



UNIVERSIDAD NACIONAL DE COSTA RICA

Sede Regional Brunca

**“PIMI: PLATAFORMA INTEGRAL PARA EL MANEJO DE INSPECCIONES PARA
LA MUNICIPALIDAD DE PÉREZ ZELEDÓN”**

Para optar al grado de Licenciatura en Informática con énfasis en

Sistemas de Información

ING. LUIS BARRANTES MORA.

ING. ERICK FLORES SOLÍS.

Campus Pérez Zeledón

2019

HOJA DE APROBACIÓN

Trabajo final de graduación presentado el 27 de agosto de 2019, en la ciudad de San Isidro de El General, Pérez Zeledón, San José, Costa Rica, como requisito para optar al grado de Licenciatura en Informática con énfasis en Sistemas de Información.

El trabajo presentado se da por aprobado por los miembros del Tribunal Examinador

Miembros del Tribunal Examinador

M.A. Yalile Jiménez Olivares
Vicedecana, Sede Regional Brunca

M.I. Joseph Montenegro Bonilla
Director Académico, Campus Pérez Zeledón

Msc. Pablo Chaves Murillo
Profesor tutor

Msc. Rubén Mora Vargas
Lector interno

Msc. Carlos Carranza Blanco
Lector interno

DEDICATORIA

Dedicado a mi familia, la cual siempre ha estado a mi lado para todo tipo de situación adversa que se haya presentado. A mis padres, Lesmes y Neydi, quienes han sido un apoyo incondicional durante toda esta trayectoria y que, sin ellos, hubiese sido imposible llegar hasta estas instancias. A mis hermanos, Pablo y Claudia, que han sido quienes siempre han estado para ofrecer consejos y motivación. Finalmente, a mis sobrinos María Paula, Joz Daniel y Zoé, los cuales han sido el motor que me ha motivado para seguir adelante y desear superarme día con día.

Luis Barrantes Mora

Dedicado a mis padres Ana y Roger por ser pilar fundamental en todo lo que soy, en toda mi educación, tanto académica como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo, todo este trabajo ha sido posible gracias a ellos. A mi compañera de vida Joselyn Elizondo quien me acompañó y fortaleció en cada momento con una gran sonrisa. A mis hermanos y a Dios, quien me fortaleció e iluminó y por haber puesto en mi camino a aquellas personas que han sido mi soporte y compañía durante todo el periodo de estudio.

Erick Flores Solís

CARTA DEL FILÓLOGO

Los suscritos, Elena Redondo Camacho, mayor, casada, filóloga, cédula de identidad número 3 0447 0799 y Daniel González Monge, mayor, casado, filólogo, cédula de identidad número 1 1345 0416, vecinos de Quebradilla de Cartago, en calidad de filólogos, revisamos y corregimos el trabajo final de graduación que se titula: *PIMI: Plataforma Integral para el Manejo de Inspecciones para la Municipalidad de Pérez Zeledón*, sustentado por Luis Barrantes Mora y Erick Flores Solís.

Hacemos constar que se corrigieron todos los aspectos de forma, redacción, estilo y otros vicios del lenguaje que se pudieron trasladar al texto.

Esperamos que nuestra participación satisfaga los requerimientos de la Universidad Nacional.

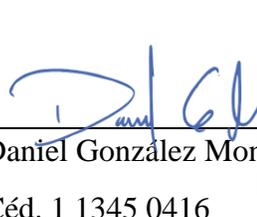


Elena Redondo Camacho

Céd. 3 0447 0799

Bachiller en Filología Española

Carné ACFIL 0247



Daniel González Monge

Céd. 1 1345 0416

Bachiller en Filología Española

Carné ACFIL 0245

TABLA DE CONTENIDO

Hoja de Aprobación.....	i
Dedicatoria.....	ii
Carta del Filólogo	iii
Tabla de Contenido.....	iv
Índice de Figuras	ix
Índice de Tablas.....	x
Resumen Ejecutivo.....	xi
Capítulo 1. Introducción.....	1
1.1. Antecedentes.....	1
1.1.1. Parso.....	1
1.1.2. E-Park.....	2
1.1.3. Municipalidad de Escazú	3
1.2. Justificación	4
1.3. Objetivos del proyecto.....	6
1.3.1. Objetivo General.....	6
1.3.2. Objetivos Específicos.....	6
1.4. Metodología.....	7
1.4.1. Planificación del Proyecto.....	8

1.4.2. Diseño y Arquitectura	9
1.4.3. Desarrollo (<i>sprints</i>)	10
1.4.4. Pruebas de Usuario.....	10
1.4.5. Controles de SCRUM	11
1.4.6. Implementación y Cierre.....	12
Capítulo 2. Marco Teórico.....	13
2.1. Marco Referencial.....	13
2.1.1. Misión	13
2.1.2. Visión	13
2.1.3. Objetivos Organizacionales	14
2.1.4. Beneficios Esperados con la Plataforma	14
2.2. Marco conceptual.....	15
2.2.1. Conceptualización Inicial.....	15
2.2.2. Estructura Modular.....	18
2.3. Marco Metodológico.....	20
2.3.1. Levantamiento de Requerimientos.....	20
2.3.2. Recolección de Datos	22
2.3.3. Análisis.....	23
2.3.4. Estudio de Factibilidad.....	24
2.3.5. Factibilidad Operacional	24

2.3.6. Factibilidad Técnica	25
2.3.7. Factibilidad Económica.....	25
2.3.8. Modelaje de Datos.....	25
2.3.9. Desarrollo.....	25
2.4. Pruebas.....	26
2.4.1. Manual del Usuario.....	27
2.4.2. Manual Técnico.....	27
Capítulo 3. Procedimiento Metodológico.....	28
3.1. Levantamiento de requerimientos.....	28
3.1.1. Aplicación móvil para registro de infracciones de tránsito.....	28
3.1.2. Aplicación <i>web</i> para venta de tiempo de estacionamiento	31
3.1.3. Aplicación <i>web</i> de administración	32
3.2. Recolección de datos	33
3.3. Análisis	34
3.3.1. Aplicación para registro de infracciones: Inicio de sesión.....	34
3.3.2. Aplicación para registro de infracciones: Registrar infracción.....	35
3.3.3. Aplicación para registro de infracciones: Consultar vehículos.....	36
3.3.4. Aplicación para registro de infracciones: Sincronización de infracciones	37
3.3.5. Aplicación para registro de infracciones: Registro de dispositivos	37
3.3.6. Aplicación para registro de infracciones: Sincronización de parámetros Generales	

.....	38
3.3.7. Aplicación para venta de tiempo para estacionamiento: Compra de tiempo	39
3.3.8. Sistema de administración PIMI: Inicio de sesión.....	40
3.3.9. Sistema de administración PIMI: Registro de usuarios	41
3.3.10. Sistema de administración PIMI: Edición de usuarios	42
3.3.11. Sistema de administración PIMI: Listado y edición de infracciones	42
3.3.12. Sistema de administración PIMI: Generación de archivo para SICSOA.....	43
3.3.13. Sistema de administración PIMI: Registro de vehículos para MPZ Pass	44
3.3.14. Sistema de administración PIMI: Localización de inspectores.....	45
3.4. Estudio de Factibilidad	46
3.4.1. Factibilidad Técnica.....	46
3.4.2. Factibilidad Operativa.....	49
3.4.3. Factibilidad Económica.....	51
3.5. Modelaje de Datos	52
3.6. Desarrollo.....	52
Capítulo 4. Conclusiones.....	56
4.1. Agilización de procesos	56
4.2. Accesibilidad	56
4.3. Seguridad	56
4.4. Impacto ambiental.....	57

4.5. Innovación	57
4.6. Aporte UNA a la sociedad	58
4.7. Recomendaciones de los Autores	58
Capítulo 5. Bibliografía y Glosario	60
5.1. Bibliografía	60
5.2. Glosario de Términos.....	62
Capítulo 6. Anexos	65
6.1. Anexo 1	65
6.2. Anexo 2.....	65
6.3. Anexo 3.....	66
6.4. Anexo 4.....	67

ÍNDICE DE FIGURAS

Figura 1. Arquitectura general de la plataforma.....	17
Figura 2. Diagrama de flujo para el inicio de sesión.....	35
Figura 3. Diagrama de flujo para el registro de infracción.....	36
Figura 4. Diagrama de flujo para la consulta de vehículos.....	36
Figura 5. Diagrama de flujo para la sincronización de infracciones	37
Figura 6. Diagrama de flujo para el registro de dispositivos.....	38
Figura 7. Diagrama de flujo para la sincronización de parámetros generales.....	39
Figura 8. Diagrama de flujo para la compra de tiempo	40
Figura 9. Diagrama de flujo para el inicio de sesión	41
Figura 10. Diagrama de flujo para el registro de usuarios.....	42
Figura 11. Diagrama de flujo para la edición de usuarios	42
Figura 12. Diagrama de flujo para el listado y edición de infracciones	43
Figura 13. Diagrama de flujo para la generación de archivo para SICSOA	44
Figura 14. Diagrama de flujo para el registro de vehículos para MPZ Pass	45
Figura 15. Diagrama de flujo para la localización geográfica de inspectores	45

ÍNDICE DE TABLAS

Tabla 1. Especificación de hardware y software de la institución.....	46
Tabla 2. Especificación de herramientas de desarrollo	47
Tabla 3. Costo de recurso humano	51
Tabla 4. Comparativa de gastos.....	52

RESUMEN EJECUTIVO

El proyecto de Plataforma Integral para el Manejo de Inspecciones para la Municipalidad de Pérez Zeledón (PIMI), consiste en una plataforma *web* y móvil que brinde una solución innovadora a diferentes procesos que se realizan de manera manual en la institución, específicamente en los departamentos de Licencias y Patentes comerciales, Seguridad Ciudadana y Planificación Territorial; el *software* citado propone su funcionamiento en cuatro diferentes ramas: registro de actas de inspección de patentes en locales comerciales, registro de multas por infracción a la ley de parquímetros, venta de boletas de estacionamiento por medio de la página *web* municipal y, por último, registro de actas de las inspecciones realizadas a las construcciones; esto aunado a una plataforma *web* global que permite la administración de estos y la generación de diferentes reportes.

La plataforma brinda la disponibilidad de cada una de las aplicaciones ante inconvenientes de comunicación por medio de mecanismos de almacenamiento local y sincronización que aseguran su funcionamiento de manera independiente al Internet.

A continuación, se detalla cada uno de los módulos que componen el proyecto:

Infracciones por estacionamiento

Elaboración del módulo *Infracciones de tránsito que incumplan la ley de parquímetros*, que permite a los usuarios (oficiales de parquímetros de la municipalidad) realizar las boletas de infracción a los vehículos que infrinjan la ley de Parquímetros a través de una plataforma móvil, específicamente en un dispositivo PDA (asistente digital personal). El ciclo de vida del módulo consistirá en:

- Identificación del inspector, por medio de un usuario y contraseña. Este proceso es gestionado por el servidor de datos de la municipalidad.
- Integración con el módulo *Venta de boletas de permiso de estacionamiento (2)*, por medio del servidor de la municipalidad, que permite la consulta de permisos de estacionamiento por vehículo al ingresar el número de placa del vehículo a consultar. Este proceso podrá realizarse por medio del escaneo de un código QR

que se le facilitará a las personas que deseen adquirirlo.

- Registro de infracción al vehículo al ingresar los datos del vehículo, específicamente: tipo de vehículo, placa, color, marca, artículo infringido, observaciones, fotografías, ubicación, fecha y hora.
- Integración con el servidor de la municipalidad para el registro en tiempo real de la infracción.
- Impresión de comprobante a la infracción realizada, por medio de una impresora portátil, vinculada vía Bluetooth al dispositivo inteligente; este comprobante mostrará los datos del vehículo, así como fecha y hora, además del monto de la infracción e información meramente de la municipalidad.
- Generación de reporte que muestre las infracciones realizadas en un rango de fechas establecidas por medio de parámetros.

Venta de boletas de permiso de estacionamiento

Implementación del módulo *Venta de tiempo para estacionamiento en zonas permitidas*, que le permite a los usuarios (ciudadanos) la compra de boletas de estacionamiento mediante la plataforma *web* de comercio electrónico de la Municipalidad, la cual se integrará con la plataforma móvil de *Infracciones por estacionamiento (I)*, lo que le permitirá al inspector consultar en tiempo real las boletas de permisos de estacionamiento registradas y con vigencia. La elaboración del módulo se detalla a continuación:

- El módulo permite a los ciudadanos la consulta de boletas de permiso de estacionamiento registradas que se encuentren en estado vigente. La plataforma requerirá la placa del vehículo, para posteriormente mostrar la fecha, hora y duración del permiso de estacionamiento consultado.
- El usuario podrá comprar tiempo de estacionamiento en zonas permitidas mediante este módulo.
- El módulo se integrará con la plataforma de comercio electrónico de la municipalidad, la cual consiste en un convenio con el Banco Costa Rica, lo que

permitirá brindarle al usuario una vía confiable, para que se realice la transacción.

- El módulo registra el tiempo de estacionamiento en el servidor de datos de la municipalidad, la cual se integra con el módulo de *Infracciones por estacionamiento (I)* para brindarle al inspector la veracidad de que el vehículo cuenta con un permiso de estacionamiento.
- El módulo generará un comprobante en formato PDF que muestre al usuario el movimiento realizado, específicamente con el número de transacción, placa del vehículo, fecha y hora del permiso y, por último, su duración.

Inspección de patentes

Se ha desarrollado una aplicación móvil, la cual permite la generación de actas digitales con la información recolectada por los inspectores de la municipalidad, mediante las inspecciones de patentes realizadas a los locales comerciales. Esta aplicación permite registrar la información recolectada en la base de datos de la Municipalidad, de manera que puedan procesarse por el departamento de Licencias y Patentes.

La aplicación, además, permite que el inspector de la municipalidad consulte si ya existe un acta de patente registrada con anterioridad, con el fin de evitar pérdida de tiempo al registrar información ya existente. Además, se debe tomar en cuenta la disponibilidad de conexión a la red, pues existen zonas alejadas en las cuales no hay forma de registrar los datos recopilados en tiempo real; por lo tanto, existe un mecanismo de almacenamiento local en el dispositivo móvil que se sincroniza con la base de datos principal una vez que el dispositivo cuente con cobertura telefónica y el inspector decida sincronizarla. El ciclo de vida con el que cuenta la aplicación es el siguiente:

- El inspector se dirige al lugar en el que realizará la inspección (visita rutinaria o por solicitud de un tercero).
- Una vez que el inspector se encuentra en el local comercial en el que realizará la inspección, procederá a revisar los detalles que corresponden a la patente del lugar.
- Posteriormente, se procede a seleccionar el tipo de acta que realizará (tasación,

clausura, etc.) y se dispone a llenar el formulario con los datos pertinentes.

- Cuando el usuario decide registrar el acta realizada, el sistema verificará si existe disponibilidad de conexión; en caso afirmativo, se establecerá la conexión con el API de la Municipalidad y se registrará inmediatamente; en caso contrario, el sistema registrará el acta de forma local esperando disponibilidad en la red para que la información sea sincronizada.

Inspección de permisos de construcción

Se desarrolló un módulo de *Inspección de permisos de construcción* mediante una plataforma móvil, que permite a los funcionarios el registro de la información recolectada durante las inspecciones realizadas para las construcciones de obras.

La aplicación le permite al inspector de la municipalidad consultar si existe una inspección realizada con anterioridad a la construcción, además de visualizar datos sobre las inspecciones de obras alojadas en el sistema.

Al igual que el módulo de inspecciones de patentes, se tomaron en cuenta los inconvenientes con la conexión a Internet, ya que existen muchas zonas rurales en las cuales la señal telefónica es muy débil y resulta imposible establecer una conexión directa con el API de la municipalidad, por lo tanto, se desarrolló un mecanismo de sincronización de datos que almacena el acta de forma local en caso de que el dispositivo detecte la imposibilidad de establecer la comunicación y cuando tenga conexión a Internet y el inspector así lo desee, se sincronicen los datos previamente recolectados.

En síntesis, el ciclo de vida de las inspecciones a construcciones es el siguiente:

- El inspector se dirige al lugar en el que realizará la inspección (visita rutinaria o por solicitud de un tercero).
- Una vez que el inspector se encuentra en lugar en el que se realizará la inspección, verificará los detalles que corresponden a los permisos necesarios para llevar a cabo la construcción.
- Una vez que se obtienen los detalles necesarios, se procede a crear un acta digital en la que se detalla toda la información recopilada.

- Cuando el usuario decide registrar el acta realizada, el sistema verificará si existe disponibilidad de conexión; en caso afirmativo, se establecerá la conexión con el API de la municipalidad y se registrará inmediatamente; en caso contrario, el sistema registrará el acta de forma local esperando disponibilidad en la red para que la información sea sincronizada.

Aplicación *web* para la administración de aplicaciones

Se ha desarrollado una aplicación que engloba la administración de las aplicaciones anteriormente descritas; en esta se puede controlar todo el funcionamiento que respecta a los usuarios que hacen uso de las aplicaciones, así como consultar toda la información que se recolecta de manera cotidiana a través de la plataforma.

Las funcionalidades de la aplicación descrita se detallan a continuación:

- Registro de usuarios con sus respectivos permisos, los cuales determinarán los accesos que posean en las diferentes aplicaciones.
- Registro de vehículos para el uso de la aplicación MPZ Pass, lo cual genera un código único a partir de los datos del vehículo y lo plasma en un código QR, que podrá consultar posteriormente el inspector de parquímetros.
- Monitoreo de localizaciones de los diferentes inspectores municipales, a partir de las localizaciones registradas por el dispositivo, las cuales se captan cada vez que se desplaza una cantidad de metros parametrizable.
- Consulta sobre las infracciones de tránsito, filtradas por inspector, vehículo y rango de fechas.
- Generación de diferentes reportes, por medio de las localizaciones registradas en cada acta de inspección o infracción de tránsito.
- Generación de archivo para el sistema SICSOA, el cual pertenece al Instituto Nacional de Seguros (INS), que solicita las boletas sin cancelar una vez al año, para posteriormente cargarse al marchamo de cada vehículo.
- Gestión de parámetros generales en cada aplicación, como los artículos a la ley

de parquímetros, colores de vehículos, marcas de vehículos, entre otros.

Capítulo 1. Introducción

1.1. Antecedentes

Con la carrera tecnológica que ha surgido en los últimos años se han creado nuevos conceptos ligados a esta, una de las ramas novedosas de la tecnología pertenece al término de ciudades inteligentes, que se define como aquella ciudad que hace uso de los avances tecnológicos para mejorar la calidad de vida de sus habitantes (El País, 2010).

