite el flujo de trabajo del equipo de diseño.

-
- Diseño de la página web con el objetivo de obtener mayor número de clientes y una mejor experiencia de usuario.

LatamReady (julio 2022 – abril 2021)

Cargo Diseñador UX/UI

- Funciones
- Diseño de una plataforma para la gestión de proyectos y tareas del equipo de desarrollo, llamada Smartready, centrado el diseño en las necesidades de los usuarios.
 - Diseño de una plataforma para el manejo de proyectos para el equipo de consultoría, llamada Wormax, centrando el diseño en las necesidades de los usuarios.
 - Mejoras de flujo, de interacción en sistemas ya existentes, reduciendo pasos, eliminando inconsistencia y aumentando su usabilidad.
 - Creación de UI Kits y componentes que estén acordes a la línea gráfica de la empresa.

Kushki (enero 2022 – junio 2022)

Cargo Diseñador UI

- Funciones
- Diseño de mockups que cumplan con los objetivos de cada proyecto y requerimientos de los stakeholders, respetando el Sistema de diseño de la empresa.
 - Creación de prototipos que cumplan con los objetivos de cada proyecto definiendo también la interacción.
 - Propuesta y diseño de mejoras para el BackOffice de la empresa.

-
- Creación de librerías para mejorar el flujo de trabajo del equipo de diseño.
 - Uso de Figma a nivel Avanzado.

LatamReady (enero 2021 – diciembre 2021)

Cargo Front End Developer

- Funciones
- Diseño de interfaces para proyectos de uso interno (Wireframes y mockups) mediante el uso de Figma.
 - Propuesta y Desarrollo de soluciones que permitan tener conocimiento del estado actual de los clientes respecto al uso de las diferentes herramientas o modulo que se les proveen.
 - Desarrollo de herramientas externas mediante el uso de React JS.
 - Manejo de Servicios Web.
 - Manejo de APIs con Postman.

LatamReady (junio 2019 – diciembre 2020)

Cargo Development Trainee

- Funciones
- Diseño de interfaces para proyectos de uso interno mediante el uso de Adobe XD.
 - Desarrollo de soluciones para mejorar el flujo de trabajo interno.
 - Manejo de la ERP Netsuite y el lenguaje de programación Suitscript.
-

-
- Manejo del módulo Electronic Bank Payment y elaboración de archivos de transferencia de pagos de Bancos Latinoamericanos para comunicación entre la compañía y el proveedor.
-

Nota. Fuente: Elaboración Propia

CAPITULO II - CONTEXTO EN EL QUE SE DEARROLLÓ LA EXPERIENCIA

2.1. Empresa – Actividad que realiza

LatamReady es una empresa especializada en brindar soluciones de software que permitan a otras empresas cumplir con los requisitos fiscales y contables del país o países donde operan, principalmente ubicados en Latinoamérica.

Sus servicios se centran en la implementación del ERP Oracle NetSuite, con personalización adaptada a las necesidades específicas de cada empresa. Además, ofrece soporte técnico, consultoría, programas de formación y capacitación para garantizar su correcto uso. Estas soluciones ayudan a las empresas a cumplir con los requisitos fiscales de 18 países en América Latina y España, abarcando la gestión de impuestos, la generación de informes financieros y la automatización de procesos contables, entre otros.

Fundada en 2009 por Carlos Zumaeta Hernández, LatamReady tiene su sede principal en Miami y oficinas en Perú. La empresa cuenta con 150 empleados y colabora estrechamente con multinacionales que operan en América Latina y España. Su facturación anual asciende a 10 millones de dólares, y gestiona una cartera de más de 200 clientes en estas regiones. Además, ha sido reconocida por Oracle como uno de sus principales socios de implementación y ha sido nominada en varias ocasiones al premio Solution Provider Partner del Año de Oracle.

2.2. Misión de la empresa

La misión de LatamReady es impulsar el crecimiento y la expansión de las empresas en la región latinoamericana, ofreciendo una variada gama de servicios de implementación y

soporte de ERP, con el fin de optimizar los procesos, mejorar la eficiencia y aumentar la rentabilidad de sus clientes.

2.3. Visión de la empresa

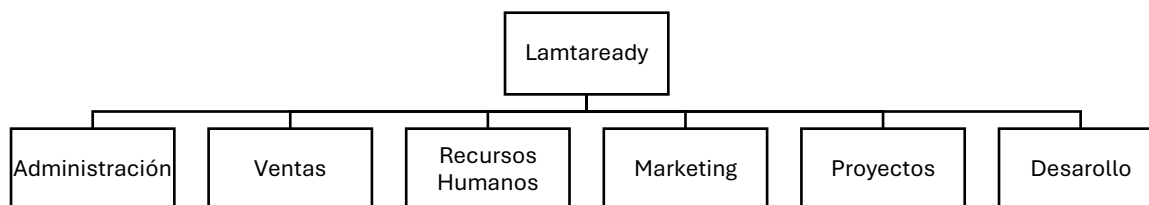
La visión de LatamReady es convertirse en el principal socio de implementación de ERP Oracle Netsuite en América Latina. Además, se compromete a ofrecer soluciones ERP innovadoras y personalizadas que ayuden a sus clientes a alcanzar sus objetivos comerciales.

2.4. Organización de la empresa

En la Figura 1 se presenta un organigrama que muestra las 6 principales áreas que conforman la empresa LatamReady: Administración, Marketing, Ventas, Proyectos, Recursos Humanos y Desarrollo.

Figura 1

Organigrama de la empresa LatamReady



Nota. Fuente: elaboración propia

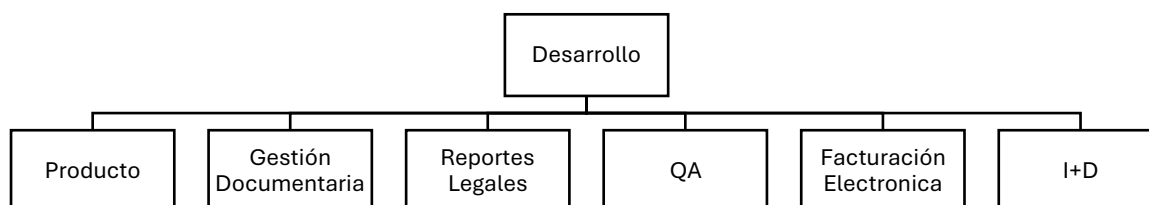
El área de proyectos es la encargada de analizar las necesidades de los clientes en cuanto a la personalización y/o implementación del ERP Oracle Netsuite y, a partir de ello, generar

nuevos requerimientos y proyectos que atiendan y solucionen dichas necesidades. Por otro lado, el área de Desarrollo se encarga de tomar estos requerimientos y ejecutar la planificación, diseño, pruebas e implementación de la solución, siempre en colaboración con el área de Proyectos.

El área de Desarrollo está compuesta por 7 subáreas: Producto, Gestión Documentaria, Reportes Legales, QA, Facturación Electrónica, Personalizaciones e Investigación y Desarrollo (I+D). En la Figura 2 se presenta un organigrama que ilustra dichas subáreas.

Figura 2

Subáreas del área de Desarrollo



Nota. Fuente: elaboración propia.

La subárea de I+D es la encargada de brindar apoyo y soporte a las demás subáreas del equipo de desarrollo. Sus principales tareas son: Investigación de nuevas tecnologías y tendencias que Oracle Netsuite implementa en cada una de sus actualizaciones y el desarrollo de módulos y herramientas que mejoren el flujo de trabajo de los distintos equipos de la empresa.

El autor de este trabajo participó con el rol de Diseñador UX/UI de la subárea de I+D, quien, junto con su equipo, se encargó del rediseño de un sistema para la gestión de proyectos y tareas del área de desarrollo, denominado Smartready.

2.5. Área, cargo y funciones desempeñadas

El autor de este trabajo ocupó el rol de Diseñador UX/UI en la subárea de I+D de la empresa LatamReady, teniendo como principales funciones:

- Establecer una estructura visual que sea consistente con la línea gráfica de la empresa y que pueda ser aplicada a todos los módulos y desarrollos realizados por el área o áreas externas.
- Análisis de módulos existentes, para identificar oportunidades de mejora tanto visuales como funcionales, con el objetivo de mejorar la experiencia del usuario final.
- Ideación, análisis y diseño de nuevas herramientas que mejoren y potencien el flujo de trabajo de la empresa.
- Acompañamiento y soporte al equipo de desarrolladores del área para el cumplimiento de los lineamientos establecidos en cada diseño.
- Creación de librerías que mejoren el flujo de diseño de soluciones del área.
- Trabajo en conjunto con el líder del equipo para la resolución de problemas, velando siempre por mantener la mejor experiencia de usuario.

2.6. Experiencia profesional realizada en la organización

El autor participó desde el campo del diseño UX/UI en los siguientes proyectos:

- Implementación del sistema de gestión de clientes, requerimientos y proyectos para el equipo de Proyectos, denominado Wormax.

- Implementación del sistema de gestión de tareas y proyectos para el equipo de Desarrollo, denominado Smartready.
- Implementación de un sistema para generar pagos masivos y su conciliación bancaria.
- Implementación de un sistema para generar pagos de retenciones.
- Implementación de un sistema para la gestión de clientes potenciales de la empresa.
- Implementación de un sistema para la programación e instalación de bundles en las instancias de producción y sandbox de los clientes.
- Implementación de un sistema para la gestión documentaria de la empresa.

CAPITULO III – ACTIVIDADES DESARROLLADAS

3.1. Situación problemática

En el ámbito empresarial, las herramientas de gestión de proyectos desempeñan un papel importante y fundamental para garantizar la eficiencia y el cumplimiento de los objetivos organizacionales. Según Verecol (s.f.), los sistemas de gestión de proyectos han transformado la manera en que las empresas planifican y supervisan actividades, permitiendo una asignación más efectiva de recursos y un seguimiento más preciso del progreso. Por otro lado, la implementación de sistemas de gestión adecuados no solo impacta positivamente en la productividad, sino que también en la competitividad de las organizaciones, como lo destaca un estudio de Ortiz et al. (2016), que enfatiza cómo estas herramientas pueden ser determinantes en la mejora continua de los procesos internos. Asimismo, el análisis de los procesos de gestión también resulta clave para optimizar los costos operativos y mejorar la eficiencia organizacional, como señala Mora (2023). Estas evidencias subrayan la importancia de contar con sistemas de gestión que no solo sean funcionales, sino que también se alineen con los flujos de trabajo específicos de cada empresa, permitiendo maximizar los recursos y alcanzar los objetivos estratégicos de manera más efectiva.

En este contexto, LatamReady trabaja con la ERP Netsuite para resolver necesidades de gestión internas de la empresa, como la gestión de empleados, la gestión de clientes potenciales y la gestión de proyectos. Para este último, Netsuite cuenta con funcionalidades integradas, como gestión de tareas y cronogramas, asignación de recursos y seguimiento de tiempos. Sin embargo, estas funcionalidades no se ajustaban a los flujos de trabajo del equipo. Por ello, a nivel gerencial, se decidió personalizar Netsuite de acuerdo con las

necesidades actuales y crear un nuevo sistema, llamado Smartready, para la gestión de proyectos y tareas, que utilizaría la ERP como base de datos. Este sistema permitió al equipo visualizar sus tareas de manera sencilla y acceder a todos los detalles relacionados con cada una de ellas.

Si bien inicialmente esta solución funcionó de manera adecuada, el crecimiento del equipo y la creciente complejidad de sus procesos y flujos de trabajo expusieron las limitaciones del sistema. Esto resultó en una gestión ineficiente del equipo de desarrollo y en una dependencia de herramientas externas, como Netsuite, para poder completar los flujos de trabajo, lo que a su vez incrementó los costos operativos.

3.1.1. Problema general:

LatamReady enfrentaba dificultades para mejorar el flujo de trabajo del equipo de desarrollo debido a que su sistema de gestión de proyectos, Smartready, presentaba una mala experiencia de usuario, problemas de usabilidad y carecía de funcionalidades que resolviesen las necesidades actuales del equipo. Esto generaba deficiencias en la gestión del equipo y en el uso de los recursos de la empresa.

3.1.2. Problemas específicos:

- **Mala experiencia de usuario:** Smartready era poco intuitiva y presentaba inconsistencias visuales, lo que dificultaba la navegación, y hacía engorrosa la experiencia del usuario final.
- **Falta de nuevas funcionalidades para una gestión adecuada:** Smartready carecía de funcionalidades esenciales para la correcta gestión de tareas y proyectos, como la creación de issues (errores), la gestión de subtareas, seguimiento de tiempo (time

tracking) y un sistema efectivo de notificaciones, entre otras funcionalidades requeridas.

- **Funcionalidades existentes no alineadas a las necesidades del usuario:** El sistema tenía funcionalidades limitadas o mal adaptadas que no cumplían con las expectativas y necesidades del equipo de desarrollo.

3.2. Solución

3.2.1. Objetivos

3.2.1.1. Objetivo general

Rediseñar la experiencia de usuario (UX) y la interfaz (UI) del sistema de gestión de proyectos llamado Smartready, incluyendo mejoras que resuelvan las necesidades del equipo y la usabilidad del sistema, para mejorar la gestión del equipo de desarrollo y racionalización de los recursos de la empresa.

3.2.1.2. Objetivos específicos

- **Mejora de la interfaz y experiencia de usuario de Smartready:** Rediseñar la interfaz de usuario para hacerla más intuitiva y consistente con la finalidad de mejorar la navegación y la usabilidad, lo que contribuye a una experiencia de usuario más satisfactoria.
- **Diseño de funcionalidades esenciales para la gestión de proyectos y tareas:** incorporación de nuevas funcionalidades que permitan la correcta gestión de proyectos y tareas en el equipo de desarrollo.
- **Análisis y mejora de funcionalidades existentes:** Evaluación de las funcionalidades actuales de la plataforma para identificar las mejoras que se alineen con las necesidades de los usuarios.

