

## Proyecto para mejorar la calidad de vida de personas con celiaquía mediante TICs



### **Autores**

- ❖ Laserna, Lautaro Diego (lasernalauta@gmail.com)
- ❖ Viejo, Julián Darío (julyvjo@gmail.com)

### **Director**

- ❖ Guccione, Leonel



RINFI es desarrollado por la Biblioteca de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata.

Tiene como objetivo recopilar, organizar, gestionar, difundir y preservar documentos digitales en Ingeniería, Ciencia y Tecnología de Materiales y Ciencias Afines.

A través del Acceso Abierto, se pretende aumentar la visibilidad y el impacto de los resultados de la investigación, asumiendo las políticas y cumpliendo con los protocolos y estándares internacionales para la interoperabilidad entre repositorios



Esta obra está bajo una [Licencia Creative Commons Atribución- NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).

# Autorización Repositorio Institucional - RINFI

Se presenta conjuntamente con la versión final del Trabajo Final

Repositorio Institucional RINFI, Facultad de Ingeniería, UNMDP

En calidad de TITULARES de los derechos de autor de la obra que se detalla a continuación, y sin infringir según mi conocimiento derechos de terceros, por la presente informo a la Facultad de Ingeniería de la UNMDP mi decisión de concederle en forma gratuita, no exclusiva y por tiempo ilimitado la autorización para:

- 1) Publicar el texto del trabajo más abajo indicado, exclusivamente en medio digital, en el sitio web de la Facultad y/o Universidad, por Internet, a título de divulgación gratuita de la producción científica generada por la Facultad, a partir de la fecha especificada.
- 2) Permitir a la Biblioteca que, sin producir cambios en el contenido, establezca los formatos de publicación en la web para su más adecuada visualización y la realización de copias digitales y migraciones de formato necesarias para la seguridad, resguardo y preservación a largo plazo de la presente obra:

Autor 1: Julián Darío Viejo. Documento: 41783676      Teléfono: 2236162177 E-mail: julyvjo@gmail.com	 Firma 1
Autor 2: Lautaro Diego Laserna Documento: 41073071      Teléfono: 2262533937 E-mail: lasernalauta@gmail.com	 Firma 2
Director: Leonel Domingo Guccione Documento: 17.982.189      Leg. 8234	 Firma Director
Codirector/a: ..... Documento: .....      Leg. ....	Firma Codirector/a

2. Título obtenido: Ingeniero en informática

3. Identificación/Título de la Obra: Proyecto para mejorar la calidad de vida de personas con celiacía mediante TICs



4. AUTORIZO la publicación bajo con la licencia Creative Commons BY-NC-ND Atribución-NoComercial-Sin Obra Derivada.

5. **Nota de Embargo:** Para aquellas obras que NO pueden ser de acceso a texto completo por razones de acuerdos previos con empresas o instituciones; por razones de índole comercial u otras razones; se procederá según lo establecido en Art. 6 de la Ley 26899 de Repositorios digitales institucionales de acceso abierto:

**ARTICULO 6º** — En caso que las producciones científico-tecnológicas y los datos primarios estuvieran protegidos por derechos de propiedad industrial y/o acuerdos previos con terceros, los autores deberán proporcionar y autorizar el acceso público a los metadatos de dichas obras intelectuales y/o datos primarios, comprometiéndose a proporcionar acceso a los documentos y datos primarios completos a partir del vencimiento del plazo de protección de los derechos de propiedad industrial o de la extinción de los acuerdos previos antes referidos.

Asimismo, podrá excluirse la difusión de aquellos datos primarios o resultados preliminares y/o definitivos de una investigación no publicada ni patentada que deban mantenerse en confidencialidad, requiriéndose a tal fin la debida justificación institucional de los motivos que impidan su difusión. Será potestad de la institución responsable en acuerdo con el investigador o equipo de investigación, establecer la pertinencia del momento en que dicha información deberá darse a conocer. A los efectos de la presente ley se entenderá como “metadato” a toda aquella información descriptiva sobre el contexto, calidad, condición o características de un recurso, dato u objeto, que tiene la finalidad de facilitar su búsqueda, recuperación, autenticación, evaluación, preservación y/o interoperabilidad.

En razón de lo expuesto, si el Trabajo se encuentra comprendido en el caso de que su producción esté protegida por derechos de Propiedad Industrial y/o acuerdos previos con terceros que implique la confidencialidad de los mismos, el/la directora/a debe indicar a continuación motivos y fecha de finalización del embargo:

NO SE AUTORIZA la publicación antes de la fecha \_\_/\_\_/\_\_\_\_ por lo siguientes motivos:

.....  
.....  
.....  
.....  
.....  
.....  
.....

Cumplido el plazo del embargo, estará accesible a texto completo según contempla la normativa vigente.

---

Director/a del TF

# Agradecimientos

A nuestras familias, por su apoyo incondicional, su paciencia y su presencia constante a lo largo de toda nuestra formación. Su acompañamiento fue esencial para llegar hasta acá.

A nuestros amigos y amigas, por estar siempre cerca, por cada charla, cada aliento y cada momento compartido que hizo más llevadero este camino.

A nuestros compañeros de cursada, por tantos trabajos en equipo, aprendizajes y desafíos compartidos durante estos años. Y a quienes nos acompañaron en los momentos más intensos del proyecto, incluso sin formar parte formal del equipo.

A los docentes de la Facultad de Ingeniería, por guiarnos con compromiso y generosidad, y por ser parte fundamental de nuestra formación.

Agradecemos especialmente a Leonel Guccione, por su guía constante como director del proyecto, a Estela Mota y Silvana Barragán, por su acompañamiento como referentes funcionales, y a Aureliano Vega, por su colaboración técnica en las primeras etapas.

Finalmente, al Hospital Interzonal General de Agudos "Dr. Oscar Alende", por confiar en nosotros y permitirnos aportar a una problemática concreta desde nuestra profesión.

# Índice

Agradecimientos.....	1
Resumen.....	5
Introducción.....	6
Objetivos.....	7
Objetivo general.....	7
Alcance del proyecto.....	7
Aporte del proyecto.....	8
Beneficiarios.....	8
Impacto.....	9
Análisis.....	10
Dominio del problema.....	10
Requerimientos.....	10
Requerimientos funcionales.....	11
Administrador web.....	11
Aplicación mobile.....	11
Requerimientos no funcionales.....	12
Aplicación web.....	12
Aplicación mobile.....	12
Proyecto.....	13
Antecedentes.....	13
Análisis de riesgos.....	14
Plan de contingencia.....	15
Análisis FODA.....	16
Fortalezas.....	16
Oportunidades.....	17
Debilidades.....	17
Amenazas.....	17
Organización del proyecto.....	18
Subproyecto Web.....	18
Subproyecto Mobile.....	18
Subproyecto Backend.....	19
Estimación inicial.....	19
Desvíos y ajustes de cronograma.....	23
Primer ajuste — Agosto de 2024.....	23
Segundo ajuste — Enero de 2025.....	24
Proceso.....	26
Metodología de trabajo.....	26
Metodología de diseño.....	27
Metodología de desarrollo.....	27

Uso de contenedores Docker.....	28
Comunicación.....	28
Control de tiempos y avances.....	30
Gestión del colaborador técnico.....	31
Diseño del sistema.....	33
Arquitectura.....	33
Tecnologías.....	34
Backend.....	34
Frontend Web.....	34
Aplicación Mobile.....	35
Motor de base de datos.....	36
Backend.....	36
Uso de estructuras reutilizables.....	36
Capa de datos.....	37
Gestión de errores.....	38
Pruebas.....	39
Pruebas automáticas.....	39
Unit testing.....	39
Pruebas de integración.....	40
Pruebas manuales.....	40
Testing web.....	41
Testing mobile.....	41
Resultados y mejoras.....	41
Seguridad.....	42
Logs.....	43
Logs en archivos del sistema.....	43
Trazabilidad en base de datos.....	43
Ambiente de pruebas.....	44
Validación funcional.....	45
Producto: Senda.....	46
Introducción a Senda.....	47
Cuestionario de ingreso.....	49
Cuidados para el hogar.....	51
Alimentos.....	52
Recetas.....	54
Puntos de compra.....	56
Estadísticas.....	57
Método del plato.....	58
Participación en concurso.....	60
Memoria del proyecto.....	61
Tecnologías.....	61

Trabajo en equipo.....	62
Gestión del colaborador técnico.....	63
Obstáculos externos.....	65
Planificación vs ejecución.....	66
Estimación inicial.....	68
Primer ajuste.....	69
Segundo ajuste.....	70
Comparación de horas por subproyecto.....	71
Análisis del Subproyecto Web.....	73
Análisis del Subproyecto Backend.....	75
Análisis del Subproyecto Mobile.....	77
Cumplimiento de objetivos.....	79
Aprendizajes.....	79
Trabajo a futuro.....	80
Conclusión.....	82
Anexo.....	84
Anexo I: Glosario.....	84
Anexo II: Diagrama entidad-relación simplificado (DER).....	87
Bibliografía.....	88

## Resumen

En la ciudad de Mar del Plata, se presenta una problemática vinculada a las personas que padecen celiaquía, quienes enfrentan diversas dificultades al intentar mantener una dieta libre de gluten (DLG).

Frente a esta situación, surge desde el Hospital Interzonal General de Agudos (HIGA) la necesidad de contar con una herramienta que brinde apoyo a esta población y, al mismo tiempo, permitir a los profesionales de la salud la posibilidad de conocer el estado de estas personas y ofrecer información útil en forma de guía y acompañamiento.

A modo de solución nace *Senda*, un sistema informático integrado por tres aplicaciones: un sistema web codificado en *Angular* orientado a los profesionales de la salud, una aplicación mobile codificada en *Kotlin* orientado a la población celíaca y un *backend* desarrollado con *Spring Boot*, que actúa como intermediario entre ambas aplicaciones, gestionando la lógica de negocio y el acceso a los datos. Este sistema busca facilitar el cumplimiento de la DLG y promover hábitos alimenticios seguros y saludables.

Como resultado, *Senda* cumplió con las expectativas de los profesionales de la salud y además agregó valor adicional por medio de funcionalidades propuestas por el equipo que contribuyen a la mejora de la calidad de vida de las personas con celiaquía. Además, se continuará trabajando en conjunto con los profesionales en la participación de un concurso sobre proyectos de investigación y tecnologías aplicadas, organizado por la Asociación de Celíacos y Sensibles al Gluten de España, lo cual representa una gran oportunidad para dar visibilidad al proyecto, validar su impacto a nivel internacional y continuar fortaleciendo el vínculo entre la tecnología y la salud.

El proyecto se destaca por su enfoque social en el ámbito local, aprovechando la accesibilidad tecnológica de las personas celíacas para abordar una problemática real. No solo se cumplió con el objetivo principal de brindar una herramienta informática que contribuya a la mejora de la calidad de vida de la población celíaca, sino que este proyecto contribuyó a fortalecer las competencias profesionales de sus integrantes, mejorando aspectos técnicos del desarrollo de software — como el aprendizaje y aplicación de nuevas tecnologías —, además de brindar experiencia práctica en la toma de decisiones, gestión de proyectos, estimación de plazos y trabajo en equipo.

## Introducción

La celiaquía es una enfermedad autoinmune cuyo único tratamiento consiste en seguir estrictamente una dieta libre de gluten (DLG). En Argentina, la cantidad de casos continúa aumentando y actualmente afecta entre el 0,5 % y el 1,0 % de la población adulta.

Por su parte, en el Partido de General Pueyrredón, la mayoría de las personas celíacas enfrentan dificultades para seguir una dieta libre de gluten, principalmente relacionadas a la desinformación. Frente a esta situación, surge en el año 2020, desde el Hospital Interzonal General de Agudos "Dr. Oscar Alende" (HIGA), un Proyecto de Desarrollo Tecnológico y Social (PDTS), en el cual se buscó establecer los requerimientos mínimos para crear una herramienta informática orientada a brindar soporte tanto a la población celíaca como a los profesionales de la salud involucrados. Este proyecto fue continuado y discontinuado en distintas etapas, hasta ser finalmente retomado por el equipo actual, que reorganizó sus objetivos y definió una nueva estrategia de desarrollo.

El propósito del presente informe es documentar el desarrollo del sistema informático denominado *Senda*, cuyo objetivo fundamental es aportar a la mejora de la calidad de vida de las personas con celiaquía. Este sistema busca acompañar a los usuarios celíacos, brindando apoyo y facilitando el seguimiento de una dieta libre de gluten, al mismo tiempo que provee a los profesionales de la salud una plataforma para monitorear y asesorar adecuadamente a la población afectada.

El sistema *Senda* está compuesto por tres componentes principales: una **aplicación web** —codificada en Angular— destinada a profesionales de la salud, orientada a la gestión y análisis de datos; una **aplicación mobile** —implementada en Kotlin— diseñada para brindar acompañamiento cotidiano a la población celíaca; y una **aplicación backend** —implementado con Spring Boot— que integra ambos *frontends*, administra la lógica del sistema y asegura la persistencia de los datos.

En este contexto, y gracias a la participación activa de Leonel Guccione como director del trabajo, Estela Mota como referente funcional y Aureliano Vega como colaborador técnico, se pretende desarrollar un sistema capaz, no solo de cumplir con los requerimientos iniciales establecidos por la PDTS, sino también de incorporar funcionalidades adicionales y valor agregado basados en propuestas generadas por los desarrolladores principales del proyecto, Lautaro Laserna y Julián Viejo.

# Objetivos

## Objetivo general

El propósito fundamental de este proyecto es desarrollar un sistema informático que contribuya a mejorar la calidad de vida de las personas con celiaquía en el Partido de General Pueyrredón. La solución busca facilitar el seguimiento de una dieta libre de gluten y brindar herramientas tecnológicas gratuitas tanto a la población celíaca como a los profesionales de la salud del Hospital Interzonal General de Agudos "Dr. Oscar Alende". Asimismo, el sistema busca proporcionar información confiable y centralizada, respaldada por el trabajo de los profesionales, que promueva prácticas alimentarias seguras.

## Alcance del proyecto

El sistema desarrollado contempla el diseño e implementación de múltiples funcionalidades específicas. Entre ellas se incluye un cuestionario de ingreso, orientado a relevar información clave de los usuarios celíacos, y un apartado de cuidados para el hogar, que promueve buenas prácticas alimentarias y brinda pautas claras para evitar la contaminación cruzada en el entorno doméstico.

También se contempla un apartado dedicado a la gestión y consulta de alimentos aptos, donde se centraliza información confiable, respaldada y validada. Estos datos alimentan otras funcionalidades del sistema, como la creación de recetas seguras y equilibradas. Las recetas pueden visualizarse con sus instrucciones paso a paso y valores nutricionales calculados automáticamente, a partir de los ingredientes seleccionados por los profesionales.

Adicionalmente, se incorpora un apartado de puntos de compra con información validada sobre comercios locales, un módulo de estadísticas que permite a los profesionales analizar el uso de la aplicación y el comportamiento de los usuarios celíacos.

Una sección especialmente relevante es el método del plato, que ofrece una herramienta visual para confeccionar platos balanceados por medio de una interfaz intuitiva. A medida que se incorporan ingredientes, el sistema ofrece recomendaciones personalizadas, detecta carencias nutricionales y sugiere mejoras, aportando valor educativo y práctico a la planificación de comidas seguras y completas.

Todos estos apartados se desarrollan bajo criterios de usabilidad, accesibilidad y eficiencia, priorizando una experiencia clara y sencilla para personas celíacas y profesionales de la salud. El alcance del proyecto se limita al desarrollo de la herramienta, sin incluir su mantenimiento posterior. La carga y validación de los datos será responsabilidad de los profesionales de la salud, quienes designarán a los involucrados adecuados, tanto profesionales como usuarios finales. El equipo de desarrollo se mantendrá disponible para colaborar en caso de ser necesario.

# Aporte del proyecto

## Beneficiarios

Este proyecto tiene un impacto social significativo, orientado principalmente a la población celíaca, que se constituye como la principal beneficiaria del sistema. A través de una herramienta diseñada específicamente para acompañarlas en el seguimiento de su dieta libre de gluten, las personas celíacas pueden acceder a información validada, recursos educativos y funcionalidades prácticas que promueven la autonomía y facilitan la adopción de hábitos saludables. Este aporte resulta especialmente valioso para aquellas personas recién diagnosticadas o con acceso limitado a atención médica especializada.

En segundo lugar, el Hospital Interzonal General de Agudos (HIGA) se beneficia al incorporar una plataforma que permite centralizar la información clínica de personas con celiaquía, generando datos relevantes para la toma de decisiones y optimizando el seguimiento médico.

A su vez, los integrantes del equipo se ven beneficiados, ya que el proyecto trascendió el ámbito académico para convertirse en un producto real, con potencial de comercialización entre locales y comercios interesados en sumarse a la plataforma. La capacidad de escalar la aplicación y establecer alianzas con actores del sector genera un escenario adecuado para la sostenibilidad del producto y la proyección profesional del equipo. Además, la inscripción al concurso promovido por la Asociación de Celíacos y Sensibles al Gluten representa una oportunidad valiosa para ampliar el currículum de los participantes y dar visibilidad a esta iniciativa.

Finalmente, a nivel académico, el sistema constituye una base sólida para que futuros estudiantes puedan expandir la plataforma en colaboración con los integrantes actuales, incorporando funcionalidades adicionales —como la implementación de inteligencia artificial en determinados módulos— y adaptando el sistema a nuevas necesidades. La visión colaborativa que inspira el proyecto fomenta la continuidad del trabajo dentro de un enfoque interdisciplinario y con impacto social, promoviendo la participación activa de diversos actores en la mejora continua del sistema.

## Impacto

El impacto del proyecto se presenta principalmente a nivel local, ya que se inserta en un contexto puntual de atención médica a personas con celiaquía en el sistema de salud pública de la ciudad de Mar del Plata. Al ofrecer una herramienta tecnológica desarrollada específicamente para las necesidades de este entorno, se promueve una atención accesible y basada en evidencia.

A nivel educativo y científico, la plataforma genera datos estructurados que podrían ser utilizados en estudios o investigaciones futuras sobre la enfermedad celíaca en la región. También promueve la incorporación de tecnologías en entornos tradicionales de atención médica, lo que puede motivar a otras instituciones a adoptar soluciones similares.

Desde una perspectiva social, el sistema fomenta un estilo de vida saludable como eje central, no solo para las personas celíacas sino también para su entorno. A través de recursos educativos accesibles e integrados, se busca sensibilizar a la comunidad sobre la importancia de la alimentación segura, la prevención de la contaminación cruzada y el acompañamiento respetuoso de quienes conviven con esta enfermedad. En este sentido, *Senda* no solo funciona como una herramienta médica o técnica, sino también como un medio de concientización, promoviendo prácticas más empáticas e inclusivas en la vida cotidiana.

A nivel institucional, este proyecto fortalece los lazos entre la universidad y el sistema de salud, demostrando cómo la informática aplicada puede aportar soluciones a problemáticas reales, desde un enfoque interdisciplinario y con compromiso social.

# Análisis

## Dominio del problema

Como se mencionó anteriormente, en la ciudad de Mar del Plata existe una creciente problemática relacionada con las dificultades que enfrentan las personas celíacas para mantener una dieta libre de gluten. Estas dificultades incluyen la identificación de alimentos aptos para el consumo, el conocimiento sobre comercios que ofrecen productos sin gluten, el manejo adecuado de alimentos para evitar la contaminación cruzada y la planificación de una dieta equilibrada que cumpla con sus necesidades nutricionales.

Esto se debe, en primer lugar, a la falta de información precisa sobre la celiacía y la DLG, y, en segundo lugar, a factores socioeconómicos: los productos especialmente elaborados para celíacos son caros, mientras que los alimentos naturalmente sin gluten pueden llevar a confusiones si no se tiene mucha experiencia.

Como consecuencia, muchos pacientes eliminan el gluten sin encontrar sustitutos que cubran sus necesidades nutricionales, lo que puede llevar a deficiencias de nutrientes esenciales. Esta problemática se intensifica en dos grupos específicos: por un lado, las personas con recursos económicos limitados, que enfrentan barreras para acceder a asesoramiento nutricional; y, por otro, los pacientes recién diagnosticados, que carecen de la orientación necesaria para gestionar correctamente su alimentación.

Ante esta problemática, surge desde el Hospital Interzonal General de Agudos (HIGA) la necesidad de contar con una herramienta que brinde apoyo a esta población y que permita proporcionar información accesible y recursos prácticos para que puedan adoptar una dieta equilibrada y segura, además de concientizar a la comunidad sobre la importancia de un tratamiento adecuado de la celiacía.

## Requerimientos

Este apartado expone los requerimientos funcionales y no funcionales del sistema, considerando que el proceso de elicitación no fue llevado a cabo por el equipo actual, sino que fue heredado de equipos anteriores. En estos, ya se establecía explícitamente la necesidad de dos aplicaciones complementarias: una aplicación web destinada a profesionales de la salud y una aplicación mobile orientada a la población celíaca.

Ante esta situación, el equipo realizó un análisis del conjunto de requerimientos heredados, con el objetivo de adaptarlos a los lineamientos actuales del proyecto. Algunos fueron descartados, mientras que otros fueron incorporados o reformulados a partir del nuevo enfoque funcional y técnico adoptado.

## Requerimientos funcionales

### Administrador web

**RFW1.** Ingreso al sistema. El sistema permitirá el ingreso a los usuarios registrados que proporcionen nombre de usuario y contraseña.

**RFW2.** Registro de acciones. El sistema registrará en un archivo de tipo log todas las acciones de los usuarios registrados, incluyendo: fecha, hora, nombre de usuario y acción.

**RFW3.** Gestión de usuarios. El sistema permitirá a un usuario con perfil de administrador realizar altas, bajas, modificación y consultas de usuarios del sistema.

**RFW4.** Gestión de cuenta. Cada usuario podrá cambiar su clave de acceso. El sistema permitirá a cada usuario darse de baja.

**RFW5.** Cuidados para el hogar. El sistema permitirá crear y editar instructivos, dando recomendaciones y sugerencias, orientados a los cuidados de los alimentos en el hogar y evitar la contaminación cruzada.

**RFW6.** Gestión de alimentos: El sistema permitirá gestionar la información de alimentos, incluyendo su grado de accesibilidad. Cada alimento deberá estar asociado a un grupo nutricional.

**RFW7.** Gestión de recetas: El sistema permitirá realizar el alta, baja, modificación y consulta de recetas, incluyendo ingredientes y cantidades. Los ingredientes se seleccionarán a partir de los alimentos definidos en RFW6, junto con su cantidad y unidad de medida.

**RFW8.** Gestión de los puntos de compra. El sistema permitirá administrar comercios donde adquirir alimentos para celíacos, incluyendo altas, bajas, modificaciones y consultas.

**RFW9.** Gestión del cuestionario. Integrar o utilizar una herramienta para gestionar preguntas y respuestas. Las preguntas creadas deberán poder ser respondidas desde la aplicación mobile.

**RFW10.** Estadísticas. El sistema permitirá visualizar estadísticas sobre las búsquedas de alimentos y recetas realizadas por los usuarios, con el fin de identificar los elementos más consultados.

### Aplicación mobile

**RFM1.** Registro. La aplicación permitirá a un nuevo usuario registrarse en el sistema.

**RFM2.** Login. La aplicación permitirá el acceso a usuarios registrados mediante nombre de usuario y contraseña.

**RFM3.** Cuidados para el hogar. La aplicación debe mostrar consejos relacionados con los cuidados en el hogar (vinculado a RFW5).

**RFM4.** Cuestionario de ingreso. La aplicación permitirá responder al cuestionario definido en la funcionalidad RFW9.

**RFM5.** Consulta de alimentos. La aplicación permitirá consultar alimentos disponibles, sus valores nutricionales y nivel de accesibilidad.

**RFM6.** Consulta de Recetas. La aplicación permitirá visualizar y buscar recetas en base al nombre o a los ingredientes que contengan.

**RFM7.** Método del Plato. La aplicación permitirá confeccionar un plato utilizando el método del plato. Informará si está nutricionalmente balanceado según los grupos alimentarios, y sugerirá recomendaciones en base al resultado. Esta funcionalidad no se utilizará para crear recetas.

**RFM8.** Puntos de Compra. La aplicación permitirá visualizar los puntos de compra disponibles para personas celíacas.

## Requerimientos no funcionales

### Aplicación web

**RNFW1.** Interfaz de usuario moderna e intuitiva. El sistema web debe presentar una interfaz limpia y fácil de utilizar.

**RNFW2.** Diseño responsive. El sistema debe adaptarse correctamente a diferentes tamaños de pantalla.

**RNFW3.** Compatibilidad con navegadores. El sistema debe funcionar correctamente en las versiones actuales de los navegadores más utilizados.

### Aplicación mobile

**RNFM1.** Rendimiento. La aplicación debe responder en menos de 2 segundos ante acciones comunes del usuario. Las operaciones que requieran comunicación con el backend no deben superar los 5 segundos bajo condiciones normales de red.

**RNFM2.** Interfaz amigable y atractiva. La aplicación debe contar con una interfaz centrada en el usuario, priorizando la claridad, simplicidad y una estética visual atractiva.

**RNFM3.** Compatibilidad. La aplicación deberá ser compatible con una amplia gama de versiones del sistema operativo Android.

**RNFM4.** Tamaños de pantalla. La aplicación deberá visualizarse correctamente en dispositivos con diferentes tamaños, resoluciones y proporciones de pantalla.

# Proyecto

## Antecedentes

El proyecto que dio origen al sistema *Senda* se remonta al año 2020, a partir de una propuesta impulsada por profesionales del HIGA, en el marco de un Proyecto de Desarrollo Tecnológico y Social (PDTS). La iniciativa surgió como respuesta a las dificultades que enfrenta la población celíaca para mantener una dieta libre de gluten, especialmente en sectores socialmente vulnerables, y propuso el desarrollo de una herramienta informática que ofreciera acompañamiento tanto a pacientes como a profesionales de la salud.

En sus primeras etapas, el proyecto fue abordado por un grupo de estudiantes de la carrera de Ingeniería en Informática, quienes avanzaron en el análisis de requerimientos junto con la referente funcional Estela Mota y el director Leonel Guccione. Como resultado de ese trabajo se elaboró un documento con un conjunto de requerimientos validados que sirvieron como base para futuras etapas. Sin embargo, debido a diversas circunstancias —entre ellas la rotación del equipo original y la falta de continuidad en el desarrollo técnico— el proyecto quedó inconcluso y sin una implementación funcional.

Durante el año 2023, el proyecto fue retomado por el equipo actual, que decidió reorganizar su estructura y redefinir la estrategia de desarrollo. En este proceso, se realizó una revisión exhaustiva de los requerimientos funcionales existentes y de la documentación previamente elaborada, los cuales fueron ampliados y reorganizados para adecuarlos al nuevo enfoque técnico adoptado. Este trabajo pudo realizarse de manera ágil gracias al nivel de avance alcanzado por los grupos anteriores en colaboración con la referente funcional, lo que permitió reducir la necesidad de reuniones frecuentes.

En esta etapa se realizaron los primeros encuentros con el director del trabajo final, se avanzó en la redacción del protocolo, y se confirmó la participación de un colaborador técnico para el desarrollo mobile. Sin embargo, factores personales y laborales postergaron el inicio formal del proyecto. Lautaro se encontraba en proceso de mudanza a la ciudad de Bariloche para asumir un nuevo empleo en la Comisión Nacional de Energía Atómica (CNEA), lo cual implicaba una oportunidad muy valiosa y poco frecuente. Este cambio de contexto requirió una reorganización de tiempos y prioridades. Como consecuencia, el desarrollo fue pospuesto hasta abril de 2024, cuando finalmente se presentó y aprobó el protocolo definitivo. A partir de ese momento, se formalizó el comienzo del proyecto.

## Análisis de riesgos

Se llevó a cabo un análisis de riesgos con el objetivo de identificar aquellas situaciones que podrían dificultar el desarrollo o la finalización del proyecto. Este análisis contempla la descripción de cada riesgo potencial, sus consecuencias y la asignación de un valor numérico que permite evaluar la necesidad de implementar un plan de contingencia.

Dicho valor, denominado peso, se obtiene multiplicando la probabilidad de ocurrencia por el impacto del riesgo, utilizando una escala del 1 al 3 para ambos parámetros (donde 1 representa el nivel más bajo y 3 el más alto). Cuando el peso resultante es igual o superior a 6, se define un plan de contingencia, que consiste en una medida preventiva o correctiva destinada a mitigar los efectos del riesgo en caso de materializarse.