En todo este ambiente tecnológico se encuentra la ciudad de San Isidro, cabecera del cantón de Pérez Zeledón, la cual necesita de avance tecnológico en su municipalidad en temas relacionados con trámites que actualmente se realizan de forma manual (en papel), como el caso del trabajo de campo (inspecciones). En distintos municipios de Costa Rica han entendido a la perfección esta necesidad, por lo que han implementado sistemas hechos por terceros, entre los cuales se encuentran Parso y E-Park (Setex), que se enfocan puntualmente en la gestión de parquímetros. A continuación, se hará una breve referencia de las características con las que cuentan estas aplicaciones:

1.1.1. Parso

En el ámbito nacional existe una empresa que brinda una plataforma electrónica semejante a la que se planteó al inicio de este proyecto (módulos que involucran parquímetros), la cual se denomina Parso; esta empresa fue fundada en el año 2013 y según su sitio web tiene como misión “Contribuir en convertir a Latinoamérica en un referente en el ámbito mundial en ciudades inteligentes, mediante el uso de herramientas tecnológicas para la gestión de estacionamientos” (s. p.).

Parso brinda un servicio muy completo con cuanto a la gestión de información sobre temas de parquímetros y brindan una aplicación móvil para los inspectores de parquímetros y otra aplicación de este tipo para las personas que desean comprar las boletas para estacionamiento de forma digital. Actualmente, la empresa opera en la Municipalidad de Montes de Oca en San José y cuentan con otra sucursal en Santiago de Chile.

La empresa define su labor en tres grandes ramas, las cuales se describen en su sitio *web* de la siguiente manera:

- **Aplicación móvil:** Los conductores pueden usar nuestra aplicación para buscar estacionamientos, usando la información de disponibilidad generada en tiempo real por nuestro sistema. Una vez estacionados, los conductores usan la misma aplicación para realizar el pago de tiempo de estacionamiento, de forma rápida y conveniente. Además, ofrecemos la opción de enviar un SMS para que sea adquirida la boleta en caso de que el usuario no cuente con un teléfono inteligente.
- **Sistema de administración:** Nuestras herramientas permiten a empresas, operadores y a gobiernos locales hacer una gestión eficiente de las zonas públicas de estacionamiento, ver reportes en tiempo real del estado de todas las áreas y configurar la operación del sistema.
- **Fiscalización e inspección:** Tenemos la mejor tecnología de fiscalización e inspección de estacionamientos, la cual usa algoritmos predictivos e información en tiempo real para hacer más eficiente las labores de los fiscalizadores (s. p.).

1.1.2. E-Park

E-Park es una aplicación creada por la empresa española Setex. Esta aplicación les permite a los usuarios la compra de boletas de parquímetros de forma digital desde cualquier dispositivo móvil que cuente con un sistema operativo Android, iOS, Windows Phone o BlackBerry. En Costa Rica, la Municipalidad de San José utiliza este sistema desde marzo del 2013 y su implementación cuenta con las siguientes características:

- *Hardware:* cuenta con un sistema de aparatos electrónicos conectados entre sí, los cuales permiten la compra de boletas directamente desde lugar donde se estaciona la persona.
- **Venta de boletas:** dentro del sistema implementado en la Municipalidad de San José existen tres distintas modalidades para adquirir las boletas de parquímetro: directamente desde los aparatos ubicados en las calles, mediante un mensaje de

texto o bien, desde cualquiera de los BN Servicios del Banco Nacional (La Nación, s. f., s. p.).

Además, según el sitio *web* de Setex, se encuentra implementada en municipios españoles como: Madrid, Santander, Santiago de Compostela, Coruña, Granada, Segovia, Marbella, Córdoba, Girona, Teruel, Almuñécar, Orihuela y Jerez.

1.1.3. Municipalidad de Escazú

En la investigación que se llevó a cabo en busca de implementaciones de sistemas similares a los que se desarrolló en este proyecto, se encontró el caso de la Municipalidad de Escazú, que trabaja las inspecciones de patentes y permisos de construcción de forma digital a través de *tablets* que establecen una comunicación con los sistemas de almacenamiento de la institución. Esta aplicación consiste en los siguientes aspectos:

- **Inspecciones de patentes:** este *software* fue realizado por el departamento de TI de la municipalidad. Su funcionamiento utiliza una plataforma en Android para registrar la información de una manera digitalizada y en tiempo real.
- **Inspecciones de permisos de construcción:** este caso es un poco diferente con respecto a las inspecciones de patentes, pues la municipalidad contrató una empresa externa que desarrolló el sistema. La empresa costarricense que se dedica a servicios de *outsourcing* y que realizó este trabajo se denomina Yaipan.

Como conclusión, se destaca que las soluciones de *software* presentadas anteriormente son sistemas genéricos, por lo que el uso de estas herramientas provocaría que la Municipalidad de Pérez Zeledón deba adaptarse a la forma de trabajar de estas herramientas. Esto podría implicar un cambio en la metodología y los reglamentos con los que se trabaja en la actualidad.

Caso contrario se encuentra con la propuesta del presente proyecto, que se desarrolló desde cero y en el cual se prefirió adaptarse completamente a las necesidades y el reglamento de la Municipalidad de Pérez Zeledón, aprovechar los recursos tecnológicos actuales y brindar una mejor atención a la población.

1.2. Justificación

Es importante recordar la finalidad de las municipalidades en Costa Rica, la cual se basa en gestionar los recursos cantonales, los intereses y servicios municipales, propiciar el bienestar general y la calidad de vida de las personas y promover el desarrollo integral de las empresas, instituciones y organizaciones, con sostenibilidad ambiental (Municipalidad de Pérez Zeledón, 2017).

Actualmente, la municipalidad realiza procesos de gestión de inspecciones de forma manual, en específico:

- Registra y consulta actas de permisos de construcción y patentes de negocios de manera manual y en papel, además de trasladarse físicamente hasta la municipalidad para entregar los documentos para que estos sean registrados y puestos a la disposición de cualquiera que lo requiera.
- Consulta permisos de estacionamiento de manera visual, observa si los vehículos tienen una boleta física con la fecha y hora permitida, además, registra estas boletas de forma manual. Este proceso también incluye trasladarse hasta la municipalidad para que sean registradas y posteriormente canceladas por el contribuyente. Cabe destacar que este proceso puede atrasarse hasta tres semanas, según la cantidad de trabajo con el que cuenten los administrativos del departamento de Seguridad Ciudadana.
- Venta de boletas de estacionamiento en formato físico y en negocios autorizados por la municipalidad; esto implica que los ciudadanos deberán localizar estos negocios para adquirir las boletas.

A partir de lo anterior surge la interrogante: ¿por qué realizar este proyecto de gestión de patentes e inspecciones?

El desarrollo de estos procesos se ha realizado de forma manual durante muchos años y los funcionarios ya se encuentran adaptados a trabajar de esta forma, sin embargo, la demanda y crecimiento de la población de la zona requiere del aprovechamiento de recursos tecnológicos que permitan innovar la realización de estos procesos.

En la actualidad, la industrialización de la zona sur es un aspecto importante, con base en un trabajo que realizó el MEIC, en conjunto con el Fondo para el logro de los objetivos del desarrollo del milenio y el PNUD, llamado *Encuesta a establecimientos de la Región Brunca 2011*, se determinó que para ese año el cantón de Pérez Zeledón contaba con un aproximado de 1.275 empresas o negocios y se encuentra en constante crecimiento. Esto implica que la intervención de la municipalidad es vital para el crecimiento del sector productivo en el cantón, por esto, una gestión de patentes en tiempo real permite que la municipalidad disminuya el tiempo en que se realizan las inspecciones de patentes, lo que genera un beneficio para los ciudadanos que se dedican al comercio y para los inspectores que realizan esta labor.

En el caso de las inspecciones de permisos de construcción, su control y monitoreo es vital para garantizar la seguridad y la responsabilidad, según lo establece el reglamento municipal actual en cada uno de los casos de construcciones o remodelaciones que se llevan a cabo diariamente, por lo que realizar en tiempo real estas inspecciones de permisos fortalecerá su gestión y beneficiará tanto a la municipalidad como al ciudadano.

En el tema de inspecciones de infracción a la ley de parquímetros, que realizan los inspectores de la municipalidad, el proceso de elaboración de una infracción implica el uso doble del papel (copia para la municipalidad y para el conductor), esto produce un impacto ambiental mayor. Además, resulta un proceso mucho más lento escribir los datos a mano si se compara con la escritura a través de un dispositivo digital. La sección de la infracción que incluye adjuntar fotografías es un punto bastante alto en la aplicación, ya que se puede documentar evidencia en caso de que existan dudas por parte del dueño del vehículo. Esto aunado a que los inspectores podrán consultar si un vehículo cuenta con permiso para estacionamiento a través de la sección de MPZ Pass.

En cuanto a la ley de parquímetros, existe un aspecto sumamente importante: la venta de boletas de estacionamiento a los ciudadanos. Este proceso se realiza de manera física, por medio de comercios autorizados por la municipalidad, lo que significa que el ciudadano debe desplazarse hasta estos comercios y comprar la boleta en papel. Lo anterior justifica la realización de un módulo que permita la venta de estas boletas de forma digital, esto permitiría que personas con capacidades disminuidas puedan acceder y comprar las boletas de manera digital desde dispositivos móviles o computadoras. Además, se brinda un proceso alternativo y

amigable con el medio ambiente.

Por otro lado, un módulo que le permita a la municipalidad la visualización de gráficos estadísticos sobre los lugares en los que se realicen las inspecciones de patentes, permisos de construcción e infracciones a la ley de parquímetros, permitiría que la institución encuentre puntos clave para tomar medidas con la realización de estos procesos.

Los argumentos anteriores justifican la necesidad de desarrollar e implantar el sistema para el manejo y procesamiento de patentes e inspecciones de la Municipalidad y aprovechar los beneficios tecnológicos para bienestar de todos los ciudadanos de Pérez Zeledón.

1.3. Objetivos del proyecto

1.3.1. Objetivo General

Apoyar la gestión de la información de las inspecciones de patentes de negocios, permisos de construcción, infracciones a la ley de parquímetros y la venta de boletas de estacionamiento mediante el desarrollo de una plataforma informática *web* y móvil para la Municipalidad de Pérez Zeledón.

1.3.2. Objetivos Específicos

- Proporcionar un mecanismo que registre las infracciones a la ley de parquímetros, por medio de una aplicación móvil que permita la gestión de estas en tiempo real, con el fin de llevar un control detallado de estas infracciones.
- Elaborar un módulo que permita la compra de boletas de parquímetros, a través de una herramienta *web* vinculada a la plataforma de comercio electrónico de la Municipalidad.
- Establecer un mecanismo para la administración digital de actas de inspección de patentes para los locales comerciales de la zona, por medio de una herramienta móvil que le permita al colaborador registrar los detalles de la inspección en la ubicación del establecimiento.

- Brindar un mecanismo que registre las inspecciones de construcción, a través de una herramienta móvil que les permita a los colaboradores recopilar la información de manera digital en el lugar de la inspección.
- Mostrar los datos recopilados de las inspecciones de parquímetros, patentes comerciales y permisos constructivos, por medio de un módulo de reportes que facilite la toma de decisiones.

1.4. Metodología

Con base en análisis realizados por el equipo del proyecto (desarrolladores, lectores y tutor), se estableció el uso de la metodología ágil SCRUM (Schwaber, 2013) para desarrollar el proyecto.

SCRUM se basa en la teoría de control de procesos empírica o empirismo. El empirismo asegura que el conocimiento procede de la experiencia y de tomar decisiones con base en lo que se conoce. SCRUM emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo (Schwaber, 2013).

Utilizar la metodología de Desarrollo SCRUM permitió potenciar los siguientes aspectos para el desarrollo del proyecto:

- **Agilidad:** la división del trabajo en pequeñas unidades funcionales (*Sprints*) permitió mantener una política de entregas frecuentes de *software* que ofrecen una visión clara del estado del proyecto y permite realizar pruebas de usuario a corto plazo.
- **Simplicidad:** se centra en facilitar el desarrollo rápido y mantiene un acercamiento con el usuario, para adaptarse completamente a sus necesidades y controlar el desarrollo del producto de manera más simple.
- **Flexibilidad:** todo el desarrollo se contempla como un ciclo de iteraciones continuas de desarrollo, lo que facilitó la introducción de modificaciones *en media etapa de desarrollo* y mejoró continuamente el proceso.
- **Productividad:** se centra en el desarrollo del producto como un todo, con una

planificación sólida, con las políticas de calidad requeridas, pruebas técnicas y su debido proceso de desarrollo.

El desarrollo del proyecto se dividió en cinco fases que establece la metodología SCRUM y se describen a continuación:

1.4.1. Planificación del Proyecto

- **Especificación de requerimientos de *software*:** el proceso de levantamiento de requerimientos estará a cargo del grupo de desarrolladores integrado por los ingenieros Luis Barrantes Mora y Erick Flores Solís, las técnicas que se utilizaron para el levantamiento de requerimientos fueron: reuniones formales, entrevistas y observación de los procesos de la municipalidad.
- **Desarrollo de un *backlog* completo:** con base en la especificación de requerimientos se creó una lista de tareas que el equipo elaboró en las reuniones de planificación con diferentes funcionarios municipales, como plan para completar los objetivos del proyecto.
- **Evaluación y control adecuado de los riesgos:** se realizó una matriz de riesgos para el proyecto, con el fin de gestionarlos adecuadamente, luego de valorar los posibles impactos, por último, se creó un plan de contingencia para controlar cada riesgo.
- **Selección de las herramientas *software* y *hardware* para el desarrollo del proyecto:** se analizó la arquitectura tecnológica actual de la municipalidad, con el fin de que el *software* y *hardware* se utilicen de una manera adecuada.
- **Determinación de cronograma de trabajo:** se creó un plan de trabajo que permitió dividir el proyecto en fases, de manera que se visualizaran las tareas de este y se evaluaran aspectos vinculados directamente con el tiempo y los recursos.

1.4.2. Diseño y Arquitectura

En esta sección, se estableció un esquema global del proyecto y se definió una arquitectura de trabajo que permitió establecer lineamientos acordes con lo realizado a lo largo del proyecto. Según la guía que establece SCRUM, los pasos para esta etapa son los siguientes:

- **Revisión de los elementos del *backlog* incluidos en la versión:** se hizo una revisión de la lista completa de *backlog* creada en la planificación y se estableció el *backlog* del próximo incremento a realizar. Se establecieron prioridades para cada funcionalidad del sistema y, a partir de eso, se dieron a conocer, con certeza, las próximas actividades a realizar.
- **Identificación de los cambios necesarios para implantar el *backlog*:** Después de que se realizó la inspección de la lista completa de *backlog* y se estableció la lista de actividades a realizar para la siguiente entrega, se procedió a identificar los cambios necesarios para llevar a cabo el proyecto. Estos cambios, generalmente, surgieron a raíz de las propias limitaciones que la tecnología utilizada impone y de las solicitudes del usuario final.
- **Análisis del dominio:** además, se realizó un análisis del entorno completo del proyecto, esto con el fin de identificar puntos que pudiesen verse afectados a partir de las modificaciones acordadas por el equipo de desarrollo.
- **Acotar la arquitectura del sistema:** se establecieron límites en la arquitectura del sistema con el fin de apoyar el nuevo contexto y las necesidades de la próxima entrega, esto se realizó con el fin de enfocar todo el esfuerzo en lo que compromete de forma directa el desarrollo del proyecto.
- **Identificar problemas del desarrollo:** durante el desarrollo del proyecto, todos los problemas que el equipo de desarrollo encontró se discutieron entre los miembros del equipo de desarrollo con el fin de brindar apoyo e intentar solucionarlo con la mayor brevedad posible, esto sin afectar el flujo normal en el desarrollo del proyecto.
- **Reunión de revisión de diseño:** cada miembro del equipo presentó los cambios

que consideró oportunos realizar para implantar los elementos del *backlog* e identificar posibles reasignaciones. Este tipo de reuniones se realizaron con el fin de que todos los miembros aportaran ideas que consideraban que podían contribuir con el desarrollo del proyecto.

1.4.3. Desarrollo (*sprints*)

Un *sprint* es un conjunto de actividades de desarrollo llevado a cabo durante un periodo predefinido, por lo general, entre una y cuatro semanas. En el desarrollo de este proyecto, se realizaron las siguientes actividades en cada uno de los *sprints* establecidos:

- **Desarrollo:** en esta fase se aplicaron los cambios necesarios para la implementación de los requisitos del *backlog* del *sprint*, además del diseño, desarrollo, implementación, pruebas y documentación de los cambios. El desarrollo consiste en el micro proceso de descubrimiento, invención e implementación (SCRUM Manager, 2013).
- **Envoltura:** en esta etapa se procedió a hacer un cierre de los módulos y a crear una versión ejecutable con los cambios que implantan los requisitos del *backlog*.
- **Revisión:** además, se realizó una reunión de todos los miembros del equipo de desarrollo para presentar el trabajo y revisar el progreso, en esta se identificaron y resolvieron posibles cuestiones y se añadieron nuevos elementos al *backlog*. Además, se revisaron los riesgos y las respuestas apropiadas (SCRUM Manager, 2013).
- **Ajuste:** finalmente, se llevó a cabo un último ajuste con el fin de consolidar la información de la revisión de los módulos afectados a partir de los cambios que se identificaron en la revisión.

1.4.4. Pruebas de Usuario

Este segmento es un factor fundamental en el desarrollo del proyecto, ya que le permitió al usuario verificar y validar el producto desarrollado, esto con el fin de realizar un *software* de calidad y cumplir los objetivos planteados.

La metodología de SCRUM permitió, gracias a las iteraciones de desarrollo, generar entregables concretos del proyecto en cada iteración, esto facultó que el usuario pudiese realizar las pruebas del *software* una vez finalizada cada interacción y también que el usuario lograra visualizar un progreso en el desarrollo del proyecto.

Para el desarrollo del proyecto se planeó un flujo de pruebas para el *software* descrito a continuación:

- El equipo de desarrollo finaliza el *Sprint* o iteración, posterior e internamente, se realizan pruebas de control de calidad de *software* para verificar un producto estable y robusto.
- Una vez finalizado el proceso anterior, se realizan pruebas de usuario, en las que se utiliza el producto en un entorno de desarrollo, con datos de prueba, pero con los procesos normales de operación, lo que permite un acercamiento del usuario con la aplicación, esto ayuda con el proceso de capacitación.
- Una vez realizadas las pruebas de usuario de cada iteración del proyecto, se procede a realizar las pruebas de integración, en las que cada módulo del proyecto se prueba como un todo, con la finalidad de observar el funcionamiento del producto final.
- Este flujo descrito anteriormente permitió controlar los entregables del desarrollo del proyecto, además de mostrarle al usuario un avance en el proyecto y realizar ajustes o modificaciones en momentos claves y, por último, lograr el cumplimiento de los objetivos estipulados.

