

Desarrollo de Software

Tarjeta Vecino

Francisco Javier Barraza Rifo

Bayron Andrés Garri Mora

Fernanda Constanza Morales Guzmán

Abel Ignacio Rosales Valenzuela

Octubre de 2023

Concepción – Chile

Índice General

[**1**](#_heading=h.30j0zll) **Introducción 6**

[**2**](#_heading=h.1fob9te) **Definición del Software 7**

[**2.1**](#_heading=h.3znysh7) **Contexto – Procesos 7**

[**2.2**](#_heading=h.2et92p0) **Objetivo del software: General y específicos 7**

[**2.3**](#_heading=h.tyjcwt) **Límites 7**

[**2.4**](#_heading=h.3dy6vkm) **Requerimientos Funcionales 7**

[**2.5**](#_heading=h.4d34og8) **Requerimientos No Funcionales 7**

[**2.6**](#_heading=h.2s8eyo1) **REQUERIMIENTOS de Interfaces externas de entrada 8**

[**2.7**](#_heading=h.3rdcrjn) **REQUERIMIENTOS de Interfaces externas de salida 8**

[**3**](#_heading=h.lnxbz9) **ANÁLISIS 9**

[**3.1**](#_heading=h.35nkun2) **Modelo de casos de uso 9**

[3.1.1](#_heading=h.1ksv4uv) Diagramas de casos de uso 9

[3.1.2](#_heading=h.2jxsxqh) Actores-Roles 9

[3.1.3](#_heading=h.3j2qqm3) Especificación de casos de uso 9

[**4**](#_heading=h.4i7ojhp) **DISEÑo 11**

[**4.1**](#_heading=h.2xcytpi) **Modelo de datos 11**

[**4.2**](#_heading=h.3whwml4) **Diseño de interfaz y navegación 11**

[**5**](#_heading=h.3as4poj) **Arquitectura de software 12**

[**6**](#_heading=h.2p2csry) **VERIFICACIÓN 15**

[**6.1**](#_heading=h.147n2zr) **Verificación requisitos (Interfaces de Entrada) 15**

[**6.2**](#_heading=h.3o7alnk) **Verificación requisitos (Interfaces de Salida) 15**

[**6.3**](#_heading=h.23ckvvd) **Verificación requisitos CU 15**

[**7**](#_heading=h.ihv636) **BIBLIOGRAFÍA 16**

[**8**](#_heading=h.32hioqz) **ANEXO: DEL PROYECTO 17**

[**8.1**](#_heading=h.1hmsyys) **Objetivos del Proyecto, general y específicos 17**

[**8.2**](#_heading=h.41mghml) **Programación de actividades 17**

[**8.3**](#_heading=h.2grqrue) **Estimación de tamaño 17**

[**9**](#_heading=h.vx1227) **Anexo: ambiente de ingeniería 20**

[**9.1**](#_heading=h.3fwokq0) **Método de Desarrollo 20**

[**9.2**](#_heading=h.1v1yuxt) **Técnicas 20**

[**9.3**](#_heading=h.4f1mdlm) **Herramientas y estándares 20**

[**10**](#_heading=h.2u6wntf) **Anexo: Esfuerzo real después de terminado el proyecto 21**

[**11**](#_heading=h.3tbugp1) **ANEXO: Especificación de las Pruebas 22**

[**12**](#_heading=h.nmf14n) **ANEXO: Diccionario de datos del modelo de datos 23**

[**13**](#_heading=h.37m2jsg) **ANEXO: Entrevistas 24**

[**14**](#_heading=h.1mrcu09) **ANEXO: Cuestionarios 25**

[**15**](#_heading=h.46r0co2) **ANEXO: otros Documentos recopilados 26**

Índice Tablas

[Tabla 1: Requerimientos funcionales. 7](#_heading=h.1t3h5sf)

[Tabla 4: Interfaces externas de entrada 7](#_heading=h.17dp8vu)

[Tabla 5: Interfaces externas de salida. 8](#_heading=h.26in1rg)

[Tabla 5: Actor "Usuario invitado" 9](#_heading=h.z337ya)

[Tabla 6: Caso de uso " **NNNNN<CU001>**". 9](#_heading=h.1y810tw)

[Tabla 8: Resumen esfuerzo requerido. 21](#_heading=h.19c6y18)

[Tabla 9: Detalle de prueba "**Editar institución social**". **ejemplo** 22](#_heading=h.28h4qwu)

Índice Figuras

[Ilustración 1: Diagrama de casos de uso 1 9](#_heading=h.44sinio)

[Ilustración 2: Modelo entidad relación (MER). 11](#_heading=h.1ci93xb)

[Ilustración 4: Mockup "Página de inicio - web". 11](#_heading=h.2bn6wsx)

[Ilustración 5: Mockup "Página de inicio - móvil". 11](#_heading=h.qsh70q)

[Ilustración 6: Arquitectura Sitio Web **ejemplo** 13](#_heading=h.1pxezwc)

[Ilustración 7: Estructura de directorios app web. **ejemplo** 14](#_heading=h.49x2ik5)

# Introducción

El siguiente informe tiene como objetivo presentar en detalle las especificaciones técnicas correspondientes al desarrollo del software “Tarjeta Vecino”. El diseño de este software busca satisfacer las necesidades de las municipalidades y sus habitantes, brindando un ambiente simplificado y óptimo para el proceso de obtención y utilización de la Tarjeta Vecino.

