

Software empaquetado: Un desarrollo para Asociaciones Mutuales

Trabajo Final de Carrera
Licenciatura en Sistemas



RÍO NEGRO
UNIVERSIDAD NACIONAL

Alumno: Braian Lisandro Graff
Director: Lic. Mauricio Tassara

Índice

Introducción.....	4
Antecedentes.....	5
Inicio Ágil.....	6
User Story Mapping.....	7
Herramientas y tecnologías.....	8
Testing y Control de Calidad.....	10
Arquitectura.....	11
Requerimientos para el Despliegue.....	12
Modelado Relacional.....	13
Product backlog.....	15
Sprint Uno.....	16
User Stories.....	16
Iniciar Sesión.....	17
Alta y modificación de Empresas.....	18
Alta y modificación de Asociados.....	18
Alta, Baja y Modificación de Proveedores de Servicios.....	20
Definición de Roles del Sistema.....	21
Prototipos de Interfaz de Usuario Sprint Uno.....	22
Sprint Dos.....	23
User Stories.....	23
Alta, baja y modificación de Grupo Familiar.....	24
Interfaz basada en Roles y Permisos.....	25
Alta y baja de Créditos.....	26
Reembolso de Créditos.....	27
Alta, baja y modificación de Servicios.....	28
Alta, baja y modificación de Retenciones de Haberes.....	29
Prototipos de Interfaz de Usuario Sprint Dos.....	30

Sprint Tres.....	31
User Stories.....	31
Baja de Asociados.....	32
Consultar padrón de Asociados.....	33
Refinanciar Créditos.....	33
Cancelar Créditos.....	35
Archivo de retención.....	36
Registrar usuario.....	37
Prototipos de Interfaz de Usuario Sprint Tres.....	38
Sprint Cuatro.....	39
User Stories.....	39
Reporte de Asociado.....	40
Informar pago de cuota.....	41
Reporte de descuentos.....	42
Administrar perfil de usuario.....	43
Prototipos de Interfaz de Usuario Sprint Cuatro.....	44
Conclusiones.....	44
Referencias.....	46

Introducción

En el presente trabajo se describe cómo se gestiona el proceso de desarrollo de un software empaquetado para la administración de asociados, convenios, créditos y descuentos de haberes en una asociación mutual¹.

Las mutuales ofrecen prestaciones a sus asociados a través de sus contribuciones, ahorro o cualquier otro recurso lícito. Su objetivo es satisfacer las necesidades de los socios, ya sea mediante atención médica o farmacéutica, otorgamiento de préstamos, promoción cultural, educativa, deportiva y turística, o servicios fúnebres, entre otros.

Cada mutual envía mensualmente a las empresas donde trabajan sus socios el monto que debe descontarse de sus haberes por estar asociado y utilizar los servicios ofrecidos.

Actualmente, en general, las mutuales gestionan toda la información sobre los socios y los servicios que brindan mediante planillas de cálculo. Este enfoque es insostenible, ya que limita el crecimiento y la mejora de las prestaciones.

Además, al momento de realizar convenios con otras instituciones, resulta esencial tener la información centralizada y disponible en todo momento.

Por todos estos motivos, se ha solicitado un servicio profesional para lograr una gestión más eficiente de las asociaciones mutuales mediante la implementación de un sistema. Dado que estas necesidades son comunes en las asociaciones, se propone un sistema empaquetado: un software preconfigurado que puede ser implementado con mínimas modificaciones.

¹ Son asociaciones mutuales las constituidas libremente sin fines de lucro por personas inspiradas en la solidaridad, con el objeto de brindarse ayuda recíproca frente a riesgos eventuales o de concurrir a su bienestar material o espiritual mediante una contribución voluntaria. (Art. 2º, Ley 20.321).

Antecedentes

El desarrollo de este proyecto surge a partir de la necesidad planteada por una mutual, la cual gestionaba sus asociados y créditos mediante hojas de cálculo en Excel. Aunque funcional en sus inicios, esta solución presentaba múltiples problemas. La información no estaba centralizada, lo que dificultaba su actualización entre todos los integrantes de la organización. Además, el acceso estaba limitado a dispositivos específicos y horarios determinados, lo que impedía la disponibilidad inmediata de los datos. Por otro lado, la gestión del archivo que se enviaba a las empresas para procesar descuentos era una tarea manual compleja y propensa a errores, lo que generaba confusiones, pérdidas económicas y cobros incorrectos.

Para resolver estos problemas, se desarrolló un sistema a medida que permitió centralizar la información, automatizar procesos y mejorar la gestión operativa de la mutual. La implementación fue un éxito y generó un impacto positivo en su administración, lo que llevó a que otras mutuales de la provincia tomaran interés en la solución. Algunas de ellas incluso lograron obtener financiamiento provincial para su adquisición.

A partir de esta experiencia, y con la mejora continua del sistema en curso, que incluyó el desarrollo de una aplicación móvil para asociados, surgió un nuevo desafío: no solo desarrollar un sistema a medida para una mutual en particular, sino diseñar una solución empaquetada que pudiera ser utilizada por cualquier mutual con necesidades similares. La propuesta busca ofrecer módulos básicos como gestión de asociados, administración de ayudas económicas y descuentos de haberes, para permitir futuras adaptaciones según los requerimientos específicos de cada entidad.

Este enfoque representa un reto adicional, ya que implica pasar de un desarrollo específico a una solución estandarizada, modular y escalable, lo que demuestra la aplicación de los conocimientos adquiridos en la universidad y la evolución del trabajo profesional realizado hasta el momento.

Inicio Ágil

En las metodologías ágiles se utiliza el término 'agile inception' o inicio ágil para establecer una visión clara del producto. Se identifican los usuarios y stakeholders, se especifica el alcance inicial, se planifican las entregas y se crea el primer backlog de historias de usuarios.

La experiencia obtenida en el desarrollo de software y entrega exitosa de dos sistemas personalizados para mutuales, con adaptación a las necesidades particulares de cada una, ha permitido conocer de primera mano las particularidades del sector, los desafíos operativos y las necesidades específicas de los usuarios. El objetivo es empaquetar ese conocimiento en un sistema robusto y escalable que pueda adaptarse rápidamente a diversas mutuales, manteniendo la flexibilidad y personalización donde sea necesario.

Asumiré el rol de Product Owner, aprovechando la experiencia directa en el desarrollo de estos sistemas previos. El conocimiento de los requerimientos funcionales y las expectativas de las mutuales me permitirá definir el backlog, priorizar historias de usuarios y asegurar que el producto entregue valor real a los usuarios.

El sistema tendrá cuatro módulos principales: asociados, ayudas económicas, descuentos de haberes y ajustes del sistemas.

El módulo de asociados se centrará en la gestión de los asociados a la mutual, para permitir un manejo completo de la información de cada miembro y de su grupo familiar. Se dividirá en cuatro subcomponentes: empresas, asociados, familiares y padrón.

Las ayudas económicas representan una parte esencial de los servicios ofrecidos por las mutuales. Este módulo está dividido en dos áreas importantes: créditos y servicios de proveedores.

El módulo de descuentos de haberes es clave para la automatización y eficiencia en la recuperación de fondos prestados o servicios provistos a los asociados, ya que maneja la deducción de cuotas directamente de los haberes de los empleados. Debe permitir la integración con sistemas externos de recursos humanos, para facilitar la aplicación de

descuentos de manera automática y garantizar la correcta liquidación de los montos adeudados.

Por último el módulo de ajustes del sistema es el encargado de configurar el perfil del usuario y aspectos del sistema que no están relacionados directamente con la lógica de negocio.

User Story Mapping

Como resultado de inicio ágil, se obtiene el User Story Mapping (ver Figura 1), donde se pueden visualizar las historias de usuario organizadas para cada sprint.



Figura 1. User Story Mapping. Elaboración propia.

El mapa permite organizar el trabajo en “releases” o entregas incrementales, a partir de las historias mínimas necesarias para construir un producto mínimo viable.

En la primera entrega se habilita el alta y modificación de empresas con sus empleados o asociados, además se permite el alta, baja y modificación de proveedores de

servicios. Los usuarios pueden ingresar al sistema mediante un sistema de autenticación con un rol definido.

En la segunda entrega, se añade el alta, baja y modificación del grupo familiar de cada asociado, así como las operaciones de alta, baja y reembolsos de créditos. También se permite el alta, baja y modificación de los servicios de proveedores. Además, se incorporan funcionalidades para generar retenciones.

En la tercera entrega, se suma la funcionalidad de baja de asociados, la refinanciación y cancelación de créditos, y se incluye la posibilidad de generar el archivo con los descuentos para enviar a las empresas. También se implementa el registro de usuarios.

Por último, se agregan los reportes necesarios, la administración del perfil de usuario y la opción de informar el pago de una cuota de crédito.