3.2.2. Alcance

3.2.2.1. Alcance funcional:

- En este trabajo de suficiencia profesional, se priorizará el rediseño UX/UI de manera integral de la interfaz y la experiencia de usuario de Smartready, aplicando las mejores prácticas del diseño UX/UI, para mejorar la interfaz. El producto final servirá como referencia para que los desarrolladores lleven a cabo la ejecución y programación del sistema. Sin embargo, no se abordará ninguna etapa del desarrollo ni se discutirá el proceso de programación de la aplicación.
- En este trabajo se detalla el primer MVP del Proyecto que no incluye algunas funciones como creación de proyectos y creación de tareas dentro de Smartready ya que estas aún se hacían directamente desde Netsuite.

3.2.2.2. Alcance técnico:

- En este trabajo se emplearán herramientas especializadas en diseño UX/UI, como Figma, para la creación de wireframes, mockups y prototipos de alta fidelidad, y FigJam, para la colaboración visual y el mapeo de ideas. Además, para la recopilación de datos de las encuestas realizadas, se utilizarán formularios de Google, que facilitarán la obtención y la organización eficiente de las respuestas. Estas herramientas serán clave para garantizar un proceso de diseño estructurado y centrado en las necesidades del usuario.

3.2.2.3. Alcance Organizacional:

- El sistema de gestión de proyectos a rediseñar es principalmente para el equipo de Desarrollo de la empresa LatamReady, ya que está especializado en los flujos de trabajo de este.

3.2.3. Etapas y Metodología

Para el rediseño y desarrollo de Smartready se aplicó la metodología ágil Scrum, la cual es usada para diferentes proyectos de la empresa.

De acuerdo con Schwaber y Sutherland (2020) el equipo de Scrum es multifuncional y trabajan de manera cohesionada con un enfoque claro en alcanzar un objetivo en común: el producto. Además, señala que el equipo está conformado por:

- Desarrolladores: Son los miembros comprometidos con la creación de un incremento utilizable en cada Sprint. Son los responsables de crear el Sprint Backlog y adaptarse diariamente a el objetivo del Sprint.
- Product Owner: Es el responsable de maximizar el valor del producto y gestionar el Product Backlog. Sus tareas incluyen establecer el objetivo del producto, organizar los elementos del backlog y garantizar su transparencia a todos los miembros del equipo. Además, representa las necesidades de los stakeholders y decide sobre cambios en el backlog.
- Scrum Master: Es el encargado de guiar al equipo hacia la autogestión, facilita la creación de incrementos de alto valor, elimina impedimentos y asegura que los eventos de Scrum se realicen de manera efectiva y positiva.

Menciona Schwaber y Sutherland (2020), además, que los artefactos de Scrum son los siguientes:

- Product Backlog: Es la lista ordenada de elementos necesario para mejorar el producto. La responsabilidad de dimensionarla recae en el equipo de desarrollo, en colaboración con el Product Owner.

- Sprint Backlog: Es el conjunto de elementos del product backlog que son seleccionados para el sprint.

Por último, según Schwaber y Sutherland (2020) existen los siguientes eventos en Scrum:

- Sprint: Son ciclos fijos de trabajo que permiten transformar ideas en valor.
- Sprint Planning: Aquí inicia el Sprint, el equipo define el trabajo a realizar seleccionando elementos del product Backlog.
- Daily: es una reunión corta que permite al equipo dar seguimiento al progreso hacia el objetivo del Sprint y, de ser necesario, ajustar el sprint backlog.
- Sprint review: tiene como objetivo analizar los resultados del Sprint. Permite al equipo presentar su trabajo a los stakeholders y discutir el progreso.
- Sprint retrospective: busca planificar mejoras en la calidad y efectividad del equipo. El equipo identifica puntos de mejora y propone cambios para mejorar el trabajo en futuros sprints.

Menciona Sassa et al. (2023) que algunas de las principales ventajas de usar Scrum son:

- Flexibilidad: permite a los equipos adaptarse a los cambios que ocurren durante el proyecto, de manera que existe una respuesta rápida ante cualquier imprevisto.
- Retroalimentación y mejora continua: permite retroalimentaciones a través de las revisiones en cada Sprint, donde los entregables o productos son inspeccionados por los stakeholders permitiendo encontrar y realizar mejoras además de garantizar que el producto final cumpla las expectativas.

- Autogestión del equipo: permite al equipo gestionar su propio trabajo y tomar decisiones de manera colectiva. Lo que fomenta una colaboración más efectiva y mejora la productividad.

Para el rediseño de SmartReady, con el objetivo de que cumpla con los requerimientos y necesidades del equipo, se empleó la metodología Design Thinking dentro de cada Sprint, la cual de acuerdo con Nausation y Nusa (2021) tiene las siguientes etapas:

- Empatizar: En esta etapa se busca comprender los desafíos y necesidades del usuario, con el objetivo de profundizar en sus motivaciones y las actividades cotidianas que realiza.
- Definir: Esta etapa tiene como objetivo ver el problema real a solucionar. A partir de los hallazgos obtenidos en la etapa de empatía, se define el problema con el objetivo de mejorar la experiencia del usuario manteniendo siempre el enfoque humano en el producto. Idealmente, esta etapa permite establecer el problema central que guiará el desarrollo del prototipo de UI/UX.
- Idear: Esta etapa se enfoca en la generación de ideas, prestando atención a las suposiciones y fomentando la creatividad. A lo largo de esta etapa, se elaborará una lista de propuestas que serán exploradas. Estas ideas se transformarán en mockups de alta fidelidad, las cuales evolucionarán en prototipos.
- Prototipar: En esta fase, se trabajará en la materialización de la idea seleccionada, desarrollándola en un prototipo. Después de que el prototipo sea creado, se le agregarán interacciones para transformarlo en un producto real.

- Evaluar: El prototipo creado en la etapa anterior será probado por los usuarios. A partir de la experiencia de los usuarios al interactuar con los prototipos, se recopilará información que ayudará a mejorar los productos y a realizar ajustes.

Para la óptima construcción del prototipo de alta fidelidad se utilizó la metodología de Atomic Design, la cual permite una construcción estructurada y eficiente de los elementos de la interfaz. De acuerdo con Frost (2016), esta metodología se compone de cinco niveles jerárquicos que guían el proceso de diseño:

- Átomos: Son los elementos más básicos de una interfaz, poseen propiedades únicas como dimensiones o tamaños de fuente que influyen cuando se aplica en un sistema más amplio. Estos elementos, no existen de forma aislada y cobran vida solo cuando se utilizan en contexto, sirviendo además como referencia útil en una biblioteca de patrones para el desarrollo y mantenimiento del diseño.
- Moléculas: Es la agrupación de los átomos para funcionar como una unidad, como una etiqueta, un input y un botón formando un formulario de búsqueda. Esta agrupación da propósito a los elementos individuales, creando componentes simples y reutilizables que facilitan las pruebas y promueven la coherencia. Esto simplifica la interfaz y permite construir elementos más complejos en etapas posteriores del diseño.
- Organismos: Son componentes complejos formados por grupos de átomos, moléculas u otros organismos, y constituyen secciones diferenciadas de una interfaz. Un ejemplo sería un encabezado que contiene una imagen de logotipo, una lista de navegación y un formulario de búsqueda. Estos organismos ofrecen

contexto al reunir componentes más pequeños y simples, permitiendo su reutilización en diferentes partes de un diseño, como una cuadrícula de productos en un sitio de comercio electrónico. Los organismos ayudan a diseñadores y desarrolladores a visualizar cómo se comportan esos componentes en escenarios más amplios y prácticos.

- Plantillas: Son estructuras que organizan los componentes dentro de una página, mostrando cómo encajan y funcionan juntos en un diseño completo. Por ejemplo, al agregar un encabezado a una plantilla de página de inicio, podemos ver cómo todos los elementos (como la navegación o el formulario de búsqueda) interactúan entre sí. Esto ayuda a entender cómo los componentes más pequeños se combinan para crear un diseño funcional. Además, las plantillas se enfocan en la estructura general, no en el contenido final, lo que permite ajustar fácilmente elementos dinámicos como el tamaño de las imágenes o la cantidad de texto.
- Páginas: Las páginas son versiones específicas de plantillas que muestran cómo se ve una interfaz con contenido real, lo que permite visualizar cómo funcionará el producto final para los usuarios. Este es el punto donde se prueban todos los componentes juntos, mostrando cómo el contenido real afecta el diseño y ayudando a identificar si se necesitan ajustes en moléculas, organismos o plantillas. Las páginas también permiten probar variaciones del contenido, como la cantidad de productos en un carrito o la longitud de un título, para asegurar que el sistema de diseño sea flexible y se adapte a diferentes escenarios y necesidades de los usuarios.

Es importante mencionar que en este proyecto se decidió trabar las plantillas y páginas de manera conjunta, denominándolas mockups.

3.2.4. Fundamentos utilizados

3.2.4.1. Fundamentos teóricos

a. Diseño UX

Hace referencia al diseño de la experiencia del usuario. Soegaard (2023) menciona que el diseño UX tiene como objetivo crear experiencias utilizables, significativas y relevantes. Esto se logra mediante un análisis del producto de manera integral, entendiendo el contexto en el que los usuarios utilizan dicho producto y centrándose en satisfacer sus necesidades. El diseñador UX se encarga de analizar la accesibilidad, jerarquía de información, flujos de navegación y diseño responsivo en constante colaboración con el diseñador UI garantizando así que el producto funcione con la mayoría de los usuarios y reduzca la dificultad al utilizar el producto.

b. Diseño UI

Hace referencia al diseño de la interfaz del usuario. Soegaard (2023) menciona que el diseño UI se enfoca en los aspectos visuales y la interacción del usuario con el producto. Trata del diseño detallado de todos los elementos con los que está familiarizado el usuario como botones, animaciones, menús, entre otros. El diseño UI no solo busca ser estético, si no también ser intuitivo y facilitar la interacción fluida con el producto.

c. UI Kit

Según Naseva (2024) un UI Kit es una compilación extensa de componentes de interfaz de usuario, plantillas, estilos, recursos gráficos y otros elementos de fácil uso y reutilizables para crear interfaces de usuario. Puede estar compuesto por

iconos, botones, formularios, estilos de texto, paleta de colores, imágenes, entre otros.

Según MockFlow (s.f) los principales beneficios de usar un UI Kit son:

- Diseño rápido: al aprovechar los componentes prediseñados en el UI Kit se reduce significativamente el tiempo y el esfuerzo invertido para crear interfaces de usuario, lo que permite crea.
- Coherencia de diseño: Los UI Kits establecen un estilo visual consistente en toda interfaz donde sea aplicado. Esto genera una apariencia profesional y una mejor experiencia de usuario ya que hace que la interfaz sea intuitiva y predecible.
- Mejora de calidad: los UI Kits ofrecen componentes correctamente diseñados y probados, lo que eleva la calidad y estética del producto final.

López (2022) destaca que un beneficio adicional de usar UI Kits es su versatilidad, ya que facilitad la adaptación de sus sus componentes a las necesidades específicas de un proyecto, permitiendo modificar propiedades o estilos de manera rápida y eficiente.

d. Wireframes

De acuerdo con Interaction Design Foundation (s.f.) los wireframes son esquemas visuales simplificados que representan la estructura de una interfaz de usuario. Son una herramienta esencial en el diseño de experiencia de usuario (UX), ya que proporcionan un modelo claro para que diseñadores, desarrolladores y otras partes interesadas comprendan la ubicación de los elementos como encabezados, botones, menus y otras secciones de contenido. No incluyen aspectos estéticos como

colores, tipografía o imágenes, centrándose únicamente en la funcionalidad y organización de elementos.

Los wireframes permiten a los equipos:

- Visualizar ideas: permite plasmar visualmente ideas abstractas o conceptos, lo que permite a los diseñadores poder conocer como funcionarían los elementos juntos en una pantalla.
- Detectar limitaciones y oportunidades: permiten descubrir problemas de usabilidad, limitaciones técnicas y oportunidades de mejora.
- Ahorrar tiempo y recursos: Identificar y abordar los problemas en la etapa de wireframing permite a los equipos prevenir modificaciones costosas en etapas posteriores del proceso, especialmente en el desarrollo.

e. Mockups

De acuerdo con Interaction Design Foundation (s.f.), los mockups son herramientas utilizadas por los diseñadores para presentar, demostrar, probar y refinar sus conceptos e ideas. A diferencia de los wireframes, estos incluyen elementos visuales como colores, texto, formas y otros detalles gráficos, lo que permite visualizar de manera concreta tanto la estética como la funcionalidad de un proyecto. Además, facilitan la comunicación entre diseñadores, desarrolladores, partes interesadas y usuarios. También contribuyen a mejorar la eficiencia del proceso de desarrollo al proporcionar pautas claras sobre cómo debería lucir finalmente una interfaz.

f. Prototipos

De acuerdo con Interaction Design Foundation (s.f.), los prototipos son representaciones preliminares de un producto simulando su diseño y funcionalidad. Estos modelos se crean con el fin de realizar pruebas, recopilar feedback y realizar iteraciones en el diseño, todo antes de proceder con el desarrollo del producto final.

Además del uso en pruebas de diseño, los prototipos también juegan un papel fundamental en la comunicación entre diseñadores y desarrolladores ya que ayudan a clarificar la visión del producto y alinean expectativas.