A continuación, se presenta la tabla correspondiente al análisis de riesgos realizado.

	Riesgo potencial	Consecuencia	P	I	Peso
R01	Baja del colaborador técnico.	Atrasos en el cronograma por la necesidad de conseguir un reemplazo o asumir el desarrollo del componente tercerizado.	2	3	<b>6</b>

	Riesgo potencial	Consecuencia	P	I	Peso
R02	Baja de un integrante del proyecto.	Atrasos en el cronograma o la posible baja del proyecto si no se consigue reemplazo.	1	3	<b>3</b>

	Riesgo potencial	Consecuencia	P	I	Peso
R03	Dificultad en la curva de aprendizaje de nuevas tecnologías	Atrasos en el cronograma o deficiencias en las funcionalidades por dificultades en la implementación.	1	2	<b>2</b>

	Riesgo potencial	Consecuencia	P	I	Peso
R04	Estimación incorrecta de plazos por falta de experiencia.	Incumplimiento del cronograma, lo que conlleva reprogramaciones y cambios en las fechas límites.	3	3	<b>9</b>

	Riesgo potencial	Consecuencia	P	I	Peso
R05	Adopción por parte de la población celiaca.	Poco alcance social o baja popularidad de la aplicación.	2	3	<b>6</b>

	Riesgo potencial	Consecuencia	P	I	Peso
R06	Mala interpretación de requisitos del usuario.	La aplicación final no cumple con las necesidades del usuario.	2	3	<b>6</b>

## Plan de contingencia

En base a los riesgos detectados y su respectivo análisis, se propone el siguiente plan de contingencia.

Riesgo	Estrategia
R01	Reorganizar el trabajo, absorber el trabajo tercerizado por el equipo principal, y llevar un control del estado del suproyecto mobile para poder abordarlo de la manera más rápida posible.

Riesgo	Estrategia
R04	Realizar estimaciones pesimistas, considerando tiempos extra para abarcar posibles complicaciones. Además, realizar revisiones frecuentes del cronograma y modificaciones anticipadas si fueran necesarias.

Riesgo	Estrategia
R05	Centrar el diseño del sistema en la experiencia del usuario, mediante interfaces simples, intuitivas y visualmente atractivas. Evitar implementaciones que comprometan la usabilidad o dificulten la interacción con la aplicación.

Riesgo	Estrategia
R06	Mantener un contacto fluido con las referentes funcionales, compartiendo avances mediante videos demostrativos que expongan las funcionalidades desarrolladas. Asimismo, realizar consultas oportunas ante dudas específicas, con el objetivo de recibir retroalimentación y alinear el desarrollo con las expectativas del proyecto.

## Análisis FODA

Durante las etapas iniciales del proyecto se realizó un análisis FODA (Fortalezas, Oportunidades, Debilidades y Amenazas) con el objetivo de evaluar tanto las capacidades internas del equipo como el contexto externo en el que se desarrollaría el sistema. Esta herramienta permitió anticipar desafíos, identificar fortalezas y definir estrategias para orientar mejor las decisiones de planificación y desarrollo.

### Fortalezas

**Beneficio para la comunidad:** El proyecto, que forma parte de, aborda una necesidad real y significativa en la comunidad al proporcionar una herramienta que mejora la calidad de vida de las personas con celiaquía, así como a profesionales de la salud y empresas relacionadas.

**Enfoque multidisciplinario:** La combinación de una aplicación mobile y una aplicación web permite un enfoque que abarca tanto las necesidades de los usuarios celíacos como las de los profesionales de la salud involucrados.

**Respaldo profesional:** Se cuenta con la validación de profesionales de la salud del HIGA, especialistas en el tema y futuros usuarios finales, lo que constituye un beneficio significativo para el proyecto y el producto.

**Trabajo en equipo:** Los integrantes del proyecto final son compañeros desde hace muchos años, han realizado muchos trabajos juntos y se conocen muy bien. Además, comparten una relación profesional y una amistad muy estrecha. Todo esto hace que el trabajo en equipo sea algo natural para ambos.

**Tercerizar el trabajo:** Se decidió tercerizar el desarrollo del frontend mobile para acotar los tiempos del proyecto y poder realizar las tres aplicaciones en paralelo (backend, frontend mobile y frontend web). Esto aporta una gran ventaja a los tiempos del proyecto y les permite a los alumnos entrenar aún más sus habilidades en gestión de proyectos.

**Escalabilidad del producto:** Se planea concebir el producto de manera que, al finalizar el proyecto, sea viable expandir sus funcionalidades de manera eficiente por otro equipo de desarrollo.

**Perspectiva en primera persona:** Uno de los integrantes del equipo, Julián, es celíaco. Esto permitió contar con una mirada auténtica y cercana a la problemática desde el inicio del proyecto, aportando no solo conocimiento técnico sino también experiencia personal en el uso de productos, seguimiento de la dieta y desafíos cotidianos.

## Oportunidades

**Población de usuarios en crecimiento:** Gracias a los avances en la ciencia y tecnología, el diagnóstico de la celiaquía es cada vez más frecuente. Este progreso ha llevado a un aumento en el número de personas diagnosticadas con celiaquía, lo que amplía la base de usuarios potenciales para el sistema.

**Inserción dentro del ámbito profesional:** El sistema podría establecerse como una herramienta útil dentro del ámbito de la salud. De esta manera, los profesionales podrían promover su uso y aumentar su adopción en la comunidad médica local.

**Potencial de expansión:** A partir de la instalación del sistema en el Partido de General Pueyrredón, se podría considerar su expansión a otras regiones de la provincia de Buenos Aires, o potencialmente en toda la Argentina.

**Modelo de negocio:** Posibilidad de incorporar un modelo de negocio con empresas proveedoras de productos aptos para celíacos.

## Debilidades

**Inexperiencia en gestión de proyectos de envergadura similar:** Los miembros del equipo carecen de experiencia previa en la gestión de proyectos, por lo que es altamente probable que deban enfrentarse a nuevos desafíos relacionados con la planificación, ejecución y control de las tareas, así como la gestión de los recursos y los plazos.

**Dependencia de un colaborador externo:** La tercerización es un arma de doble filo, ya que es altamente dependiente del esfuerzo y compromiso de una persona externa al grupo. La estimación de tiempos se vuelve más compleja, y la coordinación del grupo en su totalidad también.

**Carga académica y laboral simultánea:** Ambos integrantes del equipo tienen compromisos laborales y académicos, lo que puede dificultar la dedicación sostenida y comprometer los tiempos estimados de desarrollo.

## Amenazas

**Separación del grupo:** Aunque existe una fuerte amistad y compromiso entre los miembros del grupo, es importante considerar la posibilidad de separación debido a situaciones externas. La mudanza de uno de los integrantes a otra ciudad, la virtualidad, o cambios en el trabajo actual de los integrantes pueden ser factores a tener en cuenta.

**Problemas de seguridad de datos:** Aunque la aplicación maneja datos que no son altamente críticos, aún contiene información sensible que debe permanecer íntegra. La carencia de técnicas avanzadas de seguridad informática podría convertirse en un problema si la criticidad de estos datos aumenta con el tiempo.

**Falta de interés o adopción por parte de los usuarios finales:** A pesar de diseñar una herramienta pensada para resolver una necesidad real, existe el riesgo de que las personas celíacas o profesionales de la salud no adopten activamente el sistema, por desconocimiento, falta de tiempo o resistencia al uso de nuevas tecnologías.

**Incertidumbre institucional en el HIGA:** El proyecto cuenta con el apoyo de profesionales específicos del hospital, pero no existe garantía en la continuidad del interés o respaldo por parte de la institución en caso de cambios de personal o prioridades internas.

## Organización del proyecto

El proyecto fue organizado en tres subproyectos con la intención de desarrollarlos de forma paralela, con objetivos bien definidos y responsables asignados para cada uno. Esta estrategia permitió dividir tareas de manera eficiente, favorecer el trabajo simultáneo y asegurar un seguimiento claro del avance de cada módulo. La organización modular también facilitó la adaptación a imprevistos, como la rotación de roles, y brindó mayor flexibilidad en la planificación general.

A continuación, se describen los tres subproyectos, sus principales objetivos y sus responsables.

### Subproyecto Web

Este subproyecto, con Lautaro como responsable, tiene por objetivo el desarrollo de una herramienta robusta destinada a profesionales de la salud del HIGA que permita gestionar la información del sistema. A través de una interfaz web, esta herramienta debe permitir a los profesionales cargar y mantener datos actualizados sobre alimentos aptos, recetas seguras, puntos de compra recomendados y cuidados para el hogar. Adicionalmente, debe proporcionar información estadística que permita analizar el comportamiento de los usuarios celíacos, facilitando así la toma de decisiones clínicas basadas en datos reales y actualizados.

### Subproyecto Mobile

Este subproyecto, inicialmente asignado a un colaborador técnico, y posteriormente asumido por el equipo principal, tiene como objetivo el desarrollo de una aplicación mobile diseñada para acompañar a las personas celíacas en su día a día, ofreciendo una plataforma accesible y gratuita desde donde pueden consultar alimentos validados, explorar recetas seguras, y localizar puntos de compra confiables en la ciudad de Mar del Plata. Además, busca proporcionar una herramienta visual intuitiva para confeccionar

platos balanceados nutricionalmente, promoviendo hábitos saludables y adaptándose a distintos tipos de dieta como la vegetariana o la vegana.

## Subproyecto Backend

El Subproyecto Backend, a cargo de Julián, tiene como objetivo coordinar la comunicación entre las distintas aplicaciones y servicios del sistema. Su objetivo abarca el desarrollo del núcleo de la lógica de negocio, integrando los módulos construidos en los demás subproyectos, aplicando mecanismos de seguridad y garantizando la persistencia y consistencia de los datos.

## Estimación inicial

La estimación de tiempos del proyecto se estructuró en cinco grandes bloques: uno para cada uno de los tres subproyectos técnicos (backend, frontend web y frontend mobile), y dos adicionales correspondientes a tareas de gestión y a la redacción del presente informe. Esta división permitió organizar el trabajo de forma clara y facilitar el seguimiento de avances, distribuyendo de forma equilibrada las responsabilidades a lo largo del desarrollo.

En lugar de aplicar una estimación tradicional basada en fases secuenciales (análisis, diseño, implementación y pruebas), se optó por un enfoque modular. Esta elección respondió a la necesidad de obtener mayor granularidad y precisión en las estimaciones, considerando la limitada experiencia previa del equipo en proyectos de esta envergadura.

La estimación de un proyecto es inherentemente compleja. Sin embargo, en este caso particular, el equipo ya contaba desde el inicio con un análisis avanzado heredado de equipos anteriores, lo que permitió identificar claramente los módulos a desarrollar. Además, se determinó que este enfoque permitiría llevar un mejor control del colaborador técnico, ya que se podría llevar un seguimiento más detallado de sus avances.

Cada módulo fue abordado individualmente, partiendo del análisis previo ya disponible y contemplando específicamente sus etapas de diseño, desarrollo y testing. Este enfoque permitió un manejo más claro y efectivo de cada módulo, aunque incrementó la complejidad en la gestión general del proyecto. Este *trade-off* se asumió conscientemente, ya que era necesario establecer un método viable para realizar estimaciones precisas de los tiempos en las etapas tempranas del proyecto.

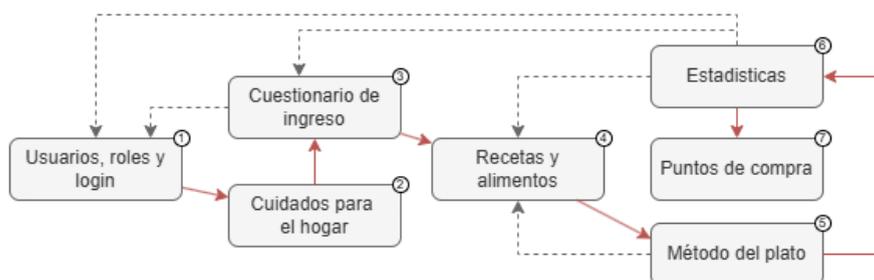
A partir de este enfoque se identificaron siete módulos principales: gestión de usuarios, cuestionario de ingreso, cuidados para el hogar, recetas y alimentos, puntos de compra, estadísticas y método del plato. Esta organización, si bien aportó claridad, introdujo un nuevo desafío: establecer un orden lógico y técnicamente viable para su implementación. Para esto, se realizaron distintos análisis que ayudaron a definir un **camino crítico** de desarrollo.

El primer análisis consistió en clasificar los módulos según su nivel de complejidad técnica e incertidumbre funcional. El objetivo fue priorizar aquellos módulos más simples y de menor ambigüedad, de modo que el equipo pudiera ganar experiencia y consolidar conocimientos tecnológicos antes de abordar componentes más desafiantes. La **Figura 1** muestra esta clasificación, donde módulos como Cuestionario y Cuidados para el hogar fueron identificados como los más accesibles, mientras que Estadísticas, Puntos de compra y Método del plato se posicionaron como los más inciertos o técnicamente exigentes, siendo programados para etapas posteriores.

	Dificultad baja	Dificultad media	Dificultad alta
Baja incertidumbre	Cuidados	Cuestionario	
Media incertidumbre		Login y usuarios Recetas y alimentos	
Alta incertidumbre		Estadísticas Puntos de compra	Método del plato

**Figura 1.** Clasificación de módulos según su complejidad e incertidumbre

En paralelo, se llevó a cabo un segundo análisis enfocado en las dependencias funcionales entre módulos. A partir de un esquema de interrelaciones, se identificó qué componentes requerían el soporte de otros para su funcionamiento, permitiendo establecer un camino crítico de desarrollo. La **Figura 2** ilustra esta interdependencia: en ella, el módulo de gestión de usuarios aparece como componente inicial del sistema, siendo la base a partir de la cual se construyó la secuencia general de implementación. En ella, se muestran las dependencias funcionales en flecha punteada gris, y el camino crítico establecido con una flecha roja y una enumeración.



**Figura 2.** Esquema de interrelaciones entre módulos y camino crítico de desarrollo.

Finalmente, se consideró una tercera dimensión: la coordinación entre subproyectos en un entorno de desarrollo paralelo. En las primeras etapas, se priorizó el trabajo conjunto entre backend y frontend mobile, con el objetivo de que Julián — desarrollador backend — pudiera colaborar estrechamente con el colaborador técnico asignado al desarrollo mobile, especialmente durante la etapa de aprendizaje y asentamiento. Esto permitiría preparar todos los endpoints necesarios para que, en las siguientes etapas, el frontend web pudiera avanzar con mayor autonomía.

Esta planificación en etapas, basada en criterios de complejidad, dependencia y sinergia entre roles, sentó las bases del diagrama de Gantt inicial del proyecto. La **Figura 3** muestra este diagrama, con los tiempos estimados para cada tarea y la proyección de finalización en la semana 32.

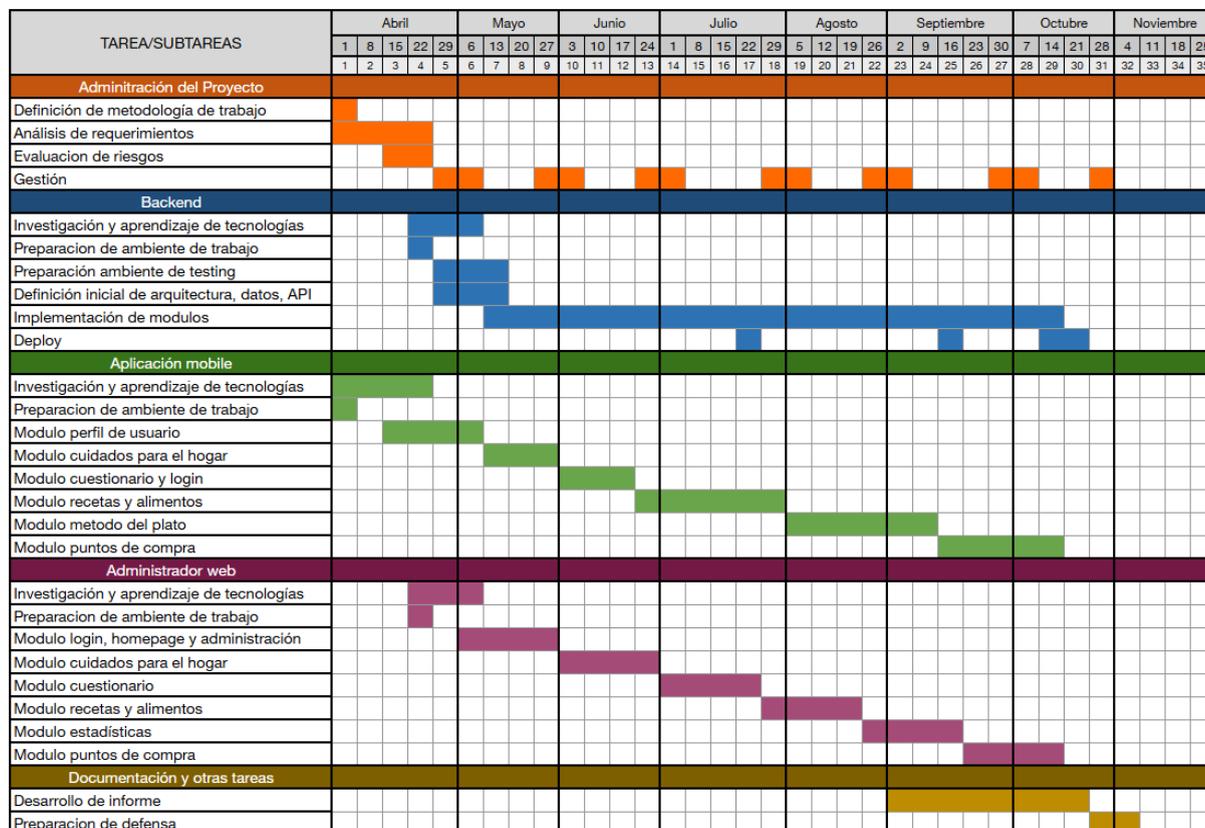


Figura 3. Diagrama de Gantt inicial.

Se estimó una dedicación promedio de **10 horas semanales por integrante**, con una fecha de inicio formal el 1 de abril de 2024 y una finalización prevista para el **28 de octubre** del mismo año (32 semanas).

Durante las primeras etapas del proyecto, la atención se centraría en tareas exclusivas de gestión. Estas actividades contemplarían la organización inicial del proyecto, la planificación detallada de las reuniones semanales con el colaborador técnico y la definición de los tiempos dedicados al aprendizaje y práctica de las tecnologías seleccionadas. Esta etapa sería crítica para asegurar una base sólida sobre la cual construir el desarrollo posterior, garantizando que todos los integrantes tuvieran claros los objetivos técnicos y metodológicos, además de adquirir las habilidades mínimas necesarias para avanzar eficientemente en etapas futuras.

Respecto a la estimación de tiempos para los subproyectos frontend, considerando los análisis realizados por el equipo, se llegaron a las siguientes conclusiones: en el frontend mobile, los análisis identificaron claramente que los módulos de "Recetas y alimentos" y "Método del plato" serían significativamente más complejos a nivel técnico, lo que implicó asignarles tiempos considerablemente mayores respecto a otros módulos menos

exigentes. Esta diferencia permitiría ajustar de manera precisa la carga laboral, brindando al colaborador técnico el margen suficiente para enfrentar desafíos técnicos mayores sin comprometer la calidad o los plazos generales.

Por su parte, en el frontend web, la estrategia de estimación resultó en una distribución equilibrada de 4 semanas por módulo. Esto se debe a que los módulos iniciales fueron identificados como más simples desde una perspectiva lógica y técnica, pero se asumió que durante estas primeras etapas las habilidades técnicas del desarrollador serían todavía limitadas. En contraste, para los módulos finales, aunque representarían mayores retos en el diseño, la complejidad lógica sería relativamente estable, ya que se trata de un sistema de gestión. Sin embargo, la expectativa era que, a medida que avanzara el proyecto, el desarrollador incrementaría notablemente su expertise, lo que permitiría mantener constante la estimación de tiempos al compensar una mayor complejidad técnica con una mayor destreza adquirida en las etapas previas.

En resumen, la metodología aplicada para la estimación inicial combinó criterios de complejidad técnica, incertidumbre funcional, interdependencia entre módulos, y capacidades evolutivas del equipo técnico. Este enfoque permitió aprovechar todo el análisis realizado y plasmarlo en un diagrama que, aunque parecía ser sólido en la teoría, se evidenciará a continuación que en la práctica los proyectos rara vez se ajustan exactamente a lo planificado, requiriendo flexibilidad y capacidad de adaptación frente a imprevistos.

## Desvíos y ajustes de cronograma

A pesar de la planificación detallada, el proyecto enfrentó una serie de imprevistos personales y técnicos que motivaron dos reestimaciones sucesivas, las cuales impactaron significativamente en los plazos de entrega. La **Figura 4** refleja la primera replanificación parcial, mientras que la **Figura 5** muestra el Gantt final, con tiempos de baja marcados en gris para cada subproyecto afectado.

### Primer ajuste — Agosto de 2024

Uno de los primeros factores que impactó en el ritmo del proyecto fue el cambio de residencia de Lautaro. A fines de 2023, fue convocado a trabajar en el Centro Atómico de Bariloche, lo que implicó múltiples mudanzas, adaptación al nuevo entorno laboral y viajes recurrentes a Mar del Plata para rendir exámenes. Durante este período (aproximadamente entre las semanas 4 y 8), el Subproyecto Web quedó prácticamente detenido, tal como se observa en el Gantt final.

En ese contexto, se tomó además la decisión de migrar el stack tecnológico del frontend web de React a Angular, principalmente porque Lautaro comenzó a trabajar en un proyecto laboral que utilizaba Angular como tecnología principal. Unificar la tecnología entre los proyectos permitió optimizar el tiempo de aprendizaje y evitar la dispersión que implicaba abordar dos stacks diferentes al mismo tiempo. Esto implicó necesariamente una reestimación de las tareas, ya que se amplió el tiempo en aquellas actividades que presentaban mayor complejidad o incertidumbre.

Por otro lado, la incorporación del colaborador técnico para el Subproyecto Mobile exigió una dedicación mucho mayor a la prevista inicialmente. Además de afrontar la curva de aprendizaje propia de las tecnologías mobile, fue necesario capacitarlo en herramientas como Docker, API REST y en conceptos clave del stack técnico, lo que impactó de lleno en la autonomía y en el avance esperado de ese subproyecto.

El análisis de los desvíos acumulados durante estas primeras etapas llevó a la conclusión de que la baja del colaborador era cada vez más probable y, aún si se mantenía, el ritmo de avance no era suficiente para cumplir con los plazos estimados. Por este motivo, se agregaron las horas necesarias para la gestión y acompañamiento del Subproyecto Mobile, las cuales no habían sido consideradas en la estimación inicial, y se diseñó un plan de contingencia realista para afrontar la situación del colaborador. Este plan contemplaba el siguiente escenario: mientras el colaborador continuara siendo parte del proyecto, los integrantes avanzarían en sus propios subproyectos y luego de terminar sus desarrollos, se sumarían para pulir detalles de la aplicación mobile; si se producía la baja antes de la finalización del desarrollo mobile, el tiempo estimado permitiría reorganizarse, cubrir la curva de aprendizaje y trabajar en conjunto para terminar la aplicación a tiempo.

Este plan fue diseñado considerando el contexto particular del proyecto: se trataba de un entorno académico, con recursos sumamente limitados y un colaborador que participaba de manera voluntaria. Por esto, se optó por realizar una estimación de tiempos del desarrollo mobile considerando un tiempo adicional de aproximadamente un 20% para poder cubrir ambas situaciones planteadas en el plan de contingencia. Si bien esta no es una práctica recomendable en el ámbito profesional, fue la solución más realista para poder cubrir las distintas situaciones previstas en el plan de contingencia.

En paralelo, el backend logró sostener un flujo de trabajo relativamente constante, consolidando las funcionalidades necesarias para que los demás módulos pudieran retomarse más adelante. Frente a todos estos factores, se realizó la primera reestimación del cronograma, extendiendo la fecha de entrega al 3 de marzo de 2025 (semana 48).

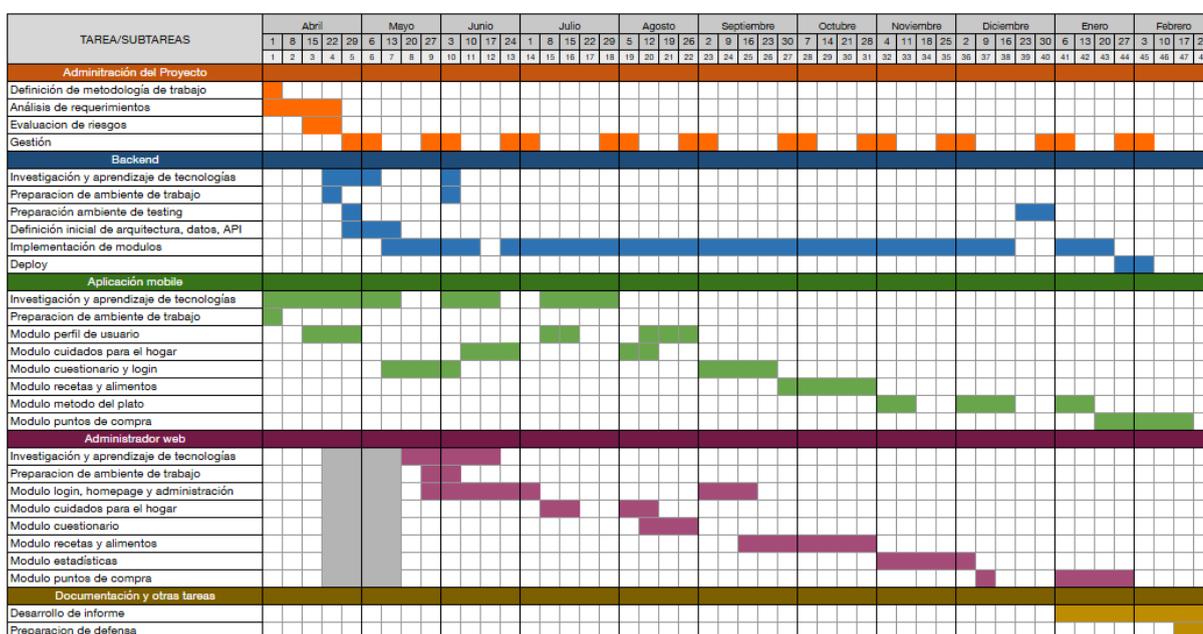


Figura 4. Diagrama de Gantt del proyecto luego del primer ajuste.

## Segundo ajuste — Enero de 2025

A fines de noviembre, Lautaro recibió la muy poco posible noticia de que su contrato laboral no sería renovado, lo que redujo nuevamente su disponibilidad debido a la búsqueda de empleo y capacitaciones. Al mismo tiempo, ocurrió lo previsto, pero en un momento muy inoportuno: el colaborador técnico decidió desvincularse del proyecto. Ambos eventos impactaron de lleno en la continuidad de los subproyectos mobile y web.

Frente a este escenario, se activó el plan de contingencia previsto. Además, se incorporó tiempo adicional al cronograma para contemplar el período de baja de Lautaro. A su vez, se reorganizaron por completo las tareas del Subproyecto Mobile. Inicialmente, Julián asumió en soledad el avance de la lógica y la integración con el backend, mientras Lautaro estaba abocado a pruebas técnicas y entrevistas laborales. Una vez que Lautaro regularizó su

situación y pudo reincorporarse, tomó efectivamente las tareas correspondientes al diseño visual y a la interfaz mobile, trabajando en conjunto con Julián para completar el desarrollo.

Esta reestructuración, alineada con el plan de contingencia, permitió que ambos integrantes se repartieran de manera más eficiente las responsabilidades del Subproyecto Mobile, compensando la baja del colaborador externo. El Gantt final refleja en color gris las pausas de los subproyectos mobile (semanas 23–32) y web (semanas 35–40), y en color cian la reactivación del mobile, esta vez a cargo de Lautaro y Julián.

Este segundo ajuste formalizó la extensión de la fecha de finalización a la semana 57, coincidiendo simbólicamente con el 5 de mayo de 2025, Día Internacional del Celíaco.