Herramientas y tecnologías

Para el desarrollo de este proyecto se utilizan diversas herramientas y tecnologías que permiten un flujo de trabajo eficiente, escalable y alineado con las buenas prácticas del desarrollo de software moderno. Se prioriza el uso de tecnologías open source, para asegurar el acceso a una comunidad activa, la constante evolución y una reducción de costos en licencias.

El backend se desarrolla en javascript con Node.js y Express. Esta combinación permite desarrollar una api restful eficiente para manejar la lógica de negocio, facilitar la comunicación con la base de datos y asegurar que el backend responda de manera rápida y eficiente ante la conexión de múltiples usuarios. Para la gestión de la base de datos dentro del backend, se utiliza Sequelize, un ORM (Object-Relational Mapping) que simplifica la interacción con MySQL, permitiendo definir modelos y realizar consultas de manera más estructurada y mantenible.

El frontend se desarrolla como una Single Page Application (SPA) utilizando React, una biblioteca javascript enfocada en la creación de interfaces de usuarios interactivas y modulares. Permite construir componentes de interfaz reutilizables, con un renderizado

rápido y eficiente, esto mejora significativamente la experiencia del usuario y facilita una navegación ágil y dinámica en el sistema.

La base de datos del sistema se gestiona con MySQL, una base relacional que almacena y organiza los datos de forma segura, permitiendo el acceso rápido y eficiente a la información necesaria para el funcionamiento del backend. Su estructura relacional es adecuada para la lógica de negocio y facilita la escalabilidad del sistema.

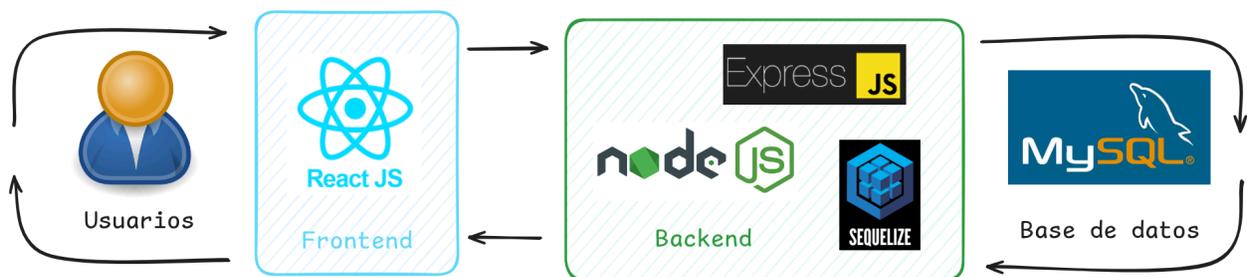


Figura 2. Tecnologías. Elaboración propia.

El control de versionados se realiza en GitHub. Git permite llevar un historial detallado de los cambios en el código, lo que facilita la colaboración entre los desarrolladores y asegura la integridad y organización del código fuente a lo largo de todo el ciclo de desarrollo.

Para el entorno de desarrollo se utiliza Visual Studio Code (VS Code), un editor de código fuente moderno, ligero y extensible. Soporta depuración de código plugins específicos para Node.js, Express.js y React. Su versatilidad facilita la productividad y mantiene el flujo de trabajo ágil.

En cuanto al modelado de base de datos, se utiliza BRModelo, una herramienta especializada en el diseño de modelos relacionales, permite crear una estructura de datos bien definida y eficiente antes de su implementación en MySQL.

Para la gestión del proyecto se utilizan varias herramientas: Jira y Trello para la organización y seguimiento de tareas. Jira es una excelente herramienta para organizar

y seguir los sprints en un entorno ágil y Trello ofrece un tablero visual accesible para la planificación y distribución de tareas y ambientes de desarrollo. Además, Miro se utiliza para facilitar la creación de diagramas y mapas mentales, mejorando la planificación y el brainstorming en equipo. Google Drive se utiliza para compartir y respaldar documentos claves del proyecto en la nube.

Por último, para publicar el sistema en internet se utiliza el hosting ofrecido por Duplika. Es un proveedor argentino por lo tanto se paga en pesos y se recibe soporte en español las 24 horas. Además, está recomendado en la página web de la Cámara de Empresas de Hosting.

Testing y Control de Calidad

La calidad del software es un pilar fundamental para garantizar su correcto funcionamiento, estabilidad y facilidad de mantenimiento. Por tal motivo durante el desarrollo se implementan estrategias de testing que abarcan distintos niveles de verificación del comportamiento esperado.

Se aplican pruebas unitarias sobre funciones específicas del backend, especialmente en la lógica de negocio, utilizando frameworks como Jest, permitiendo validar los resultados esperados de cada módulo de forma aislada.

Se implementan pruebas de integración para verificar el correcto funcionamiento entre los distintos componentes conectados entre sí. Ejemplo la interfaz con la api, la api con la base de datos.

Al final de cada sprint, se realiza una revisión manual del sistema, validando visualmente la interfaz y comprobando el correcto flujo de datos. Cada historia de usuario cuenta con criterios de aceptación que funcionan como base para definir los casos de pruebas.

En versiones futuras se plantea incluir testing automatizado en un entorno de integración continua (CI), lo que permitirá detectar errores antes de cada despliegue.

Arquitectura

Diseñar una arquitectura sólida es un paso esencial para garantizar la calidad y eficiencia de un sistema de software. Como se muestra en la imagen tres, el sistema se organiza en tres capas principales.

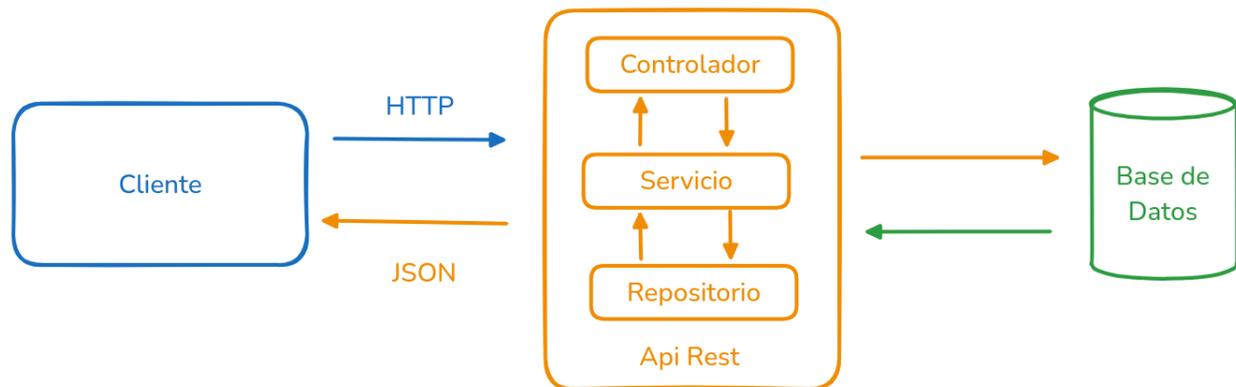


Figura 3. Arquitectura. Elaboración propia.

La capa presentación es donde los clientes interactúan con la interfaz. En este caso, es una single page application (SPA) o aplicación de una sola página creada en React para recibir y mostrar información de manera intuitiva. Esta capa se comunica con la API REST enviando peticiones http.

La capa lógica de negocio es donde se encuentra la API REST que procesa las solicitudes recibidas de la interfaz, aplica las reglas de negocio específicas consultando a la base de datos si es necesario y envía respuestas a la capa de presentación.

En la capa final de base de datos se almacena toda la información estructurada, manteniendo los datos de manera segura y accesible. En la imagen cuatro y cinco se puede ver el modelo relacional de esta capa.

Requerimientos para el Despliegue

Para desplegar la aplicación en producción se requiere un servidor virtual privado (VPS) con disponibilidad permanente en Internet y un dominio web que permita direccionar las solicitudes hacia dicho servidor

La contratación del servidor garantiza el procesamiento, almacenamiento y sistema operativo adecuados para soportar la aplicación. La ubicación del servidor resulta clave para evitar la latencia en la red. El servidor se configura con los programas necesarios para desplegar la aplicación. Es fundamental establecer medidas de seguridad que aseguren el acceso autorizado y la realización de copias de seguridad automáticas de los datos.

Se recomienda contratar un VPS con al menos 2 vCPU, 4 GB de RAM y 20 GB de almacenamiento SSD, con soporte para el sistema operativo Linux. Además, se establece un protocolo de monitoreo y mantenimiento periódico para garantizar la disponibilidad y seguridad del servidor.

Las organizaciones pueden solicitar el dominio .mutual.ar, una zona de Internet exclusiva para las mutuales argentinas registradas en el Instituto Nacional de Asociativismo y Economía Social (INAES). NIC Argentina, en colaboración con INAES, habilita este dominio para que las mutuales registren sus dominios. Este nuevo dominio de nivel superior (.mutual.ar) facilita la identificación de las mutuales argentinas en Internet.

