

**Universidad Tecnológica Nacional de Córdoba**  
**Ingeniería en Sistemas de Información**



**Documentación de Proyecto**

**Cátedra:** Habilitación Profesional

**Curso:** 4k4

**Integrantes:**

- Mezi, Mariano Gabriel                      Legajo 58508
- Pianello, Martín                              Legajo 59631

**Docentes:**

- Adjunto: Zohil, Julio César Nelson
- JTP: Aquino, Francisco Alejandro

FACULTAD REGIONAL CORDOBA Ing. en Sistemas de Información Habilitación Profesional			
<b>CÁTEDRA</b>	<b>Habilitación Profesional</b>		
<b>ÁMBITO DE APLICACIÓN</b>	Ejido municipal de la ciudad de Córdoba, Córdoba, Argentina		
<b>NOMBRE DEL SOFTWARE</b>	ParCo		
<b>OBJETIVO</b>	Control medido y tarifado de estacionamientos sobre la vía pública		
<b>METODOLOGÍA</b>	Kanban		
<b>EQUIPO DE TRABAJO</b>	<b>LEGAJO</b>	<b>APELLIDO, NOMBRES</b>	<b>Mail De Contacto</b>
	58508	Mezi, Mariano Gabriel	<a href="mailto:marianomezi@hotmail.com">marianomezi@hotmail.com</a>
	59631	Pianello, Martin	<a href="mailto:martinpianello@gmail.com">martinpianello@gmail.com</a>
<b>CURSO</b>	4K4		
<b>DOCENTES</b>	Adjunto: Zohil, Julio César Nelson JTP: Aquino, Francisco Alejandro		
<b>AÑO DE CURSADO</b>	2019		
<b>FECHA DE EXAMEN</b>	04/02/2020	<b>NOTA FINAL</b>	APR. DIRECTA – 9 (NUEVE)

<b>Estudio inicial</b>	<b>7</b>
Introducción	7
Propuesta del sistema informático	8
Objetivo del producto	8
Objetivo del proyecto	8
Alcance del sistema	8
Objetivo grupal	9
Temática	10
Problemática	10
Identificación de Oportunidad	10
Estudio de mercado	11
Condiciones iniciales	11
Marco legal	11
Aplicaciones Similares	14
Marco del proyecto	16
Metodología	16
Roles	17
User Stories	17
Calendarización	18
Estudio de factibilidad	21
Técnica	21
Operativa	24
Económica	25
<b>Plan de gestión de configuración</b>	<b>28</b>
Administración de entregables	28
Entregables	28
Identificación de Artefactos	28
Número de versión	28
Control de versiones	29
Repositorio de código	29
Dependencias	30
Repositorios de dependencias	30
<b>Plan de pruebas</b>	<b>31</b>
Objetivo	31
Estrategia de Pruebas	32

Criterios de inicio	32
Criterios de finalización	33
Niveles de Pruebas	35
Pruebas unitarias	35
Pruebas de integración	35
Pruebas de sistema	35
Pruebas de aceptación	35
Ciclos de pruebas	36
Planificación De Pruebas	37
Definición de Ambientes de Pruebas	38
Ambientes locales	38
Ambientes de testing	38
Ambiente productivo	38
Procedimiento pasaje entre ambientes	38
Niveles de Severidad de defectos	40
Plantillas	41
Plantilla de casos de prueba	41
Plantilla de ejecución de caso de prueba	42
Plantilla de defectos	42
<b>Análisis de Factibilidad</b>	<b>44</b>
Técnica	44
Requerimientos de Hardware	44
Requerimientos de Software	45
Conectividad	45
Configuración	45
Almacenamiento	45
Operativa	46
Cliente Conductor	46
Cliente Punto de carga	46
Cliente Policía de tránsito	46
Cliente Administrador	46
Personal de mantenimiento	47
Económica	48
Costos del Producto	48
Conclusión de factibilidad	49

<b>Informe de avances</b>	<b>51</b>
Iteración 1	51
Alcance	51
Resultados	51
Retrospectiva	51
Iteración 2	52
Alcance	52
Resultados	52
Retrospectiva	53
Iteración 3	55
Alcance	55
Resultados	56
Retrospectiva	57
Iteración 4	58
Alcance	58
Resultados	59
Retrospectiva	61
Iteración 5	62
Product Backlog	62
Alcance	62
Objetivo	63
Resultados	64
Retrospectiva	65
Iteración 6	66
Product Backlog	66
Alcance	67
Resultados	68
Retrospectiva	71
Iteración 7	72
Product Backlog	72
Alcance	73
Resultados	74
Retrospectiva	74
Iteración 8	75
Product backlog	75