1.4.5. Controles de SCRUM

La metodología de SCRUM define un mecanismo muy preciso en cuanto al control posterior del proyecto una vez que este se encuentra en producción, esto con el fin de brindar un mantenimiento adecuado al sistema implementado. Entre los controles que se llevarán a cabo una vez que se implemente el proyecto en su totalidad, se encuentran los siguientes:

- Riesgos: es una parte fundamental que se tomará en cuenta en el desarrollo del

proyecto y posterior a su implementación. Un seguimiento adecuado de los riesgos a lo largo de todo el ciclo de vida del proyecto permite que exista un plan de respuesta en caso de que algún riesgo se materialice.

- *Backlog*: son aquellos requisitos adicionales que no se consideraron en el momento de definir los requerimientos iniciales. Por lo tanto, este tipo de seguimiento incluye nuevas peticiones del cliente e inclusión de mejoras. Cabe destacar que estas nuevas funcionalidades serán responsabilidad del departamento de Servicios Informáticos de la municipalidad, el cual contará con el código fuente de la plataforma.
- Errores: los defectos del producto también serán contemplados como parte del control de seguimiento que se le dará al sistema una vez que este se encuentre en producción. Esta etapa se contemplará a partir del momento que se entrega el primer *sprint*.

1.4.6. Implementación y Cierre

Después de que se completaron todos los requisitos establecidos en el *backlog* inicial y, además, se solucionaron todas las peticiones adicionales del cliente, se procedió a hacer un cierre formal del proyecto, con una carta constitutiva que indica formalmente la finalización de la necesidad planteada.

Capítulo 2. Marco Teórico

2.1. Marco Referencial

El cantón de Pérez Zeledón se fundó en el año 1931 y con esto el establecimiento de su municipalidad. Actualmente, cuenta con más de 200 empleados distribuidos en diferentes departamentos, los cuales se encuentran en cinco lugares diferentes: palacio municipal, complejo cultural, mercado municipal, plantel municipal y finca municipal, además de contar con una oficina periférica para atención tributaria en Plaza Monte General. El palacio municipal es el centro de operaciones principal, este se ubica al costado de la catedral de San Isidro de El General. A continuación, se muestra el organigrama actualizado en el 2017, en el cual se observan las diferentes funcionalidades de las que se hace cargo este gobierno local, seguidamente, se mostrarán detalles más profundos y puntuales de la municipalidad, aspectos como su misión, visión, objetivos, entre otros.

2.1.1. Misión

Pérez Zeledón es un cantón multiétnico y pluricultural, caracterizado por el espíritu emprendedor de su población, que ha logrado posicionarse en distintas actividades económicas, donde sobresale la agricultura en una gran diversidad de productos, lo que ha generado un alto valor por el campesinado y su identidad local; con abundantes recursos naturales, especialmente en riqueza hídrica y belleza esencia, liderados por el Cerro Chirripó; con organizaciones dedicadas al desarrollo autogestionario de sus comunidades a partir de un incansable sentido de superación y lucha por una mejor calidad de vida para sus habitantes.

2.1.2. Visión

Ser un cantón que promueve el desarrollo humano de su población, con servicios públicos e infraestructura adecuados a las necesidades locales lo que garantiza la calidad de vida de sus habitantes, a través de la integración equitativa y solidaria, generando

espacios democráticos para la participación social, potenciando oportunidades educativas y fuentes de empleo dignos, donde se desarrollan actividades autogestionarias de proyectos agroindustriales, productivos y artesanales de pequeñas y medianas empresas locales y se promueven actividades económicas en una relación sostenible con los recursos naturales, para beneficio de las distintas poblaciones y el medio que las sustenta.

2.1.3. Objetivos Organizacionales

- Administrar y prestar los servicios públicos municipales.
- Percibir y administrar, en carácter de administración tributaria, los tributos y demás ingresos municipales.
- Concertar con personas o entidades nacionales o extranjeras, pactos, convenios o contratos necesarios para el cumplimiento de sus funciones.

2.1.4. Beneficios Esperados con la Plataforma

- Contar con una plataforma tecnológica para la automatización de procesos.
- Reducción significativa del uso del papel.
- Generación de infracciones a la ley de estacionamiento en tiempo real.
- Venta de boletas de estacionamiento de manera digital.
- Monitoreo de ubicaciones de los inspectores municipales, lo que permitirá conocer de manera adecuada su posición geográfica.
- Alternativa ecológica para la gestión de boletas de estacionamiento.
- Generación de actas de patentes en formato digital y tiempo real.
- Realizar actas de inspecciones de permisos de construcción en formato digital y tiempo real.
- Reducción de tiempo para entrega de información de patentes.

- Generación de reportes que apoyen la toma de decisiones.

2.2. Marco conceptual

2.2.1. Conceptualización Inicial

Es sumamente importante dejar claros varios aspectos que se citarán con frecuencia en el desarrollo del proyecto y que resulta vital explicar para que exista una buena comprensión del alcance del proyecto y el entorno en el cual se ha desarrollado. Para esto es preciso dividir esta sección en dos partes: el esquema técnico operacional y los términos administrativos. A continuación, se detalla lo citado anteriormente:

Esquema técnico operacional:

- *Ambiente web*: un ambiente *web* es aquel que puede accederse desde cualquier parte del mundo que cuente con conexión a Internet. La venta de boletas de estacionamiento será mediante esta plataforma, lo que permitirá realizarlas desde la página *web* principal de la municipalidad.
- *Ambiente móvil*: este concepto es relativamente nuevo y ha tomado mucha fuerza en los últimos años, ya que los teléfonos inteligentes y las *tablets* lo han permitido. Un ambiente móvil es sobre el cual se desarrollan aplicaciones que funcionan en estos dispositivos; en el caso de este proyecto, se desarrollará una aplicación móvil para el registro de actas de inspección de parquímetros, patentes comerciales y permisos constructivos.
- *Front-end*: se define como la parte de la aplicación con la que interactúa el usuario a través de una interfaz gráfica. En el caso del proyecto, este componente será desarrollado en Ionic2 para el caso de la aplicación móvil y en Angular2 para el desarrollo del componente *web* de venta de boletas de estacionamiento.
- *Back-end*: es la parte del sistema que procesa los datos ingresados en la aplicación del cliente, además, es la encargada de almacenar información en la base de datos para posteriormente consultarse por medio de un API.

- Base de datos: es el elemento que permite almacenar una cantidad de datos masivos para posteriormente consultarlos por medio de las aplicaciones que lo requieran; en el caso del proyecto se trabajará sobre bases de datos desarrolladas en plataformas como MS Access y SQL Server a nivel de servidor y SQLite a nivel de aplicación móvil.
- API: también conocido como *web service*, actualmente el API es un elemento esencial en las aplicaciones, ya que es el encargado de conectarse con la base de datos a bajo nivel y obtener la información, para luego enviarse en un lenguaje comprensible por la aplicación que lo solicitó. Para efectos del proyecto, la municipalidad cuenta con un API desarrollado en ASP.Net.
- GPS: sus siglas significan *Global Positioning System* (Sistema Global de Posicionamiento) y su utilidad es proveer la información de posicionamiento de acuerdo con nuestra ubicación en el planeta, lo que permite el funcionamiento de muchas aplicaciones en la actualidad (Kaplan y Hegarty, 2006).
- *E-commerce*: se trata de un término anglosajón, el cual puede ser traducido al español como comercio electrónico y su definición consiste en específicamente el Internet y otras redes que usan protocolos no propietarios como conducto para realizar negocios de forma electrónica (Molla y Licker, 2001).
- *Handheld*: también conocido como PDA (*Personal Digital Assistant*), es un dispositivo electrónico portátil que permite la instalación de aplicaciones móviles y realizar transacciones determinadas.

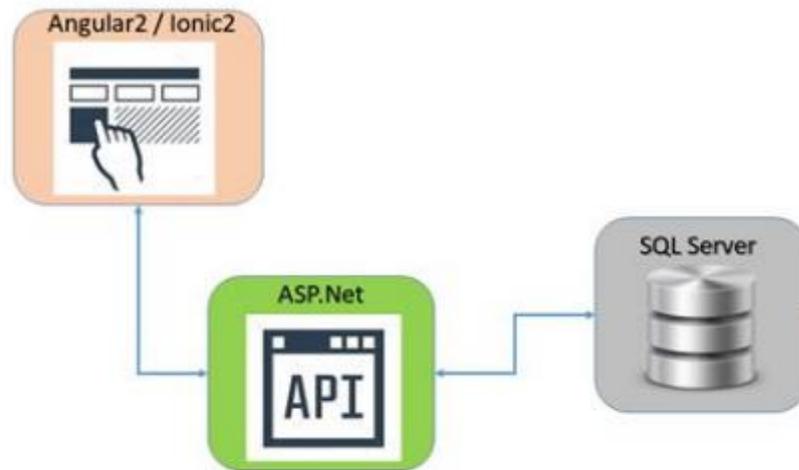


Figura 1. Arquitectura general de la plataforma. Fuente: Elaboración propia

Términos administrativos

A continuación, se hará referencia a términos administrativos que corresponden a aspectos propios de la municipalidad y que son importantes tener claros para que exista un mayor entendimiento del proyecto.

- Departamento: un departamento es una parte funcional de una organización o empresa que se dedica a atender situaciones de un área en específico. En la municipalidad existen numerosos departamentos, el proyecto involucra específicamente a los de Seguridad ciudadana, Planificación territorial y Licencias y patentes.
- Patentes comerciales: son los permisos que extiende la municipalidad para que un negocio o establecimiento desarrolle sus actividades de manera lícita y de acuerdo con la ley.
- Permisos constructivos: es la autorización que le externa la municipalidad a una persona física o jurídica que le permite llevar a cabo la construcción de un

comercio, residencia, etc., de acuerdo con los marcos de la ley.

- Boletas de estacionamiento: permiso que se adquiere en diferentes localidades de San Isidro y que brindan la posibilidad de estacionar automóviles en determinadas locaciones debidamente marcadas. Estas boletas tienen una vigencia limitada.

- Infracciones a la ley de parquímetros: es la violación a la Ley 8530 de Normas Básicas de Derecho Público que en el Artículo 1 menciona lo siguiente:

“Autorízase a las municipalidades a cobrar un impuesto cuando el tránsito así lo requiera, por el estacionamiento en las vías públicas. Además, en el Artículo 3 de la Ley 6852 se establece lo siguiente: Cuando un vehículo se estacione sin hacer el pago correspondiente o se mantenga estacionado después de vencido el tiempo por el cual se pagó, el propietario incurrirá en multa.”

- Coordinador: es la persona encargada de un departamento en la municipalidad.
- Colaborador: en cada departamento existen diferentes personas que son subordinadas del coordinador, a estos se les denomina colaboradores.
- Inspecciones: según lo define la Real Academia Española, se conoce como inspección a la acción de examinar o revisar algún aspecto en específico y asegurar su validez.

2.2.2. Estructura Modular

En cuanto a la estructura que conforma el desarrollo del proyecto, se encuentran los siguientes elementos:

- Aplicación para registro de infracciones de tránsito: esta aplicación conceptualiza su funcionalidad en la elaboración de infracciones por parte de los usuarios de la aplicación (inspectores de parquímetros de la municipalidad) hacia los vehículos que infrinjan la Ley de Parquímetros (ley que regula el estacionamiento en las vías públicas), esto por medio de una plataforma móvil,

específicamente instalada en un dispositivo PDA (asistente digital personal).

- Venta de boletas para estacionamiento: la venta de boletas para estacionamiento es controlada mediante la plataforma *web* de comercio electrónico de la Municipalidad (Sistema informático *web*) la cual se integró con la plataforma móvil de *infracciones por estacionamiento*, esta le permite al inspector consultar en tiempo real las boletas de permisos de estacionamiento registradas y con vigencia. El módulo se integra con la plataforma de comercio electrónico de la municipalidad (Sistema informático *web*), que consiste en un convenio con el Banco de Costa Rica para realizar pagos en línea y cargarlos en las cuentas del municipio.
- Aplicación para inspecciones a patentes de locales comerciales: se conceptualizó la elaboración de una aplicación móvil, que permitiese la generación de actas electrónicas (documento en formato digital) con la información recolectada por los funcionarios de la municipalidad, mediante las inspecciones de patentes realizadas a los locales comerciales; con el fin de procesarse por el departamento de Licencias y Patentes (departamento de la municipalidad encargado de gestionar las licencias y patentes en el sector comercial del cantón).
- Aplicación para inspecciones a permisos constructivos: se implementó una aplicación adicional para la inspección de permisos de construcción que controla mediante una plataforma móvil (sistema informático base en dispositivos móviles), que permitiera a los funcionarios el registro de la información recolectada durante las inspecciones que se realizan en las construcciones y remodelaciones de obras en edificios, casas, entre otros. La aplicación le permite al funcionario de la municipalidad consultar si existe una inspección realizada con anterioridad a la construcción, además de visualizar datos sobre las inspecciones de obras alojadas en el sistema.
- Aplicación *web* de administración: se desarrolló una plataforma global que permite diferentes gestiones alrededor de las aplicaciones citadas anteriormente, las cuales poseen un esquema con base en una misma base de datos, lo que permite su gestión de una forma más centralizada y ordenada. Esta aplicación

permite el registro de usuarios, asignación de permisos, registro de vehículos y generación de código QR único para su identificación, generación de archivo para plataforma SICSOA, gestión de parámetros generales de las aplicaciones, consulta de infracciones de parquímetros emitidas y monitoreo de los inspectores vía GPS.

2.3. Marco Metodológico

La metodología para emplear el desarrollo de este proyecto permite realizar, gestionar y administrar sus distintas etapas y así llevarlo a cabo con mayores posibilidades de éxito. Por lo tanto, una metodología para el desarrollo del *software* son los procesos que deben seguirse constantemente para idear, implantar y mantener una aplicación informática desde que surge la necesidad del producto hasta que se cumplen los objetivos por los cuales fue desarrollado.

El ciclo de vida de una herramienta de *software* se define como la secuencia de fases, actividades y controles para trasladarse de una fase a otra, así como los resultados que generan cada una de estas, lo que permite el desarrollo de un producto desde su concepción, entrega al usuario y evolución posterior, hasta su retiro de producción. No existe un modelo de ciclo de vida único, tanto el tipo, el orden y las actividades en cada fase pueden cambiar adaptándose a las necesidades del producto a realizar y a la propia estructura de la organización que lo desarrolla, a partir de las posibilidades que ofrece la tecnología de *software* empleado.

A continuación, se describe detalladamente la estructura metodológica que se adoptó para el desarrollo del proyecto.

2.3.1. Levantamiento de Requerimientos

Un requerimiento se concibe como una manifestación imprecisa de alto nivel de un servicio que debe proveer el sistema o como una restricción de este, por lo que se le atribuye gran relevancia (Mitec, 2010; Jacobson *et al.*, 2000). El levantamiento de requerimientos es el paso más delicado e importante en el diseño de un sistema informático. Este proceso revela las aspiraciones y requisitos que el usuario quiere en su aplicación.

La obtención de requisitos es una actividad de evolución, no todos los actores se

identifican durante la primera iteración. Es posible identificar actores iniciales durante la primera iteración y actores secundarios conforme se aprende más del sistema. Los actores iniciales interactúan para conseguir funciones del sistema y derivar el beneficio deseado del sistema (Pressman, 2005, p. 187).

No realizar en forma adecuada la recolección de datos repercutirá en forma grave la consecución del desarrollo del sistema. Como todo proceso, el levantamiento de requerimientos presenta diversos conceptos básicos, entre los que se encuentran los siguientes:

- **Requerimientos funcionales:** describen la funcionalidad o los servicios que se espera que brinde. Estos dependen del tipo de *software* y del sistema que se desarrolle y de los posibles usuarios del *software*. Cuando se expresan como requerimientos del usuario, habitualmente se describen de forma general mientras que los requerimientos funcionales del sistema describen con detalle la función de este, sus entradas, salidas, excepciones, etc.
- **Requerimientos no funcionales:** no se refieren directamente a las funciones específicas que entrega el sistema, sino a las propiedades emergentes de este, como la fiabilidad, la respuesta en el tiempo y la capacidad de almacenamiento. De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y la representación de datos que se utiliza en las interfaces del sistema. Muchos requerimientos no funcionales se refieren al sistema como un todo más que a rasgos particulares de este.
- **Requerimientos del dominio:** estos se derivan del dominio del sistema más que de las necesidades específicas del usuario. Pueden ser requerimientos funcionales nuevos, restringir los existentes o establecer cómo se deben ejecutar cálculos particulares (Mitec, 2010; Jacobson *et al.*, 2000). Si estos requerimientos no se satisfacen es imposible hacer que el sistema trabaje de forma satisfactoria.
- **Requerimientos del usuario:** describen los requerimientos funcionales y no funcionales de forma que sean comprensibles para los usuarios del sistema que no poseen un conocimiento técnico detallado. Mediante la realización de

reuniones con el usuario se aclaran los requerimientos de este y se construye una base sólida para el desarrollo del *software*.

- **Requerimientos del sistema:** son descripciones más detalladas de los requerimientos del usuario. Sirven como base para definir el contrato de la especificación del sistema y, por lo tanto, debe ser una especificación completa y consistente del sistema. Con base en los requerimientos de usuario, previamente aceptados por este, se realiza el levantamiento de requerimientos del sistema.
- **Requerimientos del *software*:** es la declaración oficial de qué es lo que requieren los desarrolladores del proyecto. Incluye tanto los requerimientos del usuario como una especificación detallada de los requerimientos del sistema.

2.3.2. Recolección de Datos

La informática, como cualquier otra ciencia, necesita de la investigación para solucionar diferentes problemas, sin embargo, en muchas ocasiones se desconoce o no se realizan de forma competente los métodos de extracción de datos. Un buen analista de sistemas conoce las técnicas para obtener información y las aplica de forma correcta, con las que adquiere el conocimiento necesario para el proyecto que debe realizar, lo que le brinda mayor valor a la información obtenida.

Si en esta etapa se comete algún error o surge algún inconveniente como la mala comprensión de los requerimientos del usuario, provocaría un retraso trascendental en el desarrollo del proyecto. Para que la fase de levantamiento de requerimientos se lleve a cabo de la mejor forma posible y debido a que, por lo general, a los usuarios les es difícil comunicar lo que realmente quieren o no saben lo que quieren, se hace uso de distintas técnicas como las que se mencionan a continuación:

- **Entrevista:** permite revelar de forma general los requerimientos del usuario respecto al sistema, lo que generará una base sólida en la cual se podrá estructurar el proyecto.
- **Cuestionarios:** este método de gran relevancia contiene en su mayoría

interrogantes relacionadas con las incertidumbres del analista en cuanto a la aplicación, con lo que se consigue una visión todavía más clara de lo que se pretende realizar.

- Observación: representa también una herramienta elemental en el diseño del proyecto, pues a través de esta se puede estudiar detalladamente la forma en la que se realiza en la actualidad la labor que se desea sistematizar, los datos que se deben incluir y la forma en que deben operar.
- Investigación: se debe tomar en cuenta la revisión de documentos, ya que estos aportan una cantidad considerable de conocimiento extra a la investigación de requerimientos.