3.2.4.2. Fundamentos metodológicos

a. Principios Heurísticos de Jacob Nielsen

Son los Principios que el diseñador utilizado de base para realizar evaluaciones de usabilidad ya sea de una aplicación móvil, página web, software u otro producto digital. Se llaman “Heurísticos” por qué son principios que guían el diseño en lugar de ser reglas estrictas o directrices específicas de usabilidad. A continuación, se mencionan los 10 principios Heurísticos creados por Nielsen (2024):

- Visibilidad del estado del sistema: El sistema debe asegurar que los usuarios siempre estén enterados de lo que ocurre en el sistema a través de una retroalimentación adecuada dentro de un tiempo oportuno.
 - Relación entre el sistema y el mundo real: El sistema debe utilizar un lenguaje que los usuarios reconozcan y entiendan fácilmente. Es relevante emplear palabras, frases y conceptos familiar para ellos, evitando jergas internas.
- Además, el diseño debe seguir un flujo natural y lógico, presentando la información en un orden intuitivo que coincida con las convenciones del mundo real.

- Libertad y control del usuario: Los usuarios a menudo realizan acciones por error. Por ello, es fundamental ofrecer una salida para que el usuario pueda abandonar o cancelar la acción sin tener que pasar por pasos complicados.
- Coherencia y estándares: Los usuarios no deberían dudar si diferentes palabras, situaciones o acciones tienen el mismo significado. Es importante que se respeten las convenciones establecidas tanto de la plataforma como del sector.
- Prevención de errores: Antes que un buen diseño de mensaje de error es más importante prevenir problemas desde el inicio. De esta manera se pueden eliminar situaciones propensas a errores o se pueden validar previamente, proporcionando a los usuarios una opción de confirmación antes de que realice una acción.
- Reconocimiento en lugar de recuerdo: Reduce la carga de memoria del usuario al hacer visible los elementos, acciones y opciones. El usuario no debería tener que recordar información de una sección de la interfaz. La información necesaria para el diseño debe ser visible o fáciles de encontrar cuando se requieren.
- Flexibilidad y eficiencia de uso: Los atajos (no visibles para usuarios principiantes) pueden hacer más rápida la interacción para los usuarios avanzados, permitiendo que el diseño funcione bien tanto para usuarios expertos como para principiantes. Se debe facilitar al usuario la oportunidad de personalizar las acciones que realiza con frecuencia.

- Diseño y estética minimalista: Las interfaces deben evitar incluir información irrelevante o de poco uso. Cada elemento adicional compite con la información importante, reduciendo su visibilidad y claridad.
- Ayuda a los usuarios a reconocer, diagnosticar y recuperar errores: Los mensajes de error deben mostrarse en un lenguaje entendible para los usuarios (sin códigos de error), indicar con exactitud el problema y sugerir una posible solución al mismo.
- Ayuda y documentación: A veces, es necesario ofrecer documentación para guiar a los usuarios a la realización correcta de sus tareas. Esta ayuda y documentación deben ser fáciles de encontrar y enfocarse en las necesidades del usuario.

3.2.5. Implementación de las áreas, procesos, sistemas y buenas practicas

Para el rediseño y desarrollo de Smartready se utilizó la metodología scrum adaptándola a las necesidades del equipo he incorporado el rol de diseñador UX/UI como parte fundamental del proceso. El equipo estuvo conformado por:

- 1 diseñador UX/UI
- 1 desarrollador Frontend Senior
- 1 desarrollador Frontend Junior
- 1 desarrollador Backend Senior
- 1 QA
- Líder de equipo (Product Owner)

Para un óptimo desarrollo, y debido al tamaño y complejidad de la solución el Diseñador UX/UI manejaba un sprint adelantado al resto del equipo. Esto permitía que, al finalizar

cada sprint, los módulos diseñados estuvieran listo para ser desarrollados en el siguiente sprint, generando entregables incrementales. Esta estrategia facilitó una mejor colaboración entre el equipo y permitió avanzar de forma continua y ágil la implementación de los módulos.

Para el trabajo de diseño se establecieron 8 sprints, cada uno con duración de 2 semana.

En la Tabla 3, se detallan las actividades realizadas en cada sprint desde el rol del autor como Diseñador UX/UI.

Tabla 3

Detalle de los sprints realizados durante el rediseño de Smartready

Sprint	Detalle
1	Análisis del problema y primer acercamiento a la solución.
2	Diseño del UI Kit.
3	Diseño del login y el tablero de tareas.
4	Diseño del drawer de detalle tarea.
5	Diseño del módulo de issue.
6	Diseño del módulo de time tracking.
7	Diseño de la caja de notificaciones y el perfil del usuario.
8	Diseño del timeline de tareas.

Nota. Fuente: elaboración propia

En cada uno de los sprints presentados se utilizó la metodología Design Thinking, asegurando que sus etapas sirviesen de base para el óptimo desarrollo de cada tarea. Esto permitió entregar un producto alineado con las necesidades del usuario y responder a los

requerimientos de cada sprint. Es importante destacar que, en ciertos casos y dependiendo de la complejidad del sprint, se aplicaron las etapas de Design Thinking que mejor se adaptaban tanto a las tareas de diseño como al entregable final.

Adicionalmente, al finalizar cada sprint, se realizaba un sprint review con todos los integrantes del equipo. Durante esta reunión, se presentaba el entregable final, permitiendo a los participantes comprender el contexto de la solución, su funcionamiento y los componentes que la conformaban.

A continuación, se detallará las tareas realizadas en cada sprint junto con las etapas de la metodología de Design Thinking que se aplicaron.

3.2.5.1. Sprint 1: Análisis del problema y primer acercamiento de la solución

El objetivo principal de este sprint fue analizar el estado actual de Smartready para identificar problemas de usabilidad y funcionalidades, comprender las necesidades reales de los usuarios y definir con claridad los alcances del problema y las oportunidades de mejora.

En la Tabla 4 se presenta el detalle de las tareas realizadas en el sprint.

Tabla 4

Detalle de las tareas realizadas en el sprint 1

Etapa	Tarea
Análisis	Análisis del estado actual de Smartready.
	Entrevistas con usuarios.
	Aplicación de encuestas para analizar los intereses de los usuarios.
Definición	Definición del problema y sus alcances.

Ideación	Creación de wireframes.
Prototipado	Creación de prototipos para hacer pruebas con los usuarios.
Evaluación	Pruebas de usabilidad con los prototipos creados. Análisis de los resultados obtenidos a partir de las pruebas de usabilidad en conjunto con el PO.

Nota. Fuente: Elaboración propia

A continuación, se detallarán cada una de las etapas realizadas:

a. Análisis

Para iniciar el trabajo de rediseño, se comenzó analizando el estado actual de Smartready, aplicando los principios heurísticos de Jakob Nielsen. Esto permitió identificar los problemas que se detallan en la Tabla 5.

Tabla 5

Problemas encontrados de acuerdo con los principios heurísticos de Jakob Nielsen

Principio Heurístico	Problema
Visibilidad del estado del sistema	La comunicación de errores era deficiente: el sistema no informaba adecuadamente cuando ocurría un problema. En algunos casos, no se mostraba ningún cambio visible en la interfaz y, en otros, se desplegaban mensajes de error en forma de líneas de código, lo que resultaba confuso y poco comprensible para los usuarios.

	Para informar éxito, el escenario no era diferente, la interfaz tampoco mostraba un cambio aparente, lo que obligaba al usuario a suponer que la acción realizada se había completado correctamente.
Libertad y control del usuario	Los usuarios no podían tener control sobre ciertas acciones clave, como eliminar o editar un comentario, cambiar la prioridad de una tarea, entre otras.
Coherencia y estándares	No existía consistencia visual ni una identidad de marca sólida. Entre los principales problemas se encontraban el uso de diferentes tipos de letra, iconografía con estilos variados y botones de tamaños y colores distintos.
Prevención de errores	Los formularios de ayuda y cambio de contraseña no informaban al usuario sobre las condiciones necesarias para completar ciertos campos, como el número mínimo o máximo de caracteres permitidos. Esto podía provocar errores al intentar guardar o enviar la información.
Diseño y estética minimalista	No existía una jerarquía adecuada de elementos ni un uso efectivo de los espacios en blanco. En algunas secciones, el sistema aparecía sobrecargado de información, dificultando la comprensión y navegación.
Ayuda a los usuarios a reconocer, diagnosticar y recuperar errores	Cuando ocurría un error, el sistema no informaba a los usuarios sobre lo sucedido ni ofrecía instrucciones para resolver el problema. En muchas ocasiones, los usuarios

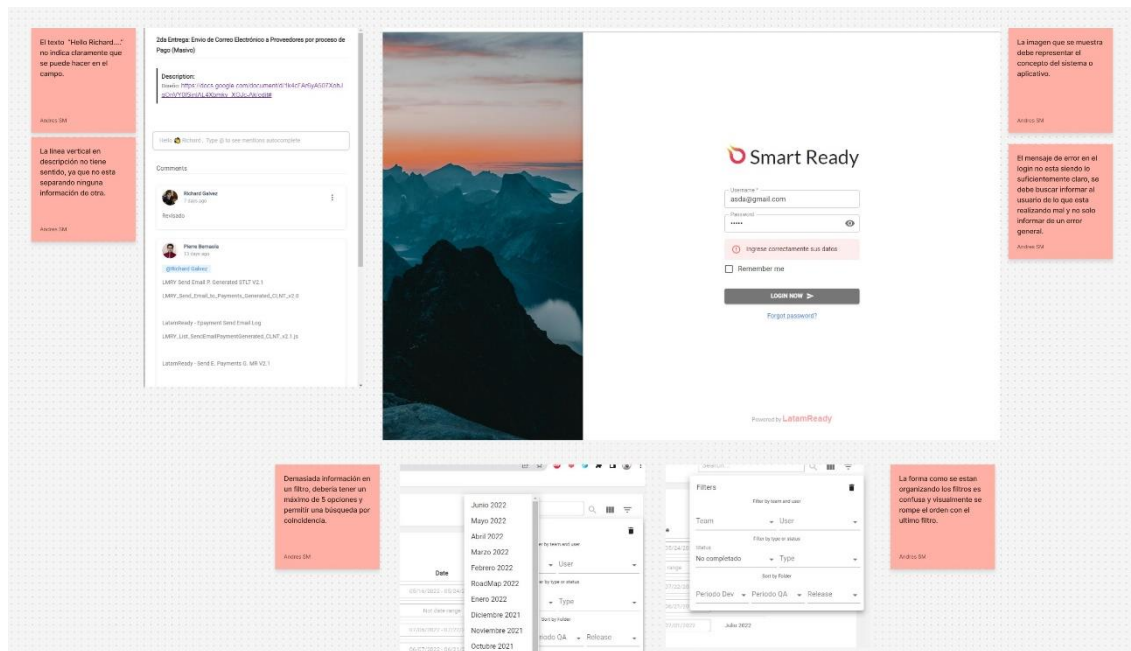
se veían obligados a consultar con un desarrollador para recibir ayuda.

Nota. Fuente: Elaboración propia

En la Figura 3 se muestra una ilustración del análisis realizado en Smartready, utilizando las heurísticas de Nielsen.

Figura 3

Análisis heurístico de Smartready



Nota. Fuente: Archivo en FigJam del análisis heurístico.

Como siguiente paso, era igualmente importante conocer las necesidades de los usuarios. Por ello, se llevaron a cabo entrevistas con seis personas del equipo de

LatamReady, incluyendo desarrolladores, QAs y líderes de equipo. Además, se envió una encuesta a todo el equipo para complementar el análisis.

Los principales hallazgos de las entrevistas fueron los siguientes:

- Información incompleta: los usuarios mencionaron la falta de información importante en las tareas, como información de las subtareas y documentos relacionados. Además, señalaron que existía una mala jerarquización de la información y errores de visualización en los campos.
- Necesidad de control de issue: los usuarios, especialmente los QAs, indicaron que era necesario contar con una correcta gestión de issues, ya que estos solían reportarse a través del chat, lo que ocasionaba pérdida de seguimiento y, en muchas ocasiones, desconocían el estado del issue.
- Necesidad de seguimiento de tiempos: Los usuarios, especialmente los líderes de equipo mencionaron la necesidad de gestionar los tiempos, conocer de manera clara que actividades está realizando su equipo y realizar el seguimiento correspondiente para asignar las tareas de manera más efectiva.
- Notificaciones sin funcionamiento: los usuarios indicaron que las notificaciones no funcionan, lo que les impide estar al tanto de los cambios o actualizaciones en una tarea.

Los principales hallazgos de las encuestas fueron los siguientes:

- Falta de información en las tareas: El 75% de usuarios coincidió en que es crucial tener la información detallada de las tareas para mejorar el flujo de trabajo y la comunicación entre todas las subáreas del equipo de desarrollo.

- Filtrado confuso y poco preciso: el 80% de los usuarios señaló que la forma en que se agregan los filtros no es óptima y que, además, necesitan filtros adicionales.
- Timeline tedioso y en desuso: el 55 % de los usuarios considera que el timeline es confuso y carece de la información necesaria para su correcto uso. El porcentaje restante no utilizaba esta funcionalidad.

b. Definición

Gracias a la información recopilada en la etapa de análisis, complementada con los requerimientos de los stakeholders, se logró comprender con mayor detalle el problema y las oportunidades de mejora para Smartready, las cuales fueron las siguientes:

- Optimización de la experiencia de usuario, redefiniendo toda la interacción, agregando una identidad sólida y conservando el minimalismo.
- Rediseñar todos los casos de éxito y error, manteniendo al usuario informado en todo momento del estado de las acciones que realice.
- Necesidad de incluir nuevas funcionalidades, como la gestión de issues, la carga de documentos, la visualización de la carga de trabajo (subtareas) y el seguimiento de tiempos (time tracking).
- Mejora de funcionalidades existentes:
 - Drawer de detalle de tarea: incluir toda la información de una tarea dentro del drawer, de manera que esté al alcance de cualquier usuario.
 - Filtros: mejorar todo el filtrado de información haciéndolo más completo y menos tedioso.