Alcance	76
Resultados	77
Retrospectiva	78
Iteración 9	79
Product Backlog	79
Alcance	80
Resultados	81
Retrospectiva	82
Iteración 10	83
Product Backlog	83
Alcance	83
Resultados	84
Retrospectiva	85
Iteración 11	86
Product Backlog	86
Alcance	86
Resultados	87
Retrospectiva	88
Iteración 12	89
Product Backlog	89
Alcance	89
Resultados	90
Retrospectiva	91
Iteración 13	92
Product Backlog	92
Alcance	92
Resultados	92
Retrospectiva	93
<b>Anexo</b>	<b>94</b>
Marco legal	94
Links a Aplicaciones similares	94
Integrantes del Equipo	95

# Estudio inicial

## Introducción

Durante el presente trabajo, nos disponemos a abordar el objetivo de llevar a cabo un proyecto propio, valiéndonos de los conocimientos y experiencias adquiridas a lo largo de la carrera. Observando una problemática general, la cual será explicada en detalle en los sucesivos puntos que conforman el presente trabajo, decidimos buscar una solución efectiva a la misma, que sea ventajosa por sobre las existentes actualmente en el mercado. Es de nuestro conocimiento la dificultad adicional que supone la construcción de un producto por sobre un desarrollo a medida, pero es una característica propia de nuestro grupo de trabajo la pasión que nos suponen los desafíos cuando nos resultan interesantes, y consideramos también que contamos con los conocimientos necesarios para poder sortear estas dificultades como así también la experiencia técnica que nos permitirá realizar un producto robusto y completo.

Esperamos que, al concluir nuestro proyecto, el mismo cumpla con los requerimientos de la cátedra, como así también nos sirva a nivel personal para poder aplicar los conceptos adquiridos a un caso real, pudiendo tener un contacto práctico con la gestión integral de un producto, desde su definición hasta su construcción y mantenimiento.

## Propuesta del sistema informático

### Objetivo del producto

Proporcionar a los conductores de automóvil una plataforma donde pagar (con medios digitales), consultar tiempo restante, extender plazo y brindar información miscelánea sobre el estacionamiento en las calles habilitadas. Además, permitir a los policías de tránsito consultar la vigencia del estacionamiento de un automóvil. Por último, se permite configurar las cuadras habilitadas para estacionamiento, así como el precio del mismo por unidad de tiempo.

### Objetivo del proyecto

Brindar una solución económica y de fácil implementación a la problemática actual que representa el obsoleto sistema de parquímetros implementado en los principales municipios de la provincia de Córdoba; a través de una alternativa escalable desde un reducido grupo de usuarios hasta una gran cantidad (suponiendo un crecimiento paulatino de su uso). Por último, tener en cuenta las necesidades y problemáticas de todos los potenciales usuarios en ámbitos urbanos.

### Alcance del sistema

Es conveniente subdividir el alcance en los diferentes actores del sistema para mejorar la comprensión del mismo desde sus diferentes perspectivas:

- Conductor
  - Consultar calles permitidas para el estacionamiento público y su respectivo costo de estacionamiento por fracción de tiempo.
  - Registrar el estacionamiento físico de un vehículo asociado a una ubicación específica.
  - Permitir el pago con diferentes medios de pago electrónicos.
- Policía de tránsito
  - Consultar la vigencia del estacionamiento de un vehículo.
- Punto de carga
  - Posibilitar el registro de estacionamiento de forma anónima.
  - Posibilitar el registro de estacionamiento en una ubicación diferente a la real del punto de carga.
- Administrador del sistema
  - Configurar tramos habilitados, prohibidos y gratuitos de en las calles de la ciudad.
  - Registrar una tarifa única de estacionamiento por fracción de tiempo.
  - Brindar Información para favorecer la toma de decisiones a través de reportes.



## Objetivo grupal

Como equipo, buscamos una necesidad no resuelta en nuestra vida cotidiana que suponga un dominio de problema desafiante a resolver aplicando tecnologías de punta; queremos investigar y formar experiencia en la propuesta y el uso de una arquitectura que consideramos soporta el dominio planteado. Además, queremos aplicar una metodología ágil que nos permita guiar el ciclo de vida del proyecto permitiéndonos enfocarnos y aplicar mayor esfuerzo en las cuestiones que consideramos son importantes a la hora de hacer un producto, contando siempre con el feedback necesario para saber qué grado de avance presenta el proyecto, y para poder tomar decisiones sobre cómo continuar.