Para solicitar el dominio, la mutual debe ceder la responsabilidad de administrar NIC.ar en la plataforma de Trámites a Distancia.

Toda la información sobre la contratación de servidores y la gestión del dominio queda documentada y accesible para la organización y quien la solicite. El servidor y la gestión de dominio deben cumplir con las normativas vigentes de protección de datos y privacidad aplicables a las mutuales argentinas.

Modelado Relacional

En el diseño de bases de datos, el proceso de modelado se divide en tres etapas principales: conceptual, lógico y físico. Cada uno de estos niveles de modelado cumple un rol específico en la creación de un sistema estructurado y bien organizado, asegurando que los datos se ajusten tanto a las necesidades del negocio como a los requisitos técnicos.

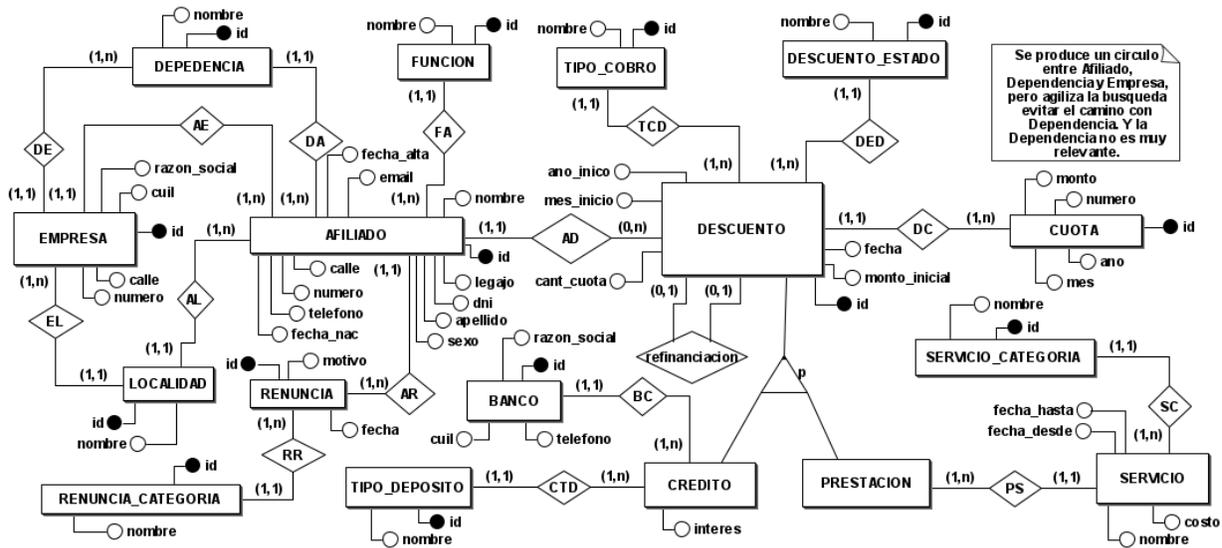


Figura 4. Modelo Relacional - Conceptual. Elaboración propia.

Modelo Conceptual: Esta es la fase inicial y se centra en una representación de alto nivel de los datos. En este modelo se identifican las principales entidades (como socios, transacciones o préstamos en el caso de una mutual), sus atributos básicos y las relaciones entre ellas, sin entrar en detalles técnicos. El objetivo es capturar el “qué” del sistema, es decir, cuáles son los datos relevantes y cómo están relacionados, usando diagramas simples que pueden entender tanto usuarios de negocio como desarrolladores.

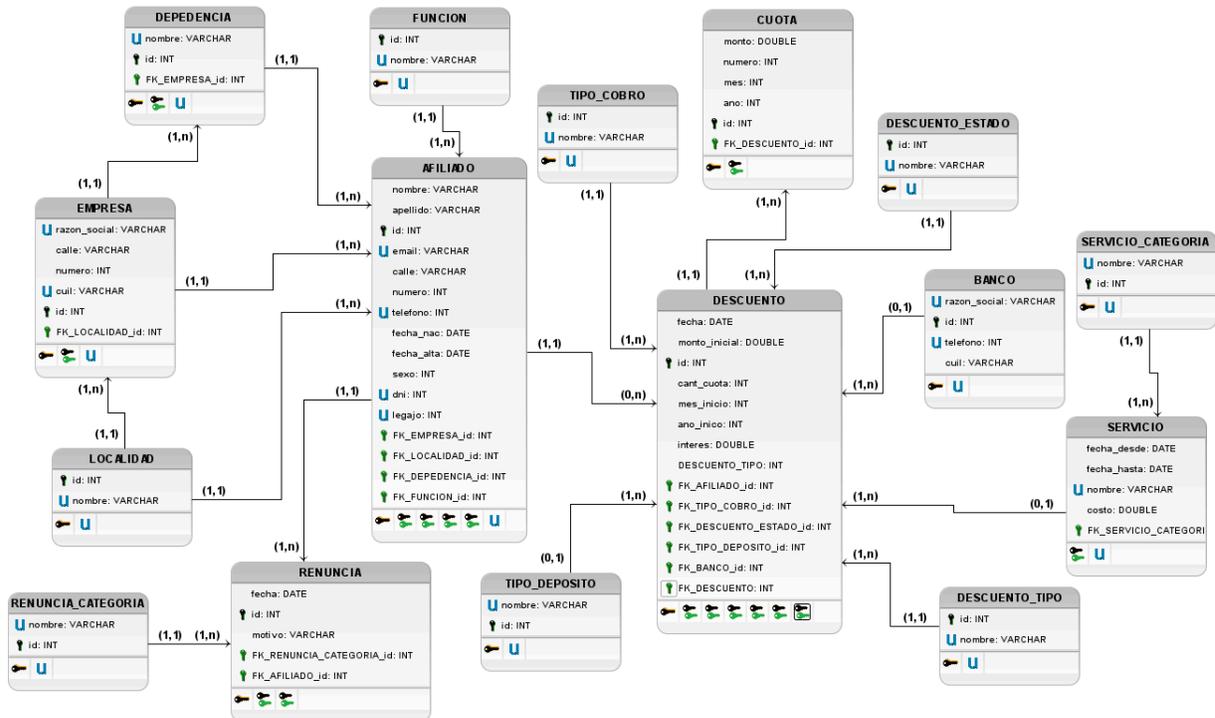


Figura 5. Modelo Relacional - Lógico. Elaboración propia.

Modelo Lógico: Este modelo toma el diseño conceptual y lo traduce en un esquema más detallado, estructurado en tablas y relaciones que siguen las reglas de la lógica relacional. Aquí se identifican las claves primarias, claves foráneas y las relaciones específicas entre las tablas, y se aplican reglas de normalización para evitar redundancias y asegurar la integridad de los datos. El modelo lógico aún no considera aspectos físicos de implementación (como índices o particiones) y se mantiene independiente del sistema de gestión de bases de datos (SGBD) que se utilizará.

Modelo Físico: En esta etapa se implementa el modelo lógico en el sistema de base de datos específico, como MySQL. Aquí se definen los detalles de almacenamiento físico, como los tipos de datos, los índices, los tamaños de los campos, y se optimizan las tablas para el rendimiento en el SGBD. El modelo físico no solo se enfoca en cómo se almacenarán los datos, sino también en cómo se accederá a ellos de manera eficiente, aplicando estrategias para consultas rápidas y minimizando el uso de recursos.

Product backlog

El Product Backlog es una herramienta clave en la gestión ágil de proyectos. Es una lista de todas las historias de usuarios necesarias para el desarrollo del software.

#	Historia de Usuario	Sprint
1	Iniciar Sesión	1
2	Alta y modificación de Empresas	1
3	Alta y modificación de Asociados	1
4	Alta, baja y modificación de Proveedor	1
5	Roles	1
6	Alta, baja y modificación de Familiar	2
7	Interfaz (layout)	2
8	Alta y baja de Créditos	2
9	Reembolso de Créditos	2
10	Alta, baja y modificación de Servicios	2
11	Alta, baja y modificación de Retenciones	2
12	Baja de Asociados	3
13	Consultar Padrón de Asociados	3
14	Refinanciar Créditos	3
15	Cancelar Créditos	3
16	Archivo de retención	3
17	Registrar usuario	3
18	Reporte de Asociados	4
19	Informar pago de cuota	4
20	Reporte de descuentos	4
21	Administrar perfil de usuario	4

Tabla 1. Product Backlog. Elaboración propia.

Sprint Uno

User Stories

El primer sprint está compuesto por cinco historias de usuarios, las cuales se detallan en la tabla 2. Al finalizar este conjunto de tareas, el usuario podrá iniciar sesión en el sistema con un rol asignado, registrar las empresas donde trabajan los asociados, cargar la información de los asociados y de los proveedores de servicios.