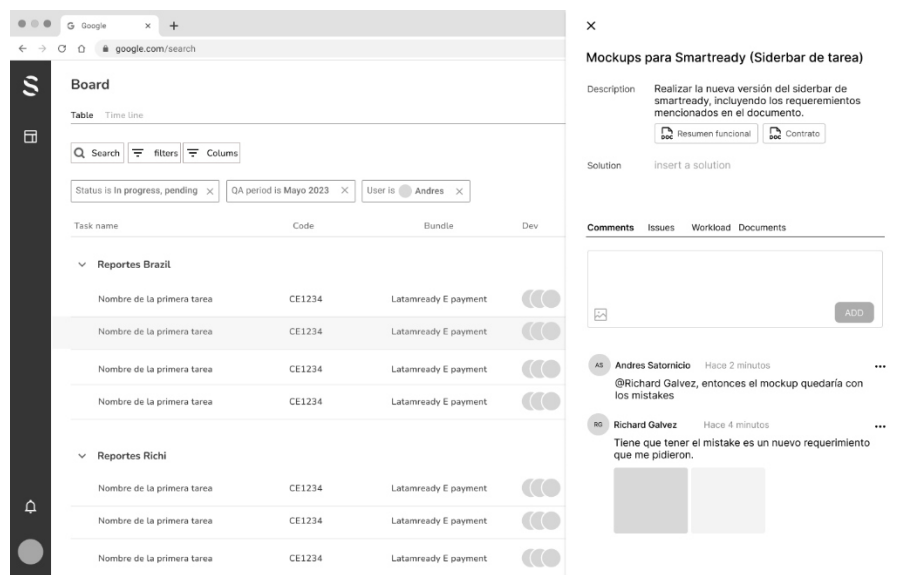
- Timeline: permitir dos formas de visualización del timeline, una para QAs y otra para un Desarrolladores.
- Perfil de usuario: incluir la información relevante del usuario, además de permitir el cambio de contraseña, validando correctamente el proceso.
- Notificaciones: permitir que el usuario elija si desea recibir notificaciones.

c. Ideación y prototipado

Con el problema claramente definido y las oportunidades de mejora identificadas, se empezaron a diseñar wireframes de baja fidelidad para abordar estas necesidades de manera rápida. La Figura 4 muestra un ejemplo de los wireframes construidos, los cuales sirvieron como base para iterar y validar ideas con usuarios. Cabe resaltar que, en este sprint, no se profundizó en el módulo de Issue, ya que se trata de un módulo complejo que requiere una investigación más extensa, la cual se detallará en el sprint 5.

Figura 4

Ejemplo de wireframe de Smartready



Nota. El wireframe muestra el detalle de una tarea. Fuente: Archivo en Figma de Wireframes creados.

Con los Wireframes creados se comenzaron a construir prototipos que simulasen los flujos más importantes que realizaría el usuario.

d. Evaluación

Con los prototipos listos, se empezó a realizar pruebas de usabilidad con 5 usuarios entre desarrolladores y QAs. Durante esta evaluación se encontraron varios puntos de mejora, entre los cuales destacan los siguientes:

- Permitir adjuntar documentos mediante enlaces de Google.
- Ordenar issues y tareas por prioridad o estado.
- Notificar por correo los cambios realizados en las tareas.
- Mostrar estado “editado” en un mensaje que fue modificado previamente.
- Separar la información general y comentarios en dos submódulos.
- Simplificar la aplicación de filtros, permitiendo combinar criterios y editarlos mientras se visualizan los resultados en tiempo real.
- Permitir la personalización del tablero agregando o quitando columnas.
- Permitir a los usuarios guardar automáticamente sus configuraciones, como criterios de búsquedas y configuración de columnas, para que se apliquen en próximos inicios de sesión.
- Permitir ingresar enlaces e imágenes en un comentario.

Con estas oportunidades de mejora, respaldadas por la investigación previa, se coordinó con el PO el ajuste de tareas para los siguientes sprints, lo que permitió

atender todas las observaciones y entregar un producto centrado en las necesidades de los usuarios.

3.2.5.2. Sprint 2: Diseño del UI Kit

El objetivo de este sprint fue construir un UI Kit que sirviera como base para el diseño de todas las pantallas y como una guía clara para los desarrolladores, detallando los lineamientos necesarios para implementar uno o más componentes de manera consistente.

En la Tabla 6 se presenta el detalle de las tareas realizadas en el sprint.

Tabla 6

Detalle de las tareas realizadas en el sprint 2

Etapa	Tarea
Análisis y definición	Benchmarking: Análisis de competencias directas e indirectas.
Ideación	Estilos primarios
	Creación de Átomos
	Creación de Moléculas
	Creación de Organismos

Nota. Fuente: elaboración propia

A continuación, se detallarán cada una de las etapas realizadas:

a. Análisis y definición

Para iniciar con la construcción de los componentes del interfaz se llevó a cabo un benchmarking de competidores directos e indirectos de Smartready. Este análisis permitió identificar y adaptar las mejores prácticas para aplicarlas en el rediseño. Entre las plataformas de gestión de proyectos evaluadas se incluyeron: Monday, Jira, Asana, Smartsheet, Notion, Teamwork y ClickUp.

b. Ideación

En esta etapa lo primero que se hizo fue crear los estilos primarios, los cuales comprendía de los siguientes elementos:

- Tipo de fuente de texto: selección de tipografía y sus variaciones de tamaño para sus diferentes usos.
- Paleta de colores: elección del color primario, grises y colores complementarios, junto con sus diferentes tonalidades.
- Sombras y efectos: definición de sombras y otros efectos visuales.
- Espaciados: especificación de tamaños para márgenes, relleno y espaciado.
- Iconografía e ilustraciones: selección y definición de iconos e imágenes ilustrativas.

En la Figura 5 se muestran los colores seleccionados para el rediseño de Smartready

Figura 5

Colores utilizados en el rediseño de Smartready

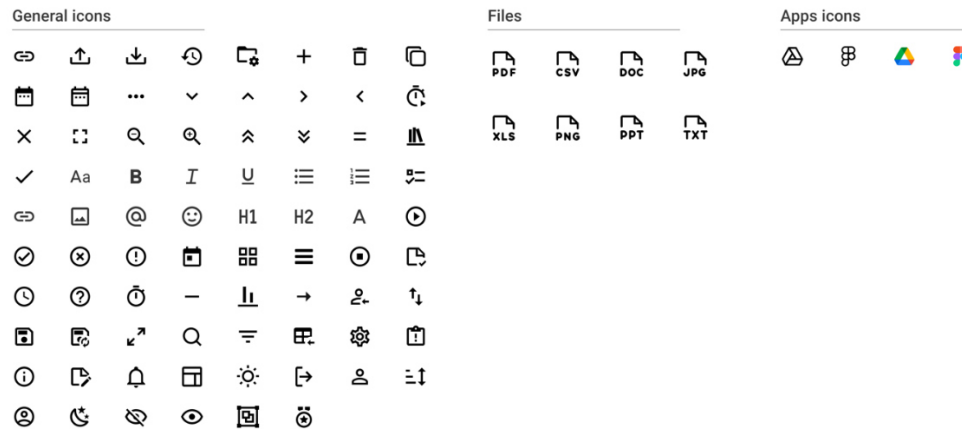


Nota. Fuente: Archivo en figma del tema diseñado

En la Figura 6 se muestran los iconos utilizados en el rediseño de Smartready. Cabe resaltar que algunos de los iconos fueron seleccionados de la librería de Material UI y se complementaron con la librería Phosphor Icons.

Figura 6

Íconos utilizados en el rediseño de Smartready



Nota. Fuente: Archivo en figma del tema diseñado.

Estos estilos primarios sirvieron de base para los componentes asegurando consistencia en todo el rediseño.

Para la creación de componentes se siguieron los niveles de la metodología de Atomic Design los cuales fueron:

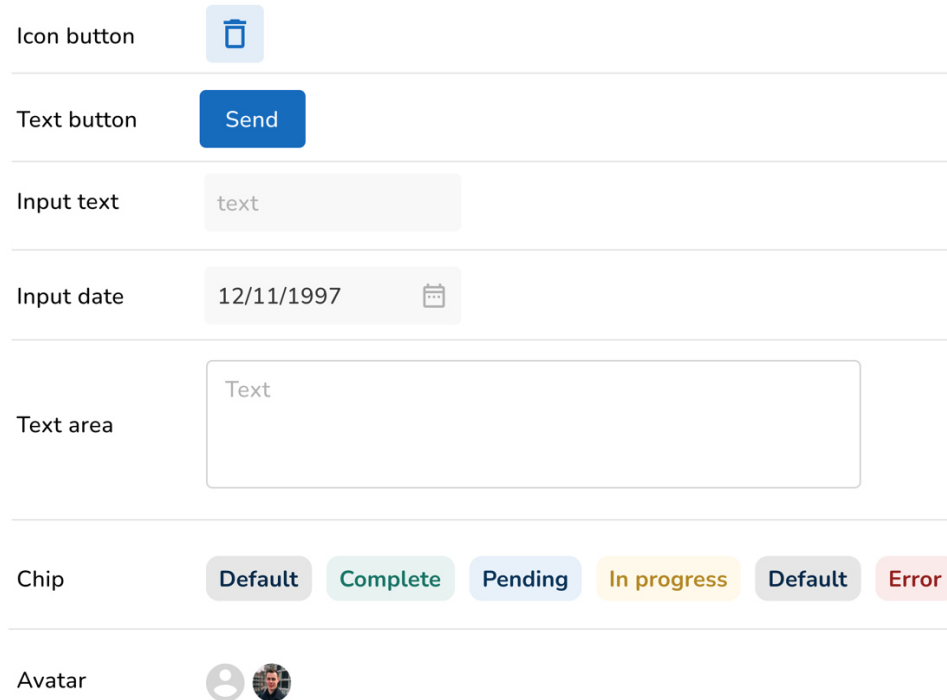
- Átomos:

Como se mencionó anteriormente, los átomos son los componentes más simples que sirven de base para la construcción de otros más complejos. En este nivel se construyeron botones, campos de texto, selectores, avatares, chips, entre otros.

En la Figura 7 se muestra algunos de estos componentes.

Figura 7

Átomos contruidos para el UI Kit



Nota. Fuente: Archivo en Figma del UI Kit

Cada uno de estos componentes fue acompañado de todos los estados necesarios para que el usuario realice una correcta interacción con el sistema. Por ejemplo, como se muestra en la Figura 8, los botones tenían los siguientes estados:

- Enabled: muestra que el botón está habilitado y listo para ser utilizado. Normalmente es el estado por defecto de cualquier botón.
- Disabled: indica que el botón esta deshabilitado y no puede ser presionado.
- Pressed: indica que el usuario ha presionado el botón.
- Hovered: se visualiza cuando el cursor del usuario pasa sobre el botón.

- Loading: indica que se está llevando a cabo una carga de información que tomara algunos segundos.

Algunos componentes tambien eran acompañados de sus diferentes variantes de manera que se establezca una jerarquía visual en los componentes contruidos.

Por ejemplo, como se muestra en la Figura 8, los botones tenían las siguientes variantes:

- Primary: usada para acciones principales.
- Outline y Outline-grey: usada para acciones secundarias.
- Text: usada para acciones terciarias.

Figura 8

Detalle del componente de botón

		Status				
		Enabled	Hovered	Pressed	Disabled	Loading
Variant	Primary					
	Outline					
	text					
	Outline-grey					

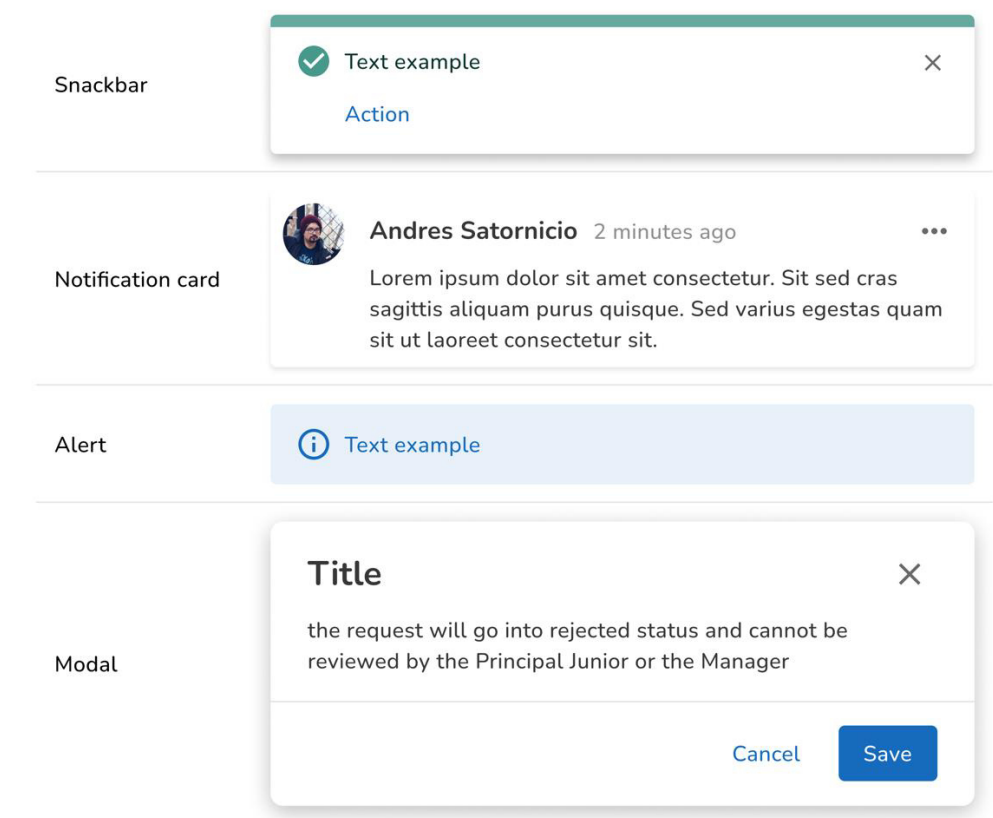
Nota. Fuente: Archivo en Figma del UI Kit

- Moléculas:

Se lograron crear en base a la combinación de los átomos diseñados anteriormente. En este nivel se construyeron tarjetas de notificación, tarjetas de issue, snackbars, alertas, modales de confirmación, editor de texto, dropdown, entre otros. En la Figura 9 se muestran el diseño de algunos de estos componentes.