## Temática

### Problemática

El sistema de tarifado de estacionamiento público manejado por parquímetros ya es obsoleto. Los problemas o desventajas más importantes de estos dispositivos son:

- Indisponibilidad del servicio por averías.
- Necesidad de disponer de monedas (en pesos argentinos).
- Costo de mantenimiento y consumo eléctrico.
- Ocupación de espacio físico.
- Necesidad de pago en el lugar.
- Indisponibilidad de datos de registro de estacionamiento confiables en tiempo real.

Por esto, consideramos que hay una necesidad insatisfecha que da lugar a la oportunidad de implementar un sistema de información con funcionalidades similares, pero con dichos problemas resueltos.

### Identificación de Oportunidad

Es posible encasillar los problemas listados en 3 categorías:

#### 1. Disponibilidad y facilidad de uso

Haciendo disponible gratuitamente una aplicación para los conductores se puede aumentar considerablemente la disponibilidad del servicio (limitándose sólo a falta de acceso a internet o a un smartphone).

#### 2. Mantenimiento de dispositivos

Los costos se reducen considerablemente; con una propuesta digital los costos de mantenimiento e instalación se limitan a: los dispositivos de hardware necesarios para ejecutar el sistema (varían según la cantidad de peticiones que deba procesar el mismo) y el mantenimiento o escalabilidad de las funcionalidades del mismo.

#### 3. Registro de datos necesarios para toma de decisiones

La información estadística que genera el sistema se basa en todos los registros de estacionamiento, consultas de policías de tránsito. Esto permite evaluar a nivel estratégico innumerable cantidad de aspectos como: tarifas, horarios, demanda según zona geográfica, evolución de demanda en el tiempo, recaudaciones.

## Estudio de mercado

### Condiciones iniciales

El alcance inicial de este proyecto pretende diseñar un sistema de información que sirva, al menos como base, para cubrir las necesidades de tarifado de estacionamiento en la vía pública de todo el país. No pretendemos hacer un producto a medida de un municipio, sino una base sólida que cubra las necesidades más importantes de todos los municipios del país. Luego, según diferentes especificaciones legales o necesidades puntuales se adaptará una solución customizada para cada ámbito.

El alcance de este proyecto pretende introducir un sistema de información que dé soporte a procesos existentes. Descartamos, al menos en el alcance de este proyecto, la posibilidad de modificar el marco legal (leyes, normas, ordenanzas); así, el enfoque está puesto en una solución que se adecue al marco reglamentario existente.

Dada la amplia cantidad de ámbitos de implementación, analizaremos las normas vigentes en la ciudad de Córdoba, no con el objetivo de analizar a fondo dicho cliente potencial, sino que, para acercarnos a la realidad y así aumentar la viabilidad del proyecto.

### Marco legal

Según la **Ley Nacional de Tránsito N° 24.449** artículo N° 2 *Competencia*: “*Son autoridades de aplicación y comprobación de las normas contenidas en esta ley los organismos nacionales, provinciales, municipales y de la Ciudad Autónoma de Buenos Aires que determinen las respectivas jurisdicciones que adhieran a ésta*”. Se interpreta que el marco legal establecido por el ámbito nacional establece las normativas básicas, habilitando a los diferentes ámbitos contenidos dentro del mismo a modificar y extender dicho marco.

De modo similar, la **Ley Provincial de Tránsito N° 8560** artículo N° 2 *Competencia*, establece: “*El Poder Ejecutivo Provincial concertará y coordinará con las Municipalidades las medidas tendientes al efectivo cumplimiento del presente régimen. Asimismo, podrá asignar las funciones de prevención y control del tránsito en las rutas provinciales y otros espacios del dominio público provincial, a la Policía de la Provincia de Córdoba y otros organismos existentes, sin que el ejercicio de tales funciones desconozcan o alteren las jurisdicciones locales.*”

En el ámbito municipal, la **Ordenanza N° 12626**, establece una serie de cláusulas para el “Sistema medido de estacionamiento en la vía pública”; de la cual, resumimos las siguientes condiciones:

- Estacionamiento permitido dentro del horario de 07:00 a 21:00 horas de lunes a viernes y de 07:00 a 14:00 horas los días sábados.
- Contar como mínimo con los siguientes medios de estacionamiento: mediante la utilización del teléfono celular y pago informatizado en Puntos de Venta.
- Contar como mínimo con los siguientes medios de pago para la compra de crédito: efectivo en Puntos de Venta y por medio de tarjetas de débito o crédito en aplicaciones de teléfono y/o mediante página WEB.
- Permitir que el usuario que utilice el celular, abone por el tiempo real estacionado con un mínimo de 30 minutos y posteriormente en fracciones por minutos. Para los usuarios de compra de tiempo de estacionamiento en Puntos de Venta, habrá un mínimo de una (1) hora y fraccionamiento cada treinta (30) minutos.
- La información de las operaciones de estacionamiento realizadas se debe poder consultar en tiempo real.