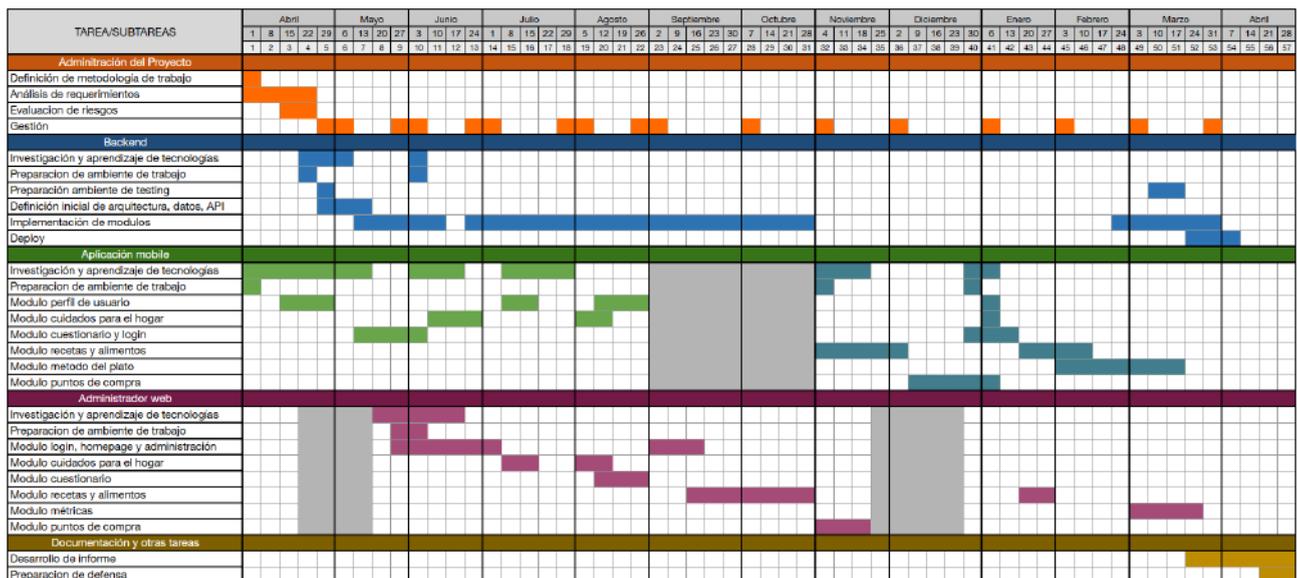


Figura 5. Diagrama de Gantt del proyecto luego del segundo ajuste.

## Proceso

Para llevar adelante el proyecto fue necesario definir las metodologías de trabajo, diseño y desarrollo a utilizar. Aun así, ciertas metodologías inicialmente adoptadas no permanecieron estáticas, sino que se fueron adaptando a medida que el proyecto fue avanzando. A continuación, se detallan cuáles fueron las metodologías elegidas para cada aspecto del proyecto y cómo estas se fueron ajustando a lo largo del proceso.

## Metodología de trabajo

El desarrollo del sistema se planteó inicialmente bajo un **enfoque en cascada por módulo**, ya que resultaba la forma más coherente de organizar un desarrollo paralelo de funcionalidades en distintos subsistemas. Aun así, este enfoque debió adaptarse al contexto previo del proyecto, donde parte del análisis ya se encontraba realizado. Por ello, se optó por una etapa inicial de análisis general, tras la cual se abordó el desarrollo de cada módulo funcional de manera secuencial, aplicando una metodología en cascada interna para las etapas de diseño, implementación y pruebas.

En etapas avanzadas del proyecto, se propuso la realización de videos demostrativos para validar funcionalidades junto a las referentes funcionales. A partir de esta iniciativa, se decidió adoptar un enfoque más iterativo en los módulos de alimentos, recetas y método del plato, anticipando posibles modificaciones futuras. Con el feedback recibido, se llevó a cabo una segunda iteración del modelo en cascada para incorporar los ajustes necesarios y volver a validar las funcionalidades implementadas.

Adicionalmente, tras la baja del colaborador técnico, el orden de prioridad de las tareas a desarrollar se vio modificado. Los subproyectos web y backend debieron ser pausados, ya que el equipo principal tuvo que concentrar sus esfuerzos en el Subproyecto Mobile. Ante este cambio, se optó por adoptar un enfoque iterativo como metodología general para dicho subproyecto, ya que resultaba la forma más simple de adaptarse a la nueva situación. El diseño del sistema fue validado inicialmente mediante videos demostrativos, y en una segunda instancia se validaron las funcionalidades específicas de cada módulo.

Como herramienta principal de gestión, y aprovechando la experiencia previa del equipo, se optó por utilizar *Notion*. Se configuró un espacio personalizado para centralizar tareas, ideas, reuniones y registrar horas trabajadas. Sin embargo, su uso resultó poco práctico, por lo que se optó por simplificar la metodología y cambiar a una planilla de *Excel* para el seguimiento de tareas, lo que agilizó la gestión y permitió enfocar el tiempo en el desarrollo.

## Metodología de diseño

El diseño cumplió un rol clave en el desarrollo del proyecto, tanto desde la perspectiva visual (interfaces y experiencia de usuario) como desde la estructuración interna del sistema (modelo de datos, componentes e interfaces de comunicación).

El proceso de diseño de cada módulo se adaptó según su complejidad y características específicas. En la gran mayoría de casos, se partió del desarrollo de las interfaces visuales, definiendo la presentación de la información y el flujo de la navegación de manera tal que resultara intuitiva y clara para el usuario. A partir de estas decisiones, se diseñaban las estructuras de datos necesarias y se planificaba la solución técnica que soportaría esas interfaces. Sin embargo, en módulos donde la incertidumbre acerca de los datos era menor, el enfoque fue inverso: se comenzó por el diseño del modelo de datos para luego construir las interfaces visuales sobre esa base. Esto aportó flexibilidad a este proceso, permitiendo adaptar el diseño de cada módulo según sea conveniente, garantizando un equilibrio entre la experiencia de usuario y la arquitectura interna del sistema.

En cuanto a las herramientas utilizadas, el proceso de diseño fue progresivo y práctico: inicialmente se recurrió a *Paint* para realizar bocetos rápidos de ideas y flujos de pantalla. Posteriormente, estos bocetos fueron trasladados a *Figma*, donde se construyeron prototipos más elaborados, colaborativos y reutilizables. El uso de Figma permitió visualizar el producto final de manera anticipada y sirvió como referencia constante para la implementación en código. Además, para la representación de diagramas de arquitectura y modelado de entidades y relaciones en la base de datos, se utilizó Draw.io, aprovechando su flexibilidad para crear esquemas claros y actualizables durante las distintas etapas del proyecto.

## Metodología de desarrollo

Para el seguimiento del estado del desarrollo, como ya fue mencionado, se optó por una planilla de Excel que listaba los requerimientos de cada módulo junto con su estado. Estas planillas se actualizaban durante las reuniones de gestión, a medida que los avances eran implementados. La definición de prioridades y el orden de implementación se acordaban en conjunto, considerando el contexto del proyecto y los posibles bloqueos entre módulos.

Adicionalmente, como los integrantes del equipo contaban con experiencia en el uso de Git, se optó por crear tres repositorios independientes (uno por subproyecto), asignando la responsabilidad de cada uno a su respectivo desarrollador.

En los proyectos individuales (backend y frontend web), los avances se gestionaron directamente sobre la rama principal, ya que no existía riesgo de conflictos al ser desarrollados por una única persona. En cambio, el Subproyecto Mobile, que finalmente

fue abordado de forma colaborativa, adoptó una estrategia distinta: se estableció una rama principal (master) que representaba la última versión estable de la aplicación, y cada integrante trabajaba en su propia rama individual. La integración de cambios se realizaba durante reuniones semanales, donde se revisaban los avances y se resolvían posibles conflictos antes de unificar los cambios en la rama principal.

## Uso de contenedores Docker

Uno de los principales desafíos en la etapa de desarrollo, fue simplificar la integración entre el backend y los demás componentes sin introducir complejidades innecesarias en los entornos de desarrollo, sobre todo considerando que uno de los integrantes contaba con menos experiencia técnica. Además, como el proceso de integración es algo que ocurre de forma constante, es importante que sea eficiente para ahorrar el mayor tiempo y esfuerzo posible. Esto motivó la búsqueda de una solución que permitiera encapsular el backend, reduciendo la necesidad de configuraciones manuales y asegurando que todo el equipo pudiera trabajar con una versión funcional y actualizada del sistema.

La herramienta elegida fue *Docker*, complementada con *Docker Hub* como repositorio remoto de imágenes. Esta combinación permitió simplificar significativamente el proceso de integración entre componentes.

Cuando el backend implementaba una nueva funcionalidad, se generaba una imagen actualizada y, en caso de ser necesario, un nuevo *dump* de base de datos. Esta imagen se publicaba en Docker Hub y el *dump* se compartía con el equipo. Luego, los desarrolladores frontend actualizaban su entorno descargando la nueva imagen y el *dump*, pudiendo así trabajar con la versión actualizada del backend sin configuraciones adicionales.

Esta estrategia no solo facilitó la integración entre los diferentes componentes del sistema, sino que también permitió agilizar las pruebas y reducir errores derivados de diferencias en los entornos de desarrollo.

## Comunicación

La comunicación entre los integrantes del equipo principal fue constante a lo largo del proyecto. Las reuniones se dividieron en dos tipos: reuniones de gestión, orientadas a la planificación y seguimiento del proyecto y reuniones de desarrollo, en las que se abordaban aspectos técnicos, se definían soluciones y se trabajaba en conjunto mediante *pair programming*.

Con respecto al colaborador técnico, se plantearon reuniones de seguimiento para analizar sus avances y planificar las actividades futuras en función de su disponibilidad. Adicionalmente, cuando fue necesario, se llevaron a cabo sesiones de capacitación para introducirlo en las tecnologías utilizadas en el proyecto.

En cuanto a la interacción con el director del proyecto, se mantuvo comunicación con distinta frecuencia según la etapa. En las primeras instancias, las reuniones se centraron en revisar y comprender los requerimientos funcionales previamente definidos por el director en conjunto con la referente. Más adelante, los encuentros estuvieron orientados a la selección de tecnologías y a la definición de aspectos técnicos clave.

Durante la mayor parte del desarrollo, las reuniones con el director se llevaron a cabo bajo demanda, es decir, cuando surgían necesidades específicas. En los casos en que las consultas no requerían un encuentro formal, la comunicación se realizó mediante mensajes de WhatsApp, esto permitió resolver dudas puntuales de manera ágil. No obstante, el director atravesó una licencia médica que se extendió a lo largo de todo el proyecto, lo que afectó considerablemente su disponibilidad. A pesar de esto, se mantuvo siempre predispuesto a colaborar y ofrecer acompañamiento. Esta situación generó cierta dificultad, ya que existía la percepción de que las consultas podían resultar inoportunas en un contexto personal complejo. Aun así, fue posible encontrar un equilibrio que permitió avanzar en el desarrollo del sistema.

Hacia el final del proyecto, la frecuencia de estos intercambios disminuyó aún más como consecuencia directa de la licencia médica. Ante esta situación, fue necesario que el equipo asumiera un mayor grado de autonomía, tomando decisiones y resolviendo problemáticas según su propio criterio. Un ejemplo fue la elaboración del presente informe: si bien al inicio se contó con el acompañamiento del director en la definición del índice, los objetivos y el alcance, posteriormente su disponibilidad se vio limitada, por lo que la redacción de la mayor parte del contenido debió ser realizada por los propios integrantes, sin posibilidad de una validación completa del contenido ni de su redacción. A pesar de esta situación, el equipo buscó métodos alternativos para validar el informe, compartiéndolo tanto con personas familiarizadas con la redacción de este tipo de documentos como con personas no técnicas. En ambos casos, se recibió un feedback positivo en relación con la redacción, la estructura y el contenido general del trabajo.

En cuanto a la comunicación con la referente funcional, se optó por enviar previamente cuestionarios y videos explicativos que contenían los avances y las dudas a tratar. Esta estrategia permitió optimizar tanto la frecuencia como la duración de los encuentros, favoreciendo instancias más dinámicas y enfocadas. Dado que el documento de requerimientos ya se encontraba elaborado y validado desde etapas tempranas, no fue necesario mantener reuniones extensas o frecuentes. Los intercambios se centraron en validar nuevas propuestas y presentar las funcionalidades desarrolladas, maximizando el aprovechamiento del tiempo disponible.

## Control de tiempos y avances

Durante el desarrollo del proyecto fue necesario controlar los avances realizados en relación con las estimaciones previamente definidas, tanto en la estimación inicial como en las correspondientes a los desvíos. Con el objetivo de monitorear el progreso y detectar posibles desviaciones respecto a lo planificado, se implementó una metodología de control basada en el seguimiento periódico del cronograma.

Como base de esta estrategia se utilizó el diagrama de Gantt inicial. A partir de él, se organizaron reuniones mensuales destinadas a revisar los avances concretos alcanzados durante cada período, comparándolos con lo planificado. Para facilitar este seguimiento, se aplicó un sistema visual directamente sobre el diagrama: las tareas planificadas se representaban en color gris, mientras que las tareas efectivamente ejecutadas se señalaban con el color correspondiente, permitiendo identificar de forma clara tanto el cumplimiento como los cambios en el orden de ejecución.

Este enfoque brindó un control más preciso sobre el estado del proyecto y facilitó la toma de decisiones, como por ejemplo al momento de gestionar solicitudes de ajustes ante la cátedra. Si bien mantener este seguimiento implicó un esfuerzo considerable —especialmente debido al grado de detalle requerido por el cronograma—, resultó útil para anticiparse a determinadas situaciones. Sin ir más lejos, uno de los fundamentos que motivó el primer ajuste de cronograma fue la identificación, a través de este control, de un ritmo de ejecución que no se correspondía con lo planificado.

En cuanto a los avances, en la práctica se presentaron múltiples situaciones que alteraron el flujo previsto del desarrollo. Las bajas inesperadas, las dificultades para coordinar reuniones y otros contratiempos derivados de la dinámica del equipo generaron modificaciones en el orden de las tareas con el objetivo de minimizar tiempos de inactividad. En varios casos, decisiones de diseño que requerían ser analizadas por ambos integrantes debieron posponerse, lo que llevó a avanzar en otros módulos o tareas independientes mientras se lograba establecer la coordinación necesaria. Esta forma de trabajar permitió aprovechar el tiempo y mantener cierto ritmo de trabajo, aunque implicó una reorganización constante de prioridades y un esquema de ejecución más flexible de lo originalmente planteado. En este sentido, el control realizado sobre el cronograma sirvió para marcar el camino de ejecución en contraposición al camino establecido en un principio.

A modo ilustrativo, en la **Figura 6** se muestra cómo se aplicó esta metodología sobre el diagrama de Gantt, visualizando los avances de un mes al siguiente mediante el sistema de colores utilizado.

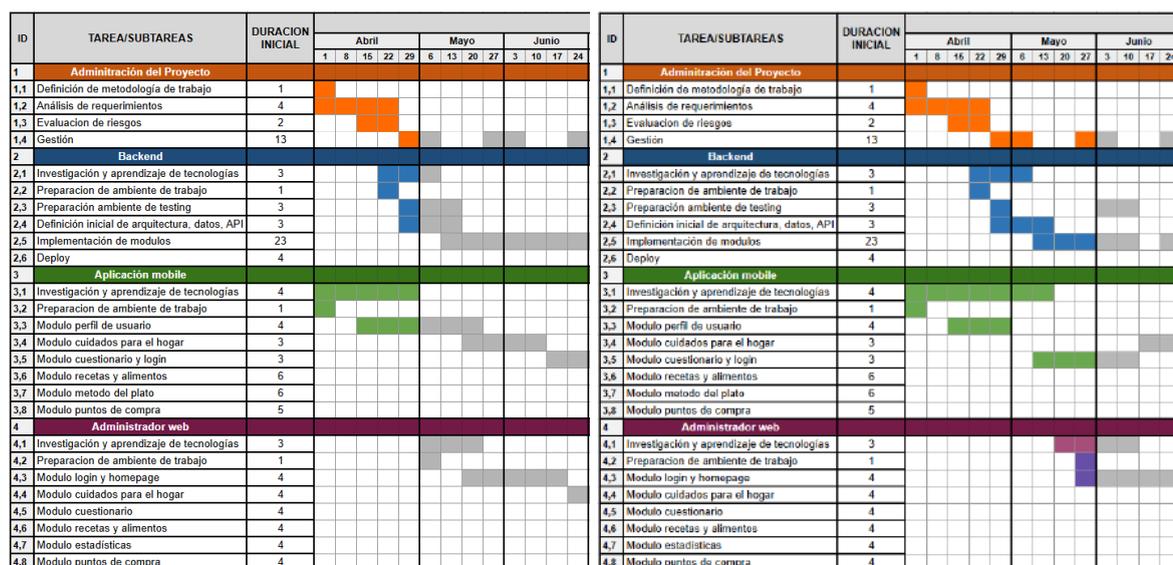


Figura 6. Metodología de control de tiempos y avances.

## Gestión del colaborador técnico

Desde el inicio del proyecto, la gestión del colaborador técnico fue un eje clave para asegurar el avance coordinado del Subproyecto Mobile. Como primera medida, se estimaron los tiempos de desarrollo en base al conocimiento previo del equipo y la disponibilidad informada por el colaborador. Dichas estimaciones fueron validadas en reuniones iniciales, donde se presentaron los objetivos, los plazos y se establecieron expectativas claras. El colaborador aprobó el plan de trabajo, dando el visto bueno para avanzar bajo esa estructura.

Durante la etapa de desarrollo, se implementaron reuniones semanales de control de avances, en las que el colaborador presentaba el progreso, explicaba cómo estaba estructurando el código y consultaba dudas funcionales. Estas instancias servían para dar feedback inmediato y resolver obstáculos técnicos que iban surgiendo.

Adicionalmente, desde las primeras etapas se organizaron sesiones específicas de capacitación y pair programming. Estas instancias fueron necesarias debido a que la curva de aprendizaje del colaborador resultó más pronunciada de lo previsto inicialmente, especialmente en aspectos técnicos clave del proyecto, como la utilización de API REST, Docker y otras herramientas fundamentales. El objetivo era acompañar al colaborador en la incorporación de nuevas herramientas y acelerar la resolución de problemas técnicos. Sin embargo, el tiempo requerido para estas capacitaciones superó ampliamente la estimación inicial, generando desvíos que impactaron en el cronograma global del proyecto.

En paralelo, se mantuvo un control mensual del avance general, en el marco de las reuniones de gestión. Allí se analizaban los progresos de todos los subproyectos y se reorganizaban tareas y prioridades según los resultados obtenidos hasta el momento. Fue

en estas instancias donde comenzaron a detectarse desvíos entre lo planificado y lo efectivamente implementado por el colaborador técnico. La diferencia entre las expectativas y los resultados obtenidos se fue acentuando con el correr de las semanas, obligando a reconsiderar los tiempos y a ajustar la planificación del equipo principal.

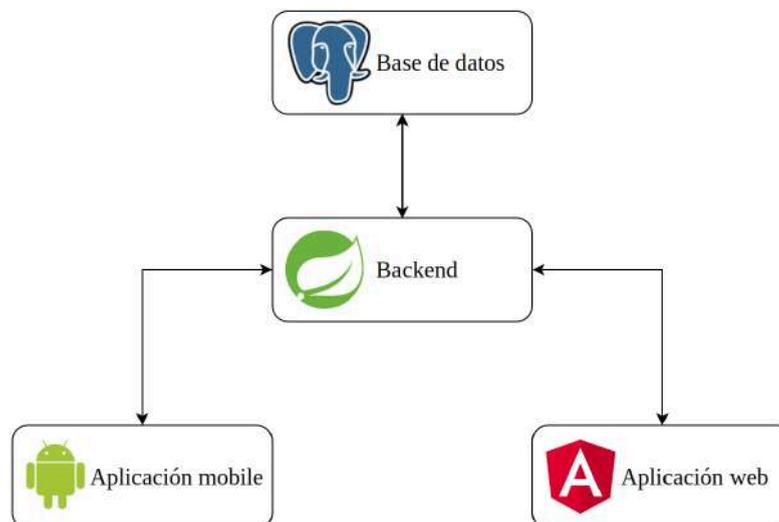
En conclusión, si bien en un principio se planificó que el colaborador pudiera desarrollar sus tareas con un alto grado de autonomía, desde las primeras etapas del proyecto se evidenció que esto no sería viable. Esta realidad obligó a reorientar la estrategia, promoviendo un trabajo mucho más colaborativo a través de capacitaciones, pair programming y acompañamiento técnico constante. Estas acciones, que no estaban previstas en la planificación original, demandaron un esfuerzo y una dedicación mucho mayor a la esperada. Aunque la desvinculación del colaborador finalmente respondió a motivos personales ajenos al proyecto, el proceso aportó aprendizajes relevantes en gestión de equipos, mentoring y adaptación ante escenarios desfavorables, así como una mirada más crítica sobre la complejidad de gestionar recursos humanos en proyectos de estas características.

# Diseño del sistema

## Arquitectura

La arquitectura implementada para este sistema, como se muestra en la **Figura 7**, es una arquitectura cliente-servidor de tres capas, un backend desarrollado en Spring Boot expone una *API RESTful* consumida tanto por el administrador web, desarrollado en Angular, como por la aplicación mobile, desarrollada en Kotlin. Esta arquitectura permite que múltiples clientes interactúen de manera simultánea con la capa de lógica y datos, promoviendo la escalabilidad del sistema.

El backend cumple el rol de orquestador principal del sistema: gestiona la lógica de negocio, administra la persistencia de datos y aplica políticas de seguridad. Los clientes, tanto web como mobile, se comunican exclusivamente con el backend a través de peticiones *HTTP*, con el objetivo de presentar la información al usuario, sin acceder en ningún momento de forma directa a la base de datos. Además, no existe comunicación directa entre las aplicaciones frontend: toda interacción o intercambio de información entre módulos del sistema debe realizarse necesariamente a través del backend, lo que garantiza una arquitectura segura y con una separación de responsabilidades bien definida.



**Figura 7.** Arquitectura del sistema en tres capas.

## Tecnologías

Para realizar la implementación del sistema se seleccionaron distintas tecnologías según el tipo de aplicación (backend, frontend web y aplicación mobile), los conocimientos previos del equipo, las necesidades del sistema y también la curva de aprendizaje y documentación disponible.

### Backend

Para el desarrollo del backend se optó por utilizar *Java* junto con el *framework Spring Boot*. Esta decisión se basó en que Julián ya tenía conocimientos previos básicos en esta tecnología. Esto redujo la curva de aprendizaje en las primeras etapas, profundizando aún más los conocimientos técnicos a medida que el proyecto avanzaba.

Spring Boot ofrece ciertas ventajas que lo convirtieron en una buena opción para este tipo de desarrollo:

- Facilita la creación de aplicaciones robustas con arquitectura basada en servicios REST.
- Permite una fácil integración con bases de datos y sistemas de autenticación.
- Posee una gran comunidad, mucha documentación y herramientas que ayudan a acelerar el desarrollo.
- Promueve el uso de buenas prácticas como la separación en capas, la inyección de dependencias y la escalabilidad.

### Frontend Web

Inicialmente se seleccionó *React* como librería para el desarrollo del frontend web, dado que Lautaro contaba con experiencia previa con esa tecnología, lo cual se consideró útil para acelerar el desarrollo. Sin embargo, aproximadamente dos meses después de iniciado el proyecto, a Lautaro se le asignaron en su trabajo nuevas responsabilidades relacionadas con el desarrollo en *Angular*, lo que motivó a reconsiderar la elección inicial.

Se optó entonces por reemplazar *React* por *Angular*, priorizando el uso de una sola tecnología en lugar de dividir esfuerzos entre dos entornos distintos. Si bien *Angular* presenta una curva de aprendizaje más pronunciada, el hecho de que Lautaro utilizara *Angular* diariamente en su entorno laboral representaba una oportunidad para capitalizar esa experiencia práctica y aplicarla directamente al avance del proyecto.

Esta elección también ofrecía ventajas importantes:

- Proporciona una estructura más robusta y escalable.
- Cuenta con *Angular CLI*, una herramienta de línea de comandos que facilita la generación de componentes y asegura buenas prácticas de desarrollo.
- Está respaldado por Google y posee una comunidad activa y en crecimiento.

## Aplicación Mobile

Para el desarrollo de la aplicación mobile se optó por una solución nativa para *Android* utilizando *Kotlin*. Esta elección respondió al objetivo de alcanzar a sectores más vulnerables de la población, donde el acceso a dispositivos de gama alta es limitado.

Si bien no es posible trazar una relación directa entre nivel socioeconómico y sistema operativo, los datos de StatCounter (abril de 2024) indican que Android concentra más del 85 % del mercado mobile en Argentina, mientras que iOS se mantiene cerca del 10 %, lo que evidencia su menor adopción a nivel general.

En función de este panorama, se priorizó el rendimiento en Android por encima de otros sistemas operativos como *iOS*. Además, se consideró que una implementación nativa ofrecía ventajas concretas frente a frameworks multiplataforma como *Flutter* o *React Native*:

- Mejor rendimiento general, especialmente en dispositivos de gama media o baja.
- Acceso directo a las funcionalidades del sistema operativo y a librerías específicas.
- Mayor control en la interfaz y la experiencia del usuario final.

Aunque el equipo contaba con experiencia previa en Java, se optó por utilizar Kotlin como lenguaje de desarrollo mobile. Esta decisión respondió a una combinación de factores. Por un lado, el colaborador técnico manifestó interés en aprender Kotlin, lo cual fue valorado como una oportunidad de crecimiento conjunto. Por otro, el equipo identificó una serie de ventajas concretas del lenguaje respecto a Java, especialmente en el contexto de desarrollo Android: una sintaxis más concisa, mecanismos de control de nulidad más seguros, mayor interoperabilidad con APIs modernas y una estructura más adecuada para escribir código limpio y mantenible.

Además, la decisión se vio reforzada por el respaldo oficial de Google, que desde 2019 estableció a Kotlin como el lenguaje preferido para el desarrollo de aplicaciones Android, brindando soporte de primera clase en la documentación, herramientas y nuevas APIs de la plataforma. Esta elección garantiza una mayor compatibilidad a futuro, mejores recursos de soporte y alineación con las prácticas actuales del ecosistema.

Si bien podría haberse optado por Java para reducir la curva de aprendizaje inicial, se consideró que los beneficios ofrecidos por Kotlin, especialmente a largo plazo, justificaban el esfuerzo adicional. En ese sentido, se asumió un trade-off consciente, priorizando la calidad y la adopción de tecnologías modernas por sobre la comodidad inmediata.

## Motor de base de datos

El motor seleccionado para la persistencia de datos fue *PostgreSQL*. Se optó por una base de datos relacional dado que sistema requería manejar grandes estructuras de datos y relaciones entre múltiples entidades.

Esta elección se basó, en primer lugar, en la experiencia previa de ambos integrantes del equipo con esta tecnología, lo que permitió reducir tiempos de configuración y evitar una curva de aprendizaje innecesaria.

Otro factor relevante en esta elección fue que se trata de un sistema de gestión de bases de datos gratuito y de código abierto, con amplio respaldo de la comunidad. Además, su compatibilidad con herramientas del ecosistema Java —como *JPA*— facilitó la integración con el backend, contribuyendo a un desarrollo más ordenado.

## Backend

El backend del sistema fue desarrollado mediante una arquitectura en capas, siguiendo prácticas estándar en aplicaciones *Java* con *Spring Boot*. Cada capa del sistema cumple un rol bien definido: la capa *Controller* gestiona las solicitudes entrantes, la capa *Service* implementa la lógica de negocio, y la capa *Repository* maneja las operaciones sobre la base de datos.

Esta separación clara de responsabilidades permitió desarrollar, probar y mantener cada parte del sistema de forma más eficiente, reduciendo la complejidad y favoreciendo la reutilización de componentes. Además, facilitó la detección y corrección de errores, ya que cada incidente podía ser rastreado a una capa específica.

## Uso de estructuras reutilizables

Durante el diseño del sistema, se anticipó que habría muchas operaciones similares para diferentes tipos de datos, como por ejemplo crear, consultar, modificar o eliminar información.