#	Historia de Usuario	Puntos estimados
1	Iniciar Sesión	3
2	Alta y modificación de Empresas	3
3	Alta y modificación de Asociados	5
4	Alta, baja y modificación de Proveedor	3
5	Roles	2

Tabla 2. Product Backlog - Sprint 1. Elaboración propia.

En el siguiente diagrama de Gantt se puede visualizar la secuencia de tareas de forma cronológica, mostrando cuándo inicia y termina cada una, y cómo se relacionan entre sí. Al representar cada tarea como una barra horizontal en una línea de tiempo, podemos identificar fácilmente qué tareas deben completarse antes de que otras puedan comenzar, y cuáles se pueden realizar en paralelo. Esto facilita la organización del trabajo, la detección temprana de cuellos de botella y la toma de decisiones informadas para cumplir con los plazos del proyecto.

HU / días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Iniciar Sesión															
Alta y modificación															

de Empresas															
Alta y modificación de Asociados															
Alta, baja y modificación de Proveedor															
Roles															

Tabla 3. Diagrama de Gantt - Sprint 1. Elaboración propia.

A continuación se detalla cada historia de usuario del sprint uno:

Iniciar Sesión

Historia de Usuario	
Número: 1	Usuario: Todos
Nombre historia: Iniciar Sesión	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 3	Iteración asignada: Iteración 1
Descripción: Como usuario del sistema, quiero poder iniciar sesión con mis credenciales, para acceder a las funcionalidades según mi rol y permisos.	
Criterio de Aceptación: <ul style="list-style-type: none"> ● El sistema debe permitir el inicio de sesión utilizando usuario y contraseña. ● Si las credenciales son correctas, el usuario es redirigido a su panel de inicio según su rol. ● Si las credenciales son incorrectas, se muestra un mensaje de error sin revelar detalles de seguridad. ● La contraseña debe ser almacenada de manera segura utilizando hashing. ● Se debe implementar un mecanismo de bloqueo temporal después de 5 intentos fallidos consecutivos. ● El sistema debe registrar la fecha y hora del último inicio de sesión del usuario. 	
Observaciones: Se recomienda utilizar JWT para la autenticación de sesiones en la API.	

Alta y modificación de Empresas

Historia de Usuario	
Número: 2	Usuario: Administrador del sistema
Nombre historia: Alta y modificación de Empresas	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 3	Iteración asignada: Iteración 1
Descripción: Como administrador del sistema, quiero poder registrar y modificar la información de las empresas que trabajan con la mutual, para gestionar correctamente sus datos y permitir su uso en otras funcionalidades del sistema.	
Criterio de Aceptación: <ul style="list-style-type: none">● El sistema debe permitir el registro de una nueva empresa con los siguientes datos:<ul style="list-style-type: none">○ Razón social (Obligatorio)○ CUIT/CUIL (Obligatorio)○ Abreviatura de la Razón social (Obligatorio)○ Dirección (Calle, Número, Localidad) (Obligatorio)● Se debe validar que el CUIT/CUIL sea único y tenga un formato válido.● Se debe permitir la modificación de los datos de una empresa existente.● No se debe permitir eliminar empresas que tengan registros asociados.● El sistema debe mostrar un mensaje de confirmación antes de guardar cambios.● La dirección debe permitir selección de localidad desde un catálogo predefinido.● Se debe permitir asignar un estado a la empresa (Activa/Inactiva).● El sistema debe registrar un log de auditoría con los cambios realizados sobre cada empresa.	
Observaciones:	

Alta y modificación de Asociados

Historia de Usuario	
Número: 3	Usuario: Administrador de afiliaciones

Nombre historia: Alta y modificación de Asociados	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 5	Iteración asignada: Iteración 1
Descripción: Como administrador de afiliaciones, quiero poder registrar y modificar la información de los asociados, para gestionar correctamente sus datos y permitir su uso en otras funcionalidades del sistema.	
Criterio de Aceptación: <ul style="list-style-type: none"> ● El sistema debe permitir el registro de un nuevo asociado con los siguientes datos: <ul style="list-style-type: none"> ○ Número de asociado (Obligatorio) Si no se ingresa, el sistema debe generarlo automáticamente a partir del último. ○ Nombre y apellido (Obligatorio). ○ DNI (Obligatorio) único y validado con algoritmo de verificación. ○ Fecha de nacimiento (Obligatorio). ○ Género (Obligatorio) con opciones predefinidas. ○ Teléfono (Opcional). ○ Email (Opcional) validado con formato correcto. ○ Fecha de alta (Obligatorio) Si no se ingresa, debe ser la fecha actual. ○ Dirección: <ul style="list-style-type: none"> ■ Calle (Obligatorio). ■ Número (Obligatorio). ■ Localidad (Obligatorio) seleccionable desde catálogo. ○ La Empresa donde trabaja (Obligatorio) Debe seleccionarse entre las empresas ya registradas en el sistema . ○ Número de legajo en la empresa (Opcional). ○ Función en la empresa (Obligatorio). ○ Dependencia donde trabaja / Localidad (Obligatorio) seleccionable desde catálogo. ● Se debe permitir la modificación de los datos de un asociado existente. ● No se debe permitir eliminar asociados con registros históricos en el sistema. ● El sistema debe mostrar un mensaje de confirmación antes de guardar cambios. ● Debe existir un log de auditoría con los cambios realizados sobre cada asociado. ● La interfaz debe permitir búsquedas y filtrado de asociados por nombre, DNI, empresa y localidad. 	
Observaciones: <ul style="list-style-type: none"> ● Se recomienda agregar validaciones en el backend y frontend para evitar datos incorrectos. ● Se debe asegurar que un mismo asociado no sea ingresado más de una vez con el mismo DNI. 	

Alta, Baja y Modificación de Proveedores de Servicios

Historia de Usuario	
Número: 4	Usuario: Administrador de ayudas económicas
Nombre historia: Alta, Baja y Modificación de Proveedores de Servicios	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 3	Iteración asignada: Iteración 1
Descripción: Como administrador de ayudas económicas, quiero poder registrar, modificar y dar de baja proveedores de servicios, para gestionar correctamente sus datos y permitir la vinculación con convenios de la mutual.	
Criterio de Aceptación: <ul style="list-style-type: none">● El sistema debe permitir el registro de un nuevo proveedor con los siguientes datos:<ul style="list-style-type: none">○ Razón Social (Obligatorio).○ CUIL/CUIT (Obligatorio, único y validado).○ Dirección:<ul style="list-style-type: none">■ Calle (Obligatorio).■ Número (Obligatorio).■ Localidad (Obligatorio), seleccionable desde catálogo.○ Tipo de servicio que ofrece (Obligatorio), seleccionable entre opciones predefinidas: Farmacia, Óptica, Alquiler, Médico, etc.○ Se debe permitir la modificación de los datos de un proveedor existente.○ No se debe permitir eliminar proveedores que tengan convenios activos con la mutual.○ Se debe permitir dar de baja lógica a un proveedor, inhabilitándolo para nuevos convenios.○ El sistema debe mostrar un mensaje de confirmación antes de guardar cambios.○ Debe existir un log de auditoría con los cambios realizados sobre cada proveedor.○ La interfaz debe permitir búsquedas y filtrado de proveedores por razón social, CUIT/CUIL, localidad y tipo de servicio.	
Observaciones: En el futuro, podría agregarse la posibilidad de adjuntar documentación de convenios firmados con la mutual.	

Definición de Roles del Sistema

Historia de Usuario	
Número: 5	Usuario: Administrador del sistema
Nombre historia: Definición de Roles del Sistema	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 2	Iteración asignada: Iteración 2
Descripción: Como administrador del sistema, quiero definir y documentar los roles disponibles en la plataforma, para establecer claramente las responsabilidades y permisos de cada usuario dentro del sistema.	
Criterio de Aceptación: <ul style="list-style-type: none">● Se deben definir los siguientes roles en el sistema:<ul style="list-style-type: none">○ Administrador: Tiene acceso completo al sistema, incluida la gestión de usuarios, configuración y auditoría.○ Afiliación: Responsable de la gestión de asociados y su grupo familiar, incluyendo altas, bajas y modificaciones.○ Créditos: Encargado de la administración y gestión de ayudas económicas otorgadas a los asociados, gestionar los proveedores y retenciones que serán enviadas a las empresas.○ Padrón: Accede a la visualización del padrón de asociados.● Los roles deben estar documentados y registrados en la base de datos del sistema.● No se debe permitir la eliminación de roles predefinidos en el sistema.● Se debe permitir la asignación de roles a los usuarios según las necesidades operativas.● No es necesario desarrollar una interfaz visual para gestionar los roles en esta fase.	
Observaciones: En futuras iteraciones, se evaluará la implementación de permisos específicos dentro de cada rol.	