- Permitir la operación a los usuarios con celular, aun cuando su saldo disponible sea negativo, hasta un monto que será determinado por el Municipio. La reposición del importe del saldo negativo será exigible al usuario en la próxima carga o compra de crédito que efectúe.
- Prever la operación de una plataforma que permita que el usuario indique su hora de llegada y abandono del lugar de estacionamiento mediante dispositivos móviles. El usuario también podrá optativamente estacionar informando la cantidad de tiempo a estacionar, sin necesidad de realizar una transacción de salida.
- Determinar si un vehículo se encuentra o no abonando su estacionamiento, permitiendo que el Inspector Municipal en ejercicio del poder de policía constate la infracción.
- Informar en qué cuadras hay lugares disponibles para estacionar.
- La Autoridad de Aplicación podrá proponer la incorporación al sistema de otras zonas dentro del ejido de la ciudad o desafectar algunas que crea conveniente, bajo las mismas condiciones de explotación.
- Cobro al titular de vehículo de una tasa en concepto de Ocupación de Espacios Públicos para Estacionamiento por el monto que establezca la Ordenanza Tarifaria Anual vigente al momento del estacionamiento. En todos los casos el sistema cobrará como mínimo treinta (30) minutos en el caso de pago con electrónico, luego de dicho lapso el costo será por cada cinco (5) minutos excedentes.

*La ordenanza municipal 9981, establece en su capítulo XII:*

*Art. 75°.- EL estacionamiento en zona urbana debe efectuarse en los lugares en que no esté expresamente prohibido, sobre el costado derecho de la calzada, y a una distancia no menor a los cinco (5) metros del borde más próximo de la senda peatonal demarcada o imaginaria. Tanto el estacionamiento sobre el costado izquierdo de la calzada, como el estacionamiento a 45 grados,*

*debe ser dispuesto por la autoridad de aplicación en forma expresa y solo permitido en aquellas vías públicas que tengan un ancho de calzada superior a los nueve (9) metros.*

#### LUGARES

#### PROHIBIDOS

*Art. 76°.- NO pueden estacionarse vehículos en los siguientes lugares, salvo motivos debidamente justificados:*

- a) En las aceras;*
- b) En los espacios verdes públicos, rotondas, plazas, parques, paseos y espacios públicos en general;*
- c) Dentro de los cinco (5) metros anteriores y posteriores de una senda peatonal o ciclística o sobre las mencionadas sendas;*
- d) A menos de diez (10) metros de las paradas de vehículos del transporte público de pasajeros;*
- e) Frente a las entradas de cocheras, garajes, estaciones de servicio y playas de estacionamiento;*
- f) En el flanco izquierdo de la calzada, atento el sentido de circulación, excepto señalización in situ en contrario;*
- g) Áreas peatonales, isletas con cordones y separadores centrales;*
- h) Frente al acceso y hasta diez (10) metros de sus respectivos lados de hospitales, escuelas, bomberos, policía, organismos de seguridad y otros servicios públicos, salvo los vehículos relacionados a la función del establecimiento;*
- i) En general, en toda zona con demarcación y/o señalización prohibitiva o selectiva.*