El desarrollo de esta aplicación responde a la necesidad del gobierno establecida en la Ley N° 21.180, la cual modifica las bases de los procedimientos administrativos para avanzar en su digitalización (https://digital.gob.cl/, s. f.), esto con el fin de facilitar el acceso de los habitantes de las diversas comunas de Chile a los diversos beneficios disponibles en su localidad adquiriendo la “Tarjeta Vecino”.

La finalidad de este software es brindar una plataforma ágil y eficaz para la gestión de solicitudes de tarjetas, un acceso simplificado a la información de los convenios y beneficios disponibles en la localidad y un sitio accesible y de fácil acceso para la postulación a la tarjeta, dando prioridad a grupos específicos de la población, como adultos mayores, personas gestantes y personas con discapacidad.

A lo largo de este informe se explorarán los aspectos técnicos, funcionales y de diseño del software, ahondando principalmente en la forma en la que éste operará, las limitaciones del sistema y los requerimientos técnicos necesarios para su implementación.

El informe se encuentra organizado de acuerdo con las siguientes secciones, las cuales buscarán abarcar los diferentes aspectos relacionados con el desarrollo y funcionamiento del sistema “Tarjeta Vecino”

1. Introducción
2. Definición del Software
3. Análisis
4. Diseño
5. Arquitectura del Software
6. Verificación

# Definición del Software

## Contexto – Procesos

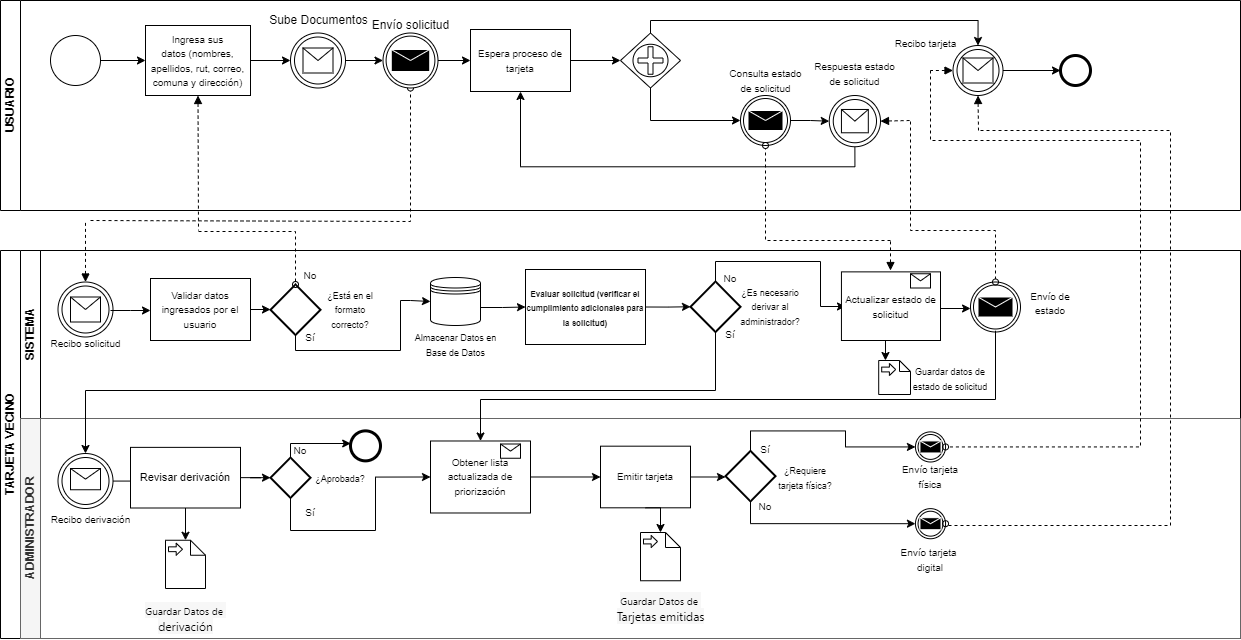
En el contexto del proyecto, se requiere realizar un software capacitado para generar y administrar las Tarjetas Vecino de las diversas comunas de Chile, con una previa solicitud y evaluación de requisitos por cada uno de los usuarios, el software proporcionará un catálogo de beneficios y convenios para aquellos que cuenten con una Tarjeta Vecino válida dependiendo de su comuna de residencia.

El usuario podrá ingresar sus datos y enviar una solicitud para acceder a una Tarjeta Vecino, además de ver el estado de su solicitud. El administrador podrá ver estas solicitudes y revisar aquellas que requieran una mayor inspección; el software se encontrará capacitado para validar cada una de las distintas restricciones según la información entregada por el usuario y, en casos particulares, solicitar la ayuda del administrador para realizar estas validaciones. Una vez aceptada la solicitud, el software se encargará de notificar al usuario que la emisión de su Tarjeta Vecino se encuentra autorizada para emisión, mientras que el administrador podrá generar dicha tarjeta en formato digital y/o solicitar su impresión en físico (dependiendo de la elección del usuario).

Se buscará priorizar las solicitudes de emisión de Tarjetas Vecino a aquellos usuarios que soliciten el formato físico de la tarjeta y se encuentren dentro de la población prioritaria, esta incluye: adultos mayores, personas con discapacidad y personas gestantes.

Adicionalmente, el software deberá almacenar las Tarjetas Vecino emitidas, las comunas a las que estas pertenecen y su periodo de vigencia, esto con el fin de llevar un registro de las mismas y poder notificar de manera oportuna a los usuarios de los periodos cercanos a la expiración de su tarjeta, con la finalidad de que pueda renovarla.