Al realizar esta serie de actividades de forma adecuada, se logró esquematizar y estructurar una solución apta para los requisitos del usuario, lo que facilitó el desarrollo de este al tener una perspectiva más determinada. Para tal efecto, se llevaron a cabo reuniones con los usuarios finales (coordinadores e inspectores de cada departamento), en las que se aplicaron los cuestionarios y entrevistas para determinar los requerimientos y aclarar dudas y cuestionamientos.

2.3.3. Análisis

Se define como el estudio que especifica las guías de una aplicación con vistas a desarrollar una solución mecanizada. En otras palabras, los pasos para elaborar una aplicación informática. Un sistema requiere el establecimiento de varios lineamientos con los cuales llevar a cabo la realización de un proyecto, los requerimientos son parte fundamental de estos, sin embargo, si no se entienden de forma idónea vuelven más complejo el diseño.

Para el análisis se realizaron reuniones entre los miembros del grupo, esto con el fin de analizar la información recopilada en las reuniones con los distintos usuarios finales. El análisis debe enfocarse en los dominios de la información, funcional y de comportamiento del problema. Para entender mejor lo que se requiere, se crean modelos, los problemas sufren una partición y se desarrollan representaciones que muestran la esencia de los requisitos y posteriormente los detalles de la implementación.

Mediante el análisis de los requerimientos se consigue especificar sistemas complejos al documentar detalles simples y breves, esto lo hace más fácil de entender y proporciona mayor flexibilidad y disposición en la etapa de programación.

Para optimizar la información formulada en los requerimientos se utiliza el análisis, no obstante, este se lleva a cabo mediante diferentes actividades como la identificación y clasificación de los requerimientos, para proporcionar un mayor interés en los requisitos indispensables del sistema. Luego se procede a dividirlos, según su prioridad dentro del sistema, ya sea porque necesitan implementarse primero o porque cubrirán en mayor parte los principales objetivos del sistema. Una vez realizado lo anterior, se continúa con la definición detallada, de manera que se puedan diferenciar los requerimientos funcionales y los no funcionales. Esto con la precaución de especificar los requerimientos precisamente.

La especificación de los requisitos del *software* se produce en la culminación de la tarea de análisis. La función y rendimiento asignados al *software* como parte de la ingeniería de sistemas se retiran estableciendo una completa descripción de la información, una descripción detallada de la función y del comportamiento, una indicación de los requisitos del rendimiento y restricciones del diseño, criterios de validación apropiados y otros datos pertinentes a los requisitos (Pressman, 1998, s. p.).

2.3.4. Estudio de Factibilidad

La factibilidad determina si es viable que el sistema de información sea desarrollado e implantado exitosamente en la empresa, por lo que se enfatiza en la importancia de conocer si el proyecto es apropiado para la organización, debido a factores técnicos, económicos y operacionales.

2.3.5. Factibilidad Operacional

Se refiere a la posibilidad de éxito que tendrá el sistema en el momento de implantarse y ser operado por el personal de la empresa. Para evaluar esta posibilidad se consultó a los usuarios si el nuevo proyecto tendrá un efecto positivo o negativo y, además, se charló con el coordinador de cada departamento para conocer si utilizarán el sistema una vez que se haya

finalizado.

2.3.6. Factibilidad Técnica

Evalúa si el equipo y *software* están disponibles, es decir, si existe la tecnología necesaria para cubrir las demandas del nuevo proyecto. Para el desarrollo del proyecto se evaluaron las herramientas tanto de *software* como de *hardware* que posee la institución, con el objetivo de ajustar el desarrollo del sistema conforme a estas herramientas para lograr una compatibilidad entre el producto y la institución. En el estudio de factibilidad técnica también se consideró si la organización tiene el personal que posee la experiencia técnica requerida para operar y mantener el sistema propuesto.

2.3.7. Factibilidad Económica

Los estudios de factibilidad económica incluyen análisis de costos y beneficios asociados al proyecto. Realizar este proyecto no tiene ningún costo para la institución, ya que es de investigativo al tratarse de un proyecto universitario, sin embargo, se hizo una investigación y estudio del *hardware* y *software* actual, para verificar si se cuenta con el equipo necesario para desarrollar el proyecto.

2.3.8. Modelaje de Datos

Al conjunto de herramientas conceptuales para describir datos, sus relaciones, significado y sus restricciones de consistencia se le conoce como modelo. La tarea de construir correctamente una base de datos se sitúa en una perspectiva adecuada al considerarla dentro del ciclo de vida del *software*.

Se desarrolló el modelo entidad-relación, luego se procedió a construir las tablas de la base de datos que requeriría la plataforma, conjuntamente se añaden nuevas tablas o funciones, según se presente la necesidad.

2.3.9. Desarrollo

Se utilizó la metodología de SCRUM la cual define un conjunto de actividades de

desarrollo llevado a cabo durante un periodo predefinido (*sprints*). En el desarrollo de este proyecto, se realizaron las siguientes actividades en cada uno de los *sprints* establecidos:

- Desarrollo: en esta fase se aplican los cambios necesarios para la implementación de los requisitos del *backlog* del *sprint*, además del diseño, desarrollo, implementación, pruebas y documentación de los cambios. El desarrollo consiste en el micro proceso de descubrimiento, invención e implementación (SCRUM Manager, 2013).
- Envoltura: en esta etapa se procede a hacer un cierre de los módulos y se crea una versión ejecutable con los cambios que implantan los requisitos del *backlog*.
- Revisión: además, se realizan reuniones de todos los miembros del equipo de desarrollo para presentar el trabajo y revisar el progreso, en estas se identifican y resuelven posibles cuestiones y se añaden nuevos elementos al *backlog*. También se revisan los riesgos y las respuestas apropiadas (SCRUM Manager, 2013).
- Ajuste: finalmente, se lleva a cabo un último ajuste con el fin de consolidar de la información de la revisión de los módulos afectados a partir de los cambios que se identifican en la revisión.

2.4. Pruebas

En esta sección se explica la metodología definida que se utilizó para el diseño de las pruebas, la cual consiste en un conjunto de pruebas por *Sprint* de desarrollo sobre cada módulo desarrollado, además de la realización de una prueba del sistema completo que brinda calidad al *software*.

La verificación de los errores y del buen funcionamiento del *software* se llevó a cabo de acuerdo con el desarrollo y terminación de los módulos, los cuales fueron probados por diferentes usuarios de los departamentos involucrados, que permitieron recolectar información valiosa como consecuencia de la interacción con el programa. Los tipos de pruebas a realizar son:

- Pruebas entre desarrolladores (integridad y modularidad).
- Pruebas con usuarios siguiendo una guía (modular).
- Pruebas de comunicación y conexión con el servidor y base de datos.
- Pruebas de interfaz en los diversos navegadores.

2.4.1. Manual del Usuario

En este documento se explica detalladamente cada uno de los formularios o pantallas que presenta el sistema, así como la función que tiene cada control, también se incluyen imágenes para que el usuario se guíe e informe sobre cuál es el elemento del que se hace referencia. Además, se cuenta con la opción de ayuda del sistema, esta complementa la forma en la que el usuario puede informarse, se creó pensando en las posibles dudas que pueda tener el usuario durante la utilización del sistema, con la posibilidad de ser accedida desde la aplicación en cualquier momento. También se explican los diferentes mensajes de alerta, información y error que se pueden presentar durante la ejecución del sistema.

2.4.2. Manual Técnico

Se confeccionó con el propósito de que cualquier otro desarrollador que lo lea pueda comprender la lógica implementada en la plataforma; en este se incluyeron todos los pasos realizados por los desarrolladores para la creación del sistema. Incluye el diccionario de datos, en el que se explica la lógica a nivel de base de datos. Este documento, al igual que el manual del usuario, será de gran ayuda en el futuro si los funcionarios de la municipalidad desean hacerle alguna modificación al sistema, ya que por lo general se tarda mucho tiempo en entender el código. Además, cuenta con una sección para realizar la instalación en la que se detallan todas las acciones que se deben implementar para que el sistema sea utilizable en nuevos dispositivos.

Capítulo 3. Procedimiento Metodológico

En este capítulo se hará una descripción detallada de cómo aplicaron todos los procesos descritos anteriormente en el marco metodológico. Este proceso se dividió en nueve partes distintas, las cuales son: levantamiento de requerimientos, análisis, análisis de requerimientos, estudio de factibilidad, modelaje de datos, desarrollo, pruebas, manual técnico y manual del usuario.

3.1. Levantamiento de requerimientos

La definición de requerimientos es una de las partes más importantes de la planificación en un proyecto, pues permite brindar una visión firme de lo necesario para desarrollar, ya que estos requerimientos delimitan el alcance del proyecto, así como sus limitaciones. Para la definición de los requerimientos se acordaron diferentes reuniones con los coordinadores de cada departamento y con sus respectivos inspectores, esto permitió obtener la información de primera mano por parte de los usuarios finales que harán uso de la plataforma. Los requerimientos se dividieron en requerimientos funcionales y no funcionales y se citan a continuación:

3.1.1. Aplicación móvil para registro de infracciones de tránsito

3.1.1.1. Requerimientos Funcionales

Inicio de sesión: se requiere un mecanismo de inicio de sesión segura en cada dispositivo que cuente con la aplicación instalada. Se acordó que la aplicación para registro de infracciones de tránsito no tenga que iniciar sesión cada vez que el usuario la abra, pues esto implicaría un gasto de tiempo innecesario, por lo que será responsabilidad de cada inspector que cada uno utilice su respectiva cuenta. En el caso de las aplicaciones para inspección de patentes y permisos de construcción, será necesario iniciar sesión cada vez que se abra la aplicación.

Sincronización de utilidades: una vez que el usuario inicie sesión, los parámetros generales de la aplicación serán sincronizados y almacenados de manera local, con el fin de que se cuente con la información de primera mano y de forma inmediata. Estos parámetros serán:

tarifas de multas, tipos de vehículos, clases de vehículos, artículos, colores y marcas. Después de que el dispositivo inicie sesión, deberá consultar al API para conocer si los datos que almacena son los más actualizados, en caso de que se determine que no es así, se procederá a informar de forma visual al usuario para que realice la respectiva sincronización.

Registro de infracciones: es uno de los requerimientos más extensos, ya que abarca casi por completo el propósito de la aplicación. Las infracciones serán registradas cuando el inspector considere que sea necesario, por lo que estas serán su responsabilidad. En el registro de cada una de las boletas se debe manejar un consecutivo que determinará el número único de boleta, este número no puede incluir signos especiales como guiones, comas, puntos, entre otros; además, cada inspector debe contar con su propio número de usuario y se verá reflejado en este consecutivo, lo que permitirá que no exista ningún tipo de inconveniente en el momento de su cancelación ya que resultará imposible un número duplicado, este consecutivo deberá estar sincronizado con la base de datos principal, pero se deberá manejar también de forma local en cada dispositivo para evitar inconvenientes cuando el dispositivo tenga problemas de comunicación vía Internet con el servidor de la municipalidad.

Aparte del número de infracción, se deberá registrar el tipo de vehículo, su clase, número de placa, color, marca, artículo infringido, ubicación exacta del vehículo (esto se hará de forma manual, al indicar el número de calle y avenida) y observaciones adicionales; además, el dispositivo deberá registrar la ubicación geográfica en la que se encuentre en el momento de la infracción.

Registro de evidencia (fotografías): este requerimiento está directamente enlazado al número 2, ya que cada infracción registrada deberá contar con una serie de fotografías que servirán como evidencia en caso de que el dueño del vehículo decida verificar si en realidad ameritaba una infracción. Estas fotografías se registrarán en un servidor de almacenamiento dentro de la municipalidad; al no contar con un servidor FTP para la transferencia de archivos, se debe desarrollar la lógica necesaria para que se registren de una manera eficiente y no se tarde mucho en el registro de cada infracción. Será necesario que el inspector registre tres fotografías exactas; no es necesario que estas fotografías se registren en una definición muy alta, pues el caso no lo amerita.

Impresión en papel de la infracción: después de registrar la infracción en la base de datos,

se imprimirá un comprobante con los datos respectivos que se colocará en el vehículo. Esta impresión se realizará por medio de bluetooth, por lo que será necesario que el dispositivo lo tenga activado. Se acordó la adquisición de una primera impresora por parte del departamento de Servicios Informáticos para la realización de pruebas durante el desarrollo.

Consulta de vehículos: es vital verificar que el vehículo no cuenta con una infracción previa por el incumplimiento a un mismo artículo en un margen de 24 horas. Además, el inspector debe verificar que el vehículo no cuente con tiempo disponible que le permite estacionarse en determinadas zonas. Esta consulta se realizará por medio del tipo de vehículo, clase y número de placa; sin estos tres datos, la consulta será imposible de realizar.

MPZ Pass: esta funcionalidad será vinculada al requerimiento anterior, ya que el inspector podrá escanear un código QR único de cada vehículo, por medio de la cámara del dispositivo; estos vehículos deberán estar registrados en la base de datos de la municipalidad, este proceso de registro queda a criterio del coordinador del departamento de Seguridad Ciudadana. Esta funcionalidad le ahorrará mucho tiempo al inspector, pues no será necesario que se escriba los datos del vehículo.

Registro de infracciones *off-line*: Es necesario que se registren infracciones de parquímetros de forma local en el dispositivo, en caso de que no se pueda establecer una comunicación correcta con el servidor de la municipalidad. Además, será necesario almacenar las fotografías tomadas y vincularlas a cada infracción.

Sincronización de infracciones: las infracciones que se registren de manera local en cada dispositivo deben mostrarse en una sección de la aplicación, de manera que el inspector las sincronice en el momento que cuente con conexión a Internet.

Captación de ubicación geográfica por desplazamiento: es necesario que la aplicación registre las ubicaciones de cada uno de los inspectores en tiempo real, por lo que se deberá almacenar en la base de datos una ubicación cada vez que el dispositivo identifique que el inspector se desplazó 30 metros (parametrizable).

Cierre de sesión: es necesario que el usuario tenga la opción de cerrar sesión, ya que se debe validar que cada inspector solo posea una sesión iniciada en un dispositivo, esto para evitar inconsistencias en los datos.

Registro de dispositivo: es necesario desarrollar un mecanismo que almacene el número único de cada dispositivo (IMEI), esto con el fin de garantizar la seguridad en la aplicación, pues solo los dispositivos registrados podrán iniciar sesión.

3.1.1.2. Requerimientos no funcionales

Sistema operativo: la aplicación deberá ejecutarse en sistemas operativos Android.

Plataforma: la aplicación deberá desarrollarse en Ionic 2, ya que es un *framework* de código abierto que permite su uso para cualquier tipo de aplicación móvil. Además, el API deberá desarrollarse en ASP.Net y en una base de datos SQL Server 2008 R2.

Interfaz: el sistema debe disponer de una interfaz gráfica efectiva y visualmente comprensible para el usuario.

Identificación: la aplicación debe mostrarse bien identificada con el logo de la municipalidad en las pantallas de inicio de sesión e inicio pantallas de iniciar sesión.

Excepciones: el sistema debe ser capaz de controlar excepciones en la medida de lo posible durante su ejecución, además de mostrar advertencias claras para el usuario.

3.1.2. Aplicación web para venta de tiempo de estacionamiento

3.1.2.1. Requerimientos funcionales

Registro de tiempo de estacionamiento: es necesario realizar un módulo en la aplicación que permita registrar los datos del vehículo que hará la compra de tiempo desde la página *web* de la municipalidad; estos datos serán: tipo de vehículo, clase de vehículo, número de placa y cantidad de tiempo a comprar (máximo 9 horas).

Conexión bancaria: es necesario que la cancelación del tiempo comprado por parte del usuario se realice a través del convenio que posee la municipalidad con el Banco de Costa Rica, para que se realice el cargo a través de los datos que esta pasarela de pago solicita. Una vez realizado el pago, deberá notificarse al usuario para que se asegure que se realizó de manera exitosa o que surgió algún tipo de error.

Integración con aplicación móvil para registro de infracciones de tránsito: es vital que la

aplicación para el registro de infracciones pueda consultar los datos registrados a través de esta aplicación que funcionará de forma paralela.

3.1.2.2. Requerimientos no funcionales

Plataforma: esta aplicación será *web*, por lo que será desarrollada en Angular 2, con una conexión a un API en ASP.Net y una base de datos SQL Server 2008 R2.

Responsiva: la aplicación debe ser responsiva, esto con el fin de que se adapte a cualquier tipo de dispositivo en el que sea acceda (computadoras, teléfonos, *tablets*, etc.).

Integración: la aplicación debe adaptarse a la página *web* de la municipalidad, por lo que se debe tomar en cuenta los estándares y la guía de estilos con la que cuentan en la municipalidad para que no exista una diferencia de contrastes entre la aplicación y el resto de los componentes en la página.

3.1.3. Aplicación *web* de administración

3.1.3.1. Requerimientos funcionales

Registro de usuarios y asignación de permisos: será necesario crear un módulo que permita el registro de nuevos usuarios en el sistema, dentro de este registro será necesario asignar permisos de forma sencilla para el usuario. Además, se tendrá la opción de editarlo e inhabilitarlo para el uso de cualquier aplicación que posea la plataforma.

Reporte de infracciones de tránsito realizadas: es necesario que la aplicación cuente con un módulo que permita la consulta de infracciones de tránsito realizadas por los inspectores. Estas infracciones podrán filtrarse por número de placa, por número de boleta o por inspectores, este último deberá permitir establecer un rango de fecha para una filtración más exacta.

Generación de archivos para SICSOA: el departamento de Seguridad Ciudadana debe enviar una vez al año un archivo al INS con las infracciones de tránsito que todavía no se han cancelado en la municipalidad, con el fin de que se carguen en el derecho de circulación y se haga el debido recargo. Este archivo se presentará en un formato predefinido; esta consulta deberá permitir establecer un rango de fechas para filtrar cierta cantidad de infracciones.

Registro de vehículos para uso de MPZ Pass: la plataforma de MPZ Pass será una idea novedosa que le permitirá al usuario el registro de su vehículo con los datos que le conciernen. Una vez registrado el vehículo con todos sus datos, la aplicación deberá desplegar un código QR único, el cual se le entregará al dueño del vehículo, esto les facilitará a los inspectores la consulta de tiempo para estacionamiento, sin la necesidad de digitar la información del vehículo. Además, queda prevista la posibilidad de comprar boletas para estacionamiento a través de este mismo código.

Monitoreo de localización de inspectores: se requiere la implementación de un módulo que permita la consulta y visualización de las ubicaciones de los inspectores en tiempo real, por lo que será necesario desplegar un mapa con marcadores que muestren la información de cada inspector. Este módulo se integrará con la información registrada, según el requerimiento 1.10.