Debido a esto, se buscó maximizar la reutilización de código para evitar la duplicación de lógica común y garantizar una estructura consistente entre los distintos módulos. Para ello, se implementó un enfoque que permitiera definir comportamientos estándar de manera genérica, de modo que pudieran ser utilizados por todas las entidades del sistema.

Las operaciones genéricas implementadas fueron las siguientes:

- Listado de entidades de forma paginada con opciones de filtrado y ordenamiento.
- Consulta directa de entidades por medio de sus identificadores.
- Creación o modificación de entidades.
- Eliminación de entidades.

Esta estrategia trajo múltiples beneficios al proyecto: redujo significativamente la cantidad de código necesario, facilitó la incorporación de nuevas funcionalidades, y aseguró una mayor coherencia en el comportamiento de los distintos módulos. Además, redujo el tiempo de codificación de las operaciones más triviales, permitiendo enfocar el tiempo en los casos donde era necesario implementar algún comportamiento particular.

## Capa de datos

De la misma forma que se planteó el desarrollo de los demás apartados ya mencionados, se optó por construir el modelo de datos de forma progresiva, siguiendo el enfoque modular del desarrollo. Esto implicó que las tablas y campos fueran incorporándose a medida que se avanzaba con cada módulo, permitiendo ajustar el modelo de forma flexible.

El modelo relacional se fue construyendo a medida que se identificaban entidades, relaciones y atributos. Para cada módulo, se analizaban los requerimientos específicos, se ajustaba el esquema existente y se incorporaban nuevas estructuras según fuera necesario.

En particular, el diseño del módulo de alimentos fue el más complejo de implementar. Esto se debió, principalmente, a la falta de conocimiento y experiencia del equipo en el área de nutrición. Para superar este desafío, se decidió tomar como referencia bases de datos nutricionales existentes y se llevó a cabo un análisis comparativo de diversas opciones encontradas. Tras esta evaluación, se seleccionó como fuente principal la base provista por la Universidad Nacional de Luján, ya que su estructura se consideró la más adecuada para respaldar las soluciones tecnológicas planteadas en función de los requerimientos del sistema. Dicha base, además, ofrece planillas descargables (en formato CSV) organizadas por categorías de alimentos, como carnes, vegetales y productos azucarados, entre otros.

Durante el proceso de análisis, se identificó que cada una de estas planillas presentaba estructuras distintas: si bien compartían algunos nutrientes básicos, también incluían campos exclusivos o carecían de ciertos datos según la categoría de alimentos. Ante esta situación, se llevó a cabo un proceso exhaustivo de unificación de datos, reorganizando y adaptando la información para conformar una estructura única, compatible con los requerimientos del sistema. Esta estructura fue posteriormente validada por la referente funcional del proyecto, quien además solicitó una ampliación que contemplara, de forma más detallada, los diferentes tipos de grasas presentes en los alimentos. Con esta

definición final, se completó una base de datos de prueba utilizando inteligencia artificial, con el objetivo de agilizar el proceso de carga durante el desarrollo.

Cabe destacar que, una vez que el sistema entre en producción, la base de datos de pruebas será reemplazada por una versión definitiva con datos reales y certificados. Para ello, se está trabajando en conjunto con los profesionales de la salud del HIGA, quienes tendrán la responsabilidad de validar la información nutricional correspondiente. Este proceso resulta fundamental, ya que cualquier error o inconsistencia en los datos podría comprometer la salud de los usuarios celíacos.

En el [anexo II](#) se incluye un *Diagrama Entidad-Relación (DER)* simplificado que refleja la estructura general de la base de datos. Cabe aclarar que el diagrama omite muchos de los atributos utilizados —por ejemplo, la tabla de alimentos, que en su versión completa contiene numerosos campos correspondientes a nutrientes específicos—, con el fin de favorecer la legibilidad.

## Gestión de errores

Con el objetivo de ofrecer una interfaz uniforme y clara para los consumidores de la API, se implementó un mecanismo centralizado para el manejo de errores y excepciones. Este enfoque permite interceptar y gestionar adecuadamente los errores que impiden el correcto procesamiento de una petición, proporcionando respuestas estructuradas y que den a entender los detalles de lo ocurrido.

Se plantean dos tipos de situaciones:

- **Errores esperados o controlados**, que pueden producirse por solicitudes con parámetros incorrectos, rutas inexistentes o datos inválidos. Para estos casos, se implementaron excepciones específicas que permiten identificar el origen del problema y devolver mensajes claros al cliente.
- **Errores inesperados o genéricos**, que pueden surgir por fallos no previstos durante la ejecución. Al igual que los anteriores, estos también son capturados de forma centralizada, evitando la devolución de mensajes inconsistentes.

Tanto los errores como las respuestas exitosas tienen definido un formato estandarizado, esto facilita el desarrollo de clientes que consumen la API, además de mejorar la mantenibilidad del sistema. En el caso de que se presenten errores, la respuesta incluye información relevante como una descripción del problema, un timestamp y el código HTTP correspondiente. Por otro lado, las respuestas exitosas utilizan una estructura similar que además encapsula el resultado de la petición, junto con los metadatos.

Este enfoque estructurado es muy útil para proporcionar una interfaz robusta, coherente y más fácil de integrar.

## Pruebas

El proceso de pruebas fue una parte fundamental del desarrollo del sistema, porque permitió detectar errores ocultos, validar el correcto funcionamiento de los distintos módulos y garantizar una mayor estabilidad. Para lograrlo, no se adoptó una única forma de realizar las pruebas sino que se aplicaron distintas estrategias, se combinaron **pruebas automáticas** y **pruebas manuales**.

### Pruebas automáticas

#### Unit testing

Como se mencionó anteriormente, el backend está estructurado en capas, donde los controladores y repositorios normalmente se mantienen como “capas finas”. Esto quiere decir que el grueso de la lógica implementada se encuentra en los servicios. Por este motivo, el enfoque principal estuvo puesto en realizar testing a la capa de servicios del backend, donde reside la lógica de negocio más relevante. Para lograrlo, se utilizaron *JUnit 5* como framework de pruebas y *Mockito* para simular el comportamiento de los repositorios, desacoplando la ejecución de las pruebas del acceso real a la base de datos.



Mockito permite crear objetos simulados (*mocks*) de las dependencias que necesita una clase, en este caso en concreto se utilizó para los repositorios. Esto significa que se puede definir, por ejemplo, qué debe devolver un método sin necesidad de tener una base de datos real conectada. De esta forma, se puede probar la lógica del servicio de forma aislada, verificando que se comporte correctamente bajo diferentes condiciones. Además, Mockito permite verificar si ciertos métodos fueron invocados, cuántas veces y con qué argumentos, lo cual es útil para asegurarse de que el flujo del código se ejecuta como se espera.

Con el objetivo de reducir la duplicación de código, fomentar la reutilización y aprovechar la estructura reutilizable mencionada en apartados anteriores, se definió también una arquitectura de pruebas similar que aplica la misma lógica. Esto significa que se definieron pruebas genéricas para las operaciones estándar que utilizan la gran mayoría de las entidades del sistema. Así pueden realizarse muchos tests con menos esfuerzo.

Gracias a utilizar este enfoque, se logró tener una buena cobertura de tests de forma sistemática para todos los servicios que siguen esta estructura común. Además, para los servicios que implementan lógica adicional específica, se desarrollaron tests concretos adicionales, permitiendo validar también estos escenarios.

## Pruebas de integración

Si bien las pruebas unitarias fueron la prioridad, también se exploró la posibilidad de realizar algunas pruebas de integración específicas, principalmente centradas en endpoints como la creación de usuarios. El objetivo fue validar la comunicación entre las distintas capas del backend y comprobar que las respuestas HTTP fueran las esperadas.

Para no comprometer la integridad de los datos de desarrollo, se creó un entorno de pruebas separado. Al ejecutar los tests, se toma una configuración específica para pruebas que apunta a una base de datos de testing que resulta de una copia manual de la base de datos de desarrollo que ya incluye ciertos datos de prueba, como usuarios básicos, alimentos, etc. Esto permitió ejecutar las pruebas de integración en un ambiente controlado y seguro, sin afectar el entorno principal del equipo de desarrollo.

## Pruebas manuales

En cuanto al frontend web y a la aplicación mobile, las pruebas se realizaron de forma manual, centradas en validar la experiencia de usuario, la interacción con el backend y el comportamiento general de la interfaz. También se realizaron pruebas manuales sobre algunos endpoints del backend utilizando herramientas como *Postman* e *Insomnia*, que son especialmente útiles para comprobar casos límite o validar endpoints que usan estructuras de datos complejas donde es más fácil definir las con un archivo json en lugar de utilizar mocks.

Para llevar un control tanto de estas pruebas manuales como de las automáticas, se utilizaron planillas de Google sheets donde se registraron los distintos casos de prueba, los resultados obtenidos, errores detectados, y observaciones relevantes. Este seguimiento permitió ordenar las pruebas, identificar errores no conocidos y lograr una documentación más sólida.

Se detallan a continuación las pruebas realizadas.

## Testing web

- Login. Ingreso al sistema con diferentes roles.
- Pruebas de responsividad en la interfaz para diferentes tamaños de pantalla.
- Cierre de sesión y cambio de contraseña.
- Movimientos de posición en cuidados para el hogar. Búsqueda.
- Movimientos de posición en cuestionario. Búsqueda.
- Cálculo automático de valores nutricionales en recetas.
- Verificación de estadísticas.
- Creación, consulta, actualización y eliminación de las entidades del sistema: cuidados, alimentos, porciones, recetas, ingredientes, puntos de compra, usuarios.

## Testing mobile

- Login. Ingreso con usuarios nuevos y existentes.
- Verificación del proceso de registro de usuarios.
- Cambio de contraseña y cierre de sesión.
- Visualización correcta de cuidados para el hogar.
- Completado del cuestionario.
- Visualización correcta de recetas y funcionamiento del buscador.
- Visualización correcta de alimentos y funcionamiento del buscador.
- Buscador de puntos de compra, redirección al mapa.
- Verificación de la interfaz del método del plato. Verificación del análisis del plato y recomendador.

## Resultados y mejoras

Gracias al conjunto de pruebas implementado, se detectaron y corrigieron varios errores que no habían sido identificados durante el desarrollo inicial. Algunos de estos errores estaban relacionados con validaciones mal implementadas, respuestas incorrectas en algunos endpoints, o comportamientos inesperados bajo condiciones no analizadas en un principio. En ese sentido, el proceso de pruebas fue clave no solo para validar la funcionalidad, sino también para mejorar la calidad general del producto.

## Seguridad

Con respecto a la seguridad, se utilizó *Spring Security*, un framework ampliamente utilizado en el ecosistema Java para la protección de aplicaciones web.

Se implementó un mecanismo de autenticación basado en tokens *JWT (JSON Web Tokens)*. Este enfoque permite validar la identidad de los usuarios en cada solicitud al servidor, sin tener la necesidad de mantener sesiones activas, lo cual es especialmente útil para arquitecturas distribuidas y ayuda a reducir la carga del servidor.

Además, las contraseñas de los usuarios se almacenan de forma segura mediante un mecanismo que combina *hashing* y *salt*. Este enfoque consiste en aplicar un algoritmo de hash sobre la contraseña del usuario concatenando un valor aleatorio (*salt*), así se garantiza que, si dos usuarios utilizan la misma contraseña, los hashes almacenados en la base de datos sean distintos. De esta forma se dificulta el uso de ataques con diccionarios de contraseñas. En este proyecto se empleó el algoritmo *bcrypt*, que automáticamente utiliza un *salt* en cada operación, ofreciendo una solución robusta.

En cuanto a la autorización, se definieron tres roles distintos que permiten controlar qué acciones puede realizar cada tipo de usuario dentro del sistema:

- **ROL\_ADMIN:** Es el rol con mayores privilegios, permite gestionar toda la información del administrador web, especialmente tiene permisos para dar de alta y baja a otros usuarios.
- **ROL\_DEFAULT:** Este rol permite gestionar la información del administrador web pero con menos privilegios, no puede modificar los datos de otros usuarios.
- **ROL\_PACIENTE:** Este es el rol otorgado a todos los usuarios que interactúan con el sistema desde la aplicación mobile. Da permisos para consumir endpoints específicos para esta aplicación.

Cada endpoint del sistema fue configurado con reglas de acceso específicas, de manera que solo los usuarios con los roles correspondientes puedan acceder a las funcionalidades permitidas. Esta configuración garantiza un control de acceso preciso y acorde a los distintos tipos de usuarios previstos en la aplicación.

## Logs

Para tener un seguimiento de las acciones realizadas dentro del sistema, se implementó un sistema de logs a nivel de aplicación y también cierta trazabilidad en la base de datos. De esta forma se puede auditar el comportamiento del sistema, detectar errores, y poder saber que usuario hizo determinada acción y en qué momento.

### Logs en archivos del sistema

Desde el backend se optó por utilizar *Logback*, una herramienta ampliamente utilizada en entornos Java que permite generar archivos de log con distintos niveles de detalle y formatos personalizables. Se definió un formato para que cada vez que un administrador del sistema realice alguna acción (como crear, modificar o eliminar entidades), esa operación quede registrada junto con el usuario que la ejecutó y una descripción de la acción realizada.

Estos registros se almacenan automáticamente en archivos de log que pueden ser consultados posteriormente para revisar los diferentes eventos ocurridos, buscar errores o analizar el uso. Además, existen distintos niveles de gravedad para los mensajes (por ejemplo, información general o errores), esto facilita el análisis en caso de tener que revisar estos archivos.

### Trazabilidad en base de datos

Además de los archivos de log, se implementó un sistema de trazabilidad interna en la base de datos. Todas las entidades principales del sistema incorporan campos que permiten saber:

- Cuando fue creado un registro
- Que usuario lo creó
- Cuando fue modificado por última vez
- Que usuario realizó esa modificación

Esto permite, por ejemplo, determinar fácilmente quién modificó una receta o cuándo se creó determinado alimento. Esta trazabilidad se diseñó para aplicarse de forma automática, sin necesidad de ser programada de forma repetitiva para cada entidad.

En conjunto con esto, también se implementó una estrategia de eliminación lógica (*soft delete*), mediante la cual los registros no se eliminan físicamente de la base de datos, sino que se marcan como inactivos. Esta decisión permite conservar y recuperar información en caso de ser necesario, y además simplifica significativamente la gestión de restricciones por claves foráneas, evitando problemas de referencias que podrían surgir al momento de eliminar registros vinculados a otras entidades.

## Ambiente de pruebas

Con el fin de desplegar el sistema en un entorno de pruebas se optó por utilizar un servidor privado virtual (VPS). Durante la etapa de análisis de opciones, se evaluaron distintas alternativas, incluyendo *Donweb*, *Hostinger*, *Dokploy*, *Github Pages*, *Render*, *Vercel*, *Digital Ocean*, etc. La selección final se basó en criterios como el nivel de control sobre el entorno, la relación costo-beneficio y la ubicación geográfica del servidor, priorizando opciones cercanas para reducir la latencia y mejorar el tiempo de respuesta. En función de esto, se optó por contratar un VPS en *Hostinger*, eligiendo *AlmaLinux 9* como sistema operativo, debido a su estabilidad, compatibilidad con las tecnologías requeridas y por contar con experiencia en su utilización.

Luego, se prepararon las herramientas a utilizar para el despliegue. Se instaló Docker, necesario para incorporar las imágenes creadas durante el proceso de desarrollo, y *Nginx*, que funciona como servidor web y reverse proxy, gestionando la comunicación entre el exterior y los distintos servicios internos.

El backend fue empaquetado como contenedor Docker y desplegado en el VPS. Esto permite asegurar la portabilidad de la aplicación, y también simplifica futuras actualizaciones y mantenimiento.

*Nginx* se configuró como reverse proxy, lo que permite enrutar las solicitudes entrantes hacia el contenedor. Además, se utilizó para servir la aplicación mobile: el archivo APK se expuso públicamente, permitiendo su descarga.

Finalmente, se registró y configuró el dominio **sendaglutenfree.site**. Esto permitió reforzar la identidad del proyecto y mejorar la experiencia de los usuarios que acceden a la plataforma. Además, con el objetivo de mejorar la seguridad, se implementó HTTPS mediante el uso de *Certbot* y los certificados gratuitos provistos por *Let's Encrypt*.

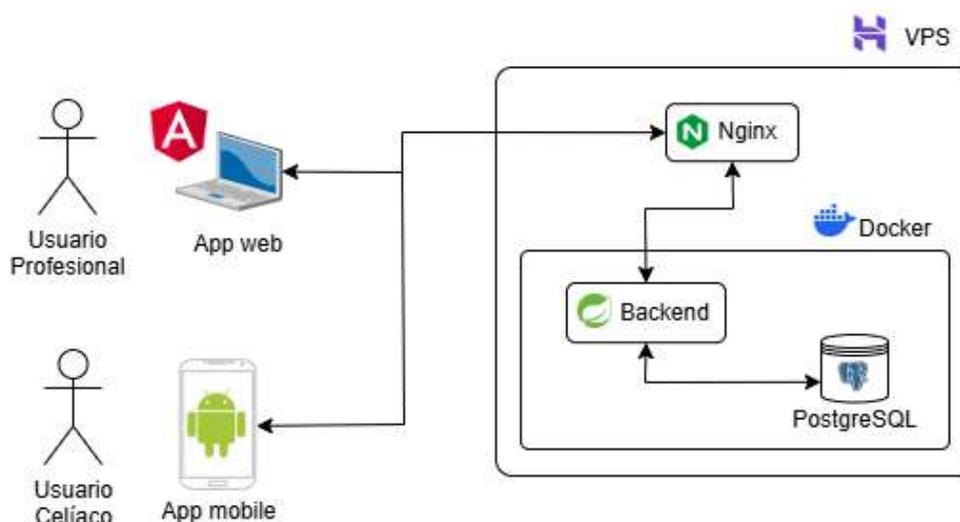


Figura 8. Diagrama de despliegue del ambiente de pruebas.

## Validación funcional

Respecto a la validación funcional del sistema, el proceso continúa en curso. Hasta el momento de la presentación de este informe, se logró completar la validación del administrador web por parte de la referente funcional. La validación de la aplicación destinada a usuarios finales aún no fue realizada, dado que los usuarios celíacos no participaron como actores activos en las etapas del desarrollo y no fueron considerados *stakeholders* directos en esta fase. En consecuencia, si bien es posible pensar en una futura prueba por parte de los usuarios finales, no puede considerarse como una validación formal dentro del alcance de este trabajo.

En cuanto al proceso de validación en curso, se entregó a la referente funcional una versión operativa del sistema para su revisión. Ella actúa como canal de comunicación entre el equipo de desarrollo y el conjunto de profesionales de la salud involucrados. También tendrá el rol de coordinar posteriormente la participación de usuarios finales. A su vez, la parametrización final de datos también se encuentra en desarrollo, en paralelo con la validación del sistema por parte del equipo profesional.

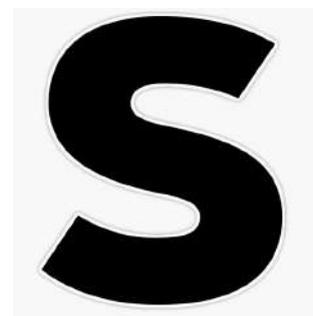
## Producto: Senda

*Senda* es el nombre elegido para el producto final desarrollado en este proyecto, el cual, como se mencionó anteriormente, está compuesto por dos aplicaciones que funcionan de forma complementaria: un administrador web para profesionales de la salud, y una aplicación mobile para la población celíaca, que buscan acompañar y asistir a las personas con celiaquía en el seguimiento de una dieta libre de gluten.

El nombre *Senda* fue elegido con un sentido simbólico: una “senda” representa un camino, una guía. En este caso, la aplicación busca acompañar a la población celíaca en un recorrido hacia una forma de vida saludable, basada en una alimentación balanceada, segura y sin gluten. No se trata únicamente de una herramienta tecnológica, sino de una propuesta de estilo de vida que promueve el bienestar desde un enfoque práctico y accesible.

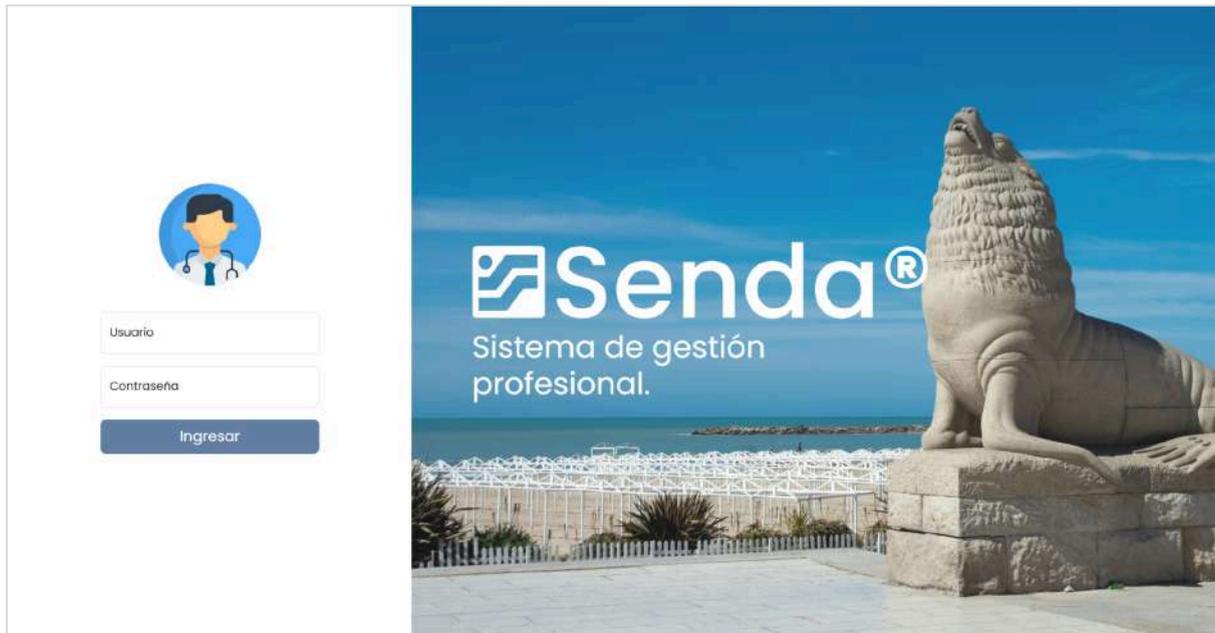


Este concepto también se refleja en el diseño de su logo: una letra “**S**” que surge de la conjunción de un **camino sinuoso** y de una **senda peatonal**. Así, tanto el nombre como la identidad visual del producto refuerzan la idea central de guiar al usuario celíaco por un trayecto claro y seguro hacia una mejor calidad de vida.

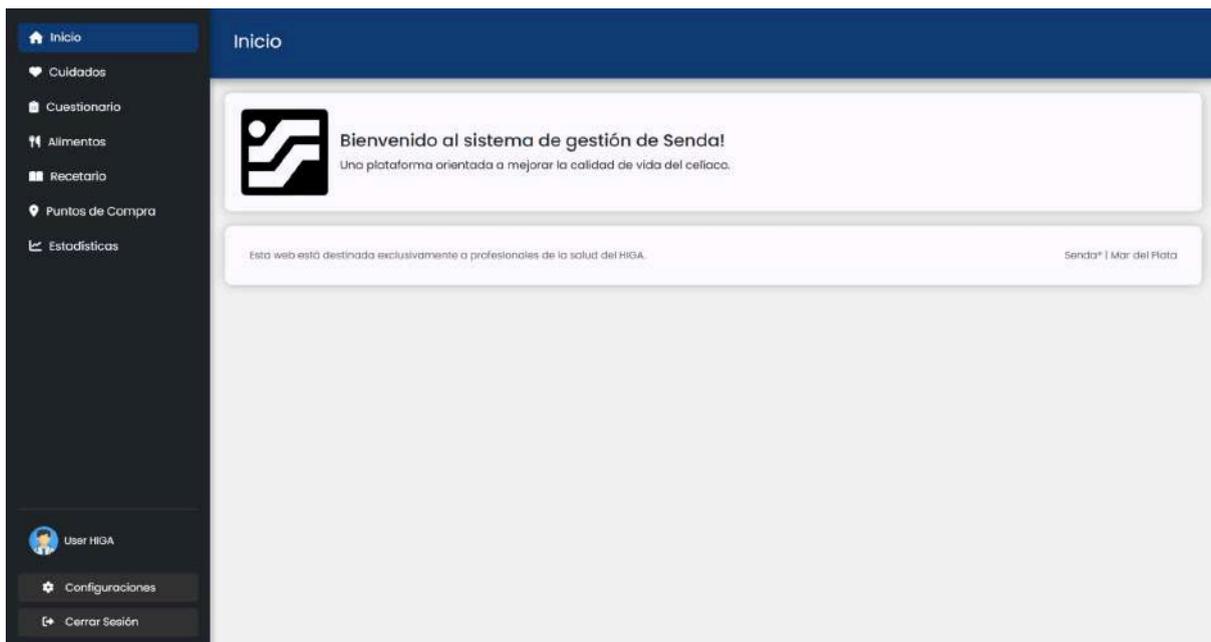


## Introducción a Senda

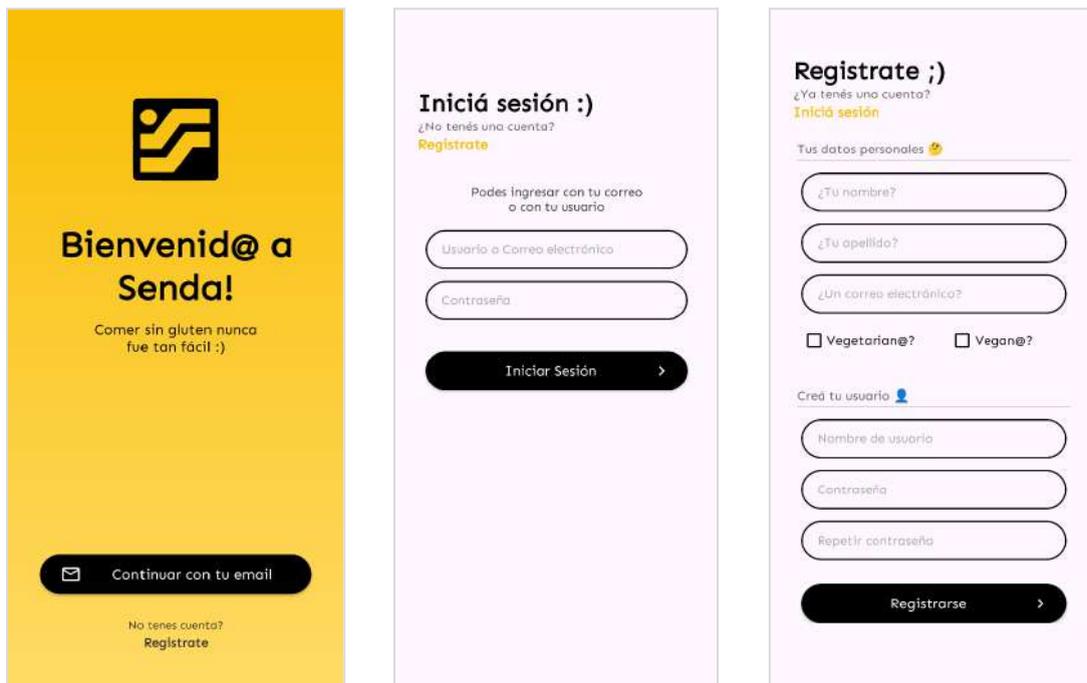
La interacción con el sistema comienza a través de una *landing page* diseñada específicamente para cada tipo de usuario. En la aplicación web, el profesional de la salud accede mediante una página de inicio de sesión (*login*), donde debe ingresar su usuario y contraseña para acceder al sistema.