Finalmente, el software deberá contener un catálogo de beneficios y convenios autorizados para que los poseedores de las Tarjetas Vecinos estén en conocimiento de ellos, y, puedan acceder los beneficios disponibles en su comuna de residencia.



*Ilustración X: BPMN Proceso general de Tarjeta Vecino*

## Objetivo del software: General y específicos

**Objetivo General:**

Desarrollar un software que permita a los habitantes de las diversas comunas del país revisar, solicitar y gestionar su Tarjeta Vecino de manera efectiva y eficiente, optimizando el acceso a los beneficios locales, una mayor difusión de estos beneficios y promoviendo la participación ciudadana.

**Objetivos Específicos:**

* **Facilitar el acceso a Beneficios Locales**

El software buscará garantizar que los habitantes puedan consultar de manera sencilla y rápida los beneficios disponibles dependiendo de la comuna en la que se encuentren y su grupo etario, promoviendo la utilización de la Tarjeta Vecino como una herramienta efectiva para acceder a descuentos y servicios locales.

* **Automatizar el proceso de solicitud y gestión de Tarjeta Vecino**

Implementar un sistema de registro y solicitud en línea que permita a los usuarios postular a la Tarjeta Vecino de manera eficiente y a los administradores gestionar dichas solicitudes de manera efectiva y centralizada. Además, se buscará garantizar que los usuarios sean notificados de manera oportuna sobre el estado de solicitud y vigencia de tarjetas

* **Optimizar proceso de emisión de Tarjeta Vecino**

Mejorar la experiencia del usuario a través del diseño e implementación de un flujo de trabajo eficiente para los administradores que permita la emisión de Tarjetas Vecino digitales de manera ágil y precisa; esto logrado a través de la generación de las tarjetas con la información del usuario solicitante (nombre, RUT, comuna, QR de validez y fecha de expiración) y una oportuna notificación y seguimiento del estado de la tarjeta.

## Límites

Dentro del sistema, no se realizan las funciones de mostrar horario del trámite, listar lista de sugerencias, categorizar tipo de trámite, generar buscador para el tipo de trámite, sino más bien todo estará relacionado con el trámite específico señalado para brindar una cantidad de procesos relevantes o prioritarios con el objetivo de realizarlo de manera satisfactoria y eficiente.

## Requerimientos Funcionales

A continuación, se presentan las funcionalidades que tendrá el sistema en ambiente web y móvil que se busca desarrollar.

*Tabla 1: Requerimientos funcionales.*

| **ID\_RF** | **Nombre** | **Descripción** |
| --- | --- | --- |
| 1 | Solicitud de la Tarjeta Vecino | El SW debe permitir el **ingreso de datos del usuario al sistema; nombres, apellido paterno, apellido materno, rut, correo, comuna y dirección, al igual que la subida de documentos necesarios siendo estos el carnet de identidad con documento de residencia, y documentos de discapacidad de ser necesario.** También si necesita la tarjeta físicamente o sólo digitalmente. Esta información será almacenada mediante la subida de archivos a la base de datos, para luego validarla posteriormente.  Las restricciones en el momento que el usuario entregue su información al sistema son:   * Nombres, apellido paterno y apellido materno en casillas correspondientes. * Rut real y entregado en formato esperado (xx.xxx.xxx-x). * Correo en formato correcto ([xxxxx@xxx.xx](mailto:xxxxx@xxx.xx)). * Dirección real (Verifica con Google Maps). * Formato de los documentos deben ser en PDF por separado. * El usuario no puede enviar la solicitud, si no están todos los campos de información y documentos rellenados como correspondan. * El usuario no puede eliminar ni modificar su solicitud una vez enviada, solo hasta que esté completa su validación y respuesta respectiva.. * El usuario solo puede enviar una solicitud a la vez.   Posteriormente, el usuario deberá esperar a la evaluación de su solicitud. |
| 2 | Evaluación de la solicitud | El SW debe permitir la **evaluación de las distintas solicitudes a una Tarjeta Vecino enviadas por los usuarios, evaluando a través de una serie de restricciones los datos entregados de forma automática.** Estas restricciones son:   * Rut real y entregado en formato esperable (xx.xxx.xxx-x) * Nombres y apellidos deben corresponder al rut entregado. * Debe ser mayor de edad. * Rut corresponde al carnet. * El carnet debe estar en vigencia. * Los documentos de residencia y discapacidad deben ser legítimos y acorde a la persona que postula a la Tarjeta Vecino. * Documento de residencia en la comuna. Esta comuna debe coincidir con la Tarjeta Vecino solicitada.   Dependiendo de los resultados de las validaciones, se entregará como estado final si fue **Aceptada, Rechazada, o Derivada.** Al ser **Rechazada**, se debe comunicar al usuario con la razón del por qué se tomó dicha decisión. Al ser **Derivada**, se enviará al administrador para que evalúe con mayor detalle la situación. Las tarjetas aceptadas, rechazadas, y derivadas, serán almacenadas en una lista a disposición del administrador. |
| 3 | Emisión de la Tarjeta Vecino | El SW **debe permitir al administrador generar un listado con orden de prioridad de emisión de Tarjetas Vecino, además de emitirlas de forma digital y enviarlas a emitir físicamente de ser necesario, previamente comprobando que sean solicitudes evaluadas y aceptadas.** Estas Tarjetas Vecino serán emitidas con una plantilla preestablecida, adaptando la tarjeta según la información del usuario (Nombre, comuna…), además de dar a conocer al usuario mediante un correo que su Tarjeta Vecino fue emitida con éxito, mostrando la fecha de vencimiento y el estado de la tarjeta (activa, bloqueada ó vencida).  Las restricciones son:   * Nombres, apellido paterno y apellido materno. - Rut real y entregado en formato esperado (xx.xxx.xxx-x). * Fecha de vencimiento en formato correcto (xx/xx). * Priorizar la emisión a adultos mayores y embarazadas.   Posteriormente, el usuario podrá ver su Tarjeta Vecino en la plataforma correspondiente con los datos anteriormente mencionados, dando la opción de consultar el estado de su Tarjeta Vecino. |
| 4 | Generación y actualización de beneficios y convenios Tarjeta Vecino | El SW debe **dar a conocer al usuario los convenios y descuentos a los cuales éste puede acceder al poseer una tarjeta vecino** y **permitir que los administradores puedan modificar/administrar los convenios de acuerdo a su municipio**.  Las restricciones para que el usuario pueda acceder a esta información son:   * Rut real y entregado en formato esperado (xx.xxx.xxx-x) * Dirección real (verificada a través de Google Maps).   Posteriormente, el usuario podrá ver la lista de beneficios y convenios a acceder, y el administrador podrá guardar los cambios realizados. |