Prototipos de Interfaz de Usuario Sprint Uno

Agregar nueva empresa

Información General:

CUIL * Razón Social * Abreviatura *

Dirección:

Calle * Número * Dependencia * VIEDMA

VOLVER AGREGAR EMPRESA

Figura 6. Alta de Empresas.

Agregar nuevo asociado

Información General:

Nro de Asociado * 340 Apellido * Nombre * DNI *

Fecha de Nacimiento * Genero * FEMENINO Teléfono

E-Mail Fecha de Alta * 14/04/2025 Afiliado al Sindicato * No

Dirección:

Calle * Número *

Empresa:

Empresa * A.R.S.A Legajo Función * ADMINISTRATIVO

Servicio * VIEDMA Dependencia * VIEDMA

VOLVER AGREGAR ASOCIADO

Figura 7. Alta de Asociados.

Sprint Dos

User Stories

El segundo sprint está compuesto por seis historias de usuarios,

#	Historia de Usuario	Puntos estimados
6	Alta, baja y modificación de Familiar	3
7	Interfaz (layout)	3
8	Alta y baja de Créditos	8
9	Reembolso de Créditos	5
10	Alta, baja y modificación de Servicios	8
11	Alta, baja y modificación de Retenciones	8

Tabla 4. Product Backlog - Sprint 2. Elaboración propia.

HU / días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Alta, baja y modificación de Familiar															
Interfaz (layout)															
Alta y baja de Créditos															
Reembolso de Créditos															
Alta, baja y modificación de Servicios															
Alta, baja y modificación de Retenciones															

Tabla 5. Diagrama de Gantt - Sprint 2. Elaboración propia.

A continuación se detallan cada historia de usuario del sprint dos:

Alta, baja y modificación de Grupo Familiar

Historia de Usuario	
Número: 6	Usuario: Administrador de afiliaciones
Nombre historia: Alta, baja y modificación de Grupo Familiar	
Prioridad en negocio (Alta / Media / Baja): Media	
Puntos estimados: 3	Iteración asignada: Iteración 2
Descripción: Como administrador de afiliaciones, quiero poder gestionar la información de mi grupo familiar dentro del sistema, para que la mutual pueda identificar a los familiares y ofrecerles servicios cuando corresponda.	
Criterio de Aceptación: <ul style="list-style-type: none">● El sistema debe permitir al asociado registrar un nuevo familiar con los siguientes datos:<ul style="list-style-type: none">○ DNI (Obligatorio) único y validado.○ Nombre y apellido (Obligatorio).○ Fecha de nacimiento (Obligatorio).○ Tipo de relación (Obligatorio), seleccionable entre: Cónyuge, Hijo, Tutor● Se debe permitir la modificación de los datos de un familiar existente.● Se debe permitir la baja de un familiar, siempre que no tenga registros activos en otros módulos del sistema.● El sistema debe validar que un familiar no pueda estar cargado más de una vez con el mismo DNI.● La interfaz debe permitir la visualización de los familiares cargados, con opciones de búsqueda y filtrado por nombre y tipo de relación.● Se debe registrar un historial de modificaciones para auditoría.	
Observaciones:	

Interfaz basada en Roles y Permisos

Historia de Usuario	
Número: 7	Usuario: Equipo de Desarrollo
Nombre historia: Interfaz (Layout) basada en Roles y Permisos	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 3	Iteración asignada: Iteración 2
Descripción: Como equipo de desarrollo, queremos diseñar y estructurar la interfaz en React de acuerdo con los roles y permisos definidos en el sistema, para asegurar que cada usuario solo acceda a las funcionalidades correspondientes a su perfil.	
Criterio de Aceptación: <ul style="list-style-type: none">● Implementar un layout dinámico que muestre opciones de navegación y contenido según el rol del usuario.● La interfaz debe considerar los siguientes roles definidos:<ul style="list-style-type: none">○ Administrador: Acceso completo al sistema.○ Afiliación: Gestión de asociados.○ Créditos: Administración de créditos.○ Padrón: Visualización de información de asociados.● Utilizar una estrategia de autenticación y autorización para controlar el acceso a los componentes.● Implementar una barra de navegación adaptable según los permisos del usuario.● Asegurar que los usuarios sin permisos adecuados no puedan acceder a rutas o funcionalidades no permitidas.● Validar los accesos en el backend para reforzar la seguridad del sistema.● La interfaz debe ser responsive y accesible en diferentes dispositivos.	
Observaciones: <ul style="list-style-type: none">● Se recomienda usar React Context API para la gestión de autenticación y permisos.● La integración con el backend debe considerar el uso de JWT	

Alta y baja de Créditos

Historia de Usuario	
Número: 8	Usuario: Administrador de ayudas económicas
Nombre historia: Alta y baja de Créditos	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 8	Iteración asignada: Iteración 2
Descripción: Como administrador de ayudas económicas, quiero poder registrar y dar de baja créditos solicitados por los asociados, para gestionar correctamente la asignación de fondos y el cobro de las cuotas.	
Criterio de Aceptación: <ul style="list-style-type: none">● Registro de Crédito:<ul style="list-style-type: none">○ Se debe seleccionar el asociado que solicita el crédito (previamente cargado en el sistema).○ Se debe ingresar el motivo del crédito, con las siguientes opciones:<ul style="list-style-type: none">■ Préstamo■ Turismo■ Salud■ Terreno○ Se debe seleccionar el tipo de cobro del crédito:<ul style="list-style-type: none">■ Retención de sueldo■ Pago en sucursal○ Se debe seleccionar el tipo de depósito para recibir el crédito:<ul style="list-style-type: none">■ Transferencia bancaria■ Efectivo■ Transferencia a tercero■ Terreno○ Se debe ingresar la fecha de alta del crédito.○ Se debe ingresar el capital solicitado (monto del crédito).○ Se debe ingresar el interés mensual, que por defecto es 2%, pero se puede modificar.○ Se debe ingresar la cantidad de cuotas, con un máximo de 10 cuotas.○ Se debe especificar desde qué mes y año se empieza a descontar el crédito.● Simulación de Cuotas:<ul style="list-style-type: none">○ Al ingresar el capital solicitado y el interés mensual, se debe mostrar un modal emergente (pop-up) con la simulación de pagos desde 1 hasta 10 cuotas.● Baja de Crédito:	

- Solo se puede dar de baja un crédito si aún no ha sido otorgado ni tiene cuotas pagas.
- Se debe registrar un historial de modificaciones y bajas.
- Todos los campos son obligatorios.

Observaciones:

- La simulación de cuotas debe calcularse con el sistema de cuota fija, aplicando el interés sobre el saldo pendiente.
- Se debe permitir la integración con el sistema de pagos y retenciones de sueldo.
- Se recomienda validar que el asociado no tenga créditos en mora antes de otorgar uno nuevo.

Reembolso de Créditos

Historia de Usuario	
Número: 9	Usuario: Administrador de ayudas económicas
Nombre historia: Reembolso de Créditos	
Prioridad en negocio (Alta / Media / Baja):	
Puntos estimados: 5	Iteración asignada: Iteración 2
Descripción: Como administrador de ayudas económicas, quiero poder procesar el reembolso de un crédito cuando el asociado ya ha pagado una o más cuotas y, por algún motivo, se debe cancelar el crédito y devolverle el dinero correspondiente.	
<p>Criterio de Aceptación:</p> <ul style="list-style-type: none"> ● Solo se puede realizar un reembolso si el crédito ya tiene una o más cuotas pagas. ● Se debe ingresar el motivo del reembolso. ● El sistema debe calcular automáticamente el monto a devolver, tomando en cuenta: <ul style="list-style-type: none"> ○ El capital abonado hasta la fecha. ○ Los intereses cobrados en las cuotas ya pagas. ○ La devolución debe excluir intereses futuros no devengados. ● Se debe seleccionar el tipo de reembolso, con las opciones: <ul style="list-style-type: none"> ○ Transferencia bancaria ○ Efectivo ● Una vez procesado el reembolso, el crédito debe quedar en estado "Reembolso" y registrarse en el historial del asociado. ● Se debe generar un comprobante de reembolso para el asociado. 	
Observaciones: Se recomienda validar que el usuario tenga los permisos adecuados antes de procesar un reembolso.	