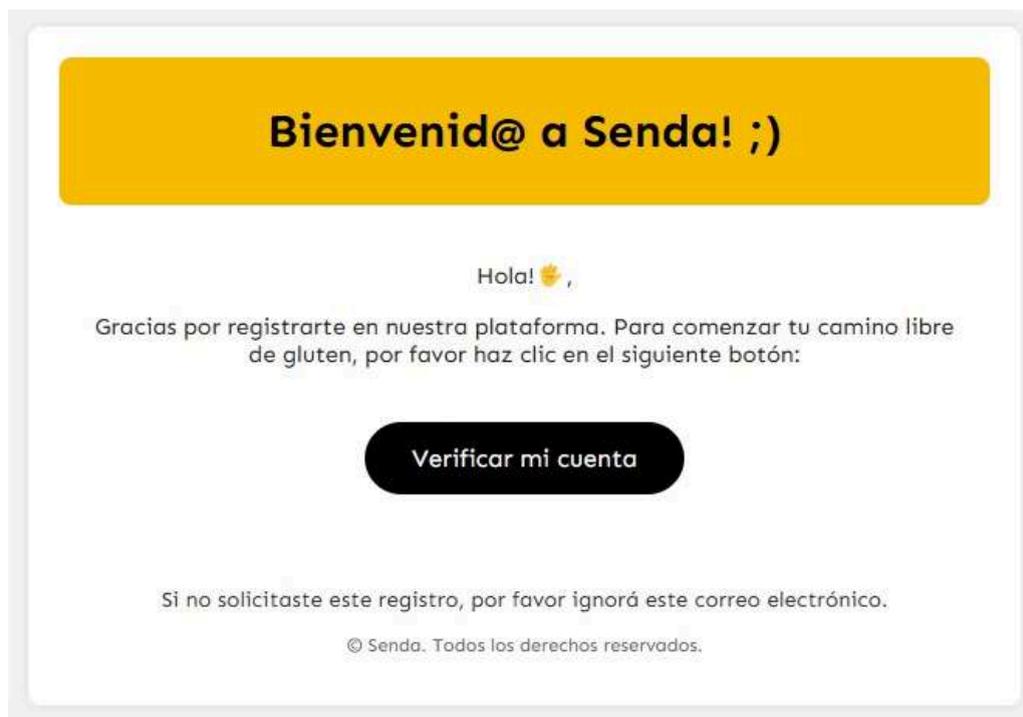
Una vez autenticado, la navegación principal se realiza a través de un panel lateral (*sidebar*), desde donde se puede acceder fácilmente a los diferentes módulos del sistema. Estos módulos serán explicados uno a uno en los siguientes apartados.



En el caso de la aplicación mobile, la experiencia inicia en una landing page amigable que da la bienvenida. Desde esta pantalla, el usuario puede optar por iniciar sesión o registrarse si aún no posee una cuenta.



En el caso de optar por registrarse, deberá completar el formulario correspondiente. Una vez finalizado el registro, recibirá un correo de verificación. Al hacer clic en el botón "Verificar mi cuenta", su usuario quedará listo y habilitado para su uso.



Tras acceder al sistema, la navegación en la aplicación mobile se organiza mediante una barra inferior con cuatro apartados principales claramente diferenciados:

1. **Hogar**, que muestra los cuidados para el hogar.
2. **Carta**, donde se encuentran los alimentos y recetas.
3. **Locales**, que muestra los puntos de compra.
4. **Mi plato**, centrado en el método del plato.



## Cuestionario de ingreso

El cuestionario de ingreso es el primer módulo funcional que interactúa con el usuario celíaco. Una vez registrado en la aplicación, se le solicita completar este cuestionario, que tiene como fin recolectar información clave para conocer mejor a la población celíaca usuaria de la app. Las preguntas apuntan a temas como hábitos alimentarios, nivel de conocimiento sobre la celiaquía y otros aspectos relevantes para el seguimiento nutricional.

Las tres pantallas muestran preguntas con opciones de respuesta:

- Pregunta 1:** Hace cuanto tiempo te diagnosticaron celiaquía?
  - Menos de 1 año
  - 1 a 3 años
  - 4 a 6 años
  - Más de 6 años
- Pregunta 2:** Con qué frecuencia seguís una dieta libre de gluten?
  - Siempre
  - La mayoría de las veces
  - A veces
  - Rara vez o nunca
- Pregunta 3:** Te resulta difícil encontrar alimentos sin tacc?
  - Nunca
  - Rara vez
  - A menudo
  - Siempre

Cada pantalla tiene un botón 'Siguiente' con una flecha a la derecha.

Desde la aplicación web, los profesionales pueden configurar este cuestionario accediendo a su apartado correspondiente en el menú lateral. En esta pantalla, se observa una grilla con las preguntas del cuestionario ya existentes, además de una barra de búsqueda y un botón para poder agregar una pregunta nueva.



Además, utilizando el botón de opciones en cada pregunta, los usuarios pueden realizar las acciones de inspeccionar, eliminar y reordenar.



Al inspeccionar una pregunta, son dirigidos a un nuevo apartado donde se les da la opción de editar el contenido. Cada pregunta incluye su enunciado y un conjunto de respuestas en formato *multiple choice*.

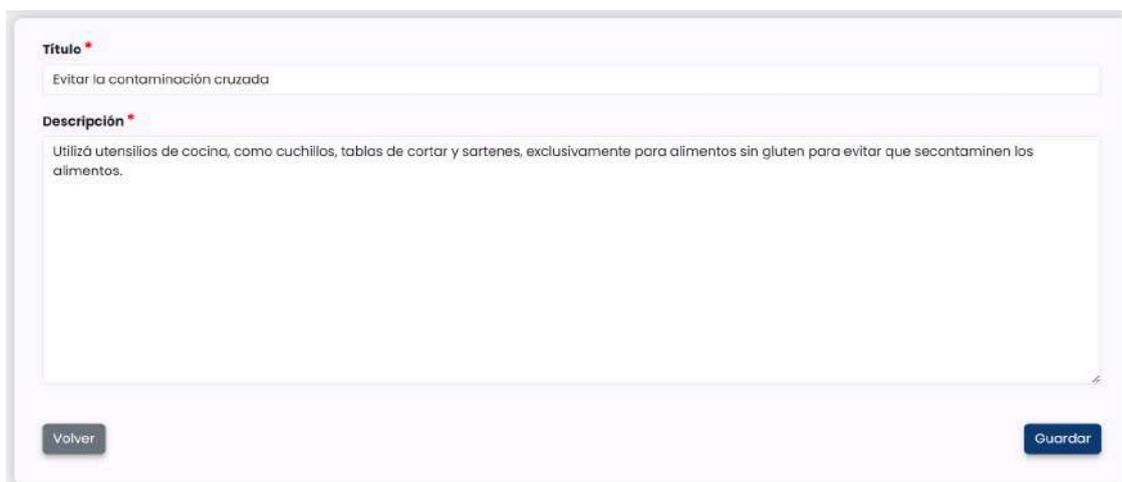


Cabe destacar que el funcionamiento de cada apartado en la aplicación web tiene un comportamiento muy similar al ya mencionado, por lo que en los demás módulos solo se prestará especial atención al contenido que se muestra al inspeccionar un elemento dentro de la grilla.

## Cuidados para el hogar

Este módulo tiene como objetivo informar y concientizar a los usuarios celíacos sobre los cuidados que deben tener en su día a día para evitar complicaciones en su alimentación, especialmente en relación con la contaminación cruzada y la manipulación de alimentos.

En la aplicación web, los profesionales de la salud definen los cuidados para el hogar que luego serán mostrados en la aplicación mobile. Para ello, los profesionales pueden crear, editar, eliminar y reordenar los distintos cuidados.



Formulario de edición de un cuidado para el hogar. El formulario tiene dos campos de texto: "Título" y "Descripción". El título es "Evitar la contaminación cruzada" y la descripción es "Utilizá utensilios de cocina, como cuchillos, tablas de cortar y sartenes, exclusivamente para alimentos sin gluten para evitar que secontaminen los alimentos." En la parte inferior del formulario hay dos botones: "Volver" y "Guardar".

Cada uno se compone de un título y una descripción. Por ejemplo, un cuidado podría ser “Evitar la contaminación cruzada”, con una explicación detallada sobre cómo prevenirla en la cocina.

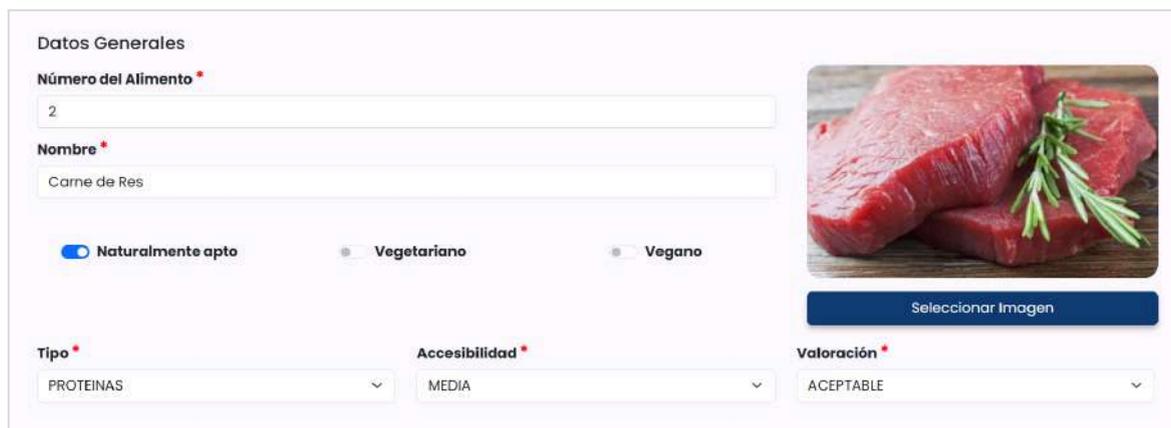
En su contraparte, en la aplicación mobile, una vez que el usuario termina el cuestionario de ingreso, accede al apartado “Hogar” que actúa como pantalla de bienvenida. En la parte superior se muestra el perfil del usuario junto a un botón de configuración.

Debajo, se listan todos los cuidados para el hogar en formato de dropdown, respetando el orden definido por los profesionales.



## Alimentos

El módulo de alimentos cumple un rol fundamental dentro de la aplicación, ya que actúa como la base de información nutricional para múltiples funcionalidades del sistema. Su objetivo es brindar a los usuarios celíacos una fuente confiable de alimentos aptos, clasificados y validados. A su vez, permite a los profesionales cargar la información con un alto nivel de detalle técnico, lo que sostiene la lógica nutricional del resto de los módulos, como el de recetas o el método del plato.



Datos Generales

Número del Alimento \*  
2

Nombre \*  
Carne de Res

Naturalmente apto     Vegetariano     Vegano

Tipo \*  
PROTEINAS

Accesibilidad \*  
MEDIA

Valoración \*  
ACEPTABLE



Seleccionar Imagen

En la aplicación web, la información de cada alimento es cargada por el profesional de la salud. Los datos ingresados incluyen el nombre del alimento, su clasificación (hidrato, proteína o verde), y detalles nutricionales específicos calculados en base a 100 gramos. Esta información contempla tanto macronutrientes (proteínas, carbohidratos y grasas) como micronutrientes (calcio, sodio, hierro, entre otros). La precisión en estos datos permite que posteriormente los alimentos sean utilizados como ingredientes en recetas o en el armado de platos dentro de otros módulos del sistema.



Valores Energéticos

Energía (kcal) \*  
124

Energía (kJ) \*  
520

Macronutrientes

Proteínas (g) \*  
26,2

Carbohidratos (g) \*  
0

Grasas (g) \*  
2

Saturadas (g) \*  
0,4

Monoinsaturadas (g) \*  
0,8

Poliinsaturadas (g) \*  
0,6

Trans (g) \*

Además, se registra si el alimento es naturalmente libre de gluten o si requiere el sello oficial “Sin T.A.C.C.”, así como también si es apto para dietas vegetarianas o veganas. También se le asigna una valoración simbólica que destaca aquellos alimentos que tienen un alto valor nutricional y que deben ser recomendados en el método del plato.

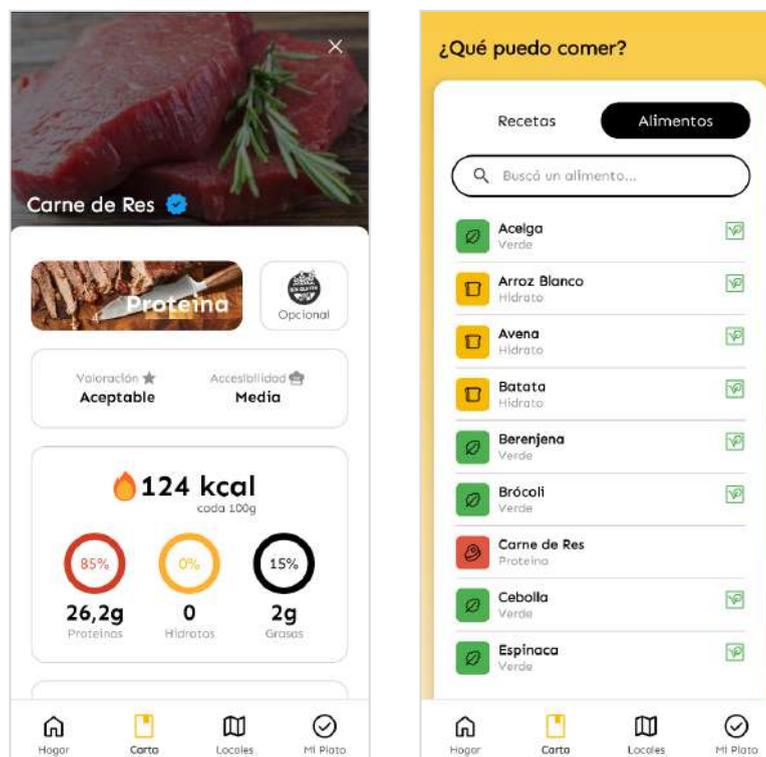
Un aspecto clave en el proceso de carga es la definición de porciones. El profesional especifica una o más porciones para cada alimento, indicando tanto su denominación (por ejemplo: “taza”, “filete”, “puñado”) como su equivalencia en gramos.

Nombre de porción Nro. 1 *	Peso equivalente (g) *
Filete chico	90
Filete mediano	125
Filete grande	160

+ Agregar porción

Este enfoque es fundamental para mejorar la experiencia del usuario celíaco, dado que manejar la información por porciones es un enfoque más intuitivo y amigable, simplificando así el uso del sistema y evitando que el usuario tenga que manejar directamente cantidades expresadas en gramos.

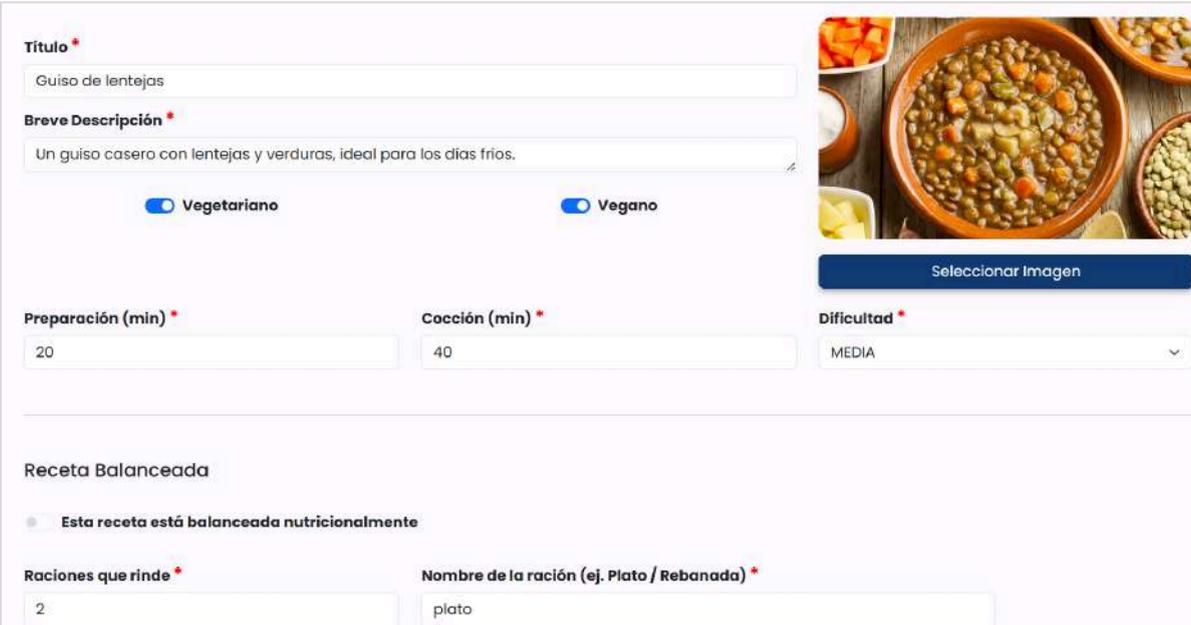
Desde la aplicación mobile, los usuarios acceden a este módulo dentro del apartado “Carta”, donde pueden explorar un listado alfabético de alimentos. Cada alimento se muestra con su nombre y un ícono de color que indica su clasificación: verde para vegetales, amarillo para hidratos y rojo para proteínas.



## Recetas

El módulo de recetas está pensado como una forma de ofrecer ideas prácticas, seguras y variadas para el día a día de las personas con celiaquía. Su propósito no es solo inspirar comidas ricas, sino también ayudar a planificar una alimentación equilibrada. Detrás de cada receta hay un trabajo del profesional que selecciona cuidadosamente los ingredientes y sus cantidades, para que el usuario tenga una guía confiable.

En la aplicación web, el profesional de la salud se encarga de crear y mantener las recetas. Al agregar una receta, el profesional debe indicar su nombre, una descripción, la cantidad de raciones que rinde, sus ingredientes y método de preparación, entre otros datos.



The screenshot shows a form for creating a recipe. It includes fields for the title, a brief description, and a 'Seleccionar Imagen' button. There are toggle switches for 'Vegetariano' and 'Vegano'. Below these are input fields for 'Preparación (min)', 'Cocción (min)', and a dropdown for 'Dificultad'. A section titled 'Receta Balanceada' has a radio button for 'Esta receta está balanceada nutricionalmente'. At the bottom, there are fields for 'Raciones que rinde' and 'Nombre de la ración (ej. Plato / Rebanada)'. An image of lentil soup is shown on the right side of the form.

**Título \***  
Guiso de lentejas

**Breve Descripción \***  
Un guiso casero con lentejas y verduras, ideal para los días fríos.

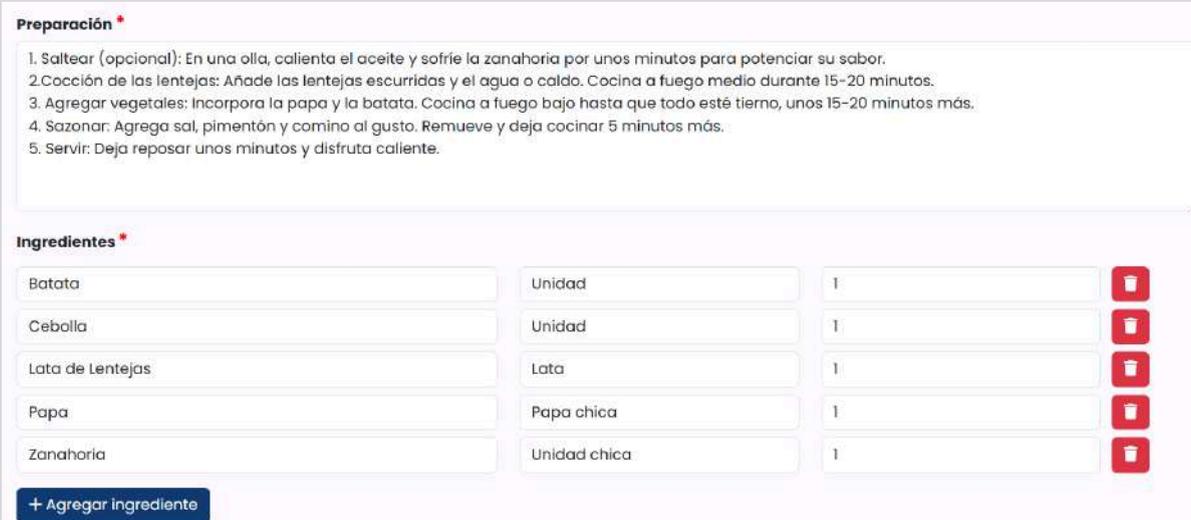
Vegetariano  Vegano

**Preparación (min) \*** 20 **Cocción (min) \*** 40 **Dificultad \*** MEDIA

**Receta Balanceada**  
 Esta receta está balanceada nutricionalmente

**Raciones que rinde \*** 2 **Nombre de la ración (ej. Plato / Rebanada) \*** plato

La carga de ingredientes se hace por medio de un modal, donde se busca el alimento a agregar, y adicionalmente se indica la porción y una cantidad.



The screenshot shows a modal for adding ingredients. It features a 'Preparación' section with a list of steps. Below is an 'Ingredientes' section with a table of ingredients, their units, and quantities. Each row has a red trash icon. A '+ Agregar ingrediente' button is at the bottom.

**Preparación \***

1. Saltear (opcional): En una olla, calienta el aceite y sofríe la zanahoria por unos minutos para potenciar su sabor.
2. Cocción de las lentejas: Añade las lentejas escurridas y el agua o caldo. Cocina a fuego medio durante 15-20 minutos.
3. Agregar vegetales: Incorpora la papa y la batata. Cocina a fuego bajo hasta que todo esté tierno, unos 15-20 minutos más.
4. Sazonar: Agrega sal, pimentón y comino al gusto. Remueve y deja cocinar 5 minutos más.
5. Servir: Deja reposar unos minutos y disfruta caliente.

**Ingredientes \***

Batata	Unidad	1	
Cebolla	Unidad	1	
Lata de Lentejas	Lata	1	
Papa	Papa chica	1	
Zanahoria	Unidad chica	1	

**+ Agregar ingrediente**

### Agregar Ingrediente ✕

**Nombre**

Nombre del Alimento

- Acelga
- Arroz Blanco
- Avena
- Batata
- Berenjena

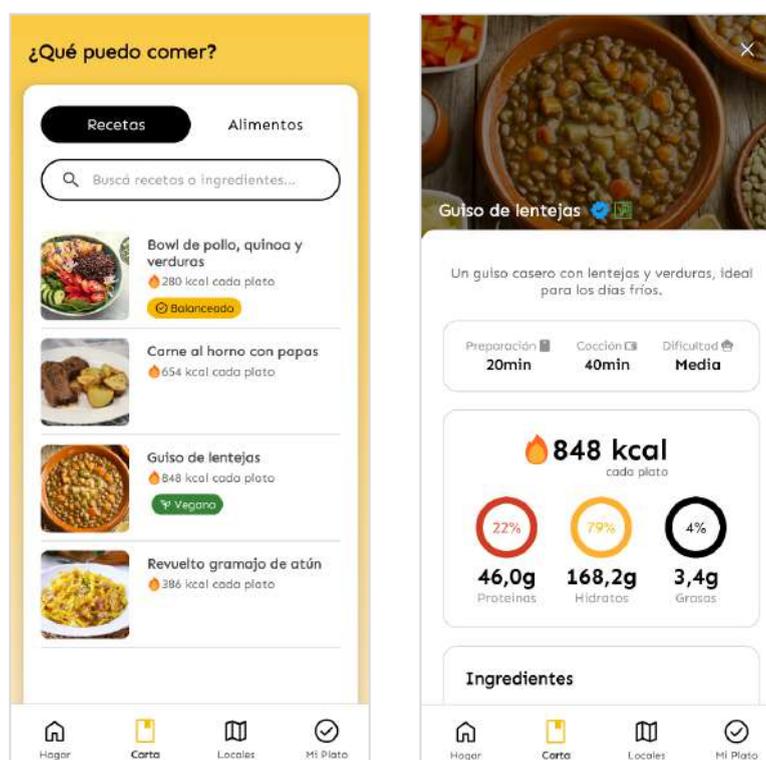
5 1 - 5 de 30 < > >>

**Porción** **Cantidad**

Taza (150g) 1 Guardar

A partir de esta información, el sistema calcula automáticamente el valor nutricional por ración de la receta, considerando tanto macronutrientes como micronutrientes. En caso de que el profesional lo prefiera, puede desactivar el cálculo automático e ingresar estos valores manualmente.

En la aplicación mobile, las recetas se encuentran también en el apartado “Carta”, junto a los alimentos. Los usuarios pueden buscarlas por nombre o por ingrediente. Al ingresar en una receta, visualizan los ingredientes necesarios, sus porciones, y el método de preparación paso a paso. Esto les permite llevar adelante preparaciones sin necesidad de buscar información en otras fuentes, y con la tranquilidad de que están basadas en alimentos aptos validados.



## Puntos de compra

El módulo de puntos de compra fue diseñado con el objetivo de ayudar a los usuarios celíacos a saber dónde pueden encontrar alimentos seguros dentro de la ciudad de Mar del Plata. Ya sea un almacén, una pastelería o una cafetería, la idea es que tengan acceso a un listado confiable de lugares que ofrecen productos aptos, y puedan llegar fácilmente hasta ellos.

Desde la aplicación web, el profesional de la salud carga cada punto de compra de forma manual. Los datos que se ingresan son: nombre del local, imagen, dirección exacta y un enlace a su ubicación en Google Maps.



Formulario de carga de un punto de compra. El formulario contiene los siguientes campos:

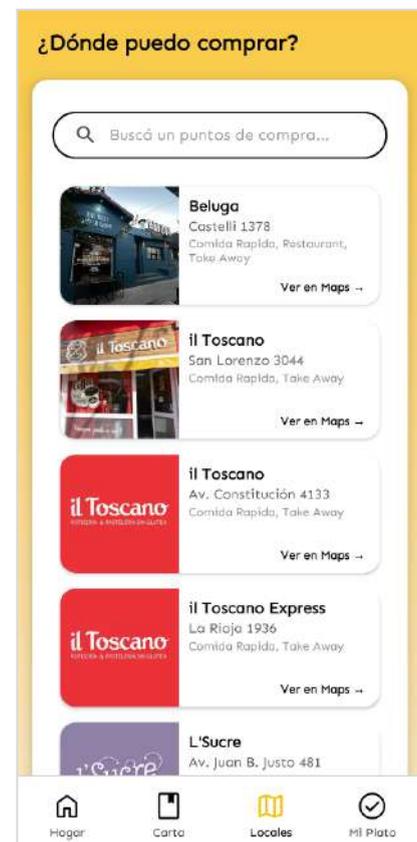
- Nombre \***: Beluga
- Dirección \***: Castelli 1378
- Link de Google Maps**: <https://maps.app.goo.gl/ewHVKETsEMgXLjwKA>
- Tags**: Selecciona Tags (Comida Rápida, Restaurant, Take Away)

El formulario incluye un botón "Volver" a la izquierda y un botón "Guardar" a la derecha. A la derecha del formulario se muestra una imagen de la fachada de un local llamado "BELUGA" con el letrero "THE BEST FISH & CHIPS". Debajo de la imagen hay un botón "Seleccionar Imagen".

Además, se pueden asignar etiquetas (tags) para clasificar el tipo de comercio: por ejemplo, “mercado”, “almacén”, “comida rápida”, “cafetería”, entre otros. Esto permite que el listado en mobile sea fácilmente filtrable y más útil para el usuario.

En la aplicación mobile, esta funcionalidad aparece bajo el nombre de “Locales”. El usuario accede a un listado completo de puntos de compra, que puede filtrar por nombre o por tag.