## Requerimientos No Funcionales

A continuación, se presentan las funcionalidades que tendrá el sistema en ambiente web y móvil que se busca desarrollar.

*Tabla 2: Requerimientos no funcionales.*

| **ID\_RNF** | **Nombre** | **Descripción** |
| --- | --- | --- |
| 1 | Plantilla de la Tarjeta Vecino | *Cada Tarjeta Vecino debe seguir un formato digital preestablecido, teniendo como únicas modificaciones los datos del usuario que la posee, la comuna autorizada, su período de vigencia y QR.* |
| 2 | Estética y diseño. | *La interfaz de usuario debe seguir una paleta de colores predefinida que cumpla con la identidad visual de la organización, garantizando una experiencia visual atractiva y coherente.* |
| 3 | Disponibilidad. | *El sistema debe estar disponible 24/7 con un tiempo de inactividad mínimo para garantizar el acceso de los usuarios en cualquier momento.* |
| 4 | Privacidad de datos. | *Se debe cumplir con las regulaciones de privacidad de datos, incluyendo la protección de información y el consentimiento del usuario.* |
| 5 | Rendimiento. | *El sistema debe ser capaz de proporcionar un rendimiento óptimo en términos de velocidad de carga y respuesta.* |
| 6 | Cumplimiento normativo. | *El software debe cumplir con las regulaciones y leyes aplicables en Chile, especialmente en lo relacionado con la emisión de la tarjeta vecino* |
| 7 | Mantención. | *Debe ser fácil de mantener y actualizar con procedimientos simples.* |

## REQUERIMIENTOS de Interfaces externas de entrada

*Tabla 3: Interfaces externas de entrada*

| **ID\_E** | **Nombre del ítem** | **Detalles de datos contenidos en el ítem** |
| --- | --- | --- |
| 1 | Datos del usuario. | *Nombre, Rut, Correo, dirección, comuna.* |
| 2 | Documentos del usuario. | *Fotos del carnet, certificado de residencia, y certificado de discapacidad.* |
| 3 | Solicitud de tarjeta. | *Datos de solicitud, como la comuna a la cual pertenece y preferencia de formato (digital o físico).* |

## REQUERIMIENTOS de Interfaces externas de salida

*Tabla 4: Interfaces externas de salida.*

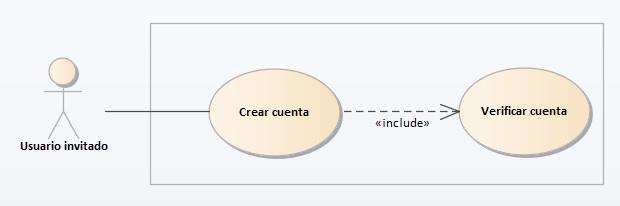
| **ID\_S** | **Nombre del ítem** | **Detalles de datos contenidos en ítem** | **Medio de salida** |
| --- | --- | --- | --- |
| 1 | Tarjeta Vecino Digital. | *Tarjeta de Vecino emitida.* | *Digital.* |
| 2 | Estado de Solicitud. | *Notificación del estado de la solicitud (pendiente, aprobada o rechazada).* | *Correo electrónico o notificación en la aplicación.* |
| 3 | Recordatorio de Renovación. | *Notificación a los usuarios sobre la proximidad de la expiración de su tarjeta.* | *Correo electrónico o notificación en la aplicación.* |
| 4 | Catálogo de Beneficios y Convenios | *Listado de beneficios y convenios disponibles en la comuna.* | *Interfaz de la aplicación o sitio web.* |
| 5 | Tarjeta Vecino Física | *Tarjeta Vecino emitida* | *Impresión física (en caso de solicitarse)* |

# ANÁLISIS

## Modelo de casos de uso

### Diagramas de casos de uso

A continuación, se presentan los diagramas de casos de uso del sistema, en donde se puede ver la interacción de los distintos actores, con los diferentes casos de uso, y de que caso de uso puede realizar cada actor.