Administración de parámetros generales: es necesario que la aplicación permita el registro de los parámetros generales. Estos parámetros serán: tarifas de multas, artículos, colores y marcas.

3.1.3.2. Requerimientos no funcionales

Plataforma: la plataforma debe ser *web*, por lo que será desarrollada en Angular 2, conectado a un API en ASP.Net y una base de datos SQL Server 2008 R2.

Interfaz: la interfaz de la aplicación debe ser clara e intuitiva, para que la curva de aprendizaje de los usuarios sea lo más corta posible y su adaptación sea correcta.

Interna: la aplicación solamente se utilizará en el ámbito institucional, por lo que se debe controlar que no pueda ser accedida desde equipos que se encuentren fuera del dominio de la municipalidad.

3.2. Recolección de datos

Para realizar esta sección se planteó establecer, en específico, tres tipos de mecanismos para recolectar datos y que son muy comunes para todo tipo de investigaciones en general. Los mecanismos acordados para la obtención de datos importantes son: investigación, observación y entrevista; se eligieron estos tres métodos ya que se ajustaron perfectamente a las necesidades

de información con las que se contaba. A continuación, se detallará la forma en la que se emplearon estas técnicas:

- **Investigación:** se planteó realizar investigaciones sobre la municipalidad y los departamentos que involucra este proyecto, por lo que se indagó sobre su forma de trabajar y sus procesos de manera general. De esta manera, fue posible crear un panorama más amplio que permitió conocer las necesidades que solventaría la implementación del proyecto.
- **Entrevista:** se coordinaron diversas entrevistas con los diferentes coordinadores de cada departamento beneficiado por el proyecto; estas entrevistas permitieron obtener más información relevante que benefició la comprensión del proyecto de manera más clara. En total se realizaron tres entrevistas con cada coordinador: la primera para recolectar los requerimientos, la segunda para confirmar los requerimientos obtenidos y una última reunión para presentar la plataforma finalizada.
- **Observación:** en la primera entrevista se solicitó información a los coordinadores de cada departamento involucrado en el proyecto, como las actas de inspección y las boletas de infracciones. Esto permitió observar de forma más detenida la información necesaria para registrar y, de esta forma, empezar la estructuración de la base de datos y el diseño de la interfaz.

3.3. Análisis

En esta sección se detallará el análisis realizado en el momento en el que se plantearon las soluciones óptimas para las necesidades obtenidas en el levantamiento de los requerimientos del sistema. Por lo tanto, a continuación, se explicará el proceso de cada una de las funcionalidades con las que cuenta la plataforma.

3.3.1. Aplicación para registro de infracciones: Inicio de sesión

Este proceso permite que el usuario se autentique una vez que la aplicación se abre en el dispositivo. En la aplicación se solicitará el nombre de usuario y contraseña, los cuales serán

enviados al API, para que estos sean validados con la base de datos y, finalmente, que se retorne una respuesta de error o de aprobación.

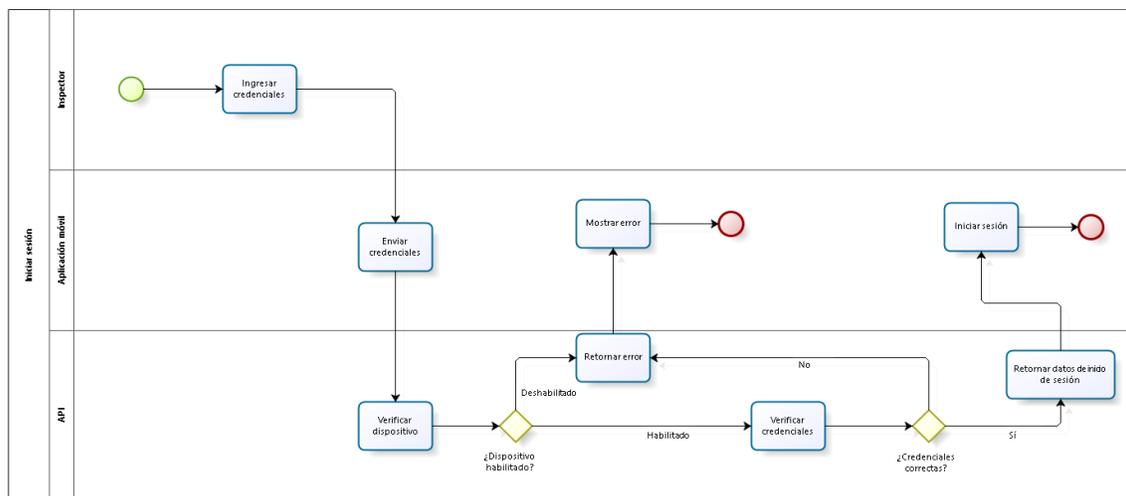


Figura 2. Diagrama de flujo para el inicio de sesión. Fuente: Elaboración propia

3.3.2. Aplicación para registro de infracciones: Registrar infracción

El flujo normal para el registro de infracciones a la ley de parquímetros consistirá en que el usuario, en primer lugar, abre la aplicación y selecciona la opción de *Nueva infracción*, después la aplicación cargará los parámetros generales como los tipos de placas, clases de placas, colores, las marcas y los artículos; seguidamente, el usuario llenará el formulario que le presenta la aplicación y la información se envía al API para registrarse. Una vez que el API confirma que la infracción se registró de manera satisfactoria, se imprime la infracción; en caso de que no se registre a través del API, se retornará un mensaje de error y la aplicación la almacenará automáticamente en la base de datos local y se imprimirá la infracción.

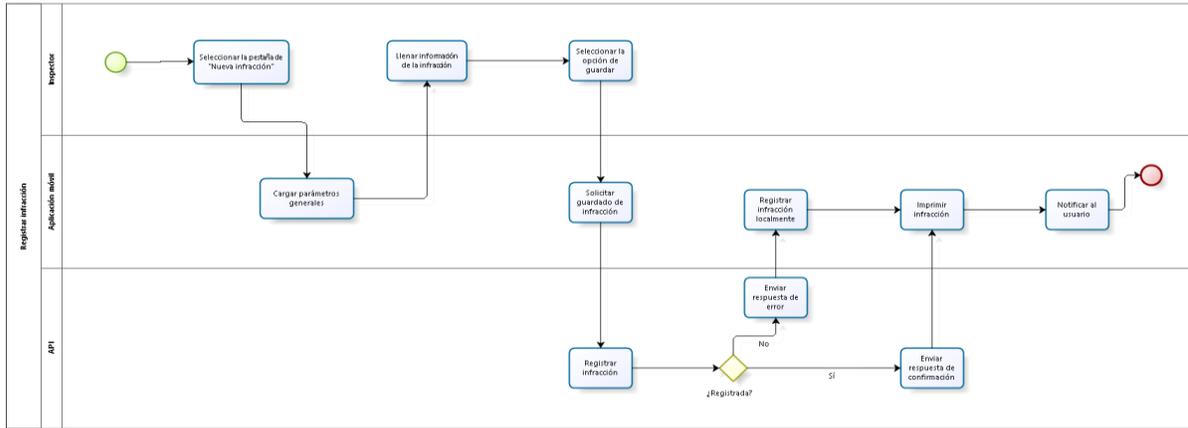


Figura 3. Diagrama de flujo para el registro de infracción. Fuente: Elaboración propia

3.3.3. Aplicación para registro de infracciones: Consultar vehículos

Para iniciar este proceso, el usuario debe seleccionar la sección para consultar vehículos en la cual la aplicación cargará los parámetros generales como el tipo de la placa y la clase; una vez cargada esta información, el usuario seleccionará el tipo de consulta: consulta regular (por número de placa) o consulta por MPZ Pass (escaneo de código QR). En caso de que el usuario elija la primera opción, se desplegará un formulario que solicite el número, clase y tipo de placa del vehículo a consultar; en caso de que seleccione la opción de MPZ Pass, se presenta un botón para activar la cámara con funcionalidades de escaneo de códigos QR. Luego de ingresar la información (datos del vehículo o código escaneado), la aplicación procede a enviar esta información al API, para que sea procesada y retorne si el vehículo amerita o no una infracción de tránsito.

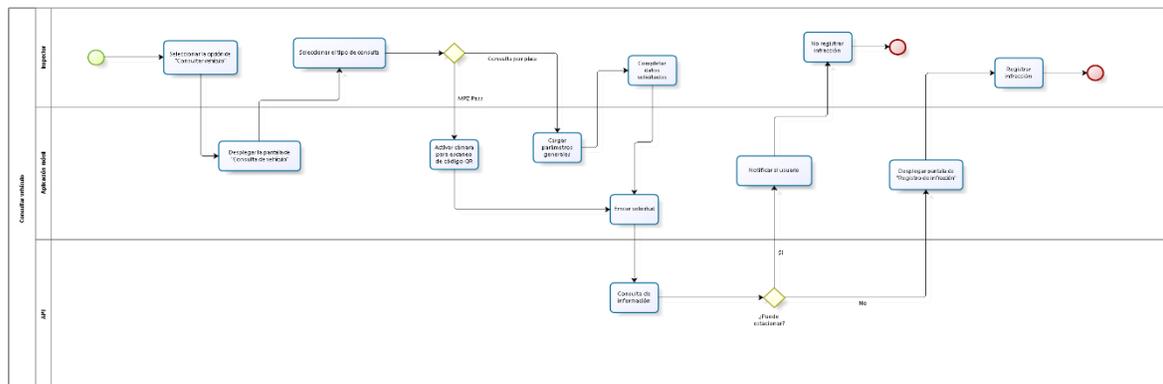


Figura 4. Diagrama de flujo para la consulta de vehículos. Fuente: Elaboración propia

3.3.4. Aplicación para registro de infracciones: Sincronización de infracciones

Para el proceso de sincronización de infracciones es necesario que el usuario, en primer lugar, ingrese a la aplicación y seleccione la opción de *Sincronización de infracciones*, la aplicación desplegará un listado de infracciones que presentaron algún inconveniente en el momento de su registro, por lo que fueron almacenadas de manera local en el dispositivo. El usuario visualizará este listado y podrá seleccionar las infracciones que desea sincronizar, por lo que, una vez seleccionada la infracción, la aplicación hará una solicitud de sincronización al API, el cual registrará la infracción con todos sus detalles y fotografías. Una vez registrada, se enviará una notificación al usuario con el estado de la transacción solicitada (registrada o error).

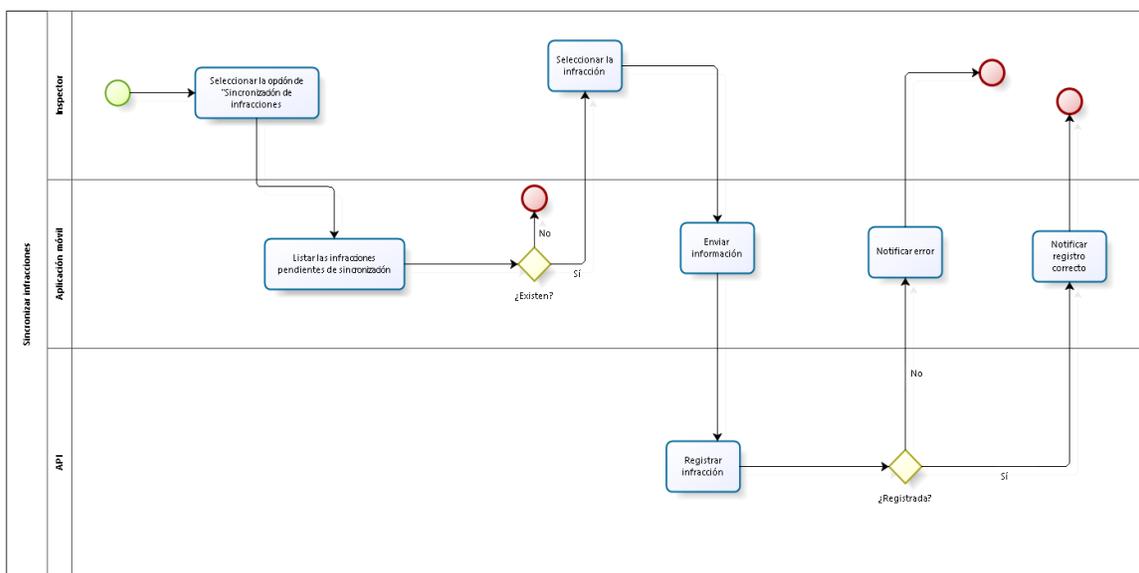


Figura 5. Diagrama de flujo para la sincronización de infracciones. Fuente: Elaboración propia

3.3.5. Aplicación para registro de infracciones: Registro de dispositivos

El registro de dispositivos es un módulo de la aplicación que permite registrar aquellos dispositivos que serán utilizados para las diferentes gestiones en la plataforma, esto para asegurar los datos almacenados. Para este proceso, el usuario ingresará a la aplicación, sin iniciar sesión, tocará el escudo de la municipalidad ubicado en la esquina superior derecha de la pantalla. Una vez seleccionada esta opción, la aplicación mostrará un formulario de credenciales, en el cual el usuario ingresará sus datos, los cuales serán consultados en la base de

datos; si el usuario cuenta con el permiso para registrar dispositivos, la aplicación tomará el número de IMEI del dispositivo y lo almacenará en la base de datos.

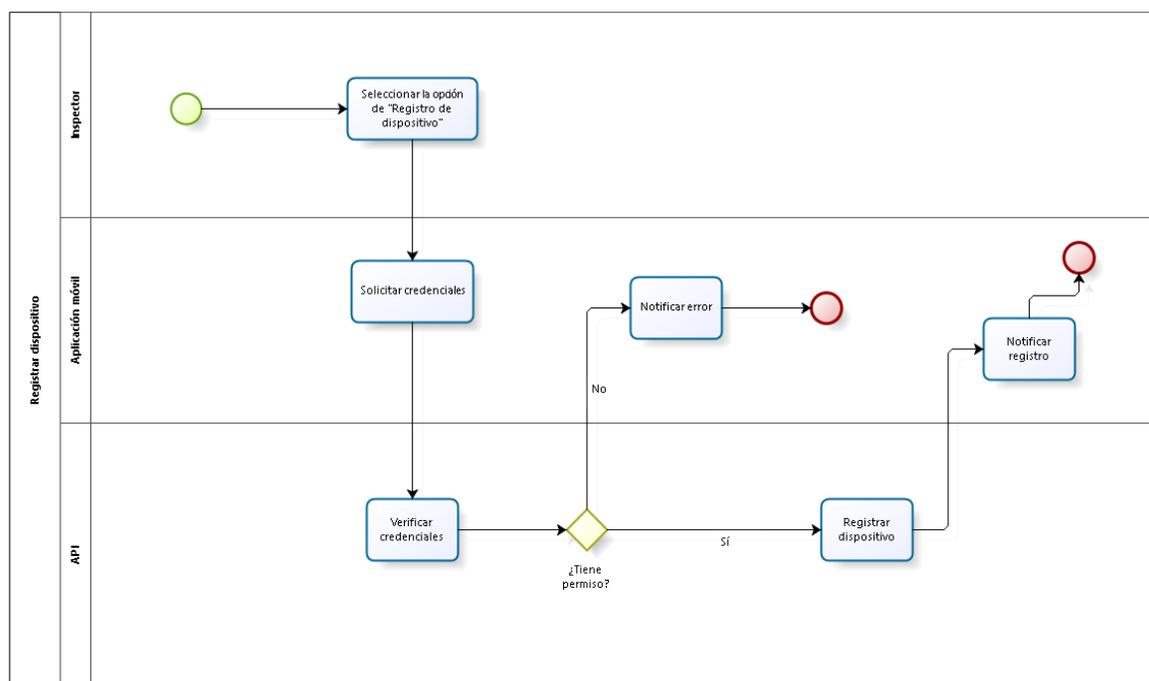


Figura 6. Diagrama de flujo para el registro de dispositivos. Fuente: Elaboración propia

3.3.6. Aplicación para registro de infracciones: Sincronización de parámetros

Generales

La sincronización de parámetros generales pretende que la información que se almacene de manera local en el dispositivo (información como tipos de vehículos, clases, colores, marcas, artículos, incisos, entre otros) sea igual a la almacenada en la base de datos principal. Cabe destacar que esta información se almacena de manera local para permitir agilidad en el proceso de registro de infracciones, ya que una consulta al API para obtener estos datos cada vez que se realiza una infracción ralentiza el proceso.

Por lo tanto, una vez que el usuario ingrese a la pantalla de inicio de la aplicación, esta enviará al API una solicitud para verificar si se encuentra sincronizado; el API tomará el número único del dispositivo (IMEI) y consultará la base de datos para conocer si se agregó, editó o eliminó algún registro existente; el API retornará un mensaje a la aplicación en el cual indicará si es necesario actualizar los datos, este mensaje será visualizado por el usuario quien decidirá

si puede realizar la sincronización de datos en ese momento; en caso afirmativo, la aplicación procederá a solicitar al API la información a sincronizar y la almacenará de manera local para que pueda utilizarla el inspector de manera inmediata.

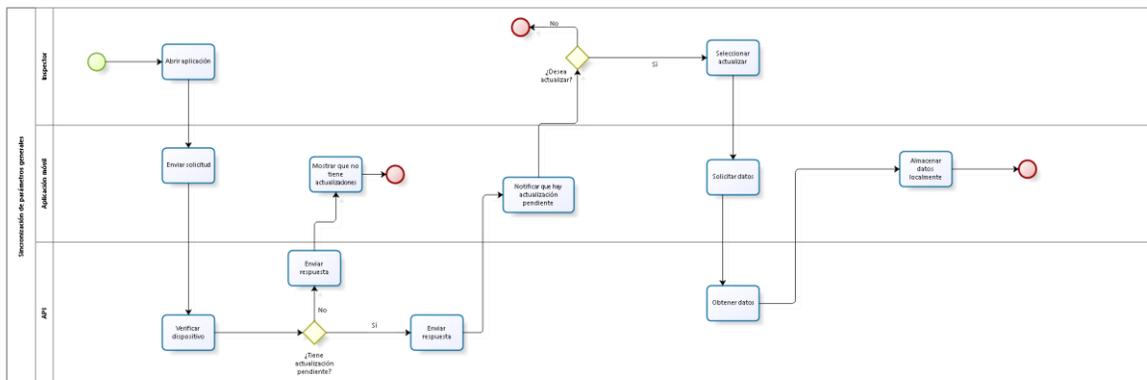


Figura 7. Diagrama de flujo para la sincronización de parámetros generales. Fuente: Elaboración propia

3.3.7. Aplicación para venta de tiempo para estacionamiento: Compra de tiempo

Como se mencionó anteriormente, la finalidad de esta aplicación es que cualquier persona pueda adquirir tiempo de estacionamiento desde cualquier lugar a través de la página web de la municipalidad. Por lo tanto, el proceso se inicia cuando el usuario se dispone a ingresar a la sección de la página para compra de tiempo, en este punto la aplicación solicitará al API parámetros generales como los tipos y clases de vehículos; después de la obtención de estos datos, la aplicación desplegará un formulario que el usuario debe completar; cuando los datos son ingresados y el usuario selecciona la opción de realizar la compra, la aplicación solicitará el registro de la información en la base de datos a través del API, la cual generará las claves necesarias para establecer la conexión bancaria.