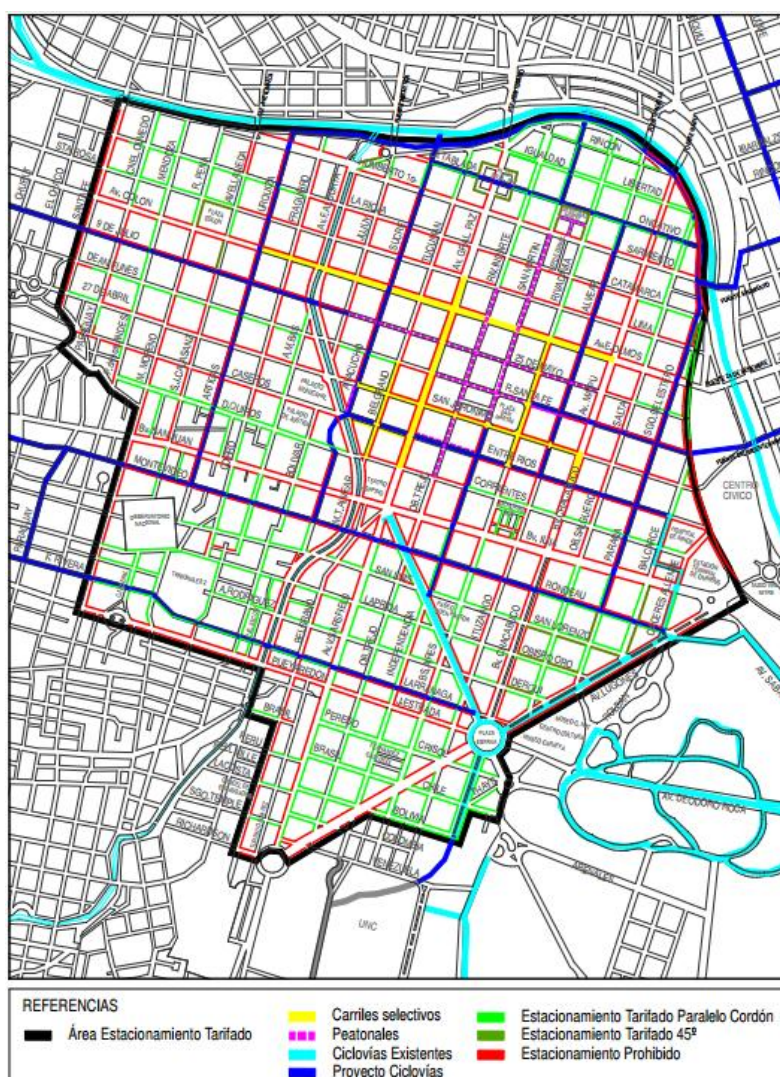
#### ESTACIONAMIENTO

#### PROHIBIDO

*Art. 77°.- SALVO expresa señalización que lo permita, queda prohibido el estacionamiento en la*

- vía pública de los siguientes vehículos, en el ejido municipal de la Ciudad:
- a) Cualquier clase de camiones y acoplados, con o sin carga;
  - b) Vehículos de transporte de pasajeros con capacidad superior a treinta (30) pasajeros;
  - c) Camiones mezcladores y de transporte de cemento a granel que no se encuentren en función específica;
  - d) Vehículos expuestos para su venta;
  - e) Vehículos en proceso de reparación, lavado o engrase, salvo que se trate de una emergencia que impida desplazar el vehículo si tal reparación no se efectúa;
  - f) Todo tipo de maquinaria especial, agrícola, vial o industrial;
  - g) Cualquier vehículo de transporte de pasajeros o carga para pernoctar su conductor o acompañantes, como casas rodantes, trailers o similares.

En la **Licitación Pública Nacional e Internacional N° 16/17** se incluye el siguiente plano, que delimita las zonas donde se aplica el control de estacionamiento dentro de la ciudad de Córdoba.



## Aplicaciones Similares

Luego de revisar comentarios en Play Store, analizamos las principales problemáticas que presentan las aplicaciones que ofrecen un servicio similar al que estamos desarrollando.

### BahiaParquímetros

- No permite registrar los vehículos, debiendo el usuario ingresar todos los datos por cada estacionamiento. Este punto es crucial para facilitar su uso y reducir el tiempo necesario para operar.
- Precisa que exista parquímetro en esa ubicación: esta es la forma en la que se conoce si el estacionamiento es tarifado o no. Pero presenta muchos problemas cuando los datos del parquímetro no son legibles. Nuestra solución deja sin efecto el uso de parquímetros y así se ahorran muchos costos.
- Se deben cargar saldo en una cuenta específica y en puntos de recarga autorizados. La interfaz es poco intuitiva.

### Apparkb

- Para renovar el estacionamiento o extender el plazo del mismo es obligatorio estar en el lugar de estacionamiento: esto supone una gran desventaja en situaciones donde se desconoce de antemano el tiempo exacto de estacionamiento-
- No se adecúa a las normas municipales de Córdoba (tarifa única y con fracciones fijas): Esta aplicación cobra por tiempo estacionado calculando el proporcional al costo de la hora, con lo cual sería necesario modificar las ordenanzas municipales o adaptar el proceso de cálculo del importe

En general, las ventajas más significativas que presenta nuestra propuesta, dado el marco legal tratado en el apartado anterior son:

- *Simplicidad de implementación.* La aplicación se implementa sobre un host Cloud cómo los es AWS (Amazon Web Services) permitiendo al cliente abstraerse de instalación, mantenimiento y actualización del hardware necesario. Además, a medida que la demanda vaya creciendo gradualmente, se puede escalar fácilmente la configuración de hardware necesaria.
- *Adaptación a marco legal existente.* Esto es fundamental para poder implementar el sistema de información sin necesidad modificar leyes vigentes.
- *Información para el ciudadano sobre las calles.* Además de poder abonar de forma digital el estacionamiento, también ofrecemos la posibilidad de consultar las calles tarifadas, las gratuitas y las prohibidas, brindando un dato esencial para la planificación de un estacionamiento y para evitar multas.
- *Comodidad para el ciudadano.* La aplicación permite agregar tiempo a un estacionamiento vigente de manera remota, evitando tener que desplazarse hasta el sitio donde el vehículo se encuentra estacionado.
- *Facilidad y seguridad en los pagos.* Al integrarse con una API de alta confiabilidad como lo es Google Wallet, los ciudadanos tendrán la tranquilidad de que su pago se transfiere de manera segura y sin salir de la aplicación, a la vez que se ofrece el servicio de Pago en Puntos de Carga, para ciudadanos que prefieran el pago en efectivo (o no cuenten con Smartphone con Android).