*Ilustración 1: Diagrama de casos de uso 1*

### Actores-Roles

*Tabla 5: Actor "Usuario invitado"*

| **Actor Roles** | |
| --- | --- |
| **Descripción** |  |
| **Nivel de conocimiento requerido** |  |
| **Funciones a las que tiene acceso** |  |

### Especificación de casos de uso

A continuación, Tablas 6-n, se presenta una especificación de cómo funcionarán los casos de uso presentados y en qué consta la interacción de los actores con el sistema.

*Tabla 6: Caso de uso "* ***NNNNN<CU001>****".*

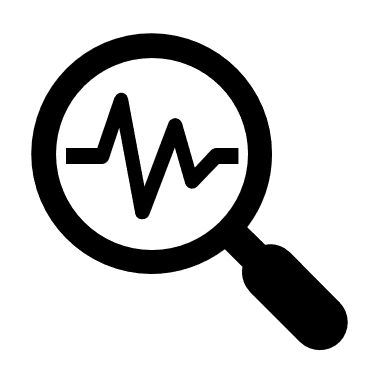
| **Precondiciones:** Ninguna. | | | |
| --- | --- | --- | --- |
| **Flujo de Eventos Básicos:** | | | |
| **Actor1** | **Actor2** | **Actor n** | **Sistema** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| **Flujo de Eventos Alternativos 1:** | |  |  |
|  |  |  |  |
|  |  |  |  |
| **Flujo de Eventos Alternativos 2:** | |  |  |
|  |  |  |  |
|  |  |  |  |
| **Flujo de Eventos Alternativos m:** | |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| **Post-Condiciones:** | |  |  |

# DISEÑo

## Modelo de datos

A continuación, se presenta el Modelo de datos. Incluya el json y un esquema con las colecciones.

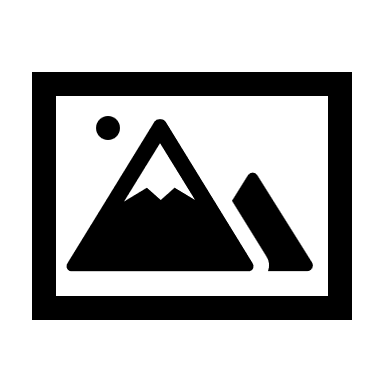
**Modelo Entidad Relación**



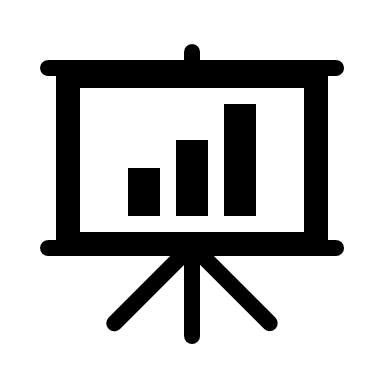
*Ilustración 2: Modelo entidad relación (MER).*

## Diseño de interfaz y navegación

A continuación, se presentan los diseños de interfaz de forma **general** de las principales páginas de la aplicación web y móvil.



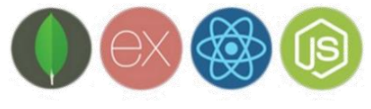
*Ilustración 3: Mockup "Página de inicio - web".*



*Ilustración 4: Mockup "Página de inicio - móvil".*

# Arquitectura de software

En el apartado de los requisitos técnicos para el desarrollo del software mencionado anteriormente, se tiene como parte de los requerimientos técnicos del proyecto la utilización de las tecnologías contenidas en el **Stack MERN.**

****

*Ilustración x: Logotipos de tecnologías MERN*

1. **Backend con Node.js y Express.js:**

* Node.js es un tipo de entorno de desarrollo basado en JavaScript, permitiendo que aplicaciones se puedan construir a la par del servidor del proyecto. Express.js es un marco de Node.js enfocado en la creación de API y aplicaciones web, ambos serán la base para llevar a cabo el desarrollo del servidor web y la API REST del software.

1. **Base de datos con MongoDB:**

* MongoDB es una base de datos NOSQL, utilizado para almacenar datos en formato JSON. Presenta una versión gratuita la cual mantiene una alta velocidad en cuanto a operaciones de Lectura y Escritura, transformándose en una alternativa ideal para almacenar la información requerida por el software.

1. **Frontend con React:**

* React es una biblioteca/framework de JavaScript con el potencial de poder construir interfaces de usuario interactivas y fluidas, dando al usuario una experiencia agradable a la hora de acceder al software.

1. **Visual Studio Code como IDE:**

* Visual Studio Code es un editor de código fuente multiplataforma con soporte para depuración y control integrado para git; gracias a su capacidad para instalar múltiples extensiones, Visual Studio Code es un IDE con gran potencial para el desarrollo de software.

1. **GitHub como Repositorio Externo oficial:**

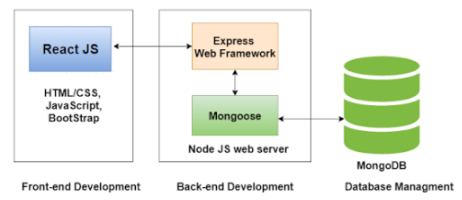
* GitHub es una plataforma en línea utilizada para alojar y gestionar proyectos de desarrollo de software a través del sistema de control de versiones Git, lo que lo vuelve una plataforma ideal para el desarrollo del proyecto.

1. **Versión de GIT de la FACE como Repositorio oficial:**

* El servidor/repositorio perteneciente a la Facultad de Ciencias Empresariales será el entorno de desarrollo en el cual el software deberá ser almacenado y ejecutado.