Antes de que la aplicación redirija el sitio al portal de pagos del banco, se solicitará que el usuario acepte los términos y condiciones del pago; una vez confirmada, se redirigirá a la pasarela de pagos del Banco de Costa Rica, en la que se le solicitará al usuario los datos de la tarjeta con la que realizará el pago. La conexión bancaria retornará una respuesta a la aplicación de compra de tiempo, que captará esta respuesta y enviará a registrar a la base de datos por medio del API, la cual devolverá el estado de la transacción, para posteriormente notificar al usuario el estado final de la transacción solicitada.

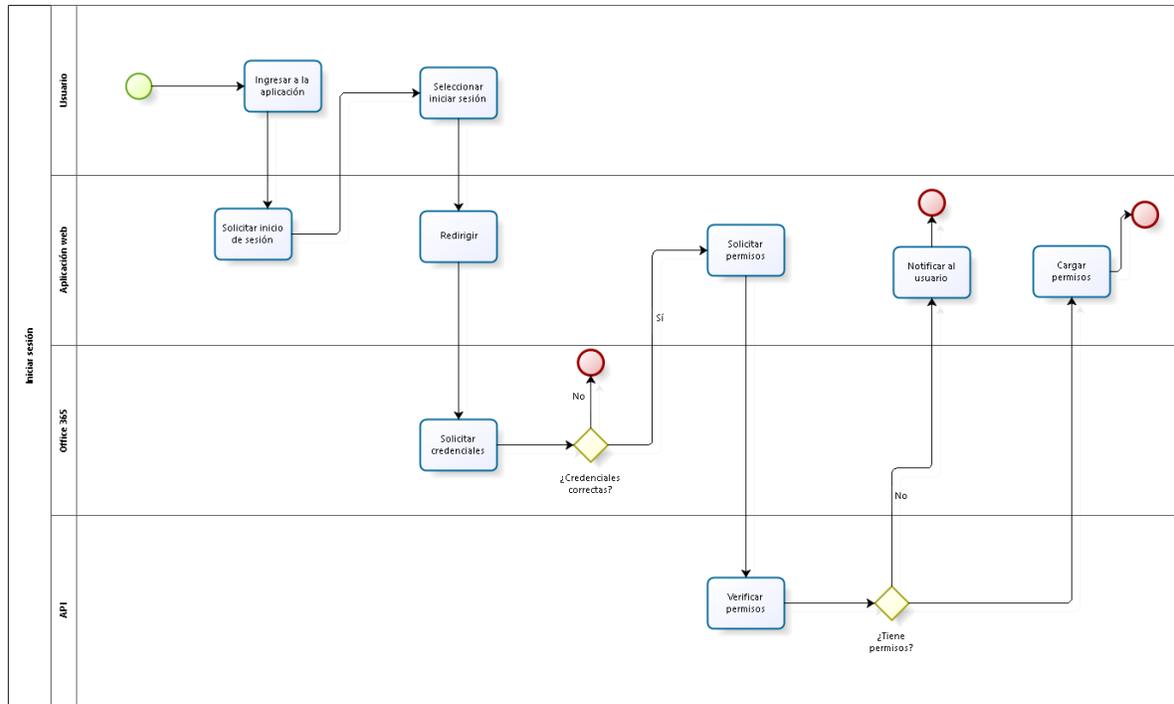


Figura 9. Diagrama de flujo para el inicio de sesión. Fuente: Elaboración propia

3.3.9. Sistema de administración PIMI: Registro de usuarios

Esta sección permitirá el registro de nuevos usuarios para diferentes funcionalidades en la plataforma. El proceso inicia cuando el usuario selecciona la opción de administración de usuarios, en la que se desplegará un listado con los usuarios existentes y la opción de agregar uno nuevo; en caso de que se elija la opción de agregar uno nuevo, la aplicación desplegará un formulario en el que se solicitará la información personal del usuario y la asignación de sus permisos. Luego de ingresar esta información, la aplicación verificará que los datos se encuentren de manera correcta y los enviará al API para que sea registrada en la base de datos; el API retornará una respuesta de confirmación o de error, lo cual será visualizado por el usuario a través de la aplicación.

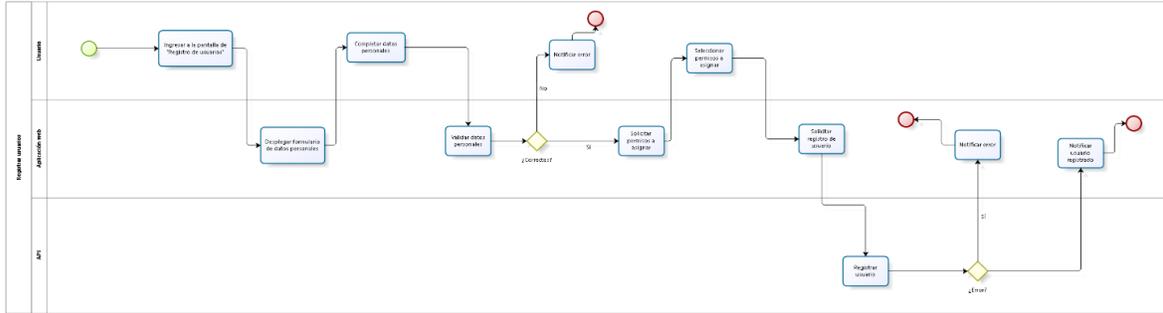


Figura 10. Diagrama de flujo para el registro de usuarios. Fuente: Elaboración propia

3.3.10. Sistema de administración PIMI: Edición de usuarios

La aplicación cuenta con un módulo que permite editar la información personal de cada usuario, así como gestionar sus permisos en la plataforma. Para iniciar el proceso, el usuario debe iniciar la aplicación y dirigirse a la sección de administración de usuarios, en la que la aplicación hará una solicitud de información al API que retornará un listado de todos los usuarios registrados, este listado lo visualizará el usuario, quien seleccionará el que desea editar, una vez que lo selecciona, la aplicación solicitará toda la información del usuario al API, la cual se visualizará en pantalla para que el usuario pueda editarla. Luego de editar y guardar la información, la aplicación enviará la solicitud de registro al API, el cual hará los cambios pertinentes en la base de datos.

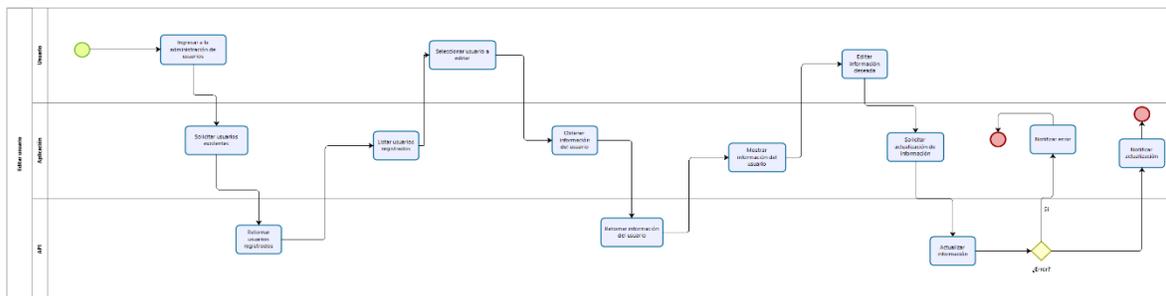


Figura 11. Diagrama de flujo para la edición de usuarios. Fuente: Elaboración propia

3.3.11. Sistema de administración PIMI: Listado y edición de infracciones

Una de las partes fundamentales de esta aplicación es la visualización de las infracciones realizadas filtradas por diferentes parámetros que se aplicarán a conveniencia del usuario. Este proceso inicia una vez que el usuario procede a acceder a la sección de *Consulta de infracciones*,

luego, la aplicación procederá a mostrarle tres opciones diferentes para filtrar las boletas: por número de boleta, datos del vehículo (número de placa, tipo y clase) o por inspectores estableciendo un rango de fechas y el nombre del inspector; una vez que el usuario selecciona el tipo de filtrado e ingresa la información que se le solicita, la aplicación procederá a realizar una consulta en el API, el cual obtendrá la información de la base de datos y la retornará, esta será presentada por la aplicación para que el usuario la visualice por medio de una tabla de datos paginada, en esta parte se cuenta con la opción de descarga del listado en formato de Excel para su manipulación.

En caso de que el usuario requiera visualizar la infracción con más detalle, podrá seleccionar cualquier infracción del listado con un doble clic, que desplegará otra pantalla con su información en la que podrá reimprimir la infracción, analizar las fotografías o visualizar su ubicación de registro en el mapa. Si el usuario cuenta con el permiso debido para editar infracciones, la aplicación mostrará un botón que al pulsarse redirigirá a una pantalla nueva en la que se podrán editar datos de la infracción como la clase, el tipo, el color, la marca y el artículo infringido; una vez que el usuario proceda a guardar la información, la aplicación enviará una solicitud de actualización de información al API, el cual establecerá la conexión a la base de datos para que la infracción quede actualizada; posteriormente, se notificará al usuario si la transacción se llevó a cabo de forma exitosa.

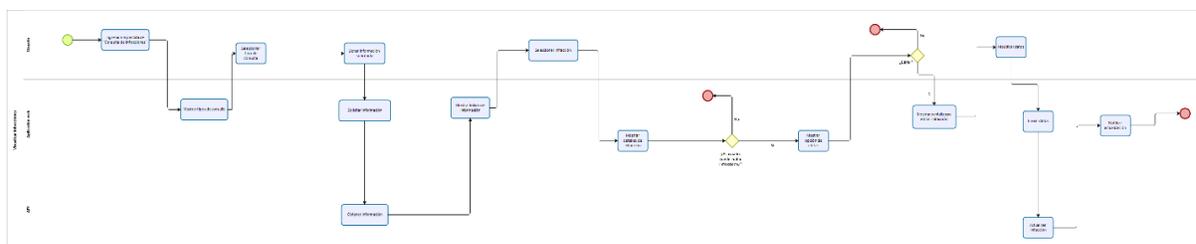


Figura 12. Diagrama de flujo para el listado y edición de infracciones. Fuente: Elaboración propia

3.3.12. Sistema de administración PIMI: Generación de archivo para SICSOA

Esta sección permite generar archivos para el SICSOA, el cual consiste en un sistema brindado por el INS para que las infracciones que no se han cancelado en la municipalidad se carguen en el derecho de circulación o marchamo anual de cada vehículo. Este módulo comienza su funcionamiento una vez que el usuario ingresa a la pantalla para la generación de reportes

para SICSOA, en el que la aplicación desplegará un formulario que le solicitará al usuario que seleccione un rango de fechas para el posterior filtrado de todas las infracciones realizadas entre los parámetros establecidos; una vez que se establecen estos parámetros, la aplicación envía una solicitud de información al API que, a la vez, obtiene la información desde la base de datos y que serán presentados a la aplicación para que sea descargado en un formato de archivo de texto plano con los datos que solicita el SICSOA.

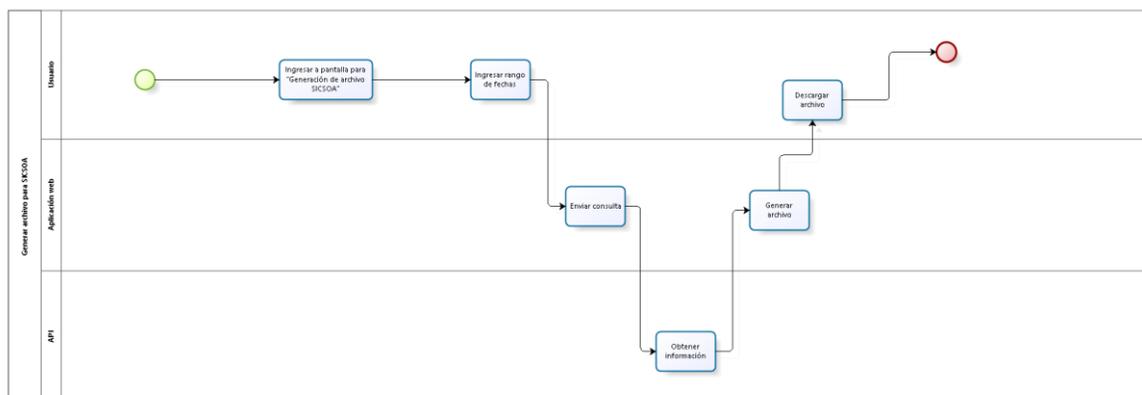


Figura 13. Diagrama de flujo para la generación de archivo para SICSOA. Fuente: Elaboración propia

3.3.13. Sistema de administración PIMI: Registro de vehículos para MPZ Pass

Entre las novedades que presenta esta plataforma se encuentra el MPZ Pass, el cual consiste en el registro de vehículos dentro de la municipalidad para que exista una clave única para cada vehículo con sus respectivos datos, lo que le facilitará al inspector la consulta de vehículos en el momento de realizar infracciones de tránsito. El registro de estos vehículos comienza cuando el usuario ingresa al módulo de registro de vehículos, en el que se desplegará un formulario que solicita información como tipo de vehículo, clase, número de placa, color, marca, nombre y cédula del propietario registral; una vez que el usuario ingrese estos datos, la aplicación solicitará al API el registro del vehículo a la base de datos y una vez que se realice la transacción se retornará una respuesta a la aplicación para que el usuario visualice si el registro se realizó de forma satisfactoria. En caso de que se haya registrado de forma exitosa, se mostrará en pantalla un código QR, el cual deberá imprimirse y entregarse al titular del vehículo.

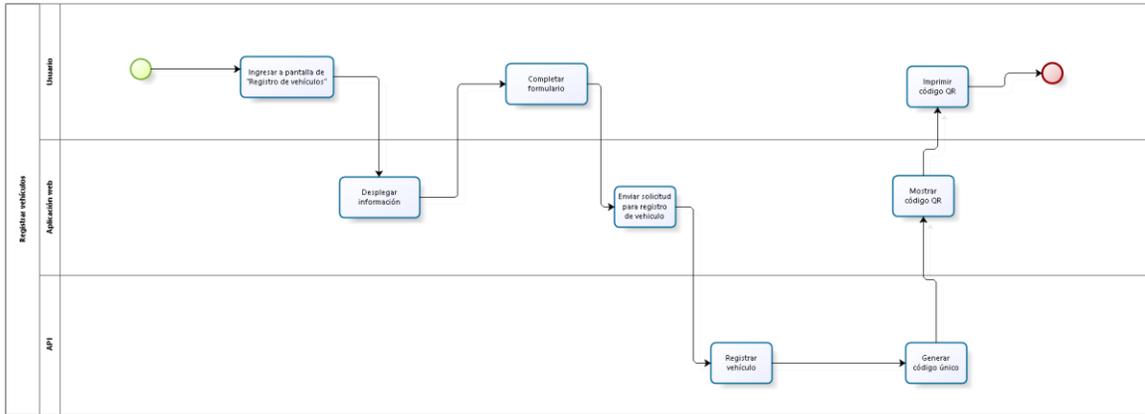


Figura 14. Diagrama de flujo para el registro de vehículos para MPZ Pass. Fuente: Elaboración propia

3.3.14. Sistema de administración PIMI: Localización de inspectores

Esta aplicación cuenta con una sección que permite visualizar las ubicaciones geográficas de los inspectores de tránsito en tiempo real en un mapa, esto con la finalidad de conocer el posicionamiento de cada inspector. El proceso inicia cuando el usuario decide ingresar al módulo de localización de inspectores, en el que la aplicación enviará una solicitud de información al API que, a la vez, hace una solicitud de información a la base de datos que obtendrá las últimas actualizaciones de cada uno de los inspectores, según su ubicación y la hora en que fue registrada; esta información la capta la aplicación y coloca los marcadores correspondientes a un mapa que se encuentra en la página consumiendo un API de Google. Esta consulta se realiza cada 30 segundos, por lo que la ubicación de cada inspector será muy precisa y se contará con información fiable y de primera mano.

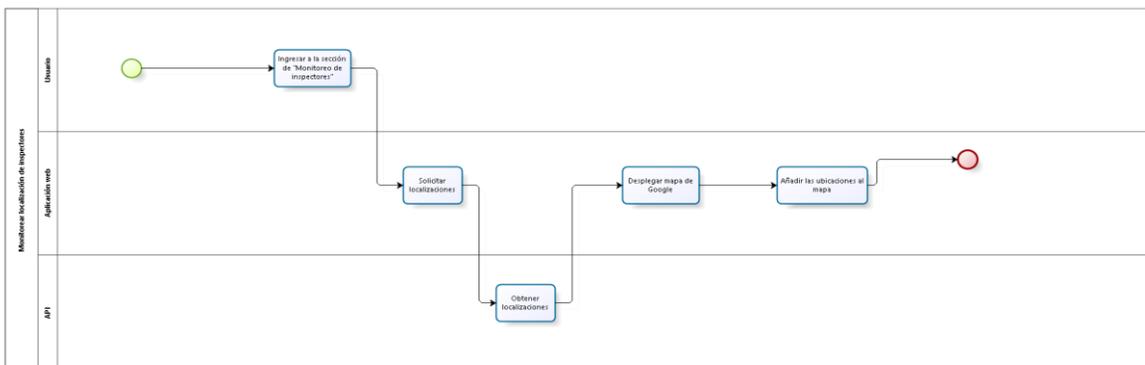


Figura 15. Diagrama de flujo para la localización geográfica de inspectores. Fuente: Elaboración propia

3.4. Estudio de Factibilidad

3.4.1. Factibilidad Técnica

Para determinar la factibilidad técnica con la que se contaba al inicio del proyecto, se tomaron en cuenta los siguientes aspectos:

3.4.1.1. Especificación de Hardware y Software de la Institución

La Municipalidad de Pérez Zeledón cuenta con los siguientes recursos:

Tabla 1. *Especificación de hardware y software de la institución*

<i>Hardware</i>			<i>Software</i>		
Tipo	Procesador	RAM (GB)	Almacenamiento (GB)	Sistema Operativo.	Otros.
Servidor	2 x Intel QuadCore Xeon X5450	16 DDR2	2048	Windows Server	SQL server 2002, Microsoft Access, Web API
Computadora	Core i5	4	500 HDD	Windows 10	Visual Studio, Android Studio, Netbeans.
Computadora	Core	2	500	Windows XP	Navegador web.
Dispositivo móvil (PDA)	Cortex-A7 1.3GHz quad core.	1	32 GB	Android 4.4.2 OS.	Controlador para bluetooth y cámara

					integrada.
Impresora portátil	N/A	N/A	N/A	N/A	Interfaz bluetooth e impresión térmica.