## MovyPark

- Maneja el concepto de “billetera” digital propia. Esto trae aparejado principalmente la desventaja de que no es posible retirar dinero de la misma, el saldo sólo puede ser consumido dentro de la aplicación.
- No permite consultar el historial de estacionamientos, por lo que no se cuenta con información histórica del uso de la aplicación.
- El mapa de consulta no brinda información precisa sobre las referencias sobre el mismo.



## Marco del proyecto

### Metodología

La metodología que aplicaremos en la realización del proyecto será una metodología Kanban aunque redefinida según las necesidades y el tamaño de nuestro equipo.

Nuestro proyecto será Estructurado por User Stories.

Se realizarán y calendarizarán Iteraciones de producto de longitud variable, en donde el resultado de cada iteración será un producto funcional (Obviamente con funcionalidad reducida) que servirá de entrada para la siguiente iteración.

Se definirá un tablero en donde se tendrán las siguientes columnas

#### **Backlog**

Backlog de la iteración en donde se encontrarán todas las User Stories que conformarán la iteración en curso.

#### **En Desarrollo**

Incidencias que están actualmente desarrollándose, no debiendo haber más que una por desarrollador.

#### **Desarrollo terminado**

Historias listas para realizar el Testing

#### **En Testing**

Historias en donde el Testing ya haya comenzado

#### **Finalizado**

Historias de la iteración en curso en donde el Testing ha concluido de manera exitosa.

#### **Iteración #**

Historias ya concluidas de iteraciones anteriores (el # representa el número de iteración al cual pertenece la historia).

#### **Bloqueado -> dependencias**

Historias que poseen un bloqueo que no permite su desarrollo. Todas las historias que conforman el producto final deben empezar en este estado, pasando a "Backlog" en cuanto comience la iteración a la que corresponden.

La **calendarización** de las Historias se hará en base a estimaciones que se realizarán consensuadas entre los integrantes del equipo, realizándose las mismas en semanas; el equipo considera que 1 (una) semana se compone de 8 (ocho) horas-hombre.



## Roles

Mezi, Mariano:

- Diseño
- Desarrollo
- Testing
- Implementación

Pianello, Martín

- Diseño
- Desarrollo
- Testing
- Implementación

## User Stories

- Como conductor, quiero contar con información sobre las calles donde está permitido estacionar, y su tarifa (si corresponde) para evitar infracciones de tránsito.
- Como conductor, me gustaría que mi dispositivo detecte mi ubicación (con precisión de cuadra) para evitar ingresar la dirección de estacionamiento de forma manual.
- Como conductor, quiero poder Cargar Tiempo de estacionamiento a través de tarjeta de Crédito para evitar el uso de efectivo.
- Como conductor, quiero que se me informe el tiempo disponible restante una vez registrado el estacionamiento, para saber con exactitud con cuanto tiempo cuento.
- Como conductor, quiero poder agregar más "Tiempo de estacionamiento" a un estacionamiento activo a través de tarjeta de Débito/Crédito, para evitar desplazarme hasta la zona de estacionamiento en caso de necesitar más tiempo.
- Como conductor, quiero poder Localizar mi vehículo para facilitar su ubicación.
- Como Policía de tránsito, quiero poder consultar el estado de un auto estacionado a través de su patente, para conocer si se encuentra en infracción.
- Como Administrador de la policía de tránsito, quiero poder registrar las calles y tramos "Tarifados", y "No permitidos" para estacionamiento, para contar con una base de datos fácilmente actualizable y mantenible.
- Como Administrador de la policía de tránsito, quiero poder actualizar la tarifa de estacionamiento de forma centralizada para adaptarme a las normativas vigentes.
- Como usuario de Android, quiero que la aplicación funcione correctamente con mi versión 4.4 *KitKat* para asegurar compatibilidad.
- Como Policía de tránsito, quiero que la consulta a través de la patente no demore más de 3 segundos, con una conexión 4G LTE.
- Como Usuario, quiero que todas las operaciones disponibles en aplicaciones Mobile funcionen con una conexión 3G, para poder operar en zonas geográficas donde la conexión 4G LTE no está disponible. Las operaciones pueden demorar más que con una conectividad mejor, pero deben poder procesarse sin arrojar un error "timeout".
- Como conductor, quiero que los datos de mi tarjeta de crédito sean enviados de manera segura, para resguardar mi información.