Figura 9

Moléculas construidas en el UI Kit



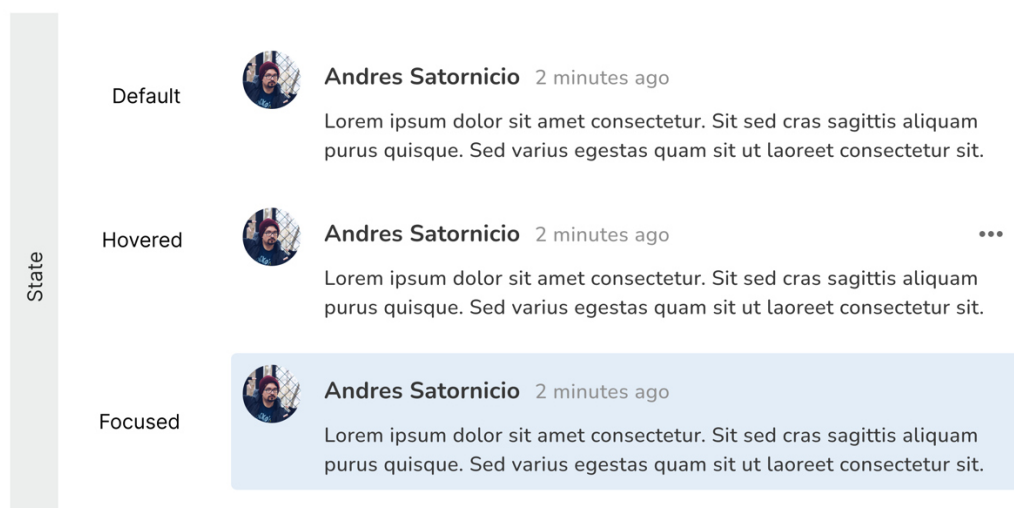
Nota. Fuente: Archivo en Figma del UI Kit

Cabe resaltar que, para algunos de los componentes, como pasó el caso de los átomos, también fue necesario definir sus diferentes estados. Por ejemplo, como se muestra en la Figura 10, el componente de Tarjeta de notificaciones tenía los siguientes estados:

- Default: estado inicial del componente.
- Hover: se visualiza cuando el cursor del usuario pasa sobre el botón.
- Focused: se visualiza cuando existe una notificación nueva.

Figura 10

Detalle del componente de Tarjeta de notificaciones



Nota. Fuente: Archivo en Figma del UI Kit

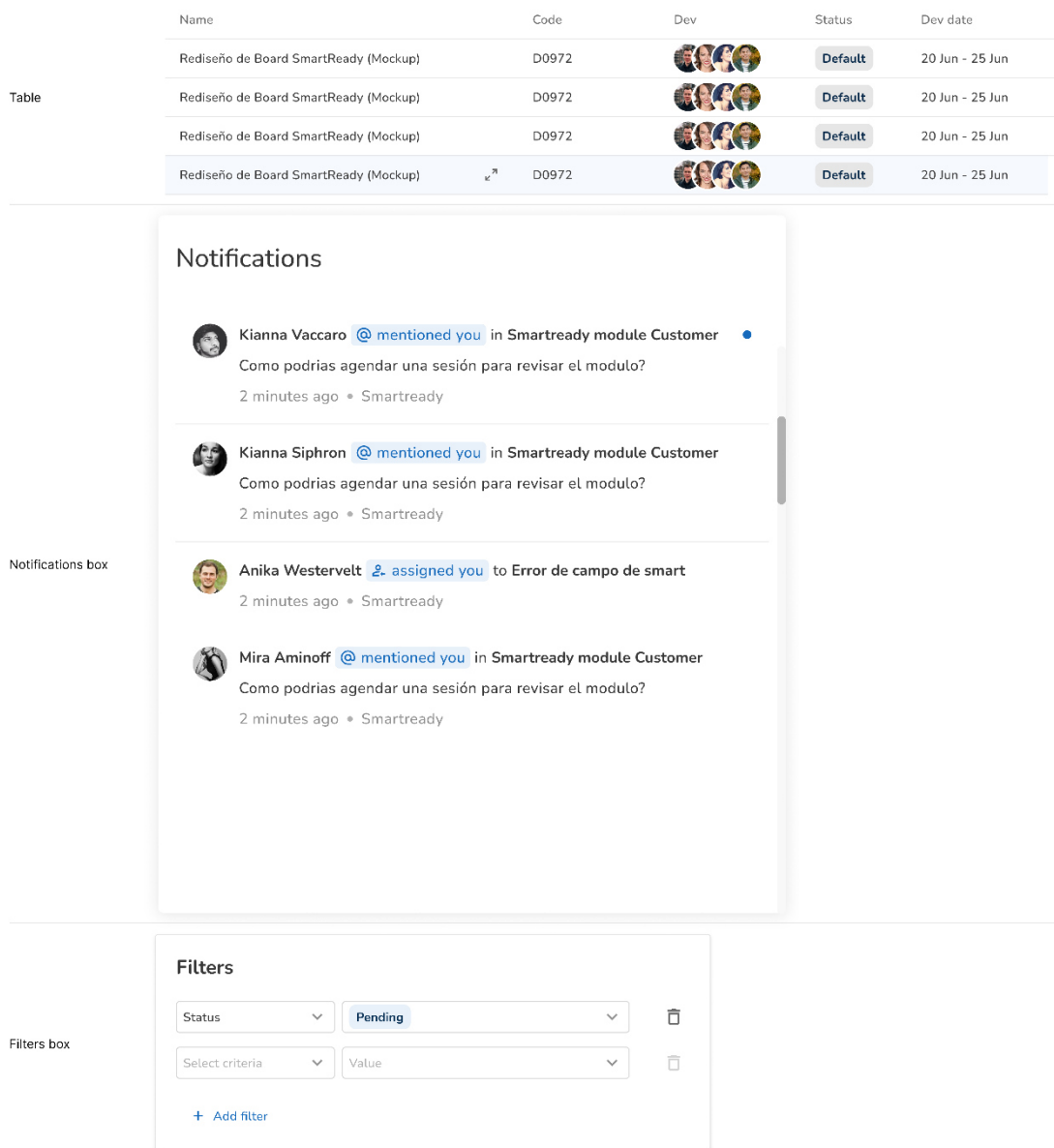
- Organismos:

Se lograron crear en base a la combinación de las moléculas y átomos diseñados anteriormente. En ese nivel se construyeron componentes como tablas, menú bar,

Navbar, caja de notificaciones, acordeón, caja de filtros, entre otros. En la Figura 11 se muestra el diseño de alguno de los componentes.

Figura 11

Organismos construidos en el UI Kit



Nota. Fuente: Archivo en Figma del UI Kit

3.2.5.3. Sprint 3: Diseño de login y tablero de tareas

El objetivo de este sprint fue construir el login, que permite al usuario acceder a Smartready, y el tablero de tareas, donde puede consultar sus proyectos y tareas asignadas.

En la Tabla 7 se presenta el detalle de las tareas realizadas en el sprint.

Tabla 7

Detalle de las tareas realizadas en el sprint 3

Etapa	Tarea
Ideación	Creación de mockups para el login y tablero de tareas.
Prototipado	Creación de prototipos interactivos que incluyeron los flujos principales, secundarios y de error.
Evaluación	Pruebas con usuarios Ajuste en mockups y flujos según el feedback recibido.

Nota. Fuente: Elaboración propia

A continuación, se detallarán cada una de las etapas realizadas:

a. Ideación

Como se mencionó anteriormente y de acuerdo con los niveles de Atomic Design, se comenzaron a construir las plantillas y páginas, que en adelante llamaremos Mockups, para el login y el tablero de tareas, utilizando los componentes creados en el sprint 2. Además, se tomaron en cuenta los puntos de mejora identificados en el sprint

1, como la simplificación de filtros y la personalización del tablero mediante la opción de agregar o quitar columnas. Esto aseguró que las soluciones propuestas respondieran directamente a las necesidades de los usuarios.

b. Prototipado

Con los mockups contruidos, se empezaron a diseñar prototipos interactivos que contemplaron los flujos principales del usuario, como también los casos secundarios y de error. El objetivo fue asegurar que el sistema cubriera todos los posibles escenarios de interacción del usuario.

Para el login, se construyeron flujos que contemplaron escenarios como:

- Primer acceso del usuario.
- Manejo de credenciales incorrectas, proporcionando retroalimentación útil y accesible.
- Manejo de errores de servidor, informando al usuario lo sucedido y posibles soluciones.

Para el tablero, se construyeron flujos que contemplaron escenarios como:

- Primer ingreso del usuario al tablero.
- Búsqueda de información específica del tablero, de manera que siempre este informado de la información similar encontrada o que la información buscada no existe.
- Aplicación de múltiples filtros por parte del usuario.
- Configuración de columnas visibles en el tablero.

c. Evaluación

Con los prototipos listos se hicieron pruebas con 5 usuarios, encontrando los siguientes puntos de mejora:

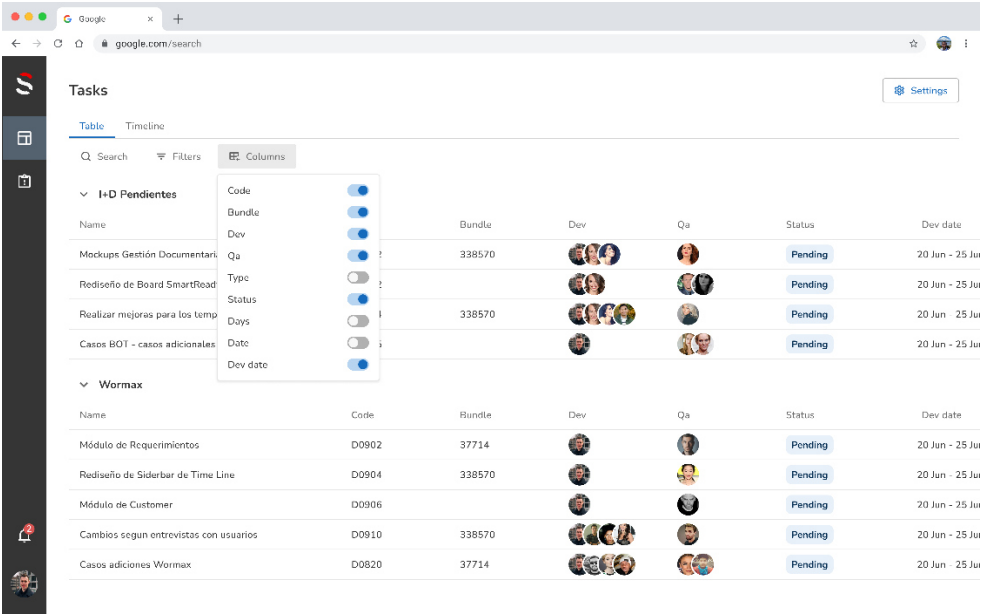
- Permitir que un usuario con una sesión iniciada previamente sea recordado automáticamente, de modo que solo deba ingresar su clave sin necesidad de escribir nuevamente su correo.
- Permitir ocultar proyectos (grupos de tareas) de manera que el usuario solo vea la información que necesita.
- Permitir visualizar tareas de periodos cerrados.

Considerando los puntos antes mencionados, se hicieron las modificaciones en las mockups y flujos, lo que permitió realizar un diseño de las dos funcionalidades correctamente validada y acorde a las necesidades de los usuarios.

En la Figura 12 se muestra el mockup diseñado para el tablero de tareas considerando el feedback obtenido.

Figura 12

Mockup del tablero de tareas



Nota. Fuente: archivo en Figma de los Mockups de Smartready

3.2.5.4. Sprint 4: Diseño del drawer de detalle de tarea

El objetivo de este sprint era construir el drawer de detalle de una tarea, permitiendo que el usuario pudiera revisar toda la información relacionada con su tarea, como datos básicos, comentarios, documentos y la carga de trabajo existente.

En la Tabla 8 se presenta el detalle de las tareas realizadas en el sprint 4.

Tabla 8

Detalle de las tareas realizadas en el sprint 4

Etapas	Tareas
--------	--------

Ideación	Creación de mockups de las subsecciones del drawer: general, comentarios, documentos y carga de trabajo (workload).
Prototipado	Creación de prototipos interactivos incluyendo flujos principales, secundarios y de error.
Evaluación	Pruebas con usuarios. Ajuste en mockups y flujos según el feedback recibido.