3.4.1.2. Herramientas de Desarrollo

Para la selección de herramientas de desarrollo, se mantuvo la línea de trabajo utilizada por la municipalidad y se aprovecharon las licencias que esta posee sobre *software* privativo como SQL Server, además, se incluyen herramientas con licencia de código abierto para el desarrollo de la aplicación de administración *web* y las aplicaciones móviles para inspecciones. A continuación, se muestra una tabla con las especificaciones de cada herramienta requerida:

Tabla 2. Especificación de herramientas de desarrollo

Herramienta	Tipo	Aplicación	Licencia
Node JS	Entorno de Desarrollo	Web y móvil	GPL
Visual Studio 2015 Community.	Entorno de Desarrollo	API	GPL
Visual Studio Code	Editor de HTML y CSS	Web	GPL
Microsoft Azure	Servidor <i>web</i> de seguridad en la nube	Web	Propietaria
SQL Server 2008 R2	Base de datos	Web y móvil	Propietaria
SQLite	Base de datos	Móvil	GPL

	móviles		
--	---------	--	--

3.4.1.3. Infraestructura a Nivel de Redes

La municipalidad cuenta con una red de conexiones de cableado estructurado y bajo los estándares más altos de certificaciones de este tipo. Además, cuenta con una seguridad perimetral muy amplia, con equipos de la marca Fortinet (<https://www.fortinet.com>), con el fin de ofrecer seguridad en la información que se aloja en las distintas bases de datos que se manejan en la institución. Además, la conexión al API de la plataforma se establecerá por medio de conexiones cifradas por SSL, lo que agrega una capa de seguridad, pues toda la información se encuentra encriptada.

3.4.1.4. Zonas de Cobertura de Redes Telefónicas para Dispositivos Móviles

El proyecto incluye el desarrollo de una plataforma móvil que se conecta mediante redes telefónicas al API de la plataforma para realizar los debidos procesos, por esta razón, contar con una amplia cobertura por parte de los proveedores como Kölbi, Movistar y Claro, brindará una alta viabilidad en el desempeño de la plataforma en general.

Mediante un mapa de las zonas de cobertura que brindan la Sutel para los proveedores mencionados, se logra visualizar que el cantón de Pérez Zeledón cuenta con una alta cobertura por parte de los tres proveedores, lo que permite una factibilidad de comunicación.

3.4.1.5. Conclusiones

Se concluye que la realización del proyecto es lo suficientemente viable. A continuación, se desglosa cada aspecto:

- La institución cuenta con el *hardware* y *software* necesario para el soporte y mantenimiento de las plataformas, tanto *web* como móvil, de manera que el proyecto cuenta con una infraestructura robusta, además, permite una importante escalabilidad del producto.

- En la sección de herramientas para el desarrollo del proyecto, la municipalidad brinda ciertas herramientas con licencias adquiridas anteriormente, se añade el uso de herramientas de uso libre, lo que permite el desarrollo del proyecto de manera económica.
- Con base en la infraestructura de las conexiones de red que presenta la municipalidad, se observa la robusta estructura con la que se cuenta, resguardada por la plataforma de ciberseguridad FortiNet, esto permite un manejo seguro de la información por la red.
- En el aspecto de zonas de cobertura de redes telefónicas para dispositivos móviles, se observa la viabilidad del desarrollo de herramientas móviles, al existir una amplia zona con conexión que permita el uso de la plataforma.

3.4.2. Factibilidad Operativa

La factibilidad operativa es la encargada de determinar las capacidades con las que se cuenta en la institución para mantener el sistema con un funcionamiento correcto y que se cumplan todos los objetivos planteados inicialmente. Esta sección se encuentra dividida en cuatro ramas diferentes: conocimientos de los desarrolladores, habilidades de los usuarios finales, disponibilidad de horarios y mantenimiento del sistema. A continuación, se detallarán más estos aspectos:

- Conocimientos de los desarrolladores: las habilidades y conocimientos del equipo desarrollador del *software* son vitales, ya que a partir de esto se puede asegurar que hay bases sólidas para llevar a cabo el proyecto y que este tenga el éxito que se requiere. Ambos ingenieros, Luis Barrantes Mora y Erick Flores Solís, cuentan con conocimientos amplios de desarrollo en plataformas *web* y móviles, siguiendo los estándares de calidad en cuanto a usabilidad. Además, es importante tomar en cuenta que el API de la municipalidad fue desarrollada por el señor Barrantes, por lo que en el equipo existe un conocimiento estructural de cómo funciona y las formas que existen para conectarse a este.
- Habilidades de los usuarios finales: según la indagación realizada a los

encargados de cada departamento involucrado en este proyecto (Seguridad Ciudadana, Patentes y Planificación Territorial), los inspectores ya están familiarizados con computadoras de escritorio y sistemas operativos como Windows XP, Windows 7 y Windows 10; además, según mencionaron, cuentan con conocimientos de herramientas de Office como Word y Excel. De acuerdo con la información mencionada anteriormente, queda claro que es vital desarrollar un *software* con base en los lineamientos de experiencia de usuario, esto para brindar una interfaz intuitiva que permita que la curva de aprendizaje sea lo más simple posible, debido a que no cuentan con conocimientos sobre manipulación de PDA. A partir de lo anterior, es importante destacar la vitalidad de una capacitación correcta a los usuarios finales para que se le dé un uso correcto a las herramientas a desarrollar.

- Disponibilidad de horarios: en este apartado se tomaron en cuenta los horarios de los desarrolladores del sistema; el ingeniero Luis Barrantes Mora trabaja en el departamento de Informática de la Municipalidad de Pérez Zeledón y trabaja de lunes a jueves de 7:00 a. m. a 4:00 p. m. y los viernes de 7:00 a. m. a 3:00 p. m. Por otra parte, el ingeniero Erick Flores Solís trabaja en el departamento de TI de Coopeagri R. L. y labora de lunes a viernes de 7:00 a. m. a 4:30 p. m. y cada dos semanas labora los sábados con un horario de 7:00 a. m. a 11:30 a. m. Esta información se refleja que el tiempo para desarrollar el *software* se limita a trabajar de lunes a viernes en horas de la noche y los fines de semana.
- Mantenimiento del *software*: lo que respecta al mantenimiento del sistema, es muy importante crear manuales de procedimientos claros tanto para los usuarios finales como para el departamento de Informática de la municipalidad, esto con el fin de evitar inconvenientes en el momento que se presente alguna eventualidad que implique el cambio del código fuente de la aplicación; además, es muy importante que exista una documentación interna en el código fuente del proyecto, esto para evitar confusiones cuando se deba hacer algún cambio. Finalmente, es primordial crear un sistema escalable que permita agregar funcionalidades nuevas que se adecúen a las necesidades de la municipalidad.

A partir de anterior, se logró concluir que el proyecto es operacionalmente factible, ya que el equipo de desarrollo cuenta con los conocimientos mínimos para llevarlo a cabo, además de tener disponibilidad de horarios para invertir el tiempo necesario que requiere el desarrollo del proyecto; también existen habilidades en los usuarios finales que les permitirá adaptarse al *software* de la manera deseable. Por último, el mantenimiento y soporte del *software* también será realizable y en coordinación con la institución.

3.4.3. Factibilidad Económica

3.4.3.1. Costo de Recurso Humano

El costo del recurso humano se estimó tomando en cuenta el valor de la hora para un bachiller universitario establecido por el Ministerio de Trabajo y Seguridad Social. Además, el tiempo estimado para el desarrollo del proyecto será de un año y se estima un total de 15 horas semanales por cada miembro del equipo de desarrollo.

Tabla 3. *Costo de recurso humano*

Miembro	Precio por Hora	Precio Mensual	Precio Anual
Erick Flores Solís	₡ 2,185.00	₡ 131,120.00	₡ 1,573,200.00
Luis Barrantes Mora	₡ 2,185.00	₡ 131,120.00	₡ 1,573,200.00
Total: ₡ 3, 146,400.00 colones.			

3.4.3.2. Tabla Comparativa de Gastos

La comparativa de gastos se realizó con referencia a un proyecto interno contra un proyecto universitario de licenciatura, que muestra el beneficio económico que presenta el desarrollo de este proyecto para la municipalidad. Solo se deberán cubrir costos adicionales para la adquisición de *hardware* como dispositivos móviles e impresoras. El costo del recurso humano no afectará los gastos para el presente proyecto, pues este es cubierto por los estudiantes

desarrolladores.

Tabla 4. *Comparativa de gastos*

Gasto a inquirir	Proyecto Interno	Proyecto de Licenciatura
Costos de Desarrollo	₪0.00	₪ 0.00
Costos de Recurso Humano	₪ 3,146,400.00	₪ 0.00
Costos Administrativos.	₪0.00	₪ 0.00
Total:	₪ 3,146,400.00	₪ 0.00

3.4.3.3. Conclusiones

Como conclusión, se observa que los costos que implica el desarrollo del proyecto los cubrieron, en su mayoría, los desarrolladores de la plataforma, lo que demuestra que existe una factibilidad económica bastante marcada.

3.5. Modelaje de Datos

El modelo de datos del sistema se encuentra en el Anexo 1, *Modelaje de Datos del sistema PIMI*.

3.6. Desarrollo

El desarrollo del proyecto se basó en la ejecución de *sprints*, los cuales especifican la elaboración de actividades con el fin de lograr los objetivos planteados.

A continuación, se detallan los *sprints* que permitieron el cumplimiento de los objetivos del proyecto:

Sistema de administración

- Inicio de sesión: se realiza el procedimiento de reconocimiento de usuarios

registrados en el sistema.

- Registro de usuarios: se realiza la funcionalidad de registro de usuarios para el sistema de administración.
- Edición de usuarios: se realiza el *sprint* que ejecuta la funcionalidad de carga y edición de usuarios del sistema.
- Listado de infracciones de tránsito: se ejecuta el *sprint* que permite listar las infracciones de tránsito registradas en el sistema de la Municipalidad.
- Generación de archivo (SICSOA): se realiza la funcionalidad que permite la generación del archivo (SICSOA), mediante el sistema de administración.
- Registro de vehículos MPZ Pass: este *sprint* permitió la elaboración del módulo de registro de vehículos en el sistema.
- Monitoreo de localización de inspectores: se desarrolla la funcionalidad que permite monitorear por medio del sistema de administración, la localización GPS de los inspectores registrados en el sistema.

Sistema de inspección de construcciones y patentes

- Autenticación de usuarios: se realiza el procedimiento de reconocimiento de usuarios registrados en el sistema de inspección de construcciones y patentes.
- Selección de módulos por usuarios: el *sprint* permite realizar una funcionalidad de control de los módulos que contiene el sistema con los usuarios.
- Carga del menú del sistema por usuario: el *sprint* permite la carga visual del menú del sistema, lo que posibilita la navegación intuitiva.
- Listado de permisos de construcción: se realiza la funcionalidad que permite consultar a la base de datos los permisos de construcción realizados por los usuarios.
- Registro de permisos de construcción: en el *sprint* se desarrolla el método que permite registrar un permiso de construcción mediante el ingreso de datos por

parte del usuario.

- Listado de inspecciones de patentes realizadas: se realiza la funcionalidad que permite consultar a la base de datos las inspecciones de patentes realizadas por los usuarios.
- Registro de acta de inspección (manual y croquis): el *sprint* permite desarrollar el método de registro de acta de inspección manual y el acta de inspección croquis.
- Registro de acta de liquidación de boletos, acta de decomiso y acta de tasado: el *sprint* permite desarrollar el método de registro de las actas de liquidación de boletos, acta de decomiso y de tasado.
- Registro de acta de determinación de categoría, acta de clausura y sellado: el *sprint* permite desarrollar el método de registro de las actas de determinación de categoría y el acta de clausura y sellado.
- Registro de acta de inspección de prevención: el *sprint* permite desarrollar el método de registro del acta de inspección de prevención.
- Desconexión de sesión del usuario: se desarrolla el método que cierra la sesión del usuario adecuadamente.

Sistema de venta de boletas de estacionamiento

- Compra en línea: el *sprint* permite el desarrollo del módulo de venta de boletas de estacionamiento mediante la plataforma *web* de la Municipalidad de Pérez Zeledón.

Sistema para registro de infracciones

- Inicio de sesión: se realiza el procedimiento de reconocimiento de usuarios registrados en el sistema de infracciones.
- Registrar infracción: el *sprint* permite el desarrollo del módulo de registro de infracciones en el sistema.

- Consulta de vehículos: se desarrolla la funcionalidad que permite la consulta de la información de los vehículos registrados en el sistema.
- Sincronización de infracciones: el *sprint* permite que el sistema brinde una solución al registro de infracciones que son realizadas sin comunicación con el servidor centralizado de la municipalidad, lo que permite una sincronización de estas infracciones una vez establecida la comunicación.
- Registro de dispositivo: por medio de este *sprint* se desarrolla la funcionalidad que permite que el sistema recolecte los datos del dispositivo en el que se encuentra alojado este sistema y se registra su información en la base de datos.
- Sincronización de parámetros: este *sprint* permite implantar la funcionalidad que posibilita sincronizar los parámetros de la base de datos con el sistema.

Capítulo 4. Conclusiones

Todos los objetivos que se plantearon al inicio del proyecto se alcanzaron de manera exitosa. A continuación, se detallarán diferentes puntos de conclusión al concluir el proyecto:

4.1. Agilización de procesos

Uno de los mayores retos que se presentaba al inicio del proyecto era el de aprovechar al máximo el tiempo laboral de los inspectores en la municipalidad, ya que el llenado de actas o infracciones de forma manuscrita resultaba lento y repetitivo. A través de las aplicaciones móviles desarrolladas, los inspectores tardarán menos tiempo en crear una infracción o acta debido a los diferentes mecanismos de autocompletado implementados en los formularios y su diseño intuitivo.

Además, se eliminó por completo el proceso de digitalización individual de actas e infracciones realizado anteriormente para su registro en la Municipalidad. Con el uso de las aplicaciones móviles, toda la información ingresada por el inspector será almacenada inmediatamente junto con sus respectivas fotografías captadas en el momento.

4.2. Accesibilidad

Un punto bastante alto dentro del desarrollo de esta plataforma es el involucramiento masivo de los pobladores del cantón de Pérez Zeledón, a través de la aplicación de venta de tiempo de estacionamiento en línea. Esta aplicación ofrece una nueva opción más rápida, sencilla y flexible al ciudadano que desee aparcarse su vehículo y no encuentre dónde adquirir su boleta de estacionamiento tradicional, o bien, que se le haya vencido el tiempo y se encuentre lejos del vehículo para renovar su boleta física. Este avance es un paso bastante importante para todo el cantón en su camino a ser una ciudad inteligente.

4.3. Seguridad

La seguridad de los datos siempre fue un tema prioritario dentro del planteamiento del proyecto debido a la sensibilidad de la información que se maneja dentro de un contexto estatal

como el de la Municipalidad. Finalmente, se logró asegurar la información de diferentes formas, asegurando su consistencia en una base de datos robusta y escalable.

También fue tomado en cuenta el ambiente en el cual serán utilizadas a diario las aplicaciones móviles y los inconvenientes de conexión a internet que existen en ciertas zonas del cantón, por lo que el sistema de sincronización permite que no se pierda ningún tipo de información por fallos de comunicación entre aplicación y servidor almacenando los datos de forma local para su posterior registro.

Finalmente, es importante destacar el mecanismo de registro de dispositivos que permite que sólo sea posible iniciar sesión en las aplicaciones móviles desde dispositivos previamente registrados por un usuario con rol de administrador.

4.4. Impacto ambiental

En nuestra actualidad existe una gran lucha ambiental que pretende frenar el cambio climático a través de diferentes mecanismos sostenibles como lo son el reciclaje y la disminución de los desechos que se emiten diariamente.

La plataforma desarrollada en el marco de esta investigación tiene también un impacto ambiental beneficioso para el cantón de Pérez Zeledón, ya que la reducción de papel por medio de la impresión de infracciones y actas de inspección ahorran recursos por el aprovechamiento del espacio que se obtiene al momento de imprimir el comprobante físico. Además, la copia para el inspector que se generaba anteriormente ya no será necesaria debido a que toda la información recaudada quedará registrada directamente en la base de datos de la Municipalidad.

4.5. Innovación

El avance tecnológico que representa la implementación de la plataforma desarrollada en este proyecto se puede establecer como un precedente en la búsqueda de convertir el cantón de Pérez Zeledón en una ciudad inteligente. Ya que la plataforma abre nichos que podrán ser aprovechados por las generaciones actuales y futuras para el desarrollo de infraestructuras tecnológicas que beneficien la gestión de procesos que generan un gran impacto en la economía y desarrollo de todo el país.

El desarrollo de aplicaciones móviles abre una gama de posibilidades nueva para la Municipalidad de Pérez Zeledón, ya que existen diferentes inspecciones de campo en otros departamentos que requieren de innovación como la alcanzada a través de esta plataforma, por lo que podrá funcionar como parámetro y guía para futuras implementaciones similares.

4.6. Aporte UNA a la sociedad

Uno de los puntos demostrados dentro de la investigación realizada, existe en la gran importancia del compromiso que tienen todos los centros educativos con la sociedad y el aporte que se puede llegar a materializar por medio de la buena comunicación y el encadenamiento de esfuerzos entre instituciones, impactando de forma muy positiva y significativa a la población de todo un cantón como es el caso de Pérez Zeledón a través de la plataforma desarrollada a través de este proyecto. El valor agregado con el que cuenta esta naturaleza de proyectos puede llegar a ser muy valioso para diferentes instituciones gubernamentales en todo el país, debido al gran ahorro de recursos que puede conllevar el desarrollo de soluciones de cualquier índole a través de proyectos universitarios.

4.7. Recomendaciones de los Autores

El producto final obtenido, como todo *software*, posee un ciclo de vida útil, el cual, además de planificación y desarrollo, contiene el despliegue del producto y su debido mantenimiento. Con base en lo anterior y con aspiraciones a obtener el máximo provecho posible del producto, se realizan las siguientes recomendaciones:

- Mantenimiento del producto: como parte del ciclo de vida de un producto *software*, se encuentra el mantenimiento adecuado que le permita ejecutar las funciones de manera correcta, por lo tanto, se recomienda su valoración constante, mantenimiento preventivo y correctivo.
- Capacitación adecuada: se recomienda el análisis respectivo de los manuales de usuarios brindados, con el fin de que el usuario del *software* obtenga el mayor provecho en la utilización de este.
- Promoción del producto: como parte del avance tecnológico que brinda el

desarrollo de este proyecto, se encuentra la promoción del sistema hacia los usuarios finales (compra de tiempo en línea), para que formen parte del avance y se logre una aceptación inmediata.