- Como policía de tránsito, quiero obtener los resultados de una consulta de manera visual, para poder verificar fácilmente el estado de un auto estacionado.
- Como conductor, quiero que la información de las calles permitidas sea similar a Google Maps, para facilitarme su uso.
- Como conductor, quiero disponer de puntos de carga accesibles en la vía pública, para registrar mi estacionamiento en caso de no contar con un smartphone o usuario registrado.
- Como punto de carga, quiero poder registrar un estacionamiento cargando la ubicación manualmente, para poder registrar automóviles estacionados en una ubicación diferente a la del punto de carga.
- Como Administrador de la Policía de Tránsito, quiero contar con un mapa de calor que me permita visualizar las zonas de mayor concentración de estacionamientos en tiempo real.
- Como Administrador de la Policía de Tránsito, quiero que el mapa de calor me permita filtrar los estacionamientos por un rango de fechas.
- Como conductor, quiero poder contar con una billetera virtual para poder pagar los estacionamientos cuando no cuente con tarjetas disponibles.
- Como conductor, quiero que la aplicación me informe cuando estoy intentando estacionar en una zona gratuita, y no se me cobre el estacionamiento.
- Como conductor, quiero que se me sugieran los dominios de los vehículos que más frecuentemente estaciono, para evitar realizar la carga en cada estacionamiento.
- Como conductor, quiero contar con un historial de estacionamientos que me informe los datos de los estacionamientos pasados.
- Como Administrador de la Policía de Tránsito, quiero poder generar y exportar un reporte en Excel, filtrado por fechas, que brinde información de la cantidad de estacionamientos registrados por hora.

## Calendarización

### **Iteración 1 Puesta en Marcha: 4 semanas (del 03/07/17 al 23/07/17)**

- Definición de Arquitectura.
- Configuración entorno.
- Modelado de dominio.
- Gestión de configuración.

### **Iteración 2: 6 semanas (24/07/17 al 04/09/17)**

- Cliente Conductor:
  - o Generación de Estacionamiento (Ingreso manual de patente)
- Cliente Policía de tránsito
  - o Consulta de estacionamiento a través de patente.

### **Iteración 3: 3 semanas (12/9/17 al 20/10/17)**

- Cliente Conductor:
  - o Consulta tiempo disponible
  - o Login
  - o Menú

**Iteración 4: 5 semanas (22/10/17 al 28/11/17)**

- Cliente Conductor:
  - o Geolocalización
- Cliente Administrador
  - o Reporte Estacionamientos por hora

**Iteración 5: 4 semanas (28/12/17 al 26/12/17)**

- Cliente Conductor:
  - o Geolocalización de precisión mejorada

**Iteración 6: 20 semanas (04/12/17 al 22/04/18)**

- Cliente Conductor:
  - o Integración con MercadoPago
- Cliente Administrador:
  - o Carga de información de calles

**Iteración 7: 9 semanas (21/05/18 al 22/07/18)**

- Cliente Conductor:
  - o Historial de estacionamientos
  - o Sugerencia dominios favoritos
  - o Validar estacionamiento tarifado
- Cliente Administrador:
  - o Clasificación de segmentos de calles

**Iteración 8: 11 semanas (23/07/18 al 07/10/18)**

- Cliente Conductor:
  - o Mapa de consulta de información
- Cliente Administrador:
  - o Definición tarifa única

**Iteración 9: 13 semanas (08/10/18 al 20/01/19)**

- Cliente conductor:
  - o Agregar tiempo de estacionamiento
  - o Redirección a estacionamiento vigente
- Cliente Administrador
  - o Mapa de Calor

**Iteración 10: 5 semanas (04/03/19 al 07/04/19)**

- Cliente Administrador
  - o Indicadores online
  - o Mapa de Calor

**Iteración 11: 10 semanas (20/05/19 al 28/07/19)**

- Cliente Punto de carga
  - o Creación App
  - o Búsqueda de ubicación
  - o Registrar estacionamiento
- Cliente Administrador
  - o Listado de calles habilitadas

**Iteración 12: 3 semanas (29/07/19 al 18/08/19)**

- Cliente Conductor:
  - o Finalizar estacionamiento
  - o Restricción estacionamiento mismo dominio
  - o Ubicación vehículo estacionado
- Administrador:
  - o Historial en mapa de calor