Nota. Fuente: Elaboración propia

a. Ideación

En esta etapa, se comenzaron a construir los mockups para el diseño del drawer y sus subsecciones: información general, comentarios, documentos y carga de trabajo (workload), utilizando todos los componentes desarrollados en el Sprint 2. Cabe destacar que, gracias al feedback recibido en el Sprint 1, se decidió separar las subsecciones de información general y comentarios para mejorar la claridad y la usabilidad. Además, se implementaron otros puntos de mejora identificados en el Sprint 1, como la posibilidad de adjuntar documentos mediante enlaces de Google, agregar enlaces e imágenes en los comentarios, e informar a los usuarios cuando un comentario sea editado.

b. Prototipado

Al igual que en el sprint anterior, en esta etapa, se comenzaron a diseñar prototipos interactivos basados en los mockups contruidos. Estos prototipos abarcaban los flujos principales del usuario, así como los casos secundarios y escenarios de error, con el objetivo de cubrir todos los posibles escenarios de interacción. Algunos de los escenarios contemplados fueron:

- Navegación del usuario entre las subsecciones del drawer.
- Edición de información dentro del drawer.
- Navegación entre comentarios, además de la creación, edición y eliminación de estos.
- Subida y gestión adecuada de documentos.
- Visualización de la carga de trabajo del usuario.

c. Evaluación

Con los prototipos listos se hicieron pruebas con 5 usuarios, encontrando los siguientes puntos de mejora:

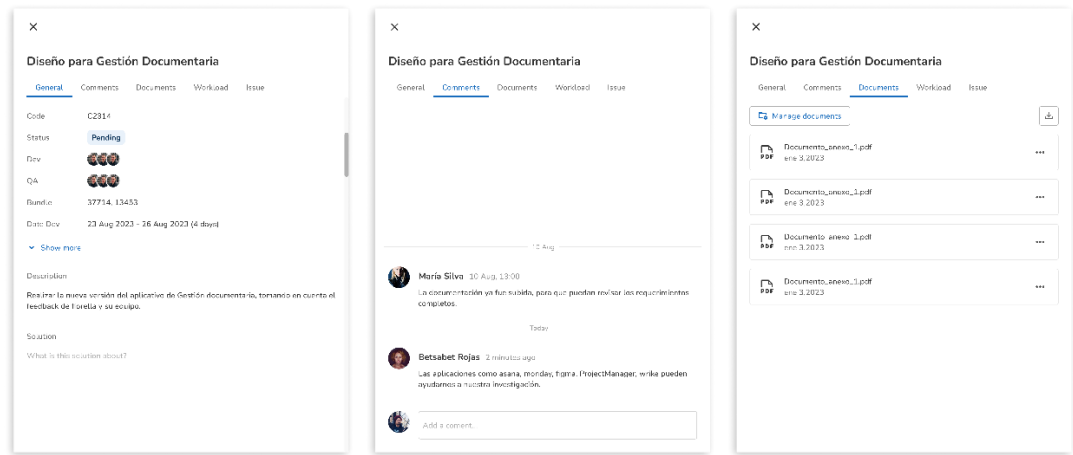
- Incluir un gráfico Gantt para representar de manera más clara la carga de trabajo.
- Ordenar la información de acuerdo con la prioridad asignada por los roles de QA y Desarrollo.
- Agrupar comentarios por fecha, mostrando primero los más recientes.
- Agregar una categoría “Otros” en los tipos de documento posibles a adjuntar, de manera que el usuario tenga la posibilidad de adjuntar documentos no categorizados.

Considerando los puntos antes mencionados, se hicieron las modificaciones en las mockups y flujos.

En la Figura 13 se muestran algunos de los mockups diseñados para el drawer, considerando el feedback obtenido.

Figura 13

Mockups del drawer de detalle de tarea



Nota. Fuente: archivo en Figma de los mockups de Smartready

3.2.5.5. Sprint 5: Diseño del módulo de issues

El objetivo principal de este sprint fue establecer el flujo de trabajo entre los equipos de QA y desarrollo para la gestión de issue, y reflejar dicho flujo en Smartready mediante el diseño de un nuevo módulo, llamado módulo de issues.

En la tabla 9 se presenta el detalle de las tareas realizadas en el sprint 5.

Tabla 9

Detalle de las tareas realizadas en el sprint 5

Etapas	Tareas
--------	--------

Análisis	Entrevistas al equipo de QA. Revisión del flujo de trabajo con el líder de equipo de QA.
Definición	Definición del problema a solucionar.
Ideación	Ideación de un flujo de trabajo óptimo para el trabajo en conjunto entre el equipo de QA y el equipo de desarrolladores. Creación de mockups.
Prototipado	Creación de prototipos interactivos incluyendo flujos principales, secundarios y de error.
Evaluación	Pruebas con usuarios. Ajuste en mockups y flujos según el feedback recibido.

Nota. Fuente: elaboración propia

a. Análisis

El módulo de issue era más extenso, por lo que se decidió realizar un análisis más profundo y específico para abordar todas las necesidades del equipo de QA, principalmente, y también las de los desarrolladores. Para iniciar este análisis, se organizó una reunión con el equipo de QA y su líder, durante la cual se identificaron las siguientes necesidades:

- Establecer un flujo de trabajo entre los equipos de Desarrollo y QA.

- Hacer un seguimiento más completo de los issues, de manera que se pueda consultar su estado en cualquier momento y conocer su duración hasta ser finalizados.
- Conocer los detalles de la solución implementada por el desarrollador para resolver el issue.
- Clasificar los issues de manera que pueda encontrarlos y gestionarlos de manera más eficiente.
- Contabilizar el número total de issues y los correspondientes a cada tarea, así como también el número de veces que un issue ha sido atendido nuevamente.

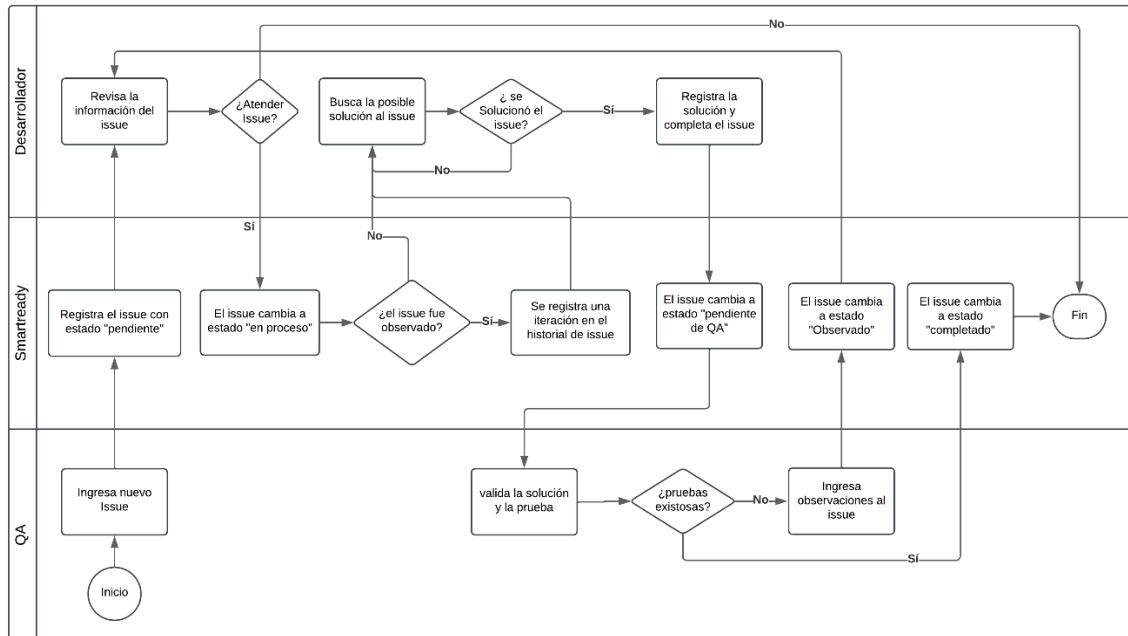
b. Definición

Con la información recopilada en la etapa anterior, se pudo conocer con más detalle las necesidades de los usuarios y los problemas a solucionar. El objetivo principal en esta etapa consistió en definir un nuevo flujo de trabajo en Smartready para la gestión de issues, con el fin de establecer una comunicación adecuada entre el área de QA y Desarrollo.

En la Figura 14 se visualiza el flujo definido por el diseñador UX/UI, el PO y el jefe de QA.

Figura 14

Flujo de trabajo definido para la gestión de un issue en Smartready



Nota. Fuente: elaboración propia

En base al flujo definido y las necesidades identificadas previamente, se incorporaron las siguientes funcionalidades al módulo:

- Manejo del estado de los issues.
- Historial de iteraciones.
- Tablero de issues.
- Filtrado de issues por diferentes criterios, como estado y tipo.
- Sección de issues en el drawer de detalle de tarea.

c. Ideación

Con la definición clara del problema a solucionar y las funcionalidades que debía incluir, se comenzó a diseñar todos los mockups, considerando el feedback recibido en

el sprint 1, como la opción de ordenar los issues por prioridad o estado, dependiendo del criterio del usuario.

d. Prototipado

Con los mockups creados se procedió a construir prototipos que simularan los flujos más relevantes para el usuario. Estos incluyeron los flujos principales, casos secundarios y escenarios de error, con el objetivo de cubrir todos los posibles escenarios de interacción del usuario. Algunos de estos fueron:

- Creación, edición y eliminación de issue por parte de un usuario con rol de QA.
- Observación de un issue por medio de un usuario con rol de QA.
- Visualización de iteraciones.
- Atención de un issue por medio de una persona con rol de Desarrollador.

e. Evaluación

Para garantizar que el módulo de issue cumpliera con todas las necesidades de los usuarios, se realizaron pruebas con 5 usuarios. Algunas de las observaciones más relevantes fueron las siguientes:

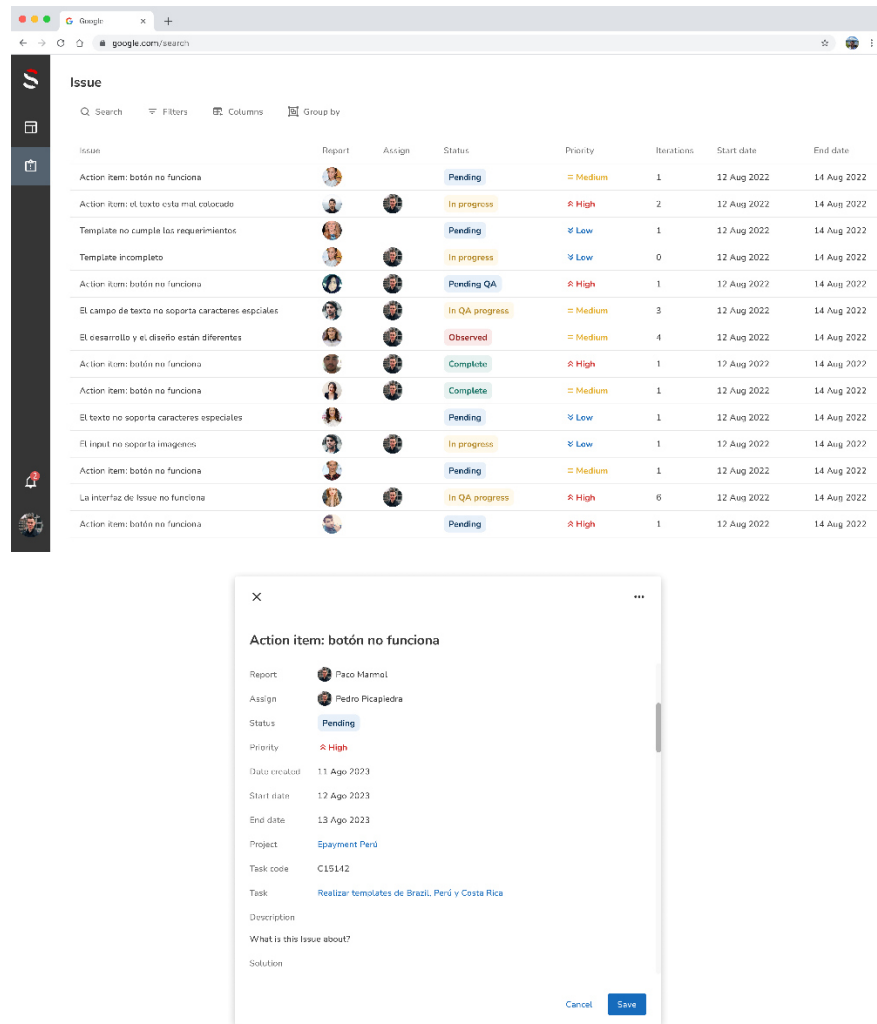
- El módulo debe permitir la edición de la descripción, mientras el issue se encuentre en estado “pendiente” o “en proceso”.
- El módulo debe incluir la opción de exportar toda la información de los issues.
- Mostrar la cantidad de issues solucionados y la cantidad total que existen.
- Permitir agrupar los issues por tarea y por proyecto.

Considerando todas las observaciones, se realizaron las modificaciones necesarias en los mockups y flujo para solucionar correctamente todos los problemas identificados.

En la Figura 15 se muestra los mockups del tablero de issue y el modal de detalle de issue, considerando el feedback obtenido.

Figura 15

Mockups de tablero de issue y modal de detalle de issue



Nota. Fuente: archivo en Figma de los mockups de Smartready

3.2.5.6. Sprint 6: Módulo de Time Tracking

El objetivo de este sprint es construir el módulo de Time tracking, permitiendo a los usuarios registrar su tiempo de trabajo y a los líderes de equipo a poder gestionar de manera más eficiente los tiempos de cada miembro, así como la asignación de tareas.