Alta, baja y modificación de Servicios

Historia de Usuario	
Número: 10	Usuario: Administrador de ayudas económicas
Nombre historia: Alta, baja y modificación de Servicios	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 5	Iteración asignada: Iteración 2
Descripción: Como administrador de ayudas económicas, quiero poder registrar, modificar y dar de baja los servicios consumidos por cada asociado, para llevar un control de los gastos y su financiamiento.	
Criterio de Aceptación: <ul style="list-style-type: none">● Registro de Servicio Consumido:<ul style="list-style-type: none">○ Se debe seleccionar el asociado que consumió el servicio (previamente cargado en el sistema).○ Se debe seleccionar el proveedor de servicio (previamente registrado en el sistema).○ Se debe ingresar una breve descripción del servicio consumido.○ Se debe ingresar el monto total gastado.○ Se debe ingresar la fecha del consumo del servicio.○ Se debe indicar si el gasto se pagó en cuotas:<ul style="list-style-type: none">■ Máximo de 3 cuotas.■ Se debe ingresar desde qué mes y año se empieza a descontar.● Modificación de Servicio: Se permite modificar sólo la descripción o la cantidad de cuotas antes del primer vencimiento.● Baja de Servicio: Solo se puede dar de baja un servicio si aún no ha comenzado el descuento de cuotas.	
Observaciones: <ul style="list-style-type: none">● Se debe permitir la integración con el sistema de pagos y retenciones de sueldo.● En futuras iteraciones, se podría agregar un límite de consumo mensual por asociado.	

Alta, baja y modificación de Retenciones de Haberes

Historia de Usuario	
Número: 11	Usuario: Administrador de ayudas económicas
Nombre historia: Alta, baja y modificación de Retenciones de Haberes	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 8	Iteración asignada: Iteración 2
Descripción: Como administrador de ayudas económicas, quiero poder generar archivos de retención de haberes para las empresas en las que trabajan los asociados, consolidando todas las ayudas económicas del período, para que la empresa descuenta los montos correspondientes del recibo de sueldo.	
Criterio de Aceptación: <ul style="list-style-type: none">● Generación de retención de haberes:<ul style="list-style-type: none">○ Se debe seleccionar la empresa a la que se enviará el archivo (previamente cargada en el sistema).○ Se debe seleccionar el mes y año del período a generar.○ El sistema debe generar una lista de asociados con el monto total a descontar.○ Cada asociado debe aparecer en una sola fila, agrupando todas sus ayudas económicas del período.● Visualización de detalle por asociado:<ul style="list-style-type: none">○ Cada fila debe contar con un botón "Ver más" para visualizar el detalle de cada ayuda económica incluida en el descuento.○ El detalle debe incluir:<ul style="list-style-type: none">■ Tipo de ayuda económica.■ Monto cuota incluida en la retención.■ Cuota que está pagando ejemplo: cuota 3 de 10.● Modificación y Baja de retención:<ul style="list-style-type: none">○ Solo se puede dar de baja la retención y generar de nuevo.○ No se permite modificar	
Observaciones: En futuras iteraciones, se podría agregar la posibilidad de enviar automáticamente el archivo a la empresa.	

Prototipos de Interfaz de Usuario Sprint Dos

<input type="checkbox"/>	Nro	Fecha	Capital	Cuotas	Asociado	Estado	Detalle	Extender	Reembolso	Eliminar
<input type="checkbox"/>	135	05/12/2022	\$6600	3	COÑUIPAN SEBASTIAN ERNESTO	PENDIENTE				
<input type="checkbox"/>	134	05/12/2022	\$22000	4	MARCHANT JOSE HECTOR	PENDIENTE				
<input type="checkbox"/>	133	05/12/2022	\$19360	4	SEPULVEDA JULIAN EZEQUIEL	PENDIENTE				
<input type="checkbox"/>	132	05/12/2022	\$22000	4	ERICE SERGIO	PENDIENTE				
<input type="checkbox"/>	131	05/12/2022	\$22000	4	Valladares Juan José	PENDIENTE				

Filas por página: 5 1-5 de 133 < >

Figura 8. Listado de Créditos.

Buscar asociado:

Asociado DNI* Nombre* Apellido*

Detalles del crédito:

Motivo* Cobro* Deposito* Fecha de Alta*

PRÉSTAMO RETENCIÓN TRANSFERENCIA 14/04/2025

Capital* Interes Mensual* Cantidad de Cuotas* Haberes* Año*

\$ 0 % 2 0 ABRIL 2025

Descripción

Figura 9. Alta de Créditos.

Información General:

DNI* Nombre* Apellido*

Fecha de Nacimiento* Genero* Vinculo*

FEMENINO HIJO/A

Figura 10. Alta de Familiar.

Sprint Tres

User Stories

El tercer sprint está compuesto por seis historias de usuarios,

#	Historia de Usuario	Puntos estimados
12	Baja de Asociados	5
13	Consultar Padrón de Asociados	3
14	Refinanciar Créditos	7
15	Cancelar Créditos	6
16	Archivo de retención	3
17	Registrar usuario	5

Tabla 7. Product Backlog - Sprint 3. Elaboración propia.

HU / días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Baja de Asociados															
Consultar Padrón de Asociados															
Refinanciar Créditos															
Cancelar Créditos															
Archivo de retención															
Registrar usuario															

Tabla 8. Diagrama de Gantt - Sprint 3. Elaboración propia.

A continuación se detallan cada historia de usuario del sprint tres:

Baja de Asociados

Historia de Usuario	
Número: 12	Usuario: Administrador de ayudas económicas
Nombre historia: Baja de Asociados	
Prioridad en negocio (Alta / Media / Baja): Media	
Puntos estimados: 5	Iteración asignada: Iteración 3
Descripción: Como administrador de ayudas económicas, quiero poder dar de baja a los asociados registrando el motivo y la fecha de la baja, para mantener actualizado el padrón sin perder información histórica ni afectar compromisos financieros.	
Criterio de Aceptación: <ul style="list-style-type: none">● Registro de baja de asociado:<ul style="list-style-type: none">○ Se debe seleccionar el asociado a dar de baja.○ Se debe ingresar una breve descripción del motivo de baja.○ Se debe registrar la fecha de baja (por defecto, la actual, pero editable).○ Se debe seleccionar un motivo de baja entre:<ul style="list-style-type: none">■ Renuncia.■ Fallecimiento.■ Jubilación.● Baja del asociado:<ul style="list-style-type: none">○ El asociado no debe ser eliminado de la base de datos.○ Su estado cambiará a "Inactivo".○ No podrá acceder a nuevos servicios ni solicitar nuevos créditos.● Impacto en ayudas económicas y créditos:<ul style="list-style-type: none">○ Si el asociado tiene créditos activos, estos deben cambiar de estado a "Pendiente de cancelación".	
Observaciones: En futuras iteraciones, se podría agregar una funcionalidad para reactivar asociados en caso de bajas por error.	

Consultar padrón de Asociados

Historia de Usuario	
Número: 13	Usuario: Administrador de afiliaciones y rol padrón
Nombre historia: Consultar padrón de Asociados	
Prioridad en negocio (Alta / Media / Baja): Baja	
Puntos estimados: 3	Iteración asignada: Iteración 3
Descripción: Como usuario con el rol de Padrón, quiero contar con una interfaz que me permita visualizar el listado de asociados, mostrando únicamente datos no sensibles, para poder consultar de manera rápida y segura el estado de los afiliados.	
Criterio de Aceptación: <ul style="list-style-type: none">● Visualización del padrón de asociados:<ul style="list-style-type: none">○ Se debe mostrar una lista de asociados con los siguientes datos:<ul style="list-style-type: none">■ Nombre■ Apellido■ DNI■ Estado (Activo / Inactivo)○ No se deben mostrar datos sensibles como teléfono, correo electrónico o dirección.● Filtros y búsquedas:<ul style="list-style-type: none">○ Se debe poder buscar por nombre, apellido o DNI.○ Se debe poder filtrar por estado (activo/inactivo).● Accesibilidad y seguridad:<ul style="list-style-type: none">○ Solo usuarios con el rol Padrón podrán acceder a esta vista.○ La interfaz debe ser clara y permitir una navegación sencilla.	
Observaciones: Se evaluará la necesidad de agregar más filtros según los requerimientos del negocio.	