1. **Desarrollo orientado a multiplataforma (Windows, Linux):**

* El software será construido bajo sistemas operativos de windows, más específicamente en sus versiones 10 y 11, y futuramente se desarrollarán pruebas de desempeño bajo el sistema linux en el cual se encuentra montado el servidor de la facultad, Ubuntu 22.04.



*Ilustración x: https://blog.facialix.com/desarrollo-mern-stac*

# VERIFICACIÓN

## Verificación requisitos (Interfaces de Entrada)

*Todos los datos de entrada son ingresados o procesados en este requisito funcional; ), solo complete en caso que todos los datos se encuentren I (ingresados) o P (procesados) en el requisito.*

|  | Interfaces externas de entrada | | | | |
| --- | --- | --- | --- | --- | --- |
|  | ID\_Entrada | ID\_Entrada | ID\_Entrada |  | ID\_Entrada n |
| Rf nombre |  |  |  |  |  |
| Rf nombre |  |  |  |  |  |
| Rf nombre i |  |  |  |  |  |

## Verificación requisitos (Interfaces de Salida)

*Todos los datos de la salida PROVIENEN de una interfaz de entrada o del procesamiento de un requerimiento funcional.*

|  | Interfaces externas de salida | | | | |
| --- | --- | --- | --- | --- | --- |
|  | ID\_salida | ID\_salida | ID\_salida |  | ID\_salida j |
| Rf nombre |  |  |  |  |  |
| Rf nombre |  |  |  |  |  |
| Rf nombre |  |  |  |  |  |
| Rf nombre i |  |  |  |  |  |
| ID\_Entrada |  |  |  |  |  |
| ID\_Entrada |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| ID\_Entrada |  |  |  |  |  |
| ID\_Entrada n |  |  |  |  |  |

## Verificación requisitos CU

*Los requisitos son implementados a través de muchos casos de uso, marque con x si el cu contribuye a satisfacer el requisito.*

|  |  | | | | |
| --- | --- | --- | --- | --- | --- |
|  | Rf nombre | Rf nombre |  |  | Rf nombre i |
| CU nombre |  |  |  |  |  |
|  |  |  |  |  |  |
| CU nombre k |  |  |  |  |  |

# BIBLIOGRAFÍA

* Documentación oficial Laravel 6 <<https://laravel.com/docs/6.x/installation>>
* Documentación oficial Vue.js <<https://vuejs.org/v2/guide/>>
* Documentación oficial Ionic <<https://ionicframework.com/docs>>
* Definiciones, “Modelo de negocio canvas” <<https://economipedia.com/definiciones/modelo-canvas.html>>
* Documentación oficial Boostrap Vue <<https://bootstrap-vue.org/>>
* Manejo del estado de la aplicación en Angular <<https://medium.com/angular-chile/manejo-de-estado-con-ngxs-en-angular-e66e11198a0>>
* Manejo del estado de la aplicación en Vue.js <<https://vuex.vuejs.org/guide/>>
* Estilo de Arquitectura API REST <<https://juanda.gitbooks.io/webapps/content/api/arquitectura-api-rest.html>>
* Límites geográficos de los países en el mundo <<https://gist.github.com/graydon/11198540>>
* Gráficos en Vue.js < <https://vue-chartjs.org/guide/#introduction>>
* Leaflet en Vue.js < <https://vue2-leaflet.netlify.app/quickstart/>>

# ANEXO: DEL PROYECTO

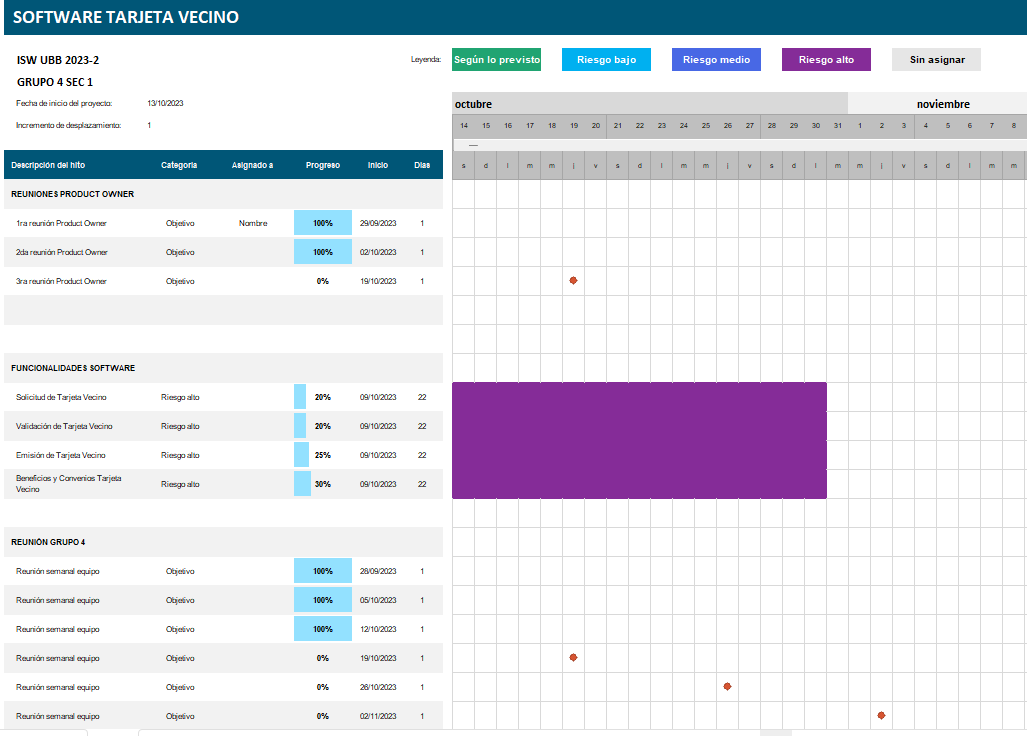
## Objetivos del Proyecto, general y específicos

**Objetivo general:** Determinar las características y requerimientos necesarios para el desarrollo del software basado en los requerimientos de la Tarjeta Vecino, registrando los avances, mejoras, y tecnologías empleadas con el fin de tener una mejor observación y organización con respecto al proceso de desarrollo.