En la tabla 10 se presenta el detalle de las tareas realizadas en el sprint 6.

Tabla 10

Detalle de las tareas realizadas en el sprint 6

Etapas	Tarea
Análisis	Reunión con stakeholders
Definición	Definición del problema a solucionar
Ideación	Creación de mockups
Prototipado	Creación de prototipos interactivos
	incluyendo flujos principales, secundarios y de error.
Evaluación	Pruebas con usuarios.
	Ajuste en mockups y flujos según el feedback recibido.

Nota. Fuente: Elaboración propia

a. Análisis y definición

Al igual que el módulo de issues descrito en el sprint anterior, el módulo de time tracking presentó una complejidad significativa, lo que requirió un análisis detallado.

Para ello, se organizó una reunión con los stakeholders con el objetivo de comprender

las razones y necesidades detrás de su implementación. La principal necesidad identificada fue lograr visibilidad sobre las tareas en progreso y el tiempo empleado en completarlas. Esto permitiría comparar el tiempo estimado con el tiempo real utilizado, además de equilibrar la carga de trabajo entre los integrantes del equipo.

Tras el conocimiento claro de las necesidades se definieron las siguientes funcionalidades para incluir en el módulo:

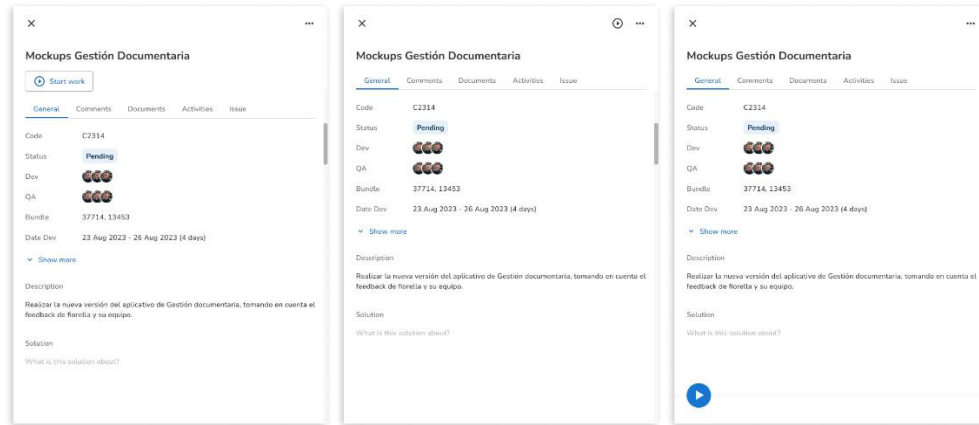
- Registro de tiempos por tareas, tanto de forma individual como masiva.
- Visualización del tiempo estimado versus el tiempo real en cada tarea.
- Historial de tiempos registrados por tarea.
- Cambio de nombre del submódulo workload por actividades, permitiendo incluir más información sobre las actividades realizadas por los usuarios, como el tiempo estimado y el tiempo real.

b. Ideación

En esta etapa, se empezaron a diseñar todos los mockups necesarios para las funcionalidades mencionadas anteriormente. Cabe resaltar que también se realizaron pruebas con algunos usuarios para validar detalles de jerarquía visual, como se muestra en la Figura 16, donde se presentan tres opciones para iniciar el registro del tiempo de una actividad. Tras las pruebas, se seleccionó la opción de la izquierda.

Figura 16

Mockups para pruebas de jerarquía de la información



Nota. Fuente: archivo en Figma de los mockups de Smartready

c. Prototipado

Con los mockups creados se comenzaron a construir prototipos que simulaban los flujos principales del usuario, así como los casos secundarios y de error, con el objetivo de cubrir todos los posibles escenarios de interacción. Algunos de estos flujos fueron:

- Registro de tiempos en tareas con actividades y sin actividades
- Búsqueda de una actividad para el registro de tiempos.
- Creación masiva de tiempos desde una tarea.
- Inicio del contador de tiempo desde una tarea.

d. Evaluación

Para asegurar que los usuarios puedan interactuar con el módulo, se hicieron pruebas con 5 de ellos. Algunas de las observaciones encontradas fueron las siguientes:

- El módulo debe permitir registrar los tiempos invertidos en la resolución de un issue.

- El módulo debe ofrecer la opción de registrar tiempo de tareas terminadas, ya que algunos usuarios mencionaron que podrían olvidar dicho registro.
- El módulo debe facilitar la corrección de errores, como el ingreso erróneo de tiempos o descripciones.

Considerando todas las observaciones, se hicieron las modificaciones en los mockups y flujos para asegurar que todos los problemas estén correctamente resueltos.

En la Figura 17 se muestra el mockup diseñado para la carga masiva de tiempos, considerando el feedback obtenido.

Figura 17

Mockup de carga masiva de tiempos

The image shows a web browser window displaying a 'Time tracking' modal form. The form includes a date selector set to '20 Dec 2023'. Below this is a table with columns: 'Task', 'Activity or Category', 'Description', and 'Duration'. The 'Task' column contains 'Mockups de Wormax'. The 'Activity or Category' column has a dropdown menu that is open, showing three options: 'Realizar estimación de tiempos', 'Entrevistas con el team de Customer Care', and 'Mejora del UI KIT y creación de nuevos componentes'. The 'Description' column has a placeholder 'Insert description'. The 'Duration' column has a placeholder 'H:mm' and a trash icon. A '+ Add' button is located below the table. At the bottom right of the modal are 'Cancel' and 'Save' buttons. The background shows a sidebar with icons and a table with columns 'Dev date' and 'Status'.

Nota. Fuente: archivo en Figma de los mockups de Smartready

3.2.5.7. Sprint 7: Diseño de la caja de notificaciones y el perfil del usuario.

El objetivo de este sprint fue construir la caja de notificaciones, permitiendo al usuario recibir cualquier notificación desde Smartready, y además, construir el perfil, donde el usuario podrá realizar configuraciones básicas y revisar su información.

En la tabla 11 se presenta el detalle de las tareas realizadas en el sprint 7.

Tabla 11

Detalle de las tareas realizadas en el sprint 7

Etapa	Tarea
Ideación	Creación de mockups para la caja de notificaciones y el perfil de usuario.
Prototipado	Creación de prototipos interactivos, flujos principales, secundarios y de error.
Evaluación	Pruebas con usuarios. Ajuste en mockups y flujos según el feedback recibido.

Nota. Fuente: elaboración propia

a. Ideación

En esta etapa se comenzaron a construir los mockups la caja de notificaciones y el perfil del usuario, utilizando todos los componentes desarrollados en el Sprint 2. Es importante destacar que en este sprint se incorporaron los puntos de mejora identificados en el Sprint 1 para el perfil del usuario, como la posibilidad de visualizar

información relevante, cambiar la contraseña validando correctamente todo el proceso y activar las notificaciones por correo.

b. Prototipado

Con los mockups listos, se comenzaron a diseñar prototipos interactivos que abarcaban los flujos principales del usuario, casos secundarios y escenarios de error, con el objetivo de cubrir todos los posibles escenarios de interacción. Algunos de los escenarios contemplados fueron los siguientes:

- Visualización del detalle de una notificación reciente.
- Creación y validación de una nueva contraseña.
- Cambio de foto de perfil.
- Activación o desactivación de las notificaciones por correo.

c. Evaluación

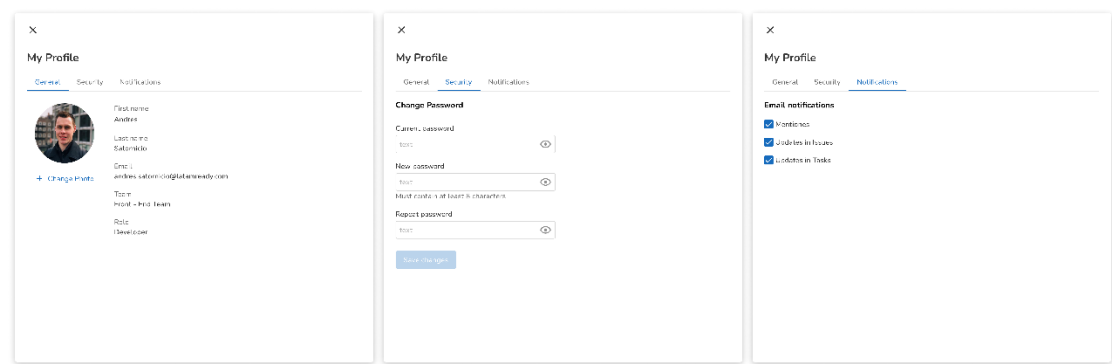
Para validar que la solución implementada en las dos funcionalidades fuera correcta, se realizaron pruebas con 5 usuarios, encontrando las siguientes observaciones:

- La información de la caja de notificaciones debe categorizarse en menciones (cuando el usuario es mencionado en un comentario) y asignaciones (cuando el usuario es asignado a una tarea o issue).
- El usuario debe tener la posibilidad de activar las notificaciones para los siguientes casos:
 - Cuando se realiza alguna actualización en un issue donde se encuentre asignado.
 - Cuando el usuario sea mencionado en algún comentario.

Considerando estas observaciones, se hicieron las modificaciones en los mockups y flujos, de manera que estén correctamente corregidos todos los problemas.

En la Figura 18 se muestran los mockups diseñados para el perfil del usuario, tomando en cuenta el feedback obtenido.

Figura 18
Mockups diseñados para el perfil del usuario



Nota. Fuente: archivo en Figma de los mockups de Smartready

3.2.5.8. Sprint 8: Diseño del timeline de tareas

El objetivo de este último sprint fue diseñar el timeline de tareas, abordando todas las necesidades de los usuarios y buscando generar un mayor uso de este.

En la Tabla 12 se presenta el detalle de las tareas realizadas en el sprint 8.

Tabla 12
Detalle de las tareas realizadas en el sprint 8

Etapa	Tarea
Ideación	Creación de mockups del timeline.

Prototipado	Creación de prototipos interactivos, flujos principales, secundarios y de error.
Evaluación	Pruebas con usuarios. Ajuste en mockups y flujos según el feedback recibido.

Nota. Fuente: elaboración propia

a. Ideación

En esta etapa, como en todos los sprints anteriores, se hizo uso de los componentes creados en el sprint 2 para la creación de todos los mockups necesarios para el timeline. Además, se buscaron considerar los puntos de mejora identificados en el sprint 1, los cuales consistían principalmente en poder visualizar el timeline de tres formas diferentes: la primera para el equipo de QA, la segunda para el equipo de Desarrolladores y la tercera para los líderes de equipo. De esta manera, se buscaba aumentar su uso al atender mejor las necesidades de cada grupo de usuarios.

b. Prototipado

Se tomaron los mockups contruidos para diseñar los flujos principales del usuario, casos secundarios y de error. Algunos de los escenarios contemplados fueron:

- Visualización de información dependiendo del rol del usuario.
- Aplicación de filtros.
- Opcion de ocultar o mostrar información como código de tarea o capacidad.

c. Evaluación

Para validar que la solución realizada estuviera correcta, se hicieron pruebas con 5 usuarios, encontrando las siguientes observaciones:

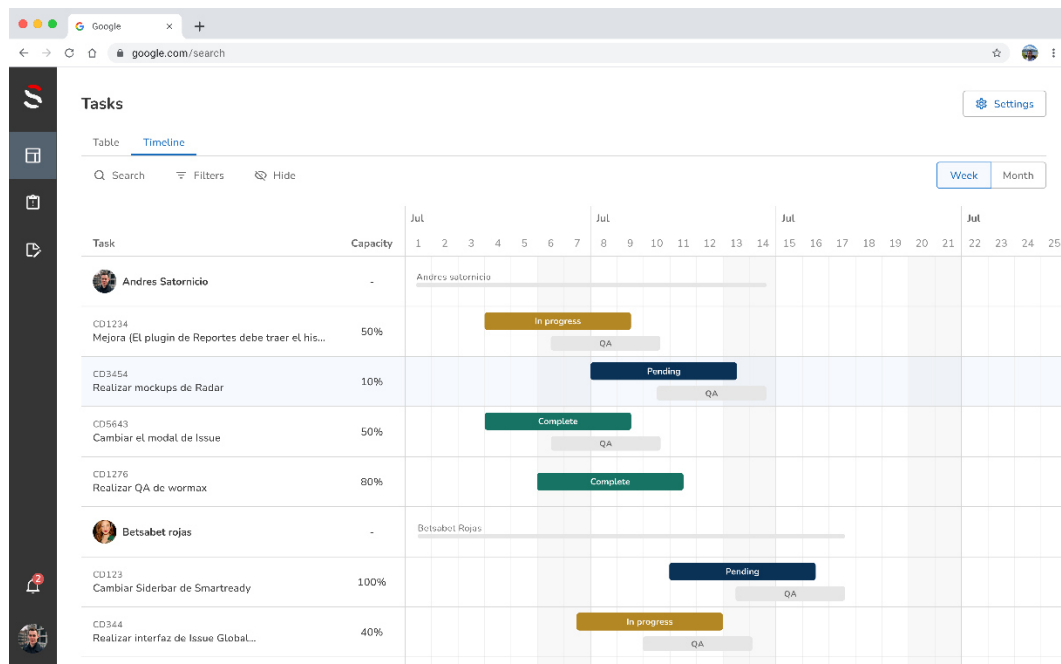
- Se debe ocultar o mostrar actividad del QA de la tarea para el perfil de desarrollador y líder.
- Se debe ocultar o mostrar actividad del desarrollador para el perfil de QA.
- Permitir abrir el drawer de detalle de tarea desde el timeline.
- Permitir mostrar la información agrupada por semana o mes.