Refinanciar Créditos

Historia de Usuario	
Número: 14	Usuario: Administrador de ayudas económicas
Nombre historia: Refinanciar Créditos	
Prioridad en negocio (Alta / Media / Baja): Media	

Puntos estimados: 7	Iteración asignada: Iteración 3
<p>Descripción: Como administrador de ayudas económicas, quiero permitir que un asociado refinance uno o más créditos activos, ajustando los intereses, cuotas y fecha de inicio de los descuentos, para brindarle mayor flexibilidad en el pago y evitar incumplimientos.</p>	
<p>Criterio de Aceptación:</p> <ul style="list-style-type: none"> ● Selección de créditos a refinanciar: <ul style="list-style-type: none"> ○ Se debe poder seleccionar uno o más créditos de un mismo asociado. ○ Validar que todos los créditos seleccionados pertenezcan al mismo asociado. ● Cálculo y configuración de la refinanciación: <ul style="list-style-type: none"> ○ El sistema debe mostrar en la interfaz: <ul style="list-style-type: none"> ■ Nombre y apellido del asociado. ■ Capital total a refinanciar (suma de los créditos seleccionados). ■ Interés a aplicar (por defecto 2%, pero editable). ■ Cantidad de cuotas (máximo 10). ■ Mes y año en que se empieza a descontar. ■ Se debe seleccionar el tipo de cobro: <ul style="list-style-type: none"> ● Retención de sueldo ● Pago en sucursal ■ Se debe seleccionar el tipo de depósito: <ul style="list-style-type: none"> ● Transferencia ● Efectivo ■ Ingresar una breve descripción con el motivo de la refinanciación, como: <ul style="list-style-type: none"> ● "Unificación de créditos en uno solo" ● "Extensión de cuotas por dificultades de pago" ■ Se debe registrar la refinanciación en el sistema con el motivo "Refinanciación". ● Confirmación y cierre de operación: <ul style="list-style-type: none"> ○ Una vez confirmada la refinanciación, los créditos originales deben quedar registrados como cancelados por refinanciación. ○ Se debe generar un nuevo crédito con las condiciones refinanciadas. ○ Opción de descargar un comprobante digital con los detalles de la refinanciación. 	
<p>Observaciones:</p> <ul style="list-style-type: none"> ● Evaluar la posibilidad de agregar simulación de cuotas antes de confirmar la refinanciación. ● En futuras iteraciones, permitir la refinanciación solo después de haber pagado al menos 3 cuotas 	

Cancelar Créditos

Historia de Usuario	
Número: 15	Usuario: Administrador de ayudas económicas
Nombre historia: Cancelar Créditos	
Prioridad en negocio (Alta / Media / Baja):	
Puntos estimados: 6	Iteración asignada: Iteración 3
Descripción: Como administrador de ayudas económicas, quiero permitir que un asociado cancele uno o varios créditos pendientes en la mutual, asegurando que los cálculos sean automáticos y correctos, para garantizar una gestión eficiente y sin errores.	
Criterio de Aceptación: <ul style="list-style-type: none">● Selección de créditos a cancelar:<ul style="list-style-type: none">○ Se debe poder seleccionar uno o más créditos de un mismo asociado.○ Validar que todos los créditos seleccionados pertenezcan al mismo asociado.● Crear cancelación:<ul style="list-style-type: none">○ El sistema debe mostrar los siguientes datos en la interfaz:<ul style="list-style-type: none">■ Nombre y apellido del asociado.■ Capital total a cancelar (suma de los créditos seleccionados).■ Interés aplicado (por defecto 2%, pero editable por el usuario).■ Monto total a pagar, calculado automáticamente según el capital e interés ingresado.■ Tipo de depósito con el que el asociado pagará la cancelación:<ul style="list-style-type: none">● Transferencia● Efectivo● Otro método según la mutual○ Los campos capital total, asociado y monto total no deben ser editables, el sistema los completa automáticamente.● Confirmación y cierre de operación:<ul style="list-style-type: none">○ Una vez confirmada la cancelación, el crédito debe quedar registrado como cancelado en el sistema.○ Se debe generar un comprobante digital de la cancelación para respaldo.	
Observaciones: En futuras iteraciones, se podría agregar la posibilidad de imprimir o enviar por email el comprobante.	

Archivo de retención

Historia de Usuario	
Número: 16	Usuario: Administrador de ayudas económicas
Nombre historia: Archivo de retención	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 3	Iteración asignada: Iteración 3
Descripción: Como administrador de ayudas económicas, quiero poder generar un archivo con el detalle de las retenciones de haberes para cada empresa, en un formato adecuado para su procesamiento, de manera que pueda enviarlo fácilmente a las empresas correspondientes.	
Criterio de Aceptación: <ul style="list-style-type: none">● Generación del archivo de retención:<ul style="list-style-type: none">○ A partir de la HU 14 donde se genera la retención, debe tener la posibilidad de descargarla● Estructura del archivo:<ul style="list-style-type: none">○ El archivo debe generarse en formato CSV.○ Cada fila debe contener:<ul style="list-style-type: none">■ Número de asociado■ Nombre y apellido■ DNI■ Monto total a retener○ El archivo debe cumplir con las especificaciones de formato requeridas por las empresas.● Opciones adicionales:<ul style="list-style-type: none">○ Se debe incluir un botón para descargar el archivo.○ Validar que si no hay datos en el período seleccionado, se muestre un mensaje informativo en lugar de generar un archivo vacío.	
Observaciones: Esta historia complementa la HU 14, separando la funcionalidad de generación del archivo de la gestión de retenciones.	

Registrar usuario

Historia de Usuario	
Número: 17	Usuario: Administrador del sistema
Nombre historia: Registrar usuario	
Prioridad en negocio (Alta / Media / Baja): Baja	
Puntos estimados: 5	Iteración asignada: Iteración 3
Descripción: Como administrador del sistema, quiero poder registrar nuevos usuarios, blanquear sus contraseñas y cambiar su estado (activo o baja), para gestionar correctamente el acceso al sistema.	
Criterio de Aceptación: <ul style="list-style-type: none">● Debe permitir registrar nuevos usuarios con nombre, apellido, correo electrónico y rol.● Solo un usuario con rol Administrador puede realizar estas acciones.● Al registrar un usuario, se debe enviar un correo con las credenciales de acceso.● Debe existir una opción para blanquear la contraseña de un usuario y enviarle una nueva por correo.● Se debe poder cambiar el estado del usuario entre activo y baja (la baja no elimina el usuario, sólo impide su acceso).● No se debe permitir registrar dos usuarios con el mismo correo electrónico.● Se debe registrar la fecha de alta y de baja.● Se requiere una interfaz en la administración del sistema para gestionar los usuarios.	
Observaciones:	

Prototipos de Interfaz de Usuario Sprint Tres

Asociado:

DNI * 44444444

Nombre * Juan José

Apellido * Valladares

Detalles de la refinanciación:

Capital * \$ 22000

Interes Mensual * % 2

Cantidad de Cuotas * 0

Haberes * ABRIL

202

Motivo * PRÉSTAMO

Cobro * RETENCIÓN

Deposito * TRANSFERENCIA

Descripción Refinanciación

VOLVER REFINANCIAR

Figura 11. Refinanciar Crédito.

Buscar asociado

DESCARGAR CSV

Legajo	DNI	Nombre	Apellido	Monto
969	XX.XXX.XXX	SEBASTIAN ERNESTO	COÑUIPAN	\$10166.67
2851	XX.XXX.XXX	JEREMIAS RUBEN	SEPULVEDA	\$5500
3211	XX.XXX.XXX	Eduardo Elias	Millaqueo	\$5500
2903	XX.XXX.XXX	JULIAN EZEQUIEL	SEPULVEDA	\$3300
2883	XX.XXX.XXX	CESAR SAUL	NAVARRO	\$8800
4594	XX.XXX.XXX	JUAN DAVID	LOPEZ	\$4400
4576	XX.XXX.XXX	JULIO NESTOR	ARANDA	\$8800
4613	XX.XXX.XXX	Mauro Javier	Montero	\$4400
4607	XX.XXX.XXX	JOSE LUIS	GOROZO	\$3520
4573	XX.XXX.XXX	ROBERTO	VAZQUEZ	\$9900

Filas por página: 10 1-10 de 60

VOLVER

Figura 12. Archivo de Retención.

Sprint Cuatro

User Stories

El cuarto sprint está compuesto por cuatro historias de usuarios,

#	Historia de Usuario	Puntos estimados
18	Reporte de Asociados	3
19	Informar pago de cuota	4
20	Reporte de descuentos	4
21	Administrar perfil de usuario	3

Tabla 9. Product Backlog - Sprint 4. Elaboración propia.

HU / días	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Reporte de Asociados															
Informar pago de cuota															
Reporte de descuentos															
Administrar perfil de usuario															

Tabla 10. Diagrama de Gantt - Sprint 4. Elaboración propia.

A continuación se detallan cada historia de usuario del sprint cuatro:

Reporte de Asociado

Historia de Usuario	
Número: 18	Usuario: Administrador de afiliaciones
Nombre historia: Reporte de Asociados	
Prioridad en negocio (Alta / Media / Baja): Baja	
Puntos estimados: 3	Iteración asignada: Iteración 4
Descripción: Como administrador de afiliaciones, quiero generar un reporte de asociados que incluya todos sus datos registrados en el sistema, con la posibilidad de aplicar filtros, para obtener información específica según mis necesidades.	
Criterio de Aceptación: <ul style="list-style-type: none">● El reporte debe incluir todos los datos del asociado:<ul style="list-style-type: none">○ Número de asociado○ Nombre y apellido○ DNI○ Fecha de nacimiento○ Género○ Teléfono (si tiene)○ Email (si tiene)○ Fecha de alta○ Dirección (calle, número, localidad)○ Empresa donde trabaja○ Número de legajo (si tiene)○ Función en la empresa○ Dependencia (localidad)○ Estado del asociado (activo/inactivo)● Debe permitir aplicar filtros por:<ul style="list-style-type: none">○ Empresa○ Estado del asociado (activo/inactivo)○ Localidad○ Fecha de alta en un rango determinado○ Otros criterios según necesidades futuras● El reporte debe mostrarse en pantalla y permitir su exportación en formato CSV y PDF.● La interfaz debe permitir visualizar los datos de forma paginada y ordenada por columnas.	
Observaciones: La generación del reporte no debe afectar el rendimiento del sistema.	