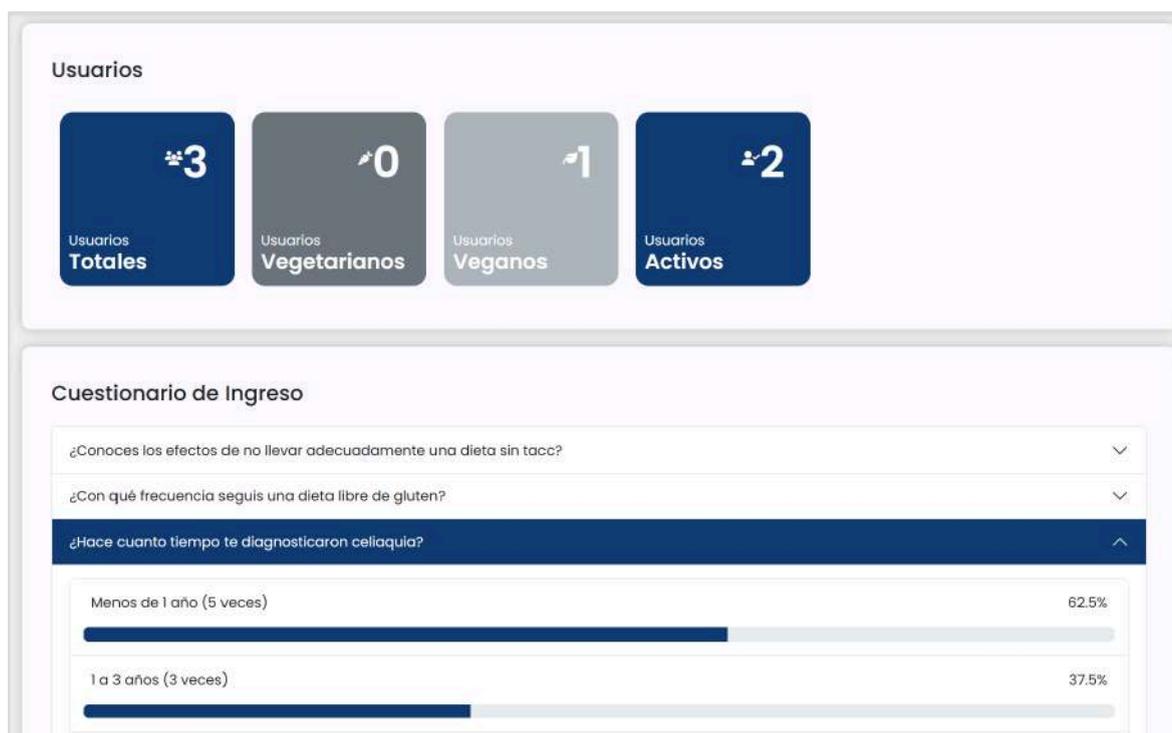
Cada entrada muestra el nombre del local, su correspondiente imagen, su dirección, y además permite abrir directamente la ubicación en *Google Maps* para facilitar el acceso.



## Estadísticas

El módulo de estadísticas tiene un perfil más analítico dentro del sistema. Está diseñado exclusivamente para los profesionales de la salud, y su objetivo es ofrecerles una herramienta que les permita observar cómo la población celíaca está utilizando la aplicación mobile. Este análisis puede ayudar a tomar mejores decisiones clínicas, ajustar contenidos, o incluso proponer nuevos enfoques.

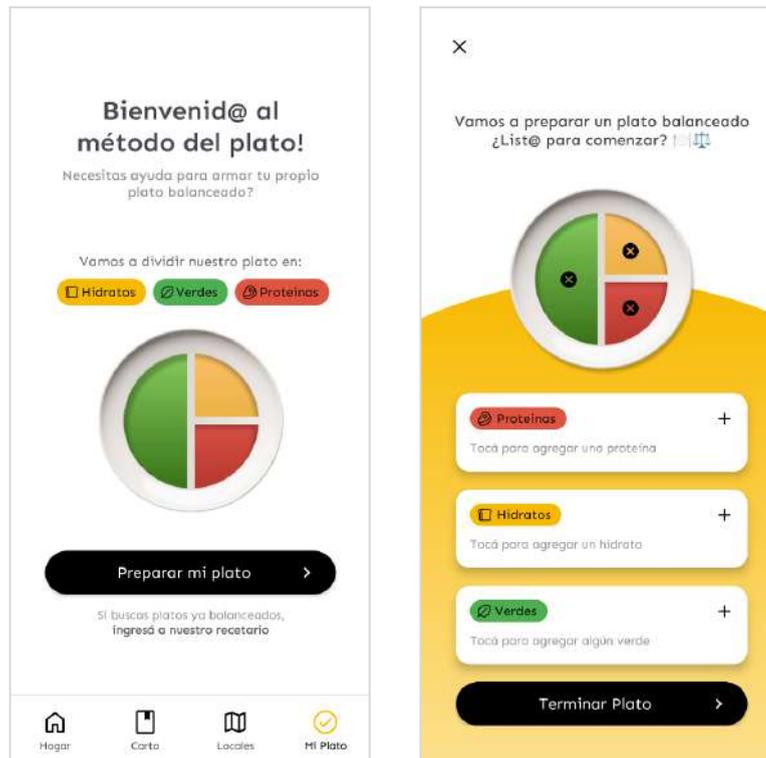
Este módulo muestra gráficos, métricas y resúmenes que reflejan la actividad de los usuarios celíacos. Estos datos, siempre anonimizados, permiten detectar patrones de uso y también áreas de mejora o de mayor interés.



Top 10 alimentos más buscados		Top 10 platos más buscados	
1. Pechuga de Pollo	77 veces	1. Bowl de pollo, quinoa y verduras	26 veces
2. Carne de Res	70 veces	2. Carne al horno con papas	18 veces
3. Huevo	52 veces	3. Guiso de lentejas	8 veces
4. Pescado Blanco	48 veces	4. Revuelto gramajo	8 veces
5. Tofu	32 veces		
6. Lata de Lentejas	25 veces		
7. Lata de garbanzos	22 veces		
8. Quinoa	20 veces		
9. Lata de atún	19 veces		
10. Pavo	15 veces		

## Método del plato

Este módulo representa una de las propuestas más innovadoras y personalizadas de toda la aplicación. Su objetivo es brindar una herramienta visual y simple para que los usuarios celíacos puedan confeccionar platos balanceados, sin necesidad de tener conocimientos avanzados en nutrición. Está basado en un concepto clásico: dividir el plato en tres partes —mitad para verdes, un cuarto para hidratos, y un cuarto para proteínas— y combinarlas de forma armónica.

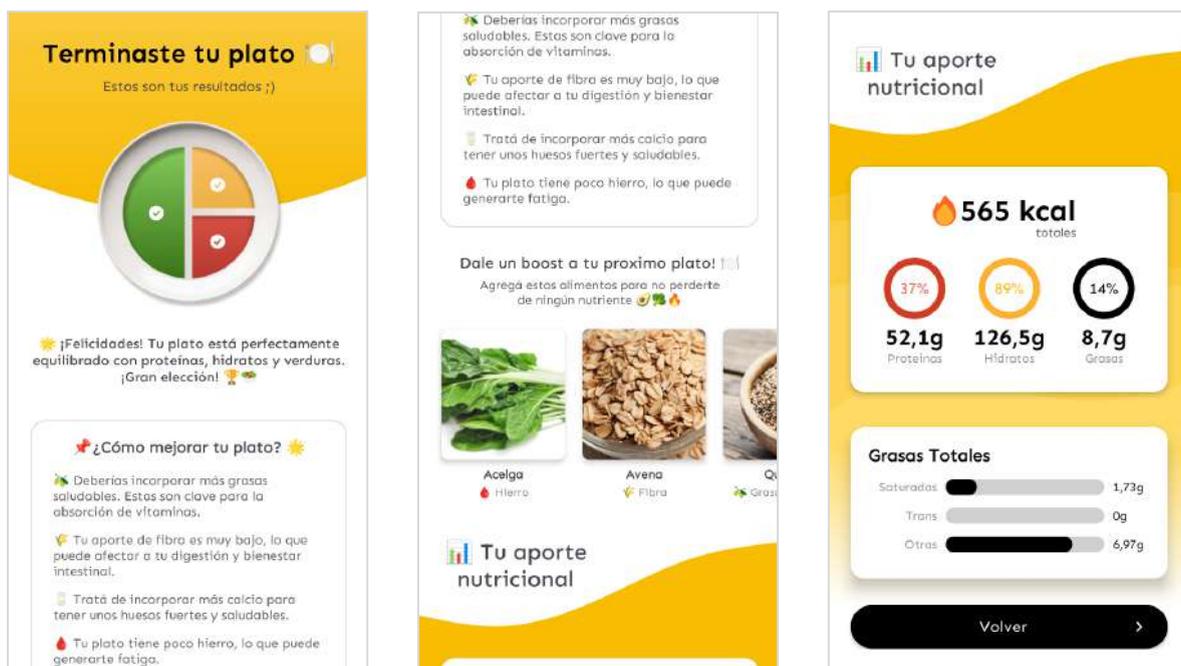


Al acceder al apartado de “Mi Plato”, el usuario puede seleccionar entre los alimentos previamente cargados por los profesionales, eligiendo porciones y cantidades para cada uno. A medida que arma su plato, un sistema de recomendación en tiempo real le indica si está alcanzando un balance adecuado o si, por ejemplo, necesita agregar más proteína o está en exceso de hidratos. El usuario puede terminar el plato en cualquier momento, aunque si logra un equilibrio, el sistema lo notificará.



Al finalizar, el sistema analiza el plato y genera un reporte con los valores nutricionales completos: macronutrientes (proteínas, hidratos, grasas), con un enfoque específico en la clasificación de grasas.

Además, el sistema identifica las carencias específicas presentes en el plato y sugiere formas de mejorarlo. En base a estas carencias, el sistema recomienda futuras comidas, considerando siempre si el usuario lleva una dieta vegetariana o vegana, y recomendando alimentos saludables con una buena valoración nutricional. Por ejemplo, si el plato es bajo en hierro, se podría sugerir agregar pescado o acelga, dependiendo si el usuario es vegetariano o no.



## Participación en concurso

Como parte del desarrollo de este trabajo final, y por iniciativa de la referente funcional, se decidió postular el sistema al XXII Premio de Investigación sobre Patologías por Sensibilidad al Gluten/Trigo, organizado por la Asociación de Celíacos y Sensibles al Gluten, con sede en Madrid, España. Este concurso tiene como objetivo fomentar investigaciones clínicas, básicas, aplicadas o de innovación tecnológica vinculadas a enfermedades como la enfermedad celíaca, la sensibilidad al gluten/trigo no celíaca, la ataxia por gluten, entre otras patologías asociadas al consumo de alimentos con gluten.

La postulación se realizó junto con la referente funcional, quien participa como Investigadora Principal en la propuesta presentada, en conjunto con los integrantes del trabajo final, Lautaro y Julián, cumpliendo el rol de desarrolladores.

La participación en este concurso constituye un hito significativo dentro del marco del proyecto, al permitir situarlo no solo en un contexto académico, sino también en un entorno de investigación internacional con proyección clínica y tecnológica. En caso de ser seleccionado, el reconocimiento se considerará un antecedente profesional relevante, tanto para el currículum como para futuras actividades académicas o laborales vinculadas al área.

Además, el hecho de que este concurso premie enfoques de innovación tecnológica refuerza el valor del sistema propuesto como una solución concreta para mejorar la calidad de vida de personas con enfermedad celíaca y otras patologías relacionadas. Más allá del premio en sí, la participación establece las bases para el desarrollo de un posible emprendimiento, con capacidad de escalar el producto a nivel regional o incluso global, dada la amplitud del problema que aborda.

Esta instancia permite validar el enfoque elegido desde una perspectiva científica y profesional, y abre puertas para futuras colaboraciones interdisciplinarias que continúen desarrollando y mejorando la herramienta.

## Memoria del proyecto

El presente apartado tiene como finalidad realizar una mirada retrospectiva sobre el desarrollo del proyecto *Senda*. A lo largo de este proceso se enfrentaron numerosos desafíos técnicos y organizativos, se tomaron decisiones estratégicas con diferentes niveles de acierto, y se atravesaron situaciones externas que afectaron el ritmo de avance. El objetivo de esta sección es identificar los principales aciertos, detallar los errores cometidos, analizar sus causas y exponer cómo fueron gestionados. Asimismo, se busca recuperar los aprendizajes obtenidos durante el desarrollo, tanto a nivel técnico como personal, que aportaron significativamente al crecimiento profesional del equipo.

## Tecnologías

Las decisiones tecnológicas tomadas durante el proyecto resultaron, en su mayoría, acertadas y coherentes con los objetivos planteados. La elección de Spring Boot para el desarrollo del backend permitió estructurar el sistema de manera sólida, facilitando la implementación de una arquitectura en capas, pruebas automatizadas, y políticas de seguridad robustas. Del mismo modo, el uso de PostgreSQL como motor de base de datos resultó ser una buena decisión debido que al contar con experiencia previa por ambos integrantes no supuso un obstáculo tecnológico.

En cuanto al frontend web, la decisión inicial de utilizar React fue reconsiderada debido a un cambio en el contexto laboral de Lautaro, optando finalmente por migrar a Angular. Esta transición resultó un notable acierto, ya que permitió alinear el stack tecnológico del proyecto con las herramientas empleadas cotidianamente en el ámbito profesional del equipo. Esto no solo redujo significativamente los tiempos en las etapas finales del desarrollo, sino que también impactó positivamente en la carrera profesional de Lautaro, formándolo con sólidas competencias y experiencia en el desarrollo web.

Para la aplicación mobile se optó por Kotlin tras un análisis exhaustivo de las tecnologías disponibles. La elección de un lenguaje nativo de Android fue estratégica y clave para cumplir con uno de los objetivos más relevantes del proyecto: garantizar una experiencia de usuario óptima en dispositivos con recursos limitados, predominantes en los sectores vulnerables a los que apunta la aplicación. Ante la baja del colaborador técnico, el equipo evaluó seriamente la posibilidad de reiniciar el desarrollo desde cero utilizando una tecnología más sencilla. Sin embargo, se priorizó mantener el objetivo principal, aún sabiendo que esto demandaría un esfuerzo adicional y el desafío de atravesar una curva de aprendizaje con una tecnología completamente nueva para todos los integrantes. Esta decisión implicó una alta carga horaria, pero terminó consolidándose como uno de los pilares fundamentales de la aplicación: no solo fue crucial para alcanzar las metas propuestas, sino que también representó una oportunidad de crecimiento profesional. Afrontar este proceso enriqueció el perfil del equipo, ampliando sus conocimientos en un

área hasta entonces desconocida como el desarrollo mobile, y formó a los integrantes como profesionales más completos, capaces de desenvolverse en distintos ámbitos tecnológicos y no restringidos únicamente al desarrollo frontend o backend.

La incorporación de Docker fue otro de los aciertos técnicos. Permitió encapsular el backend y facilitar la sincronización entre los distintos entornos de desarrollo, reduciendo tiempos de configuración y errores derivados de diferencias en los entornos locales.

En definitiva, a pesar de que los tiempos de desarrollo se extendieron más allá de lo inicialmente previsto, las decisiones tecnológicas tomadas estuvieron respaldadas por fundamentos sólidos y objetivos claros, permitiendo alcanzar resultados altamente satisfactorios. El equipo se encuentra sumamente conforme con el rendimiento obtenido y con la calidad final de la aplicación, reflejando el compromiso y la capacidad de adaptación ante los desafíos surgidos durante el desarrollo del proyecto.

## Trabajo en equipo

El desarrollo del proyecto se llevó a cabo bajo una dinámica de equipo conformada por los dos desarrolladores principales, el director y la referente funcional. La confianza mutua y el conocimiento previo entre los desarrolladores constituyeron una fortaleza inicial. La experiencia compartida en proyectos anteriores y la comunicación fluida en los aspectos técnicos facilitaron la toma de decisiones y la resolución de conflictos cotidianos.

Sin embargo, la estructura del equipo presentó debilidades significativas en otros aspectos clave. Uno de los principales errores fue la escasa comunicación directa con la referente funcional, Estela. Durante gran parte del proyecto, el contacto con Estela se realizó principalmente a través del director, Leonel, quien cumplió el rol de nexo entre los desarrolladores y la referente. Esto se debió a que, al inicio, ya existía una comunicación previa y exhaustiva entre Leonel y Estela, lo que le permitió al director conocer en profundidad todos los requerimientos funcionales y asumir la responsabilidad de transmitirlos al equipo de desarrollo. La decisión de centralizar la comunicación en Leonel buscaba evitar sobrecargar a la referente con consultas innecesarias y optimizar su tiempo, especialmente considerando que el documento de requerimientos ya se encontraba validado.

Sin embargo, esta intermediación, aunque efectiva en la primera etapa, generó una “lejanía” involuntaria con la referente funcional, lo que derivó en oportunidades perdidas para validar funcionalidades de manera más ágil y comprender en mayor profundidad ciertas necesidades del usuario final. En retrospectiva, se reconoce que priorizar un canal directo y frecuente con Estela hubiese enriquecido el proyecto, aportando mayor claridad, validación y valor a cada etapa del desarrollo.

La situación se agravó debido a la licencia médica de Leonel, que impactó directamente en la dinámica del equipo. La falta de su participación activa durante períodos críticos obligó a los desarrolladores a asumir mayor autonomía en la toma de decisiones, pero también limitó la posibilidad de contrastar enfoques y recibir devoluciones en tiempo real. Esto, por ejemplo, afectó la validación del presente informe final, que, si bien fue elaborado en su mayor parte por los integrantes principales, no contó con una revisión exhaustiva por parte del director ni con la retroalimentación continua de la referente funcional.

Por otra parte, durante la etapa de desarrollo surgieron ciertas diferencias de criterio a la hora de tomar decisiones. Mientras Julián priorizaba la entrega en tiempo y forma, Lautaro ponía mayor énfasis en el acabado final del producto y la experiencia del usuario. Estas diferencias, lejos de entorpecer el proyecto, ayudaron a equilibrar la funcionalidad y la calidad, permitiendo encontrar un punto medio que favoreciera ambas visiones. Sin embargo, en retrospectiva, se reconoce que el verdadero error estuvo en no consultar estas diferencias con el director del proyecto. Por temor a resultar inoportunos debido a su licencia médica, se optó por resolver todo internamente, perdiendo así la oportunidad de contar con una mirada externa y definida que podría haber marcado un rumbo más claro para el equipo.

En síntesis, la experiencia de este proyecto evidenció que la confianza y el trabajo fluido entre los integrantes técnicos no son suficientes para garantizar el éxito si no se acompaña de una comunicación directa y sostenida con todos los actores involucrados. Dependiendo casi exclusivamente del director como único nexo fue un error que impactó de lleno en la calidad de las decisiones y la capacidad de adaptación ante los desafíos. El mayor aprendizaje es haber entendido desde la propia práctica que un equipo de trabajo real trasciende a los desarrolladores e implica comprometer, escuchar y coordinar activamente a todos los involucrados desde el inicio y hasta el final. Estas lecciones remarcan la importancia de establecer canales directos y permanentes con todos los miembros del equipo, de asegurar la participación activa de referentes y de anticipar estrategias de reemplazo ante imprevistos como licencias o ausencias prolongadas.

## Gestión del colaborador técnico

Para gestionar el trabajo del colaborador externo se organizaron reuniones periódicas para revisar los avances, despejar dudas y capacitarlo en herramientas clave como Docker, el uso de APIs REST y conceptos del stack tecnológico general. La estrategia planificada consistió en anticipar desde el backend las funcionalidades necesarias para que el colaborador pudiera avanzar con autonomía en la implementación mobile.

La incorporación de un colaborador externo para el desarrollo del Subproyecto Mobile fue una decisión estratégica tomada con el objetivo de acortar los tiempos de implementación y permitir el avance paralelo de los distintos componentes del sistema. Sin embargo, esta forma de trabajo resultó difícil de llevar a la práctica tal como había sido definida al

comienzo del proyecto. Como se mencionó anteriormente, uno de los motivos que llevó al primer ajuste fue la identificación de que la dinámica de trabajo implementada estaba implicando una inversión de tiempo mayor a la prevista.

Aun así, la decisión de continuar el proyecto bajo esta modalidad se sustentó en la voluntad de los integrantes que, desde el comienzo del proyecto, decidieron asumir el desafío que implicaba gestionar el trabajo de un tercer integrante. A lo largo de la carrera, no se había tenido una experiencia similar, ya que este tipo de dinámicas son más propias del ámbito profesional y difíciles de replicar en un contexto académico. Por ello, se optó por afrontar esta situación como una oportunidad formativa, aun a sabiendas de las complejidades adicionales que podría introducir en el desarrollo del proyecto.

Las dificultades experimentadas pueden explicarse a partir de dos factores principales. En primer lugar, no se estimó correctamente la dedicación requerida por parte del equipo principal para gestionar el trabajo tercerizado. Aunque el cronograma original establecía una distribución clara de tareas, donde cada integrante se haría cargo de un subproyecto específico, en la práctica el Subproyecto Mobile no fue completamente autónomo. Si bien el desarrollo de la aplicación fue asignado al colaborador, aspectos como el diseño de la interfaz, la gestión de requerimientos, la validación de avances y las revisiones de código debieron ser abordados por los integrantes principales del equipo. Este esfuerzo adicional no fue correctamente anticipado, lo que llevó a una subestimación de la carga real de trabajo. A esto se suma la falta de experiencia previa en la gestión de equipos, lo que impidió identificar con claridad la distribución horas entre tareas técnicas, de gestión y de apoyo.

En segundo lugar, se presentaron dificultades asociadas a la propia naturaleza de la colaboración. El colaborador participaba de forma voluntaria, sin una relación formal ni una contraprestación económica, lo que limitaba la posibilidad de establecer compromisos de entrega o exigir el cumplimiento estricto de plazos. Esto generó una tensión entre la planificación estructurada del proyecto y la realidad de la dinámica de trabajo. Por un lado, se había planteado un esquema de desarrollo rígido, con tiempos definidos y un orden específico de ejecución de los módulos; por otro, no se contaba con mecanismos efectivos para garantizar que esa planificación pudiera cumplirse.

Para concluir, la decisión de tercerizar el desarrollo mobile terminó extendiendo más los plazos que ayudando a cumplirlos. Los numerosos imprevistos ocurridos con la gestión del colaborador técnico llevaron a una inversión totalmente mayor de tiempos a la esperada, cuando estos tiempos podrían haber sido invertidos de una forma más eficiente si se hubiesen enfocado de la forma correcta. En retrospectiva, se reconoce que la decisión correcta desde un comienzo del proyecto hubiese sido la de acortar el alcance y asumir el proyecto íntegramente por los integrantes principales, o sumar un tercer integrante principal y sumar un poco más de complejidad al sistema. Aun así, a pesar de la inversión extra de tiempo, se rescata la experiencia enriquecedora de gestionar a un tercero. Este aprendizaje es crucial para desempeñarse como un futuro profesional, tanto la toma de decisiones en

situaciones adversas, la gestión de recursos humanos, y la gestión de un equipo completo de desarrollo.

Para concluir, la decisión de tercerizar el desarrollo mobile terminó extendiendo los plazos más de lo que contribuyó a cumplirlos. Los numerosos imprevistos relacionados con la gestión del colaborador técnico implicaron una inversión de tiempo muy superior a la esperada, tiempo que, con una planificación más adecuada, podría haber sido asignado de forma más eficiente. En retrospectiva, se reconoce que la decisión más acertada desde el inicio habría sido acotar el alcance del sistema y asumir el proyecto íntegramente entre los integrantes principales, o bien incorporar un tercer integrante formal al equipo y aumentar ligeramente la complejidad funcional.

Aun así, más allá del costo en términos de tiempo, se rescata la experiencia enriquecedora de gestionar a un tercero. Este aprendizaje es crucial para desempeñarse como un futuro profesional, tanto la toma de decisiones en contextos adversos, como la coordinación de recursos humanos y la conducción de un equipo de desarrollo con roles diferenciados. Como se mencionó anteriormente, se trató de una instancia de formación práctica muy valiosa que no es usual en contextos académicos.

## Obstáculos externos

A lo largo del proyecto surgieron diversos factores externos que impactaron significativamente en el ritmo de desarrollo. La relocalización y las constantes mudanzas de Lautaro, junto con la falta de recursos básicos para iniciar el trabajo y los viajes recurrentes a Mar del Plata para rendir exámenes, generaron atrasos en la etapa inicial del proyecto.

Sin embargo, el cambio tecnológico de React a Angular fue un acierto muy valioso para recuperar parte de este tiempo. Permitió que Lautaro pudiera formarse profesionalmente durante sus mañanas laborales y, por las tardes, aplicara esos conocimientos al proyecto, avanzando con rapidez en el desarrollo de los módulos web.

El segundo factor externo que afectó considerablemente el avance del proyecto fue el cierre del contrato laboral de Lautaro, que provocó una nueva interrupción en el Subproyecto Web y una ralentización en el Subproyecto Mobile, que había sido recientemente asumido por los integrantes tras la desvinculación del colaborador técnico. Esta situación derivó directamente en la necesidad de solicitar una segunda prórroga.

Frente a este contexto adverso, el equipo mostró una alta capacidad de adaptación. Se diseñó y ejecutó un plan de emergencia: durante este período, Julián se enfocó exclusivamente en el desarrollo de la lógica de la aplicación mobile, mientras que, una vez reincorporado, Lautaro retomó el trabajo asumiendo el diseño visual y la codificación de las interfaces. Esta estrategia fue acertada, ya que no solo permitió alcanzar el producto

esperado, sino que también superó las expectativas iniciales en un período de desarrollo muy acotado.

Cabe destacar que estos avances fueron posibles gracias a la gran amistad entre los integrantes, quienes mantuvieron una comunicación constante a lo largo del proceso. En los tiempos de baja de Lautaro, Julián supo continuar con el desarrollo por su cuenta, y en los tiempos de alta de Lautaro, él supo adaptarse a los cambios realizados y mostrar gran compromiso por los avances del proyecto.

## Planificación vs ejecución

En este apartado se presenta un análisis retrospectivo de la ejecución del proyecto en relación con la planificación inicial. El objetivo es evaluar el grado de cumplimiento de los plazos y estimaciones, así como identificar aciertos y errores cometidos a lo largo del proceso.

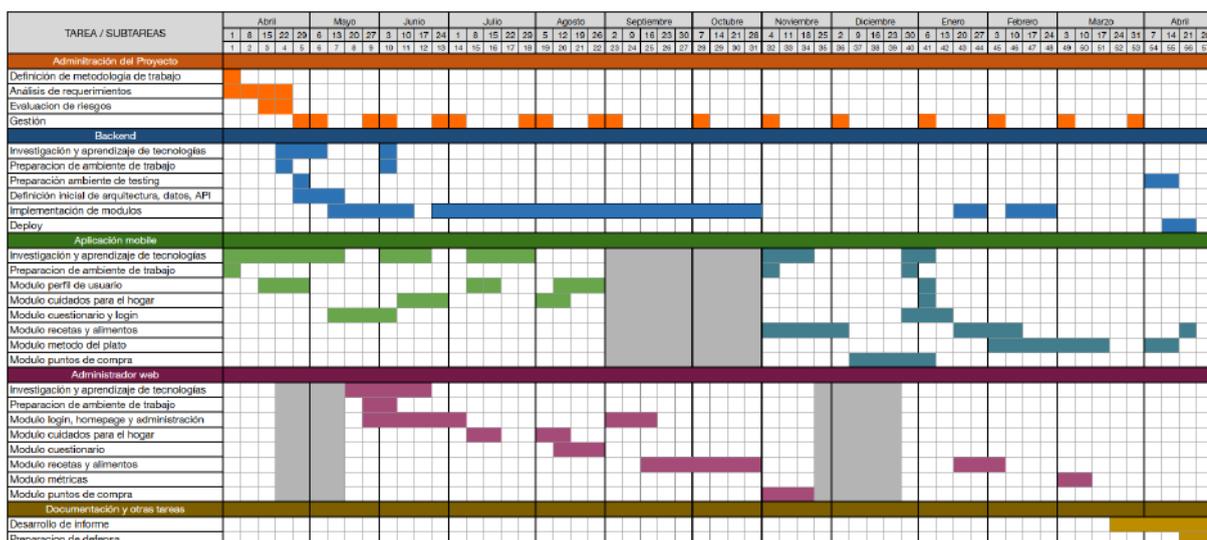
Desde el inicio, la gestión del proyecto se apoyó en el uso del diagrama de Gantt como herramienta central para el seguimiento de los avances. Se buscó plasmar con el mayor nivel de detalle posible la dedicación estimada a cada tarea, diferenciando los subproyectos y organizando el desarrollo en módulos. La expectativa era que esta planificación detallada permitiera un desarrollo ordenado y controlado.

Sin embargo, en retrospectiva, se evidenció que mantener actualizado el diagrama de Gantt y replanificar semanalmente implicaba una carga de trabajo y una complejidad considerable. Una de las principales limitaciones fue haber utilizado el Gantt como herramienta de control. Por este motivo, tras el primer ajuste, se decidió descartarlo y reemplazarlo por una checklist, con el objetivo de evitar perder tiempo recalculando el cronograma completo cada mes.

Por otro lado, la segmentación del cronograma en subproyectos sin considerar el escaso recurso humano trajo problemas de solapamiento de tareas, especialmente en el desarrollo de la aplicación mobile. Este componente había sido planificado como un trabajo tercerizado, mientras que los integrantes principales se centrarían solo en aportar con tareas de gestión y soporte. Sin embargo, no solo se subestimó el esfuerzo necesario para las tareas de acompañamiento, sino que tampoco se logró desacoplar claramente los tiempos dedicados a cada subproyecto cuando se trabajaba en conjunto al colaborador.