- Base de aprovechamiento tecnológico: en vista de que la implementación de este sistema tecnológico significa un avance importante para la Municipalidad en el desarrollo de sus labores cotidianas, se recomienda aprovechar los lineamientos que el sistema propone, con el fin de estandarizar y perfeccionar procesos.
- Mejora continua: se recomienda una mejora continua en lo que respecta al *software* desarrollado, con el fin de solventar cualquier necesidad en el futuro.

Capítulo 5. Bibliografía y Glosario

5.1. Bibliografía

Baah, A. (s. f.). *Agile software development*.

El Economista. (2014). Grupo Setex diseña la app de pago del parking de Teruel. Eva Sereno.

Recuperado de: <http://www.economista.es/aragon/noticias/6142643/10/14/Grupo-Setex-disena-la-app-de-pago-del-parking-de-Teruel.html#>

El País. (2010). ¿Qué es una ciudad inteligente? David Fernández. Recuperado de:

http://elpais.com/diario/2010/12/12/negocio/1292162603_850215.html

E-park. (2013). La Aplicación de los parquímetros. Recuperado de: <https://www.e-park.es>

Fortinet. (2017). Recuperado de: <https://www.fortinet.com/>

La Nación. (2013). Nuevo sistema de parquímetros digitales en San José opera desde este

lunes. Recuperado de: http://www.nación.com/archivo/Nuevo-parquímetros-digitales-San-Jose_0_1338466205.html

Madrigal Pana, J. (2012). Encuesta a establecimientos de la región Brunca.

Menzinsky, A., López, G. y Palacio, J. (2016). *Scrum Manager*, 2º ed.

Ministerio de Trabajo y Seguridad Social. (2017). Lista de salarios mínimos por ocupación

Año 2017. Recuperado de: http://www.mtss.go.cr/temas-laborales/salarios/lista_salarios_1_2017.pdf

Municipalidad de Escazú. (s. f.). Recuperado de: <http://www.escazu.go.cr/>

Municipalidad de Pérez Zeledón. (2017). Recuperado de: <http://mpz.go.cr/index.php>

PDA International. (2017). Recuperado de: <http://www.pdainternational.net>

Proyectos Ágiles. (2017). Recuperado de: <https://proyectosagiles.org/>

- Schwaber, K. y Sutherland, J. (2013). La Guía de SCRUM. 1° ed. Recuperado de:
<https://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf>.
- Scrum manager. (2017). Modelo original de Scrum para desarrollo de *software* - Scrum Manager BoK. Recuperado de: https://www.scrummanager.net/bok/index.php?title=Modelo_original_de_Scrum_para_desarrollo_de_software
- Sutel. (2017). Mapas de cobertura. Recuperado de: <http://mapas.sutel.go.cr/index.php>
- Vargas, G. (2017). Scrum-QA. Recuperado de: <http://scrum-qa.blogspot.com/>.
- VASS digital. (2017). SCRUM, desarrollo ágil por excelencia. Recuperado de:
<http://www.vassdigital.com/scrum-la-metodologia-de-desarrollo-agil-por-excelencia/>

5.2. Glosario de Términos

Glosario de términos	
Término	Descripción
SCRUM	Marco de trabajo para el desarrollo y el mantenimiento de productos complejos adaptativos y permite la entrega de productos del máximo valor posible productiva y creativamente (Schwaber-Sutherland, 2013).
Sprint	Bloque de tiempo de un mes o menos durante el cual se crea un incremento del producto <i>terminado</i> , utilizable y potencialmente desplegable (Schwaber-Sutherland, 2013).
Backlog	Lista de deseos del dueño del producto; existen backlogs del producto y del sprint (Schwaber-Sutherland, 2013).
PDA	Se trata de una agenda electrónica que cuenta con muchas funcionalidades similares a las de un teléfono móvil; además, permite la conexión de otros dispositivos externos como una impresora portátil.
Patentes comerciales	Según la Real Academia Española, una patente comercial se define como el título o despacho real para el goce de un empleo, honor o privilegio.

Permisos constructivos	Aquellos permisos externados por la municipalidad que certifican que un lugar es apto o no para realizar la construcción de alguna obra.
Infracción de parquímetro	Es aquella que recibe el propietario de un vehículo que se encuentra estacionado en un lugar que no es permitido o bien no cuenta con los requisitos necesarios para estacionar en ese sitio.
Boleta de estacionamiento	Permiso externado por la municipalidad que le permite a los dueños de vehículos estacionar en ciertos lugares; estas boletas establecen un límite de tiempo.
Comercio electrónico	Método de compraventa de bienes a través de una plataforma electrónica, tal como una página <i>web</i> o una aplicación móvil.
API o Web Service	Plataforma <i>web</i> que permite consultar la información de una o varias bases de datos ubicadas en distintas partes.
SMS	Mensajes de texto telefónicos
MEIC	Ministerio de Economía Industria y Comercio.
PNUD	Programa de las Naciones Unidas para el Desarrollo.

End-point	Función pública que se consulta en el API. Estas funciones permiten hacer llamadas al servidor directamente.
------------------	--

Capítulo 6. Anexos

6.1. Anexo 1

Modelaje de datos. Este archivo se encuentra anexo en un documento externo con el nombre Modelaje de datos.

6.2. Anexo 2

Manuales de Usuario

Los manuales de usuario del presente proyecto se encuentran listados a continuación, con su respectiva información:

Nombre	Descripción
Parquímetros – Manual de usuario	Contiene un manual de uso para la aplicación móvil para el registro de infracciones a la Ley de parquímetros.
Patentes y PUCC – Manual de usuario	Contiene un manual de la aplicación móvil para el registro de actas de inspección.
SSCMPZ – Manual de usuario	Contiene un manual de usuario para usar la aplicación <i>web</i> de administración y generación de reportes.
Tiempo de estacionamiento – Manual de usuario	Contiene un manual de uso para la aplicación <i>web</i> de compra de tiempo

	para estacionamiento en línea.
--	--------------------------------

6.3. Anexo 3

Manuales técnicos

Los manuales técnicos para desarrolladores de este proyecto se encuentran listados a continuación, con su respectiva información:

Nombre	Descripción
Parquímetros – Manual de usuario	Contiene un manual especializado para la aplicación móvil de registro de infracciones a la Ley de parquímetros.
Patentes y PUCC – Manual de usuario	Contiene un manual técnico de la aplicación móvil para el registro de actas de inspección.
SSCMPZ – Manual de usuario	Contiene un manual especializado para conocer el funcionamiento de la aplicación <i>web</i> de administración y generación de reportes.
Tiempo de estacionamiento – Manual de usuario	Contiene un manual técnico para la aplicación <i>web</i> de compra de tiempo para estacionamiento en línea.

6.4. Anexo 4

Plan de Trabajo (Cronograma de Actividades)

	Nombre	Inicio	Duracion	Terminado	Pre...
1	☐PIMI	4/09/17 8:00	327 days	4/12/18 17:00	
2	☐Planificación	4/09/17 8:00	37 days	24/10/17 17:00	
3	Cronograma	4/09/17 8:00	5 days	8/09/17 17:00	
4	Definición de tecnología	11/09/17 8:00	3 days	13/09/17 17:00	3
5	Definición y especificación de requerimientos	14/09/17 8:00	5 days	20/09/17 17:00	4
6	Creación de plan de riesgos	21/09/17 8:00	5 days	27/09/17 17:00	5
7	Desarrollo de backlog completo	3/10/17 8:00	5 days	9/10/17 17:00	6
8	Diseño y arquitectura	10/10/17 8:00	5 days	16/10/17 17:00	7
9	Revisión del backlog	17/10/17 8:00	4 days	20/10/17 17:00	8
10	Reunión de revisión	23/10/17 8:00	2 days	24/10/17 17:00	9
11	☐Desarrollo	13/12/17 8:00	254 days	3/12/18 17:00	2
12	☐Sistema de infracciones	13/12/17 8:00	76 days	28/03/18 17:00	
13	☐Sprint #1: Autenticación de usuario	13/12/17 8:00	9 days	25/12/17 17:00	10
14	Desarrollo de sprint	13/12/17 8:00	5 days	19/12/17 17:00	10
15	Envoltura de sprint	20/12/17 8:00	1 day	20/12/17 17:00	14
16	Revisión de sprint	21/12/17 8:00	1 day	21/12/17 17:00	15
17	Ajustes de sprint	22/12/17 8:00	1 day	22/12/17 17:00	16
18	☐Fase de pruebas	25/12/17 8:00	1 day	25/12/17 17:00	17
19	Pruebas de calidad	25/12/17 8:00	1 day	25/12/17 17:00	
20	Pruebas de usuario	25/12/17 8:00	1 day	25/12/17 17:00	
21	Pruebas de integración	25/12/17 8:00	1 day	25/12/17 17:00	
22	☐Sprint #2: Registro de infracción	26/12/17 8:00	16 days	16/01/18 17:00	10
23	Desarrollo de sprint	26/12/17 8:00	12 days	10/01/18 17:00	21
24	Envoltura de sprint	11/01/18 8:00	1 day	11/01/18 17:00	23
25	Revisión de sprint	12/01/18 8:00	1 day	12/01/18 17:00	24
26	Ajustes de sprint	15/01/18 8:00	1 day	15/01/18 17:00	25
27	☐Fase de pruebas	16/01/18 8:00	1 day	16/01/18 17:00	26
28	Pruebas de calidad	16/01/18 8:00	1 day	16/01/18 17:00	
29	Pruebas de usuario	16/01/18 8:00	1 day	16/01/18 17:00	
30	Pruebas de integración	16/01/18 8:00	1 day	16/01/18 17:00	
31	☐Sprint #3: Consulta de vehículos por placa	17/01/18 8:00	14 days	5/02/18 17:00	22
32	Desarrollo de sprint	17/01/18 8:00	10 days	30/01/18 17:00	30
33	Envoltura de sprint	31/01/18 8:00	1 day	31/01/18 17:00	32
34	Revisión de sprint	1/02/18 8:00	1 day	1/02/18 17:00	33
35	Ajustes de sprint	2/02/18 8:00	1 day	2/02/18 17:00	34
36	☐Fase de pruebas	5/02/18 8:00	1 day	5/02/18 17:00	35
37	Pruebas de calidad	5/02/18 8:00	1 day	5/02/18 17:00	
38	Pruebas de usuario	5/02/18 8:00	1 day	5/02/18 17:00	
39	Pruebas de integración	5/02/18 8:00	1 day	5/02/18 17:00	
40	☐Sprint #4: Consulta de vehículo por MPZ Pass	6/02/18 8:00	14 days	23/02/18 17:00	22
41	Desarrollo de sprint	6/02/18 8:00	10 days	19/02/18 17:00	39
42	Envoltura de sprint	20/02/18 8:00	1 day	20/02/18 17:00	41
43	Revisión de sprint	21/02/18 8:00	1 day	21/02/18 17:00	42
44	Ajustes de sprint	22/02/18 8:00	1 day	22/02/18 17:00	43
45	☐Fase de pruebas	23/02/18 8:00	1 day	23/02/18 17:00	44
46	Pruebas de calidad	23/02/18 8:00	1 day	23/02/18 17:00	
47	Pruebas de usuario	23/02/18 8:00	1 day	23/02/18 17:00	
48	Pruebas de integración	23/02/18 8:00	1 day	23/02/18 17:00	
49	☐Sprint #5: Registro y sincronización de infracciones off-line	26/02/18 8:00	14 days	15/03/18 17:00	40
50	Desarrollo de sprint	26/02/18 8:00	10 days	9/03/18 17:00	48
51	Envoltura de sprint	12/03/18 8:00	1 day	12/03/18 17:00	50
52	Revisión de sprint	13/03/18 8:00	1 day	13/03/18 17:00	51
53	Ajustes de sprint	14/03/18 8:00	1 day	14/03/18 17:00	52
54	☐Fase de pruebas	15/03/18 8:00	1 day	15/03/18 17:00	53
55	Pruebas de calidad	15/03/18 8:00	1 day	15/03/18 17:00	
56	Pruebas de usuario	15/03/18 8:00	1 day	15/03/18 17:00	
57	Pruebas de integración	15/03/18 8:00	1 day	15/03/18 17:00	
58	☐Sprint #6: Registro de dispositivos	16/03/18 8:00	9 days	28/03/18 17:00	40
59	Desarrollo de sprint	16/03/18 8:00	5 days	22/03/18 17:00	57
60	Envoltura de sprint	23/03/18 8:00	1 day	23/03/18 17:00	59
61	Revisión de sprint	26/03/18 8:00	1 day	26/03/18 17:00	60
62	Ajustes de sprint	27/03/18 8:00	1 day	27/03/18 17:00	61
63	☐Fase de pruebas	28/03/18 8:00	1 day	28/03/18 17:00	62
64	Pruebas de calidad	28/03/18 8:00	1 day	28/03/18 17:00	
65	Pruebas de usuario	28/03/18 8:00	1 day	28/03/18 17:00	
66	Pruebas de integración	28/03/18 8:00	1 day	28/03/18 17:00	

	Nombre	Inicio	Duracion	Terminado	Pre...
141	☐Sprint #6: Registro de acta de inspección con croquis	20/08/18 8:00	14 days	6/09/18 17:00	132
142	Desarrollo de sprint	20/08/18 8:00	10 days	31/08/18 17:00	
143	Envoltura de sprint	3/09/18 8:00	1 day	3/09/18 17:00	142
	Nombre	Inicio	Duracion	Terminado	Pre...
205	☐Sprint #2: Reporte por infracciones de tránsito realizadas	24/10/18 8:00	9 days	5/11/18 17:00	196
206	Desarrollo de sprint	24/10/18 8:00	5 days	30/10/18 17:00	
207	Envoltura de sprint	31/10/18 8:00	1 day	31/10/18 17:00	206
208	Revisión de sprint	1/11/18 8:00	1 day	1/11/18 17:00	207
209	Ajustes de sprint	2/11/18 8:00	1 day	2/11/18 17:00	208
210	☐Fase de pruebas	5/11/18 8:00	1 day	5/11/18 17:00	209
211	Pruebas de calidad	5/11/18 8:00	1 day	5/11/18 17:00	
212	Pruebas de usuario	5/11/18 8:00	1 day	5/11/18 17:00	
213	Pruebas de integración	5/11/18 8:00	1 day	5/11/18 17:00	
214	☐Sprint #3: Generación de archivos para SICSOA	24/10/18 8:00	12 days	8/11/18 17:00	196
215	Desarrollo de sprint	24/10/18 8:00	8 days	2/11/18 17:00	
216	Envoltura de sprint	5/11/18 8:00	1 day	5/11/18 17:00	215
217	Revisión de sprint	6/11/18 8:00	1 day	6/11/18 17:00	216
218	Ajustes de sprint	7/11/18 8:00	1 day	7/11/18 17:00	217
219	☐Fase de pruebas	8/11/18 8:00	1 day	8/11/18 17:00	218
220	Pruebas de calidad	8/11/18 8:00	1 day	8/11/18 17:00	
221	Pruebas de usuario	8/11/18 8:00	1 day	8/11/18 17:00	
222	Pruebas de integración	8/11/18 8:00	1 day	8/11/18 17:00	
223	☐Sprint #4: Registro de vehículos para uso de MPZ Pass	9/11/18 8:00	8 days	20/11/18 17:00	214
224	Desarrollo de sprint	9/11/18 8:00	4 days	14/11/18 17:00	
225	Envoltura de sprint	15/11/18 8:00	1 day	15/11/18 17:00	224
226	Revisión de sprint	16/11/18 8:00	1 day	16/11/18 17:00	225
227	Ajustes de sprint	19/11/18 8:00	1 day	19/11/18 17:00	226
228	☐Fase de pruebas	20/11/18 8:00	1 day	20/11/18 17:00	227
229	Pruebas de calidad	20/11/18 8:00	1 day	20/11/18 17:00	
230	Pruebas de usuario	20/11/18 8:00	1 day	20/11/18 17:00	
231	Pruebas de integración	20/11/18 8:00	1 day	20/11/18 17:00	
232	☐Sprint #5: Monitoreo de localización de inspectores de parquímetros	9/11/18 8:00	8 days	20/11/18 17:00	214
233	Desarrollo de sprint	9/11/18 8:00	4 days	14/11/18 17:00	
234	Envoltura de sprint	15/11/18 8:00	1 day	15/11/18 17:00	233
235	Revisión de sprint	16/11/18 8:00	1 day	16/11/18 17:00	234
236	Ajustes de sprint	19/11/18 8:00	1 day	19/11/18 17:00	235
237	☐Fase de pruebas	20/11/18 8:00	1 day	20/11/18 17:00	236
238	Pruebas de calidad	20/11/18 8:00	1 day	20/11/18 17:00	
239	Pruebas de usuario	20/11/18 8:00	1 day	20/11/18 17:00	
240	Pruebas de integración	20/11/18 8:00	1 day	20/11/18 17:00	
241	☐Sprint #6: Administración de parámetros generales	21/11/18 8:00	8 days	30/11/18 17:00	232
242	Desarrollo de sprint	21/11/18 8:00	4 days	26/11/18 17:00	
243	Envoltura de sprint	27/11/18 8:00	1 day	27/11/18 17:00	242
244	Revisión de sprint	28/11/18 8:00	1 day	28/11/18 17:00	243
245	Ajustes de sprint	29/11/18 8:00	1 day	29/11/18 17:00	244
246	☐Fase de pruebas	30/11/18 8:00	1 day	30/11/18 17:00	245
247	Pruebas de calidad	30/11/18 8:00	1 day	30/11/18 17:00	
248	Pruebas de usuario	30/11/18 8:00	1 day	30/11/18 17:00	
249	Pruebas de integración	30/11/18 8:00	1 day	30/11/18 17:00	
250	☐Sprint #7: Generación de reportes	21/11/18 8:00	9 days	3/12/18 17:00	232
251	Desarrollo de sprint	21/11/18 8:00	5 days	27/11/18 17:00	
252	Envoltura de sprint	28/11/18 8:00	1 day	28/11/18 17:00	251
253	Revisión de sprint	29/11/18 8:00	1 day	29/11/18 17:00	252
254	Ajustes de sprint	30/11/18 8:00	1 day	30/11/18 17:00	253
255	☐Fase de pruebas	3/12/18 8:00	1 day	3/12/18 17:00	254
256	Pruebas de calidad	3/12/18 8:00	1 day	3/12/18 17:00	
257	Pruebas de usuario	3/12/18 8:00	1 day	3/12/18 17:00	
258	Pruebas de integración	3/12/18 8:00	1 day	3/12/18 17:00	
259	Cierre	4/12/18 8:00	1 day	4/12/18 17:00	250
202	Pruebas de calidad	23/10/18 8:00	1 day	23/10/18 17:00	
203	Pruebas de usuario	23/10/18 8:00	1 day	23/10/18 17:00	
204	Pruebas de integración	23/10/18 8:00	1 day	23/10/18 17:00	