**Objetivo específico**: Documentar todo el proceso de desarrollo, junto a sus restricciones y requerimientos con el fin de comprender con mayor objetividad la forma de creación del software.

Visualizar las formas de trabajo involucradas en base a modelos, junto al funcionamiento, manejo de datos (Entradas y salidas), y los roles establecidos, utilizando técnicas de modelados (BPMN, Diagramas de Casos de Uso).

## Programación de actividades



## Estimación de tamaño

Esta estimación se realizó utilizando la técnica basada en Casos de Uso.

| **FACTOR DE PESO DE LOS ACTORES SIN AJUSTAR** | | | | |
| --- | --- | --- | --- | --- |
| **Tipo de Actor** | **Descripción** | **Factor de Peso** | **N° Actores** | **Subtotal (UAW)** |
| Simple | Otro Sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Aplication Programming Interface) | 1 |  |  |
| Medio | Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto | 2 |  |  |
| Complejo | Una persona que interactúa con el sistema mediante una interfaz | 3 |  |  |
| **Total Actores (UAW)** | | |  |  |

| **FACTOR DE PESO DE CASOS DE USO SIN AJUSTAR** | | | | |
| --- | --- | --- | --- | --- |
| **Tipo de CU** | **Descripción** | **Factor de Peso** | **N° Casos de Uso** | **Subtotal (UUCW)** |
| Simple | El caso de uso contiene de 1 a 3 transacciones | 5 |  |  |
| Medio | El caso de uso contiene de 4 a 7 transacciones | 10 |  |  |
| Complejo | El caso de uso contiene más de 7 transacciones | 15 |  |  |
| **Total Casos de Uso (UUCW)** | | |  |  |

| **Puntos de Caso de Uso Sin Ajustar (UUCP)** |  |
| --- | --- |

| **FACTORES DE COMPLEJIDAD TECNICA (TCF)** | | | | |
| --- | --- | --- | --- | --- |
| **Factor** | **Descripción** | **Factor de Peso** | **Valor Asignado (0-5)** | **Subtotal** |
| T1 | Sistema distribuido | 2 |  |  |
| T2 | Objetivos de performance o tiempo de respuesta | 1 |  |  |
| T3 | Eficiencia del usuario | 1 |  |  |
| T4 | Procesamiento interno complejo | 1 |  |  |
| T5 | El código debe ser reutilizable | 1 |  |  |
| T6 | Facilidad de instalación | 0,5 |  |  |
| T7 | Facilidad de uso | 0,5 |  |  |
| T8 | Portabilidad | 2 |  |  |
| T9 | Facilidad de cambio | 1 |  |  |
| T10 | Concurrencia | 1 |  |  |
| T11 | Incluye objetivos especiales de seguridad | 1 |  |  |
| T12 | Provee accesos directos a terceras partes | 1 |  |  |
| T13 | Se requieren facilidades especiales de entrenamiento a usuarios | 1 |  |  |
| **Total Factor** | | | |  |

| **FACTORES DE AMBIENTE (EF)** | | | | |
| --- | --- | --- | --- | --- |
| **Factor** | **Descripción** | **Factor de Peso** | **Valor Asignado**  **(0-5)** | **Sub total** |
| E1 | Familiaridad con el modelo del proyecto usado | 1,5 |  |  |
| E2 | Experiencia en la aplicación | 0,5 |  |  |
| E3 | Experiencia en orientación a objetos | 1 |  |  |
| E4 | Capacidad de analista líder | 0,5 |  |  |
| E5 | Motivación | 1 |  |  |
| E6 | Estabilidad de los requerimientos | 2 |  |  |
| E7 | Personal media jornada | -1 |  |  |
| E8 | Dificultad del lenguaje de programación | -1 |  |  |
| **Total Factor Ambiente** | | | |  |

| **ESTIMANDO LOS UCPs** | | |
| --- | --- | --- |
| **Parámetro** | **Descripción** | **Valor Calculado** |
| UUCP | Puntos casos de uso sin ajustar |  |
| TCF | Factor de complejidad técnica |  |
| ECF O EF | Factor de ambiente |  |
| **UCP** | Puntos casos de uso ajustados |  |
| **LOE** | Nivel de esfuerzo |  |
| **Factor de Capacidad (min 10, avg 20, max 40) hours/UCP** | |  |

| **Esfuerzo Total = UCP x LOE** | AAAA horas hombre |
| --- | --- |

# Anexo: ambiente de ingeniería

## Método de Desarrollo

La metodología a emplear en el desarrollo de este software, será la Incremental (Iterativo). Se ha decidido optar a este tipo de metodología por su desarrollo escalable y optimizado cuando constan de varias etapas, adicionalmente que ofrece una mayor versatilidad a la hora de generar alteraciones o cambios en las funcionalidades principales de nuestro software. Aún así, se reconoce la dificultad que puede generar fragmentar en distintas partes el desarrollo total del software, por lo que se requiere suma precisión en la etapa de análisis principal, para repartir las distintas tareas a completar en tiempos alcanzables con los recursos a tener disponibles.