Considerar estas cuestiones es clave para entender los resultados que se presentan en los apartados siguientes, donde se analizan en detalle los desvíos de tiempo, las decisiones tomadas durante los ajustes, las diferencias en la carga horaria y la comparación entre las estimaciones iniciales y el esfuerzo realmente invertido.

A continuación, en la **Figura 9** se muestra el cronograma real de ejecución del proyecto, destacando en color gris los períodos de baja no previstos para cada responsable de subproyecto.



**Figura 9.** Diagrama de Gantt final con la ejecución real del proyecto.

Lo primero que se observa es un corrimiento significativo respecto a lo planificado inicialmente. Si bien el proyecto fue pensado para ejecutarse entre abril y octubre de 2024, la entrega final se concretó en abril de 2025. Este desfase de seis meses refleja una acumulación progresiva de desvíos que comenzaron a hacerse visibles en las primeras etapas del desarrollo.

No obstante, es importante señalar que parte de este corrimiento estuvo vinculado a la baja temporal de Lautaro por motivos laborales, una situación que no hubiese ocurrido en un contexto profesional donde el proyecto representaría su vía de ingreso principal. Si se excluyen esos períodos de inactividad —derivados de problemas externos y no de la toma de decisiones del equipo—, puede estimarse que el proyecto habría concluido aproximadamente dos meses antes, lo que reduciría el desfase real a un máximo de cuatro meses. Esta consideración permite poner en perspectiva el desfase real y mejorar su interpretación.

Por otra parte, se evidencia una diferencia relevante entre el orden de desarrollo planificado para los módulos y el orden en que finalmente se implementaron. Aunque se había definido una secuencia lógica basada en la complejidad técnica y las dependencias funcionales, en la práctica esa lógica debió alterarse en múltiples oportunidades. En algunos casos, la incertidumbre sobre la implementación llevó a pausar ciertos módulos y avanzar con otros que ya estaban definidos. En otros casos, la baja temporal de algún integrante imposibilitó la integración entre subproyectos, lo que obligó a reordenar prioridades en función de la disponibilidad del equipo.

## Estimación inicial

En lo relacionado a la estimación inicial, hay varios aspectos a considerar. Por un lado, la planificación se basó en un análisis exhaustivo del camino crítico del desarrollo, lo que permitió establecer una visión clara del proyecto a partir de su descomposición modular. Esta estructura inicial resultó útil para organizar las tareas y guiar las primeras etapas del trabajo. No obstante, también se presentaron diversas dificultades, así como decisiones que, en retrospectiva, podrían haberse abordado de una mejor manera.

En primer lugar, si bien la forma modular de pensar el proyecto se cree que fue acertada, no ocurrió lo mismo con la división por subproyectos. Esto agregó mucha complejidad en el mantenimiento del diagrama y su seguimiento. La granularidad utilizada buscaba reflejar con precisión el avance por subproyecto y módulo, pero en la práctica resultó muy difícil de sostener. La realidad del desarrollo obligó a realizar cambios frecuentes en el orden, la duración y la asignación de tareas. Una mejor decisión hubiese sido adoptar una planificación menos rígida, centrada en hitos o que únicamente contemple módulos sin especificar subproyectos para reflejar el flujo real del trabajo.

En segundo lugar, se subestimaron las horas que los integrantes principales del equipo invertirían en tareas de acompañamiento y soporte al colaborador técnico. Durante la planificación se creyó que estas tareas requerirían un esfuerzo limitado, concentrado principalmente en cuestiones organizativas y en la provisión de requerimientos. Sin embargo, como se detalla en el apartado *Memoria del proyecto – Gestión del colaborador técnico*, la realidad fue distinta: se debieron realizar múltiples instancias de capacitación, revisiones, asistencia técnica y coordinación, lo cual representó una carga horaria significativa no reflejada en la planificación inicial.

Por último, la planificación inicial contempló superposiciones entre ciertas tareas, en particular entre el desarrollo del sistema y la redacción del informe. Se asignaron bloques de tiempo para realizar ambas actividades en paralelo, lo cual resultó inviable en la práctica. La experiencia demostró que llevar adelante estas tareas de forma simultánea requería un nivel de esfuerzo y concentración que no era sostenible. Fue necesario, en cambio, concentrar los esfuerzos primero en el desarrollo y, una vez finalizado, centrarse en la elaboración del presente informe.

Mirando en retrospectiva, queda claro que el enfoque ideal habría sido organizar la planificación directamente por módulos, sin dividir estrictamente por subproyectos ni permitir superposiciones innecesarias. Haber atravesado este proceso permitió identificar errores y entender en profundidad las necesidades reales del proyecto. Con el aprendizaje adquirido, hoy sería posible diseñar un cronograma mucho más efectivo y ajustado a la realidad, como el que se esquematiza a continuación en la **Figura 10**.

	Análisis previo	Roles y usuarios	Cuidados	Cuestionario	Recetas y alimentos
Etapa de análisis					
Etapa de diseño					
Etapa de desarrollo					
Etapa de testing					

**Figura 10.** Esquema de planificación alternativa.

## Primer ajuste

El primer ajuste de cronograma se realizó una vez analizados los desvíos acumulados en las primeras etapas del proyecto. Estos desvíos se identificaron mediante la metodología de control de tiempos ya mencionada, la cual fue crucial para detectar a tiempo los ajustes necesarios. A diferencia de la planificación inicial, en este caso sí se contempló el tiempo destinado a brindar soporte al colaborador técnico.

Adicionalmente, este primer ajuste implicó una calibración de los tiempos restantes asignados al desarrollo mobile. Esta decisión respondió a la necesidad de anticipar la posible baja del colaborador técnico y asegurar un margen de maniobra ante eventuales imprevistos. Si bien este tiempo adicional funcionó como una forma de plan de contingencia — dado que la baja efectivamente ocurrió —, se reconoce que, desde una perspectiva profesional, una decisión como esta puede considerarse una práctica poco usual. Aún así, este margen permitió absorber tanto la desvinculación del colaborador como los períodos de baja disponibilidad de los integrantes principales, garantizando la continuidad del desarrollo y el cumplimiento de los objetivos.

En un entorno profesional, donde los recursos son mayores y las decisiones se toman con objetivos económicos, probablemente la gestión hubiese sido diferente. Ante un desempeño insuficiente, y considerando un contexto similar al ocurrido durante este proyecto, es pertinente prescindir de un recurso humano y buscar un reemplazo que permita cumplir los plazos y objetivos establecidos. Sin embargo, en un contexto académico, donde los estudiantes dispuestos a colaborar de manera voluntaria son escasos y existe una relación de compañerismo, este tipo de decisiones no resultan apropiadas ni mucho menos éticas. Por este motivo, el ajuste de tiempos fue una respuesta directamente condicionada por las características del entorno en el que se desarrolló el proyecto.

En definitiva, las modificaciones introducidas en el cronograma como respuesta a las situaciones previamente mencionadas — tanto la baja del colaborador como los períodos de menor disponibilidad de Lautaro — implicaron una reconfiguración de prioridades y una extensión en los tiempos de desarrollo. En un entorno profesional, este tipo de costos resultan difíciles de asumir; sin embargo, en el contexto académico actual, representaron una respuesta razonable frente a las dificultades enfrentadas. Como aspecto positivo, se destaca que el riesgo fue correctamente identificado con anticipación, lo que permitió evitar impactos críticos sobre la continuidad y los objetivos finales del proyecto.

## Segundo ajuste

El segundo ajuste marcó uno de los momentos más difíciles y caóticos del proyecto. El colaborador técnico decidió efectivizar su desvinculación con el proyecto, y además, a esta compleja situación se sumó la baja temporal de Lautaro, dejando por un tiempo a Julián como único responsable activo frente a un proyecto cada vez más exigente. La planificación inicial había quedado obsoleta, las expectativas estaban desalineadas con la realidad y la presión por cumplir con los plazos ya estaba desbordada.

Fue en este contexto, con un panorama desordenado y una fuerte carga emocional, donde el equipo logró hallar la templanza para frenar, repensar y reconstruir. Se tomó la decisión de reorganizar completamente el trabajo restante: Julián se haría cargo del desarrollo de la lógica de la aplicación mobile y Lautaro, al reincorporarse, asumiría la responsabilidad del diseño y codificación de las interfaces visuales.

Durante este período, Julián sostuvo el desarrollo del proyecto, logrando avances en la lógica de la aplicación mobile pese a la situación adversa. Lautaro, por su parte, regresó con la determinación suficiente para sumarse a un código que no había tocado y avanzar con compromiso y eficacia. El nivel de resiliencia demostrado en esta etapa no fue casual: fue fruto de la responsabilidad, la confianza mutua y la convicción de que el proyecto debía terminarse, a pesar de todo.

En medio de este escenario, es imposible no destacar el rol del director. A pesar de estar atravesando una situación personal compleja, logró brindarle al equipo palabras de calma, claridad y respaldo. Su acompañamiento, aún a la distancia, sirvió como ancla en un momento de mucha tensión, y permitió que, aún bajo presión, el equipo no pierda el rumbo.

Es importante resaltar que, en medio de esta reestructuración, también se puso sobre la mesa la posibilidad de cambiar la tecnología elegida para el desarrollo mobile. En ese momento, se evaluó si continuar con Kotlin o reemplazarla por una alternativa más simple que permitiera reducir la curva de aprendizaje y acelerar el desarrollo. Tras analizarlo, el equipo concluyó que no quería resignar rendimiento ni usabilidad. Uno de los pilares que definía la esencia del proyecto era llegar a los sectores más vulnerables, y priorizar la compatibilidad y el rendimiento en dispositivos limitados era parte de ese compromiso. Por ese motivo, se descartó cualquier solución que implicara sacrificar performance. Esta decisión implicó asumir una dificultad técnica mayor, pero se afrontó con determinación y, en un corto período, se logró dominar la tecnología lo suficiente como para completar la aplicación en tiempo y forma. En retrospectiva, se considera que fue una decisión totalmente acertada.

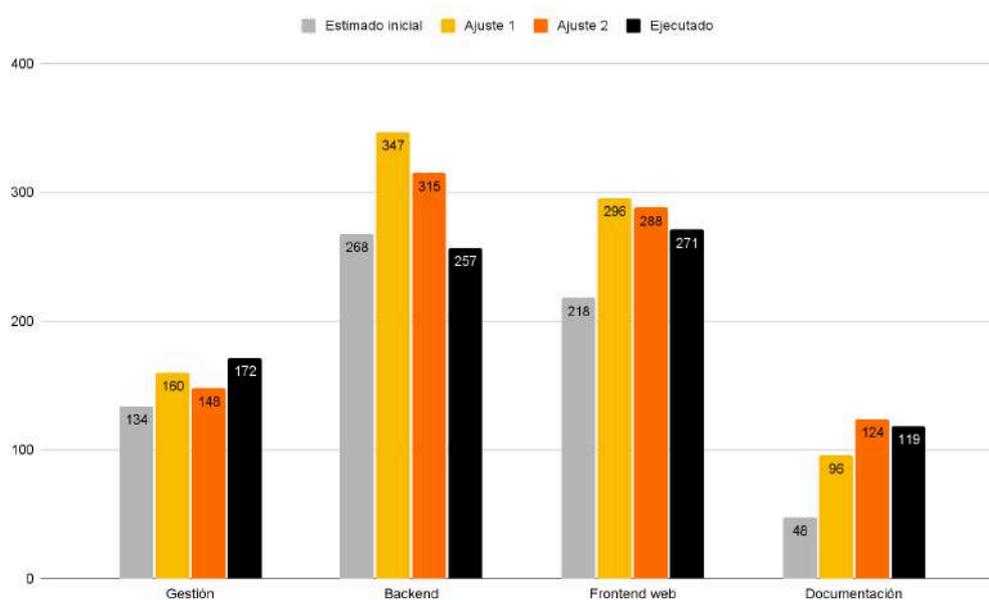
Lo ocurrido durante este segundo ajuste no fue simplemente una etapa más del proyecto, sino un punto de inflexión. El equipo se encontró ante decisiones críticas, con recursos limitados, tiempo en contra y un escenario de alta tensión. En ese contexto, fue necesario madurar de golpe: tomar decisiones firmes y asumir nuevas responsabilidades.

Fue luego de estas decisiones, cuando el equipo logró visualizar con claridad el final del proyecto. La capacidad de adaptarse, el compromiso y el trabajo en equipo no solo permitieron completar el desarrollo, sino que dejaron enseñanzas clave para el desarrollo profesional y humano de cada integrante.

## Comparación de horas por subproyecto

Se analiza a continuación la dedicación de horas de trabajo de forma general. Se contemplan tanto los tres subproyectos principales (backend, web y mobile) como también las horas destinadas a tareas de gestión y documentación. Las horas dedicadas al Subproyecto Mobile se analizarán en una sección específica por tratarse de un caso especial.

A continuación, en la **Figura 11** se presenta el gráfico con la comparativa de horas.



**Figura 11.** Comparativa de horas estimadas y ejecutadas del proyecto.

Uno de los principales aprendizajes surgidos de este análisis es la dificultad de realizar un seguimiento fiel del esfuerzo real cuando las estimaciones están estructuradas exclusivamente en torno a subproyectos asignados a personas. Como se analizó previamente, algunas tareas fueron transversales o compartidas, lo que generó solapamientos y una asignación horaria ambigua, especialmente en lo referido al soporte brindado al colaborador técnico.

Se cometió un grave error en cuanto a la contabilización de horas por parte de Julián y Lautaro en lo referido a las tareas de asistencia al colaborador técnico, particularmente por Julián quien trabajaría más directamente con él. Por considerar la asistencia al colaborador como ayuda en la integración de módulos con el backend u otros módulos, la dedicación de estas horas terminó por contabilizarse dentro del propio subproyecto de cada

integrante, cuando deberían haberse contabilizado por separado. Es por este motivo que, en los ajustes realizados, se aumentó la carga horaria en los subproyectos de Julián y Lautaro, además de otros factores como los atrasos y bajas imprevistas que alentaron en reconsiderar los tiempos. Se reconoce que hubiera sido mejor considerar estas tareas dentro del propio Subproyecto Mobile.

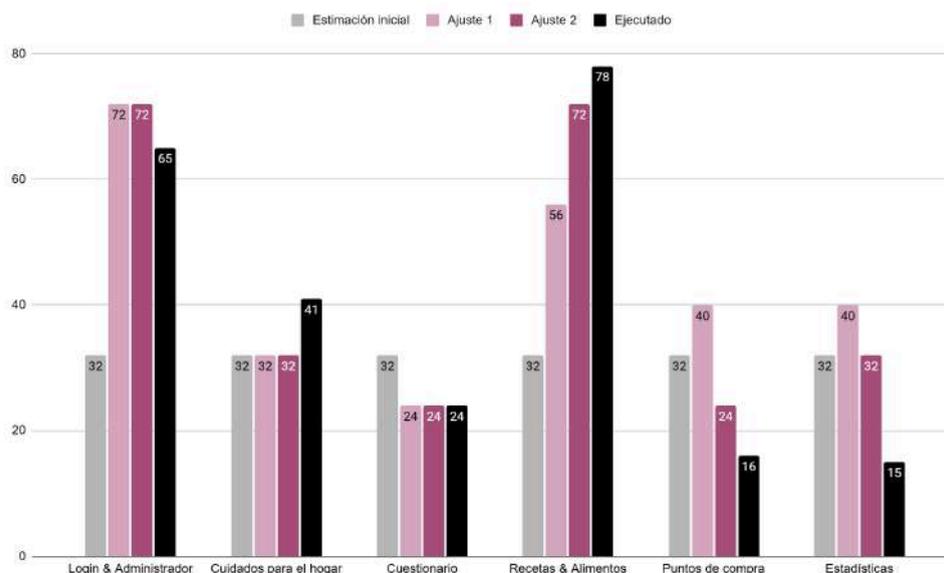
Aun así, el gráfico permite identificar algunos aciertos. En las tareas de gestión, documentación y el Subproyecto Web, las reestimaciones realizadas durante los ajustes resultaron adecuadas: se corrigieron a tiempo las proyecciones iniciales, que eran optimistas, y se alcanzó una diferencia mínima entre lo estimado en el segundo ajuste y las horas finalmente trabajadas. En particular, el Subproyecto Web mostró una desviación final de tan solo 17 horas, mientras que la documentación, inicialmente subestimada, logró una planificación cercana al valor real tras los ajustes.

En conjunto, estas diferencias evidencian tanto la utilidad de los ajustes realizados como las limitaciones del enfoque adoptado y los errores cometidos. El principal aprendizaje que surge de este análisis es la necesidad de lograr mayor coherencia entre la planificación y la ejecución. Si desde el inicio se hubiera contemplado que el Subproyecto Mobile no estaría completamente a cargo de un único integrante, esta particularidad debería haberse reflejado en el esquema de planificación.

En retrospectiva, habría resultado más adecuado adoptar un enfoque modular transversal a todo el proyecto, en lugar de segmentarlo en tres subproyectos rígidamente asociados a cada integrante. Esta estructura generó inconsistencias en la estimación, en el seguimiento y en el registro de horas, especialmente frente a tareas compartidas o interdependientes. En futuros proyectos, se buscará adoptar esquemas más flexibles y representativos de la dinámica real de trabajo, que permitan capturar con mayor precisión la distribución del esfuerzo y facilitar el análisis posterior.

## Análisis del Subproyecto Web

El desarrollo del frontend web presentó múltiples desafíos a lo largo del proyecto, especialmente condicionados por los períodos de baja en la disponibilidad de Lautaro, responsable de este componente. A continuación, la **Figura 12** presenta una comparativa de horas por módulo.



**Figura 12.** Comparativa de horas por módulo en el Subproyecto Web.

Un desacierto importante en la estimación inicial fue la asignación uniforme de 32 horas por módulo. Esta cifra surgió bajo la premisa de que los primeros módulos, aunque más simples, requerirían más tiempo por la curva de aprendizaje, mientras que los últimos, más complejos, aprovecharían la lógica previamente implementada. De este modo, se planteó una estimación promedio de aproximadamente un mes de desarrollo por módulo, donde los primeros meses se consolidarían los conocimientos tecnológicos, y los últimos meses se abordarían los módulos más complejos con mayor rapidez y soltura. Según el diagrama de Gantt planteado, se estimaron inicialmente 40 horas totales por módulo, de las cuales un 20% (8 horas) corresponden a tareas de gestión, dejando efectivamente 32 horas para desarrollo técnico.

Sin embargo, en la práctica resultó poco realista suponer que todos los módulos demandarían exactamente lo mismo. Se debería haber profundizado en el análisis funcional individual para lograr estimaciones más ajustadas. La falta de experiencia al realizar las primeras estimaciones llevó al equipo a utilizar métodos matemáticos para intentar aproximar las horas necesarias por módulo, cuando en realidad estas estimaciones suelen apoyarse en la experiencia previa que uno adquiere en tareas similares de proyectos anteriores.

Lo que efectivamente ocurrió fue que los módulos más simples, como "Cuestionario de ingreso" o "Cuidados para el hogar", se completaron sin mayores desvíos, mientras que aquellos con mayor complejidad o incertidumbre presentaron diferencias significativas respecto de lo planificado.

Al analizar detalladamente, se observa una reestimación creciente en los módulos de "Login & Administración" y "Recetas & Alimentos". El módulo "Login & Administración" debió absorber el impacto del cambio de tecnología del subproyecto. Este cambio, aunque no estaba contemplado, permitió que la dedicación de horas en los módulos finales sea considerablemente menor debido a la experiencia acumulada por Lautaro al trabajar en paralelo con la misma tecnología. Por otra parte, el módulo "Recetas & Alimentos" mostró incrementos en las reestimaciones desde etapas tempranas, tras comprobarse que requería más horas de las inicialmente previstas, basándose en la experiencia adquirida con los primeros módulos.

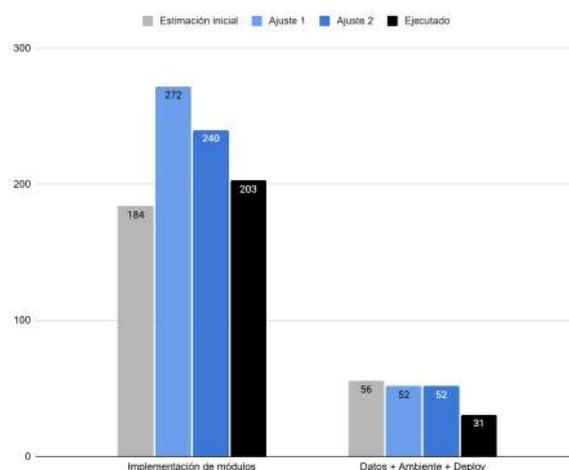
Los módulos de "Puntos de compra" y "Estadísticas" fueron identificados desde un inicio como aquellos con mayor incertidumbre (motivo por el cual se dejaron para el final), principalmente debido a su implementación. Aunque los requerimientos funcionales estaban definidos y validados, no se especificaba claramente cómo debía mostrarse la información. Esto fue detectado por el equipo, por lo que se acordó que los desarrolladores tendrían libertad para añadir los componentes que consideraran más adecuados. Esta libertad, aunque enriquecedora, también complicó las estimaciones debido a que ciertas implementaciones eran mucho más complejas que otras. Finalmente, ante la baja del colaborador técnico y un panorama ajustado, se optó por soluciones rápidas, con el fin de concentrar la atención en la parte más crítica del proyecto: el Subproyecto Mobile.

Igualmente, el equipo exploró ideas para aportar mayor valor respecto a lo inicialmente planteado, como añadir mapas para señalar locales o implementar gráficos más complejos y otros componentes visuales en las estadísticas. Estas mejoras finalmente no fueron incorporadas debido a este cambio en las prioridades del proyecto mencionado. Aun así, estos módulos lograron un alto nivel de satisfacción en las validaciones funcionales, pero estos eventos se ven reflejados en las estimaciones, donde el tiempo que se pensaba dedicar para estos módulos es considerablemente mayor al que se terminó invirtiendo.

## Análisis del Subproyecto Backend

El backend se consolidó como el subproyecto más estable. A diferencia de los demás componentes, en la planificación no se optó por dividirlo en módulos específicos, ya que su función es dar soporte a ambos frontends implementando la lógica necesaria para cada funcionalidad. Por esta razón, se lo trató como una única tarea general de desarrollo de módulos.

La **Figura 13** muestra una comparativa de las horas invertidas en las tareas más relevantes de este subproyecto.



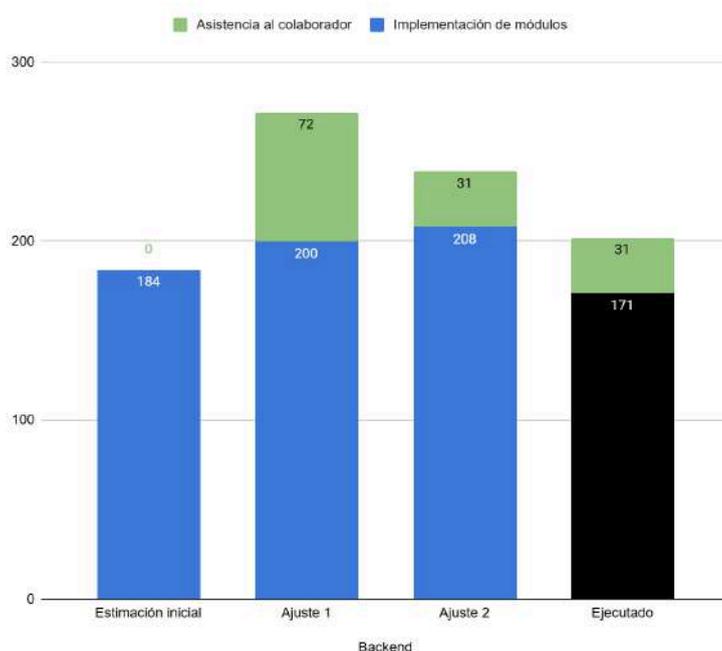
**Figura 13.** Comparativa de horas por módulo en el Subproyecto Backend.

El análisis evidencia que gran parte del tiempo fue efectivamente destinado a la implementación de endpoints y lógica del negocio, mientras que otras tareas complementarias (como configuración de base de datos, generación de datos de prueba o armado de documentación técnica) representaron una fracción menor del esfuerzo total.

El problema previamente mencionado sobre la contabilización de las horas destinadas a la asistencia del colaborador técnico se manifiesta con particular claridad en este subproyecto. A partir del primer ajuste, se incorporaron aproximadamente 72 horas asignadas a tareas de soporte en el desarrollo del colaborador, bajo la suposición de que estarían mayormente relacionadas con la integración entre el backend y la aplicación mobile. Sin embargo, una vez producida la baja del colaborador (en el segundo ajuste), dichas tareas de asistencia dejaron de ejecutarse tal como estaban previstas y las horas de trabajo se destinaron posteriormente al desarrollo mobile. Las horas ejecutadas en asistencia en desarrollo al momento de realizar el segundo ajuste fue de aproximadamente 31 horas. Esto refleja al final que las horas totales ejecutadas corresponden a 203 horas, de las cuales 171 fueron destinadas a desarrollo backend puro mientras que el resto se destinó a dar asistencia en desarrollo e integración.

Para ilustrar esta situación, la **Figura 14** presenta la distribución real de horas correspondientes a la tarea de implementación de módulos. En color azul se representan las horas estimadas para la implementación propia del backend, mientras que en verde se indican las horas agregadas para tareas de asistencia al colaborador.

En el segundo ajuste —momento en que se formalizó la baja del colaborador—, las horas en verde reflejan el tiempo efectivamente invertido hasta ese punto. Como la colaboración externa se interrumpió, se registraron únicamente las 31 horas que ya habían sido destinadas a ese acompañamiento.



**Figura 14.** Horas reales invertidas en la tarea de implementación de módulos.

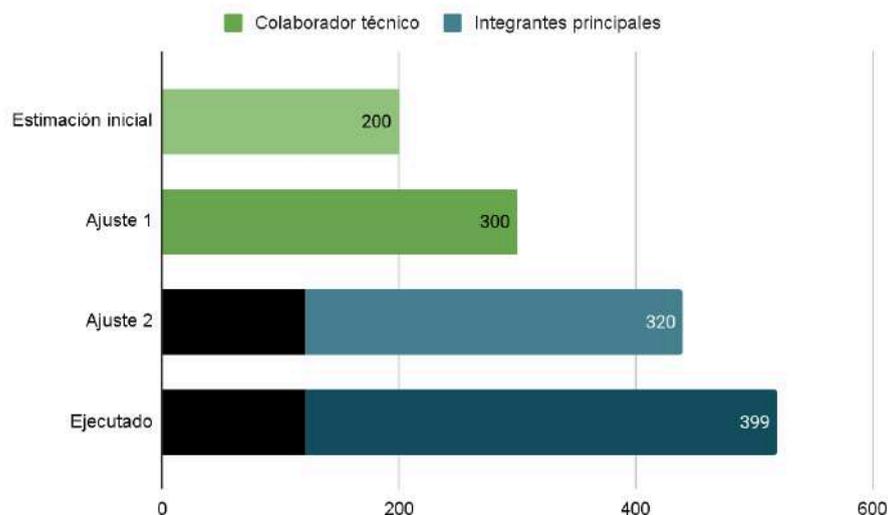
Considerando esto, se observa que las estimaciones estrictamente asociadas al desarrollo backend no experimentaron grandes variaciones respecto al punto de partida. Los ajustes posteriores reflejan leves modificaciones, propias de la incertidumbre experimentada en cada una de estas etapas.

En definitiva, los errores cometidos en la metodología de control del proyecto derivaron en ambigüedades al momento de gestionar y contabilizar las horas de trabajo en cada tarea y subproyecto. Esta dificultad también es consecuencia de la propia dinámica de trabajo, en la cual las reuniones entre los integrantes solían abordar múltiples problemáticas de forma simultánea, dificultando así la delimitación precisa del tiempo dedicado a cada actividad. Esta reflexión refuerza la idea ya planteada respecto a la conveniencia de adoptar una metodología distinta, que no divida el proyecto en subproyectos rígidos, ya que dicha segmentación introdujo complejidades innecesarias que el equipo debió resolver sobre la marcha.