Considerando estas observaciones, se hicieron las modificaciones en los mockups y flujos, de manera que se solventaran las mismas.

En la Figura 19 se muestra el mockup del timeline creado considerando el feedback recibido.

Figura 19

Mockup del timeline de tareas



Nota. Fuente: archivo en Figma de los mockups de Smartready

3.3. Evaluación

3.3.1. Evaluación económica

Para la evaluación económica del proyecto se tomaron en cuenta los costos generados por el personal a cargo del proyecto. A continuación, en la tabla 13 se muestra un desglose detallado de estos costos de manera mensual, expresados en nuevos soles y considerando cada rol que participó en el proyecto.

Tabla 13

Costos por mes del personal a cargo del proyecto

Rol	Costo mensual (en nuevos soles)
Diseñador UX/UI	S/ 4,000
Desarrollador Frontend Senior	S/ 4,200
Desarrollador Frontend Junior	S/ 2,500
Desarrollador Backend Senior	S/ 4,200
QA	S/ 3,500
Líder de equipo	S/ 9,000
Total	S/ 27,400

Nota. Fuente: elaboración propia

Ademas, tambien se consideró los costos generados por el uso de herramientas primordiales para el funcionamiento del proyecto como Netsuite, que se usaba de base de datos, y AWS que sirvió de

alojamiento web para la instancia de sandbox y producción de smartready. En la tabla 14 se muestra un desglose de los costos de estas herramientas de manera anual.

Tabla 14

Costos por año de las herramientas utilizadas

Herramienta	Costo anual (en nuevos soles)
4 licencias de Netsuite	S/ 8,880
AWS	S/ 4,440
Total	S/ 13,840

Nota. Fuente: elaboración propia

En base a la información de costos mostrada anteriormente, se determinó que el proyecto tuvo una inversión total de S/ 122,920 nuevos soles, durante los 4 meses de su duración.

3.3.2. Interpretación del VAN y TIR

El proyecto tuvo una inversión total de S/ 122,920 nuevos soles, considerando que tendrá flujos de S/ 84,360 nuevos soles anuales durante 5 años y una tasa de descuento del 10% anual, realizaremos la evaluación económica del proyecto mediante el cálculo del VAN y TIR. En la tabla 15 se muestran los cálculos realizados para hallar el VAN.

Tabla 15*Cálculo del VAN*

Tiempo	0	1	2	3	4	5
FNE	-122,920	84,360	84,360	84,360	84,360	84,360
FNE actual	-122,920	76,691	69,719	63,381	57,619	52,381
FNE acumulado	-122,920	-46,229	23,490	86,871	144,490	196,871

Nota. Fuente: elaboración propia

Se puede concluir de los resultados obtenidos, con una tasa de descuento del 10%, que el VAN es de S/. 196,871 y la TIR sería igual a 63%. Con estos datos se puede determinar que el proyecto es rentable ya que el VAN es positivo y la TIR supera significativamente la tasa de descuento.

CAPITULO IV - REFLEXIÓN CRÍTICA DE LA EXPERIENCIA

El autor del presente trabajo de suficiencia profesional ha desempeñado el rol de Diseñador UX/UI y, gracias a la experiencia obtenida durante el proyecto, ha obtenido las siguientes reflexiones:

- La metodología empleada en este proyecto combinó Scrum como marco de trabajo ágil con Design Thinking en cada sprint, logrando una integración efectiva entre una planificación iterativa y un diseño centrado en el usuario. Esta combinación permitió dividir el proyecto en entregables incrementales, asegurando avances constantes sin olvidarse del análisis de las necesidades de los usuarios en cada iteración. Además, también se incorporó la metodología Atomic Design que permitió crear componentes reutilizables y consistentes. Sin embargo, este enfoque integral no estuvo exento de desafíos: en algunos sprints, los tiempos ajustados hicieron necesario acortar ciertas etapas del Design Thinking, lo que obligó a priorizar las etapas más importantes según las necesidades del sprint, sin comprometer la calidad del resultado final.
- El trabajo en equipo fue fundamental para el éxito del proyecto, destacándose especialmente la comunicación con los desarrolladores, ya que permitió, por ejemplo, resolver dudas o conocer las limitaciones técnicas. La colaboración fue especialmente valiosa con la implementación de componentes bajo la metodología de Atomic Design, ya que la documentación detallada y las reuniones frecuentes garantizaron que los desarrolladores comprendieran correctamente las interacciones, estados y comportamientos previstos para cada

componente. De esta manera se aseguró que el diseño planteado se materializara de acuerdo con las expectativas y requisitos definidos.

- El diseño UX/UI no solo mejoró la experiencia de los usuarios finales, si no también enriqueció significativamente la forma de trabajo del equipo de I+D. Al incorporar principios de diseño centrado en el usuario y metodologías como Atomic Design, se establecieron estándares claros y componentes reutilizables que simplificaron la implementación y mantenimiento del sistema. Además, el enfoque estructurado y documentado del diseño sentó las bases para futuros proyectos, proporcionando un marco replicable y escalable que puede adaptarse a nuevas necesidades.
- Este proyecto permitió al autor entender más a fondo cómo se lleva a cabo la gestión de proyectos en el equipo de desarrollo de LatamReady, lo que le permitió tener una visión más clara de sus flujos de trabajo, fortalezas, áreas de debilidad y oportunidad. A través del análisis de las interacciones entre los usuarios y las herramientas existentes, le permitió identificar puntos clave de mejora, como la necesidad de optimizar algunos procesos y agregar funcionalidades que faciliten el trabajo del equipo de desarrollo. Este aprendizaje no solo enriqueció la perspectiva del autor como diseñador UX/UI, sino que también le permitió proponer soluciones prácticas, enfocadas en resultados y alineadas con los objetivos de la empresa.

CAPITULO V – CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- El rediseño de Smartready resultó en una mejora significativa de la experiencia de usuario, optimizando la navegación e interacción con el sistema. Esto favoreció un incremento en la frecuencia de uso por parte de los usuarios y consolidó la aceptación del sistema dentro del equipo de desarrollo.
- Las nuevas funcionalidades diseñadas e implementadas en el sistema respondieron a las necesidades de los usuarios identificadas durante cada sprint y facilitaron la organización de trabajo, cumpliendo con los requerimientos establecidos por los stakeholders, contribuyendo a la correcta gestión de proyectos de la empresa y permitiendo que los usuarios dejen de utilizar herramientas adicionales.
- La evaluación de las funcionalidades existentes permitió identificar áreas críticas de mejora. Estas mejoras, alineadas con las necesidades del equipo de desarrollo, optimizaron los flujos de trabajo internos y solucionaron las limitaciones previas del sistema.

5.2. Recomendaciones

- Se recomienda incorporar el diseño UX/UI en todas las etapas de desarrollo de producto y en todos los productos, ya sean internos o externos ya que no solo pone de protagonista al usuario final, sino que también impulsa, de esa forma, un mayor retorno de inversión al mejorar la satisfacción del cliente.
- Al utilizar componentes en el diseño, es fundamental mantener una comunicación fluida con el equipo de desarrollo para comprender las

limitaciones técnicas y garantizar la viabilidad de la propuesta. Además, establecer lineamientos claros de comunicación entre ambos equipos facilita la implementación eficiente y evita malentendidos.

- Es fundamental mantener actualizado el UI Kit, las librerías de componentes y cualquier otro recurso utilizado en el desarrollo. Esto asegura la coherencia visual y funcional del sistema. Además, es importante documentar cualquier cambio realizado en los componentes y comunicar esta información de manera efectiva a todos los involucrados en el proyecto, garantizando que siempre se utilicen las versiones más recientes.
- Es importante realizar evaluaciones periódicas de las funcionalidades y del diseño del sistema, involucrando a los usuarios para identificar posibles áreas de mejora. Este enfoque iterativo garantiza que el sistema evolucione de acuerdo con las necesidades cambiantes del equipo y del entorno empresarial.

5.3. Fuentes de información

Schwaber, K., & Sutherland, J. (2020). *La guía definitiva de Scrum: las reglas del juego*.

Scrumguides.org. <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf>

Sassa, A. C., De Almeida, I. A., Pereira, T. N. F., & De Oliveira, M. S. (2023). Scrum: A Systematic Literature Review. *International Journal of Advanced Computer Science and Applications*, 14(4). <https://doi.org/10.14569/ijacsa.2023.0140419>

Nasution, W. S. L., & Nusa, P. (2021). UI/UX Design Web-Based Learning Application Using Design Thinking Method. *ARRUS Journal of Engineering and Technology*, 1(1), 18–27. <https://doi.org/10.35877/jetech532>

Frost, B. (2016). *Atomic design*.

Nielsen, J. (30 de enero de 2024). *10 Usability Heuristics for User Interface Design*.

Nielsen Norman Group. Recuperado el 27 de noviembre de 2024, de

<https://www.nngroup.com/articles/ten-usability-heuristics/>

Soegaard, M. (2023). *UX vs UI: What's the Difference?* Interaction Design Foundation.

<https://www.interaction-design.org/literature/article/ux-vs-ui-what-s-the-difference>

Naseva, G. (3 de enero de 2024). *What is a UI Kit? Examples and Why you Should Use*

One. Untitled UI. <https://www.untitledui.com/blog/what-is-a-ui-kit>

Mockflow. (s.f.). *UI Kit*. Mockflow. Recuperado el 27 de noviembre de 2024,

de <https://mockflow.com/glossary/UI-Kit>

López, A. (13 de octubre de 2022). *¿Cómo usar UI Kits y Atomic Design al diseñar una*

interfaz? Medium. <https://medium.com/nowports-tech/usar-ui-kits-y-atomic-design-para-disenar-interfaz-6d9d9cbf5a2f>

Interaction Design Foundation (s.f.). *Wireframe*. Interaction Design Foundation.

Recuperado el 27 de noviembre de 2024, de <https://www.interaction-design.org/literature/topics/wireframe>

Interaction Design Foundation (s.f.). *Mockups*. Interaction Design Foundation. Recuperado

el 27 de noviembre de 2024, de <https://www.interaction-design.org/literature/topics/wireframe>

Interaction Design Foundation (s.f.). *Prototypes*. Interaction Design Foundation.

Recuperado el 27 de noviembre de 2024, de <https://www.interaction-design.org/literature/topics/prototypes>

- Ortiz, Y., Villamizar, A., Cepeda, O. (2016). Impacto de los sistemas de gestión en la productividad y competitividad. Estudio de caso de una industria cosmética. *Questionar Investigación Específica*, 4(1), 122–138.
<https://doi.org/10.29097/23461098.116>
- Verecol (s.f.). *Impacto de las aplicaciones de gestión de proyectos en el seguimiento de objetivos empresariales*. Recuperado el 27 de noviembre de 2024, de <https://vorecol.com/es/articulos/articulo-impacto-de-las-aplicaciones-de-gestion-de-proyectos-en-el-seguimiento-de-objetivos-empresariales-188882>
- Mora, K. (2023). *Optimización de los costos de producción, a través de la gestión de procesos productivos, para incrementar la rentabilidad del Tecnicentro Automotriz Tecnicar Rojas* [Tesis de bachiller no publicada]. Universidad Politécnica Salesiana de Ecuador

5.4. Glosario

- Racionalización: Es el proceso de reorganizar o estructura recursos, procesos o ideas de manera lógica y eficiente, buscando maximizar resultados con el menor esfuerzo o costo posible.
- ERP: Sistema de planificación de recursos empresariales que centraliza procesos clave de una organización, como finanzas, inventarios y gestión de proyectos.
- Netsuite: Sistema ERP basado en la nube que ofrece soluciones integrales para manejar áreas como finanzas, inventarios, relaciones con clientes y comercio electrónico, adaptándose a las necesidades de empresas de diversos tamaños e industrias.

- Bundle: Paquete de configuraciones o personalizaciones que se agrupan y se implementan en Netsuite para añadir funcionalidades o ajustes específicos.
- Ambiente de producción: Entorno donde un sistema funciona en tiempo real y los usuarios trabajan con datos reales para cumplir con los objetivos del negocio.
- Ambiente de sandbox: Espacio de pruebas aislado del ambiente de producción donde se realizan pruebas configuraciones y ajustes sin afectar el sistema en vivo.
- Pruebas de usabilidad: Método de evaluación de un producto, servicio o sistema en el que usuarios reales realizan tareas específicas para identificar qué tan fácil, intuitivo y eficiente es de usar, con el objetivo de mejorar la experiencia del usuario.
- Issue: Problema, error o incidencia reportado dentro de un sistema que requiere atención, solución o seguimiento por parte del equipo.
- Stakeholders: Personas clave involucradas en un proyecto, que tienen interés o influencia en su desarrollo, como clientes, usuarios o líderes de equipo.
- Drawer: Componente visual en una interfaz que permite mostrar u ocultar información o herramientas secundarias, generalmente deslizable desde un lateral.
- Librería: Colección de componentes o recursos de diseño predefinidos que facilitan la creación de interfaces visuales consistentes y escalables.
- Feedback: Opiniones, comentarios o sugerencias de los usuarios o stakeholders sobre un sistema, útiles para identificar mejoras y ajustar el diseño o funcionalidad.

- Figma: Herramienta de diseño colaborativo en la nube para crear interfaces, prototipos y sistemas de diseño en tiempo real.
- FigJam: Espacio de trabajo colaborativo para lluvia de ideas, diagramas y planificación.