Informar pago de cuota

Historia de Usuario	
Número: 19	Usuario: Administrador de ayudas económicas
Nombre historia: Informar pago de cuota	
Prioridad en negocio (Alta / Media / Baja): Alta	
Puntos estimados: 4	Iteración asignada: Iteración 4
Descripción: Como administrador de ayudas económicas, quiero buscar los créditos con modalidad de pago en sucursal, seleccionar la cuota correspondiente y confirmar como pagada, para actualizar el estado de la deuda del asociado y registrar la fecha de pago.	
Criterio de Aceptación: <ul style="list-style-type: none">● Se debe poder buscar un crédito por número de asociado o DNI.● Se deben listar las cuotas asociadas al crédito con su estado (pendiente o pagada).● Se debe permitir seleccionar una cuota pendiente y presionar un botón "Pagar".● El sistema debe solicitar confirmación antes de procesar el pago.● Una vez confirmado, el estado de la cuota debe cambiar de "Pendiente" a "Pagada".● Se debe registrar automáticamente la fecha de pago.● El sistema debe permitir visualizar el historial de pagos de un crédito.	
Observaciones: <ul style="list-style-type: none">● Solo los administradores de ayudas económicas pueden realizar esta acción.● El proceso debe ser auditable para evitar modificaciones no autorizadas.● En las próximas iteraciones se planea enviar por correo el comprobante de pago.	

Reporte de descuentos

Historia de Usuario	
Número: 20	Usuario: Administrador del sistema y ayudas económicas
Nombre historia: Reporte de descuentos	
Prioridad en negocio (Alta / Media / Baja): Baja	
Puntos estimados: 4	Iteración asignada: Iteración 4
Descripción: Como administrador del sistema, quiero generar un reporte que incluya todas las ayudas económicas otorgadas, junto con el asociado que las utilizó, para visualizar de manera clara los descuentos aplicados y facilitar su análisis.	
Criterio de Aceptación: <ul style="list-style-type: none">● El reporte debe listar todas las ayudas económicas junto con el asociado que las utilizó.● Se debe poder filtrar por rango de fechas, tipo de ayuda económica y estado del descuento.● El reporte debe mostrar los siguientes datos:<ul style="list-style-type: none">○ Nombre y apellido del asociado.○ DNI del asociado.○ Tipo de ayuda económica.○ Fecha de otorgamiento.○ Monto total de la ayuda.○ Cantidad de cuotas.○ Cuotas ya descontadas y pendientes.● El reporte debe permitir exportación en formato CSV y PDF.● Debe ser accesible solo para usuarios con permisos de administración y créditos.	
Observaciones: Este reporte servirá para auditar las ayudas económicas otorgadas y verificar los descuentos aplicados en sueldos o pagos en sucursal.	

Administrar perfil de usuario

Historia de Usuario	
Número: 21	Usuario: Todos los usuarios registrados
Nombre historia: Administrar perfil de usuario	
Prioridad en negocio (Alta / Media / Baja): Baja	
Puntos estimados: 3	Iteración asignada: Iteración 4
Descripción: Como usuario del sistema, quiero poder administrar mi perfil para modificar mis datos personales y credenciales de acceso, asegurando que mi información esté actualizada.	
Criterio de Aceptación: <ul style="list-style-type: none">● El usuario puede modificar su nombre, correo electrónico, nombre de usuario y contraseña.● La actualización de la contraseña requiere la confirmación de la clave actual.● Si se cambia el correo electrónico, el sistema debe enviar un correo de verificación.● A futuro, se podrá subir una foto de perfil.● Solo el usuario autenticado puede modificar su propio perfil.● Los cambios deben reflejarse en el sistema de inmediato.	
Observaciones:	

Prototipos de Interfaz de Usuario Sprint Cuatro

Empresa*
 Servicio*
 Dependencia*
 Función*

Genero*
 Afiliado al Sindicato*
 Activo*

Resultado de la búsqueda:

COLUMNAS FILTROS DENSIDAD EXPORTAR

Nombre	Apellido	Genero	Activo	Localidad	Tipo
	HUENCHUNAO	MASCULINO	SI	FDEZ ORO	TITULAR
	MARINAO	FEMENINO	NO	GENERAL ROCA	TITULAR
	MARINAO	MASCULINO	NO		HIJO/A
	MARINAO	FEMENINO	NO		HIJO/A
	MARINAO	FEMENINO	NO		HIJO/A
	VALDIVIESO	FEMENINO	SI	SAO	TITULAR
	MARINI	MASCULINO	SI	VIEDMA	TITULAR
	BENGOLEA	FEMENINO	SI		CÓNYUGE
	CANIUCURA	MASCULINO	SI	ALLEN	TITULAR
	CANIUCURA	MASCULINO	SI		HIJO/A
	CANIUCURA	MASCULINO	SI		HIJO/A
	CANTERO	MASCULINO	NO	GENERAL ROCA	TITULAR
	SALAS	MASCULINO	SI	CENTRAL	TITULAR
	ISABELLA	FEMENINO	SI		HIJO/A
	UMA CAROLINA	FEMENINO	SI		HIJO/A

1-15 de 666 < >

Figura 13. Reporte de Asociados.

Año*
 Mes*
 Estado*
 Empresa*

Motivo*
 Cobro*
 Depósito*

SUMA TOTAL DE LA BUSQUEDA: \$405918.34

COLUMNAS FILTROS DENSIDAD EXPORTAR

Legajo	Nombre	Apellido	Mes Inicio	Interes	\$Capital	Cant. Cuotas	Cuotas Pendientes	\$Monto Cuota	Cobro
		MONTERO	9	0	22.000	5	2	4400	RETENCIÓN
		SALAS	9	0	27.500	5	2	5500	RETENCIÓN
		CASANOBA	10	0	16.500	3	1	5500	RETENCIÓN
		MONCADA VAZQUEZ	9	0	22.000	4	1	5500	RETENCIÓN
		CARTES	9	0	17.600	4	1	4400	RETENCIÓN
		VAZQUEZ	9	0	22.000	4	1	5500	RETENCIÓN
		CABA	9	0	22.000	4	1	5500	RETENCIÓN
		CASANOBA	9	0	29.333,33	4	1	7333,33	RETENCIÓN
		NAVARRO	9	0	22.000	4	1	5500	RETENCIÓN
		CANIUCURA	9	0	22.000	4	1	5500	RETENCIÓN

1-10 de 79 < >

Figura 14. Reporte de Créditos.

Conclusiones

El desarrollo de un sistema empaquetado para mutuales representa una solución ligera y accesible no solo para estas entidades sino también para cualquier organización que necesite gestionar registros de afiliaciones y créditos de manera centralizada y eficiente. A diferencia de un desarrollo a medida, esta solución estandarizada permite una implementación más rápida, con costos reducidos y la posibilidad de escalar según las necesidades de cada institución.

La experiencia adquirida a lo largo de este proyecto demuestra que es posible diseñar un sistema flexible y adaptable, brindando herramientas tecnológicas que faciliten la gestión operativa de mutuales y otras organizaciones con estructuras similares.

Este trabajo no solo ha significado un desafío técnico y profesional, sino también un crecimiento personal. Durante muchos años trabajé en relación de dependencia en el área de sistemas, desempeñando diversas tareas que iban desde redes, servidores, impresoras, mesa de ayuda y software, hasta instalaciones eléctricas. Sin embargo, este proyecto marcó un punto de inflexión en mi carrera, ya que por primera vez desarrollé un software de manera independiente, asumiendo el rol de responsable en cada etapa del proceso y siendo la cara visible ante los clientes. Esto me llena de orgullo y es un logro que le debo especialmente a la Universidad Nacional de Río Negro, cuya formación y acompañamiento fueron fundamentales para alcanzar este objetivo.

Referencias

1. React documentación.

<https://es.react.dev/>

2. Node. documentación.

<https://nodejs.org/es>

3. Visual Studio Code documentación.

<https://code.visualstudio.com>

4. MySQL documentación.

<https://www.mysql.com>

5. Estándares de APIs - Secretaría de Innovación, Ciencia y Tecnología.

<https://github.com/argob/estandares/blob/master/estandares-apis.md>

6. Metodología Ágil.

<https://www.atlassian.com/es/agile>

7. Argentina.gob.ar. MutualAR

<https://www.argentina.gob.ar/noticias/mutualar-la-nueva-zona-exclusiva-para-mutuales-argentinas>