## Análisis del Subproyecto Mobile

El Subproyecto Mobile presentó características particulares que lo diferenciaron del resto del sistema. A lo largo del proyecto no solo atravesó distintos ajustes en su estimación, sino que también fue desarrollado por diferentes integrantes en distintas etapas. Por este motivo, a continuación, se analiza la evolución de la carga horaria proyectada y ejecutada.

La **Figura 15** que se presenta a continuación muestra comparativamente las horas estimadas en cada instancia y las horas finalmente ejecutadas. En el gráfico, los segmentos en verde representan tiempos para el colaborador técnico, los tramos en cian corresponden al trabajo asociado a los integrantes principales, y el bloque en negro refleja el avance realizado por el colaborador antes de su desvinculación. Cabe aclarar que, dado que el desarrollo fue tercerizado en un principio, no se llevó un control estricto de su dedicación horaria. Por este motivo, el bloque se estima como aproximadamente el 20 % del desarrollo total de la aplicación.



**Figura 15.** Comparativa de horas estimadas y ejecutadas en el Subproyecto Mobile

En la estimación inicial, se contempló que el desarrollo estaría tercerizado. En ese contexto, se estimaron 200 horas para el colaborador y no se desglosaron sobre este subproyecto explícitamente las horas de asistencia brindadas por el equipo principal, sino que se consideró como trabajo de gestión.

Tras el primer ajuste, se amplió la estimación de tiempo asignada al colaborador técnico. Tal como se mencionó anteriormente, también se extendieron los tiempos destinados al equipo principal, aunque sin reflejar de forma precisa las horas invertidas en tareas de asistencia. Estas horas fueron absorbidas dentro de los otros subproyectos.

El segundo ajuste es un punto de quiebre, con la baja definitiva del colaborador, se realizó una reorganización completa del subproyecto. Se evaluó el estado del desarrollo y se reestimaron los tiempos necesarios para completarlo, esta vez asumidos por los

integrantes principales. La estimación realizada contempla el trabajo restante por realizar en ese momento. Finalmente, el total de horas ejecutadas por los desarrolladores principales fue de 399 horas, superando en 79 horas la estimación del segundo ajuste.

El Subproyecto Mobile fue, sin lugar a duda, el que más aprendizajes dejó. No solo por su complejidad técnica, sino por las múltiples implicancias humanas y organizativas que surgieron a lo largo de su desarrollo. Fue también el caso más evidente en el que se experimentó la brecha entre lo planificado y lo que realmente ocurrió en la práctica.

Uno de los desafíos centrales fue la estimación del trabajo de un tercero. La falta de experiencia previa del colaborador técnico, sumada a la poca claridad respecto a su disponibilidad horaria, introdujo un nivel de incertidumbre alto que no fue debidamente anticipado. Las expectativas sobre su rendimiento no estaban del todo definidas, lo que dificultó establecer criterios realistas de avance. Se generó así una dualidad: por un lado, existía el deseo del equipo de alcanzar un producto de calidad, capaz de ir más allá de los requerimientos mínimos; por otro, el esquema adoptado dificultaba concretar esas aspiraciones.

Inicialmente se asumió que tercerizar este desarrollo permitiría optimizar tiempos y distribuir mejor el trabajo. Sin embargo, en la práctica, los integrantes principales tuvieron que involucrarse mucho más de lo previsto, no solo para gestionar los requerimientos, sino también para acompañar, capacitar y revisar código. Esta carga adicional se asumió progresivamente, corrigiendo el rumbo sobre la marcha y con un esfuerzo mucho mayor al estimado en las primeras etapas.

Más allá de las dificultades, esta experiencia fue profundamente formativa. Permitió que los integrantes ampliaran su visión sobre lo que implica realmente gestionar un proyecto: no solo desde lo técnico, sino también desde lo humano, lo organizativo y lo comunicacional. Se comprendió que un proyecto va mucho más allá que la construcción del producto, sino que requiere considerar los vínculos, las dinámicas de trabajo y la capacidad de adaptación ante escenarios cambiantes. En retrospectiva, se reconoce que las principales dificultades no provinieron de barreras técnicas, sino de aspectos de gestión que fueron subestimados.

Este subproyecto dejó como enseñanza fundamental la importancia de ver al proyecto como un todo, con sus múltiples dimensiones. En ese sentido, afrontar y resolver estas dificultades ayudó a formar a los integrantes como profesionales más completos, con una comprensión más realista y profunda del trabajo en equipo, la toma de decisiones y la responsabilidad en entornos de incertidumbre.

## Cumplimiento de objetivos

A lo largo del desarrollo del proyecto *Senda*, el equipo se propuso un objetivo general ambicioso: crear un sistema informático capaz de mejorar la calidad de vida de personas con celiaquía. Mirando en retrospectiva, puede afirmarse que este objetivo fue alcanzado en su esencia, aunque el camino para lograrlo implicó múltiples reformulaciones, aprendizajes y decisiones clave que excedieron lo estrictamente técnico.

Uno de los primeros desafíos fue convertir una idea general —con antecedentes difusos y fragmentarios— en un producto concreto y funcional. El punto de partida fueron los requerimientos heredados de versiones anteriores del proyecto, que debieron ser reinterpretados, completados y reestructurados. Esta tarea inicial no solo sentó las bases del desarrollo, sino que permitió identificar necesidades reales, priorizar funcionalidades y pensar la solución desde una lógica modular y escalable.

El cumplimiento del objetivo general también requirió una mirada amplia sobre el concepto de “mejorar la calidad de vida”. No se trataba únicamente de brindar información o construir interfaces, sino de generar una herramienta que realmente acompañara al usuario en su día a día. En ese sentido, decisiones como incluir el módulo del método del plato, priorizar la experiencia de usuario, o garantizar una carga de alimentos validada, fueron centrales para transformar una plataforma técnica en una propuesta significativa.

En definitiva, el proyecto logró cumplir sus objetivos no por seguir un camino recto, sino por adaptarse a las condiciones del trayecto. Cada ajuste, cada replanificación y cada desacierto fueron instancias que permitieron redefinir el enfoque, sin perder de vista el propósito inicial. El resultado final refleja ese proceso: un sistema sólido, funcional y alineado con las necesidades reales de su población objetivo.

## Aprendizajes

Durante el desarrollo del proyecto se pusieron en práctica conocimientos adquiridos a lo largo de la carrera, pero, sobre todo, se incorporaron nuevos aprendizajes y experiencias en diversas áreas.

A nivel general, y más allá de lo estrictamente relacionado con la informática, se adquirieron conocimientos sobre la enfermedad celíaca, su tratamiento y las dificultades que enfrenta esta población. Además, se abordaron temas de nutrición, aprendiendo sobre los nutrientes, cómo formar una dieta balanceada y errores comunes en la alimentación, tanto en personas celíacas como en la población general.

Desde el punto de vista técnico, se afianzaron conocimientos en desarrollo backend y frontend, así como en el uso de contenedores Docker y el despliegue en entornos Linux. Además, se incorporaron conocimientos completamente nuevos, como el diseño de

interfaces y la experiencia de usuario, utilizando Figma como herramienta principal; la realización de pruebas automatizadas en el entorno Spring Boot; y, fundamentalmente, un aprendizaje desde cero sobre el desarrollo de aplicaciones Android, utilizando Kotlin y Android Studio como entorno de desarrollo.

Asimismo, se fortalecieron habilidades blandas como la planificación, la gestión del tiempo, la toma de decisiones, la resolución de conflictos y la capacidad de adaptación ante imprevistos. La experiencia dejó en claro que el éxito de un proyecto no depende únicamente del conocimiento técnico, sino también de la capacidad para reorganizarse, aprender sobre la marcha y colaborar de forma eficiente.

Por otro lado, se adquirieron conocimientos sobre gestión de proyectos y estimación de plazos, que resultaron ser de los aspectos más desafiantes del desarrollo. La elaboración de diagramas de Gantt, el registro de horas de trabajo y la toma de decisiones aportaron madurez al equipo, mejores nociones sobre los tiempos reales de desarrollo y estrategias para resolver desvíos.

Estos aprendizajes no solo enriquecieron las competencias individuales, sino que también permitieron afrontar los desafíos con mayor autonomía y seguridad.

En definitiva, el proyecto no solo representó un reto académico, sino también una instancia de crecimiento personal y profesional que preparó al equipo para encarar futuros proyectos con mayor solidez y madurez.

## Trabajo a futuro

El sistema desarrollado ofrece una base sólida de funcionalidades. Sin embargo, se pueden identificar múltiples oportunidades para su evolución y mejora en futuras etapas del proyecto.

Actualmente, el proceso de importación de datos sobre alimentos y su información nutricional se encuentra en curso. En conjunto con las referentes funcionales, se está evaluando la fuente más adecuada para utilizar. Si bien existen diversas opciones, cada una presenta limitaciones particulares. Inicialmente se consideraron las tablas ArgenFoods, pero fueron descartadas por su escasa cobertura en cuanto a la variedad de alimentos. Otra alternativa es la base de datos del USDA (Departamento de Agricultura de los Estados Unidos), que ofrece una mayor amplitud de información, aunque presenta el inconveniente de estar en inglés. Las referentes funcionales indicaron que esta base se encuentra actualmente en proceso de traducción para facilitar su incorporación.

Por otro lado, también se está trabajando junto con el grupo de trabajo de nuestra referente funcional en la configuración de valores de referencia y límites nutricionales. Estos parámetros son fundamentales para definir el comportamiento tanto del método del plato como del sistema de recomendaciones. Se está considerando el uso de los Manuales

MSD, en particular la tabla de Ingesta Diaria Recomendada (RDA) de vitaminas para adultos sanos mayores de 18 años, como fuente confiable para establecer los valores nutricionales de referencia.

Una posible línea de mejora consiste en la integración de información proveniente de la ANMAT, especialmente para incorporar productos comerciales. Esta integración permitiría a los usuarios acceder a datos oficiales sobre productos registrados. Sin embargo, actualmente la base de datos pública de la ANMAT no incluye información nutricional detallada de los productos, lo que limita su uso directo dentro del sistema de recomendaciones. Como alternativa, esta información podría incorporarse en un apartado distinto de la aplicación, diferenciado de los alimentos naturales o genéricos utilizados en el resto del sistema.

Otro punto de mejora es la incorporación de técnicas de inteligencia artificial (IA) para optimizar las recomendaciones alimentarias personalizadas. Este tipo de herramientas permitiría generar sugerencias más precisas y adaptadas a las necesidades individuales de cada usuario. También se considera valioso ampliar la sección de estadísticas con indicadores más específicos para el seguimiento nutricional por parte de profesionales, y avanzar en la implementación de pruebas automatizadas —tanto unitarias como de integración— en aquellos módulos donde por cuestiones de tiempo solo se realizaron pruebas manuales. Esto contribuiría a elevar el estándar de calidad y robustez del sistema.

Si bien el foco actual está centrado en la celiaquía, existe la posibilidad de proyectar su uso para otras condiciones relacionadas, como diabetes, hipertensión o dislipemias. En ese sentido, se considera que el sistema tiene el potencial de evolucionar hacia una herramienta de apoyo integral para el abordaje de comorbilidades, ampliando su impacto sobre distintos perfiles nutricionales y clínicos.

Además del potencial social, este trabajo abre oportunidades concretas en el plano comercial. Uno de los modelos posibles sería articular el sistema con comercios que ofrezcan productos libres de gluten, permitiendo que estos puedan promocionarse dentro de la aplicación. Este enfoque no solo generaría un beneficio económico, sino que también aportaría valor al usuario final, facilitando el acceso a productos seguros y avalados. El producto desarrollado constituye una base sólida para explorar este tipo de iniciativas.

En el plano académico, el proyecto habilita nuevas líneas de investigación y desarrollo, especialmente dentro del ámbito universitario. El sistema podría servir como base para futuros trabajos finales o prácticas supervisadas, permitiendo que otros estudiantes continúen su evolución desde distintas perspectivas. Una de las líneas a seguir podría ser la incorporación de dinámicas de gamificación, con el objetivo de mejorar la adherencia del usuario al tratamiento y fomentar hábitos saludables de manera más amigable y sostenida.

## Conclusión

El desarrollo de este proyecto representó una experiencia formativa integral para el equipo, tanto en lo técnico como en lo humano y profesional. Se abordó el diseño y construcción de un sistema complejo, orientado a brindar una solución concreta a una problemática real como es la celiacía, trabajando en conjunto con profesionales de la salud y aplicando criterios funcionales validados desde el inicio. Esta interacción constante aportó una visión interdisciplinaria que enriqueció notablemente el proceso.

A lo largo del trabajo se enfrentaron múltiples desafíos que pusieron a prueba la planificación original, como la reorganización del equipo ante la baja del colaborador técnico, la dificultad de estimar tareas tercerizadas y la necesidad de tomar decisiones críticas en contextos de incertidumbre. Estos escenarios obligaron a replantear estrategias sobre la marcha, a adaptarse y a ejercer un alto grado de autonomía, particularmente debido a la delicada situación personal del director del proyecto que, si bien siempre se mostró predisposto a colaborar y brindar acompañamiento, limitó la comunicación en algunas etapas dadas las circunstancias. Esto llevó a no aprovechar al máximo su experiencia, especialmente en lo referido a aspectos organizativos, donde su aporte podría haber sido de gran valor.

También se vivieron situaciones personales complejas por parte de los integrantes. Mientras Lautaro tuvo que afrontar el desarrollo del proyecto lejos de su familia y lidiar con circunstancias personales y profesionales difíciles, Julián tuvo que asumir por momentos el avance en solitario con el objetivo de mantener con vida la iniciativa. Estas experiencias pusieron a prueba el compromiso individual, pero también fortalecieron el vínculo profesional entre ambos, dejando en claro que el apoyo mutuo y la confianza son pilares fundamentales en contextos desafiantes.

Uno de los principales aprendizajes fue entender que un proyecto no se define únicamente por su producto final, sino también por la forma en que se gestiona el proceso que lo sostiene. Factores como la comunicación, la coordinación, la claridad de roles y la capacidad de anticiparse a escenarios adversos resultaron tan determinantes como las decisiones técnicas o tecnológicas. En este sentido, se aprendió —de forma difícil pero valiosa— que cada integrante cumple un rol esencial. La falta de una comunicación fluida con la referente funcional limitó la perspectiva multidisciplinaria del equipo y, si bien no comprometió la calidad del producto final, impidió aprovechar aportes que podrían haber enriquecido significativamente el desarrollo. Este aspecto se reconoce como un aprendizaje clave para futuros proyectos.

Si bien se cometieron errores de estimación y se subestimaron inicialmente ciertos factores humanos y organizativos, esos desvíos se transformaron en oportunidades de aprendizaje profundo. La experiencia adquirida permitió al equipo madurar profesionalmente, adquirir herramientas prácticas de gestión y fortalecer su capacidad de adaptación frente a contextos cambiantes.

Finalmente, el sistema construido no solo cumplió con los objetivos funcionales planteados, sino que logró posicionarse como una solución con potencial de impacto real. Su participación en un concurso internacional, su proyección hacia futuras líneas de investigación y su posible integración con modelos de negocio orientados al sector sin TACC refuerzan su valor. Más allá del cierre académico, este proyecto deja planteadas bases sólidas para futuras etapas de desarrollo, y consolida un recorrido que formó a sus integrantes como profesionales más completos, responsables y comprometidos con el impacto de su trabajo.

# Anexo

## Anexo I: Glosario

*Alimento apto.* Producto libre de gluten, validado para el consumo de personas con celiaquía, ya sea por naturaleza o por certificación oficial.

*API REST.* Conjunto de reglas que permiten la comunicación entre sistemas informáticos a través de peticiones HTTP. En este proyecto, se utiliza para conectar el backend con las aplicaciones frontend.

*ANMAT.* Administración Nacional de Medicamentos, Alimentos y Tecnología Médica. Organismo que regula, entre otras cosas, el listado oficial de productos “Sin T.A.C.C.” en Argentina.

*Backend.* Parte del sistema encargada de procesar la lógica de negocio, gestionar los datos y comunicar el frontend con la base de datos. En este proyecto, fue desarrollado en Java utilizando el framework Spring Boot.

*Base de datos.* Sistema que almacena información estructurada, permitiendo su consulta, modificación y eliminación. En este proyecto se utilizó PostgreSQL.

*Celíaco/a.* Persona que padece celiaquía y, por lo tanto, debe evitar el consumo de gluten de por vida.

*Celiaquía.* Enfermedad autoinmune que afecta al intestino delgado. Las personas celíacas no pueden consumir gluten, una proteína presente en trigo, avena, cebada y centeno.

*Checklist.* Lista estructurada de tareas o elementos que deben verificarse o completarse sistemáticamente, utilizada frecuentemente para asegurar la calidad y consistencia en los procesos.

*Contaminación cruzada.* Presencia involuntaria de gluten en alimentos que deberían ser libres de esta proteína. Puede ocurrir durante la manipulación, el almacenamiento o la cocción.

*CSV.* (Comma-Separated Values). Formato de archivo que almacena datos tabulares en texto plano, donde los valores están separados por comas.

*Diagrama Entidad-Relación (DER).* Diagrama utilizado para representar entidades, atributos y relaciones en una base de datos.

*Docker.* Herramienta que permite empaquetar aplicaciones y sus dependencias en contenedores, facilitando su ejecución en distintos entornos sin necesidad de configuraciones manuales.

*Dump*. Archivo que contiene una copia de los datos de una base de datos. Se utiliza para importar o restaurar información en distintos entornos de desarrollo o pruebas.

*Endpoint*. Dirección específica a través de la cual una API expone sus funcionalidades. Generalmente asociada a métodos HTTP como GET, POST, PUT o DELETE.

*Expertise*. Conjunto de habilidades, experiencia y conocimientos especializados que posee una persona sobre una temática o área específica.

*Frontend*. Parte del sistema que interactúa directamente con el usuario. En este proyecto se desarrollaron dos frontends: una aplicación web en Angular y una aplicación mobile en Kotlin.

*Gluten*. Conjunto de proteínas presentes en algunos cereales (trigo, avena, cebada y centeno). Su consumo es perjudicial para personas con celiaquía.

*Hashing y salt*. Técnicas utilizadas en seguridad informática para almacenar datos sensibles, especialmente contraseñas. El hashing transforma los datos originales en un valor único e irreversible; el salt es una cadena adicional que se añade a los datos originales antes de aplicar la función hash, incrementando la seguridad.

*JWT (JSON Web Token)*. Formato de token utilizado para autenticación segura entre cliente y servidor. Permite validar la identidad del usuario sin necesidad de mantener una sesión activa.

*Kotlin*. Lenguaje de programación moderno utilizado para desarrollar aplicaciones Android. En este proyecto se empleó para la creación de la aplicación mobile.

*Macronutriente*. Nutriente que el cuerpo necesita en grandes cantidades para su funcionamiento: proteínas, carbohidratos y grasas.

*Mentoring*. Proceso en el cual una persona con mayor experiencia (mentor) asesora, guía y apoya el aprendizaje y desarrollo profesional de otra menos experimentada (mentee).

*Micronutriente*. Nutriente esencial que el organismo requiere en pequeñas cantidades: vitaminas y minerales.

*Mock*. Objeto simulado que reemplaza una dependencia real durante una prueba automatizada. Permite aislar el comportamiento del componente bajo test.

*Pair programming*. Técnica de desarrollo de software en la cual dos desarrolladores trabajan simultáneamente en un mismo código; uno escribe código mientras el otro revisa y proporciona retroalimentación inmediata, alternándose regularmente.

*PostgreSQL*. Sistema de gestión de bases de datos relacional, de código abierto. Fue utilizado para almacenar los datos del sistema Senda.

*Soft delete*. Técnica que consiste en marcar un registro como eliminado, manteniéndolo almacenado físicamente en la base de datos en lugar de eliminarlo completamente, permitiendo así una posible recuperación futura.

*Spring Boot*. Framework de desarrollo en Java que permite crear aplicaciones robustas y escalables de forma rápida, con enfoque modular y buenas prácticas integradas.

*Stack*. Conjunto integrado de tecnologías utilizadas en el desarrollo de software, incluyendo lenguajes de programación, frameworks, librerías y bases de datos.

*Stakeholders*. Personas, grupos u organizaciones que tienen interés directo o indirecto en un proyecto, y que pueden influir o verse afectados por sus resultados. Pueden incluir usuarios, clientes, referentes funcionales, patrocinadores, instituciones y miembros del equipo de desarrollo.

*T.A.C.C.* Sigla utilizada para referirse a Trigo, Avena, Cebada y Centeno, los cereales que contienen gluten y deben evitarse en la dieta de personas celíacas.

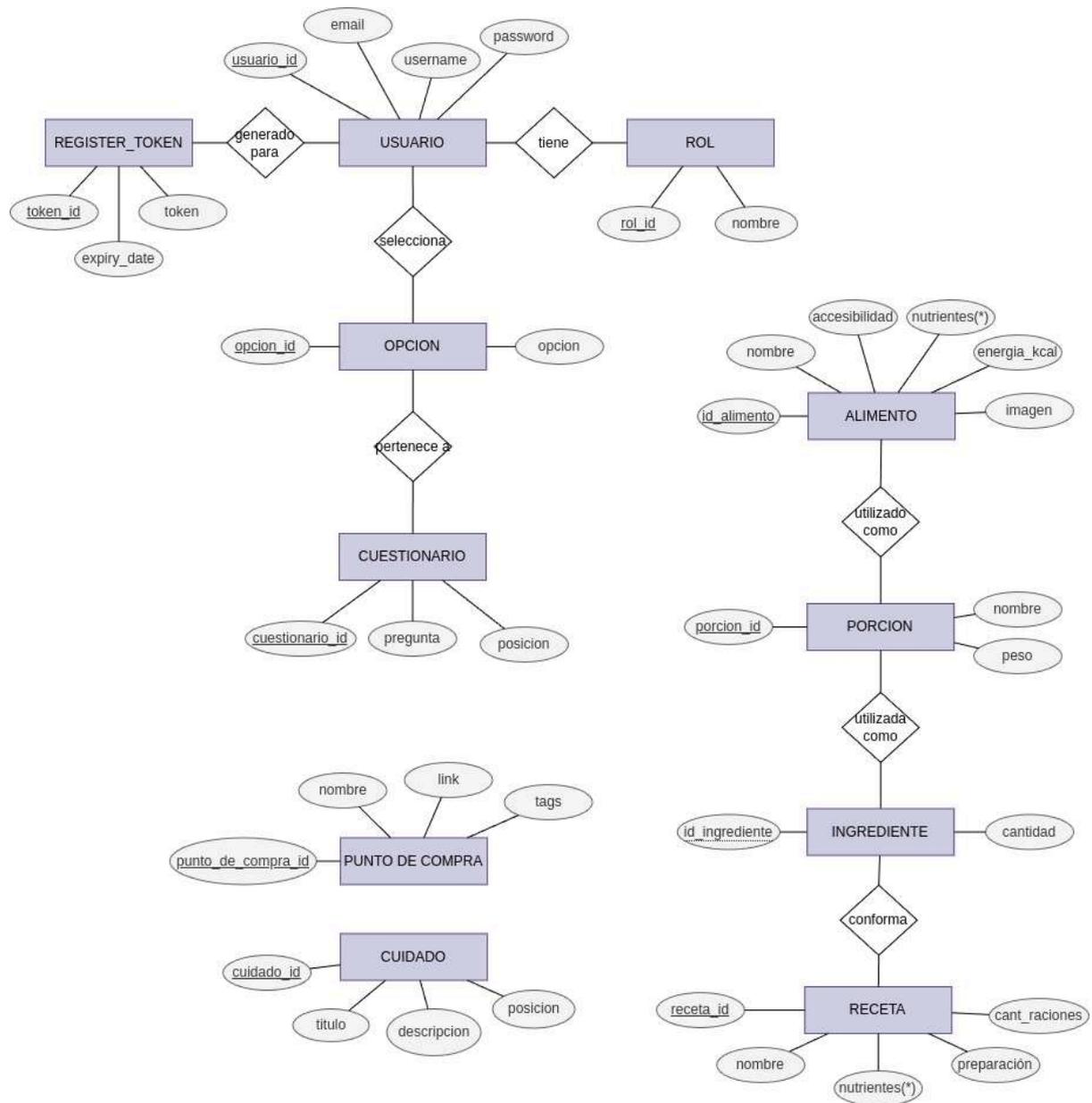
*Timestamp*. Registro temporal preciso que indica la fecha y hora exacta en que ocurre un evento específico dentro de un sistema informático.

*Token*. Identificador digital que representa una sesión autenticada. Usado para validar el acceso a recursos protegidos sin necesidad de mantener sesiones activas.

*Trade-off*. Relación de compromiso entre dos o más aspectos en conflicto, donde mejorar uno implica necesariamente sacrificar total o parcialmente otro. En el contexto del desarrollo de software, los trade-offs suelen presentarse al tomar decisiones que equilibran tiempo, calidad, rendimiento, alcance u otros factores.

*Usabilidad*. Grado en que un sistema puede ser usado con eficacia, eficiencia y satisfacción por usuarios específicos para lograr objetivos específicos.

## Anexo II: Diagrama entidad-relación simplificado (DER)



**Figura 16.** Diagrama Entidad-Relación simplificado.

## Bibliografía

1. Spring. *Spring Boot 3.2.4 Reference Documentation* [en línea]. Consultado el 5 de abril de 2024. Disponible en: <https://docs.spring.io/spring-boot/docs/3.2.4/reference/htmlsingle/>
2. Mockito. *Mockito Documentation*. Javadoc.io [en línea]. Consultado el 15 de octubre de 2024. Disponible en: <https://javadoc.io/doc/org.mockito/mockito-core/latest/org/mockito/Mockito.html>
3. Spring. *RestTemplate*. Spring Framework API [en línea]. Consultado el 15 de octubre de 2024. Disponible en: <https://docs.spring.io/spring-framework/docs/current/javadoc-api/org/springframework/web/client/RestTemplate.html>
4. Google. Android Developers [en línea]. Consultado el 4 de noviembre de 2024. Disponible en: <https://developer.android.com/?hl=es-419>
5. ANMAT. *Alimentos - Argentina.gob.ar* [en línea]. Consultado el 20 de abril de 2024. Disponible en: <https://www.argentina.gob.ar/anmat/alg>
6. USDA. *FoodData Central - Download Datasets*. U.S. Department of Agriculture [en línea]. Consultado el 26 de marzo de 2025. Disponible en: <https://fdc.nal.usda.gov/download-datasets>
7. Universidad Nacional de Luján. *Argenfoods* [en línea]. Consultado el 15 de abril de 2024. Disponible en: <https://www.argenfood.unlu.edu.ar/>
8. MSD Manuals. *Ingestión diaria recomendada de vitaminas*. MSD Professional Edition [en línea]. Consultado el 26 de marzo de 2025. Disponible en: <https://www.msdmanuals.com/es/professional/multimedia/table/ingesti%C3%B3n-diaria-recomendada-de-vitaminas>
9. NCBI. *Dietary Reference Intakes for Vitamin A, Vitamin K...* National Center for Biotechnology Information [en línea]. Consultado el 26 de marzo de 2025. Disponible en: <https://www.ncbi.nlm.nih.gov/books/NBK222881/>
10. AngularJS. *AngularJS — Superheroic JavaScript MVW Framework* [en línea]. Consultado el 25 de mayo de 2024. Disponible en: <https://angularjs.org/>
11. React. *React – A JavaScript library for building user interfaces* [en línea]. Consultado el 2 de abril de 2024. Disponible en: <https://reactjs.org/>
12. Bootstrap. *Bootstrap · The most popular HTML, CSS, and JS library in the world* [en línea]. Consultado el 12 de abril de 2024. Disponible en: <https://getbootstrap.com/>

13. Vite.js. *Vite – Next Generation Frontend Tooling* [en línea]. Consultado el 10 de abril de 2024. Disponible en: <https://vitejs.dev/>
14. MDN Web Docs. *CSS: Cascading Style Sheets* [en línea]. Consultado el 5 de mayo de 2024. Disponible en: <https://developer.mozilla.org/es/docs/Web/CSS>
15. StatCounter. *Mobile Operating System Market Share: Argentina, April 2024* [en línea]. Consultado el 5 de mayo de 2024. Disponible en: <https://gs.statcounter.com/os-market-share/mobile/argentina/#monthly-202404-202404-bar>