## Técnicas

## Herramientas y estándares

# Anexo: Esfuerzo real después de terminado el proyecto

*Tabla 8: Resumen esfuerzo requerido.*

| **Fases/Actividades principales** | **Horas hombre** |
| --- | --- |
| Definición del Proyecto |  |
|  |  |
|  |  |
| Desarrollo de Aplicación Web |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Pruebas de Sistema |  |
| Documentación |  |
| **Total** |  |

# ANEXO: Especificación de las Pruebas

*Tabla 9: Detalle de prueba "****Editar institución social****".* ***ejemplo***

| **Id** | **Característica a probar** | **Nombre** | **Salida esperada** | **Salida obtenida** | **Éxito/Fracaso** |
| --- | --- | --- | --- | --- | --- |
| **Teléfono** |
| **Dirección** |
| 1 | Funcionalidad |  | Nombre requerido | Nombre requerido | Éxito |
| 6006009000 |
| Calle X, #999, Concepción |
| 2 | Funcionalidad | Institución XYZ | Teléfono requerido | Teléfono requerido | Éxito |
|  |
| Calle X, #999, Concepción |
| 3 | Funcionalidad | Institución XYZ | Dirección requerida | Dirección requerida | Éxito |
| 6006009000 |
|  |
| 4 | Funcionalidad | Institución XYZ | Datos editados correctamente | Datos editados correctamente | Éxito |
| 6006009000 |
| Calle X, #999, Concepción |

# ANEXO: Diccionario de datos del modelo de datos

Objetos y atributos de datos

# ANEXO: Entrevistas

1ra entrevista con Product Owner (29 de Septiembre 2023 | Llamada vía Discord, 17:00hrs)

* Se inicia la reunión con el equipo completo de desarrolladores del proyecto, presentándose al Product Owner Vicente Aguilera.
* Luego de presentarse y realizar small talk entre todos, procedemos a conocer el contexto general del proyecto Tarjeta Vecino.
* A medida que se va escuchando al Product Owner explicar en lo que se debe basar el proyecto, se va tomando notas de las distintas funcionalidades que se pueden desarrollar.
* Se conoce que se requiere hacer funcionalidades para: Postular y solicitar a Tarjeta Vecino, evaluar la solicitud, emitir la Tarjeta Vecino, y enlistar los beneficios a los poseedores de la Tarjeta Vecino, correspondiendo a las comunas de cada usuario.
* Se consultan los detalles y procesos a tener en cuenta, como los datos requeridos para las solicitudes, tiempos de vigencia, verificaciones de descuentos, notificaciones con los avances de la solicitud, entre otros.
* Finalizando la 1era entrevista, se acuerda realizar una 2da entrevista para revisar los requisitos funcionales redactados.

2da entrevista con Product Owner (02 de octubre 2023 | Llamada vía Discord, 18:30hrs)

* Se inicia la reunión con el equipo completo de desarrolladores del proyecto, presentándose al Product Owner Vicente Aguilera.
* Se le presenta a Vicente los requisitos funcionales redactados.
* Verificamos detalles de validación, los formatos deseados en los que se debe entregar, y cómo este debe trabajar a través del código.
* Confirmando que se encuentran acertados y acordes a los deseos del cliente, se termina la reunión.

# ANEXO: Cuestionarios

A través de un formulario creado por los mismos estudiantes se realizaron las siguientes preguntas a un total de 30 personas:

* Seleccione el rango etario en el que se encuentra. (18 años - 30 años, 31 años - 50 años, 51 años o más).
* ¿Tiene conocimiento de los trámites que puede realizar en su Municipalidad? (Sí, No).
* Indique aquellos trámites que ha realizado en su Municipalidad.
* ¿Con qué frecuencia ha realizado trámites en su Municipalidad? (Casi nunca, Ocasionalmente, Frecuentemente, Muy frecuentemente)
* ¿Qué documentos le solicitan para realizar su trámite?
* ¿Ha tenido inconvenientes por los horarios de atención de la Municipalidad? (Sí, No).
* De presentar algún otro tipo de inconveniente, indique cuáles.
* ¿Considera que el tiempo que se demora en la Municipalidad es razonable? (Sí, No).
* ¿Qué trámites son los que más ha tardado en realizar? (Puede mencionar más de uno).
* ¿Cuáles son los pasos a seguir para realizar dicho trámite?
* ¿Considera usted que la información entregada por la Municipalidad es clara? ¿Por qué?
* ¿Existe una plataforma online de su Municipalidad? (Sí, No, Desconozco si existe).
* En los trámites que ha realizado en la Municipalidad, ¿Está disponible la opción de hacerlo online? (Sí tiene la opción online, No tiene la opción online, debe hacerse presencialmente).
* ¿Estaría interesado en realizar los trámites online? (Sí, No).
* ¿Ha recibido actualizaciones o notificaciones sobre el estado de su trámite? (Sí, No).
* ¿Cuán satisfecho se siente con el servicio que le entregan? (Muy insatisfecho, Insatisfecho, Regular, Satisfecho, Muy Satisfecho).
* De tener una calificación del 3 hacia abajo respeto a la satisfacción, ¿Por qué razones se siente así?
* ¿Algún comentario que agregar sobre su Municipalidad?

# ANEXO: otros Documentos recopilados