**Iteración 13: 4 semanas (19/08/19 al 15/09/19)**

- Cliente Policía de Tránsito:
  - o Re-styling para tablet
  - o Consulta vehículo estacionado (Optimización)
- Testing y bugfixing integral

## Estudio de factibilidad

### Técnica

#### Tecnologías Usadas

Para el desarrollo del sistema se hará uso de distintas plataformas y tecnologías, en el caso del sistema web que será brindado al administrador de Policías de Tránsito y servicios consumos por el módulo Mobile serán desarrollados bajo la tecnología **Java** (versión 1.8). La elección de este lenguaje de programación se debe a que todos los integrantes del equipo estamos familiarizados con el lenguaje y los desarrolladores particularmente trabajan a diario con la tecnología. El IDE utilizado para el desarrollo bajo este lenguaje será **Eclipse**, se eligió este IDE en primer lugar porque el mismo es gratuito, con una performance estable y los desarrolladores trabajan diariamente con el mismo, permitiendo esto una mayor agilidad y comodidad.

Por otro lado, el módulo Mobile, se desarrollará sobre la tecnología **Android**, utilizando el IDE **Android Studio**. En primer lugar, la elección de esta tecnología se debe a que claramente a nivel provincial, nacional y mundial Android y la tecnología Mobile más utilizada y popular, por otro lado, como Equipo nos hemos planteado como objetivo adicional incursionar en nuevas tecnologías que no hayamos utilizado. Android no es una tecnología que utilicemos diariamente y por ello no genera nuevos desafíos y oportunidades el desarrollo en tecnologías Mobile.

En cuanto a la Base de datos, se utilizará **PostgreSQL**, haciendo uso de *Amazon AWS* como servicio de hosting cloud. Elegimos este servicio debido a que el mismo es estable, facilita la administración de las instancias, permitiéndonos desarrollar nuestro proyecto de manera más ágil y cómoda, debido a que desarrollaremos el proyecto de manera distribuida físicamente como equipo.

La utilización de una solución cloud nos permitirá una gran capacidad de escalabilidad del sistema de acuerdo al cliente donde se implemente el sistema.

#### Requerimientos de Hardware (ambiente desarrollo)

El ambiente de desarrollo estará compuesto por:

- Instancia de base de datos centralizada, compartida por todos los integrantes del equipo. Para esto, utilizaremos una instancia *db.t2.small*. Sus características son:
  - a) CPUs virtuales: 1
  - b) Memoria RAM: 2GB
  - c) Costo: U\$S 0,076/hora (U\$S 6,08 mensuales)
- Instancias de aplicación descentralizadas, es decir que el servidor de aplicaciones donde se ejecutan las aplicaciones estará provisto en la computadora personal de cada miembro del equipo. Sus características (mínimas) son:
  - a) CPU: 2 núcleos ~ 1.7 GHz
  - b) Memoria RAM: 4GB

#### Requerimientos de Hardware (ambiente testing)

La configuración del ambiente de Testing pretende ser suficiente para poder realizar pruebas de cualquier índole, incluyendo varias pruebas concurrentes; sin embargo, no pretende poder soportar la misma capacidad de una instancia productiva, por lo que se

descartan pruebas de stress, carga y performance con volúmenes de datos similares a instancias productivas.

Usaremos una instancia de base de datos donde se alojará el motor RDBMS *db.t2.medium*, sus características son:

- El tipo de instancia t2 ofrece un rendimiento bajo pero suficiente para ambiente de desarrollo y Testing poco demandantes. Su costo es variable y asociado al uso, por lo que se ahorra dinero en momentos de inactividad.
- CPUs virtuales: 2
- Memoria RAM: 4GB
- Costo: U\$S 0,151/hora (U\$S 9,6 mensuales)

Usaremos una instancia de cómputos donde se alojará el servidor de aplicaciones *t2.medium*, sus características son:

- El tipo de instancia t2 ofrece un rendimiento mínimo estable y genera créditos de CPU cuando su uso es reducir; estos créditos se aprovechan aumentando la capacidad de cómputo del CPU en momentos de alta demanda.
- CPUs virtuales: 2
- Memoria RAM: 4 GB
- SO: Linux
- Costo: U\$S 0,081/hora (U\$S 4,5 mensuales)

#### Requerimientos de Hardware (ambiente productivo)

En cuanto a los servicios para la implementación del sistema se utilizará *Amazon Web Services (AWS)*. Consideramos que el grado de adopción de los usuarios con el producto ira creciendo gradualmente en el tiempo. Por esto, recomendamos un servicio de hosting que se pueda escalar fácilmente a medida que aumente la carga del sistema, y por ende la necesidad de hardware más potente, crezca.

Para una configuración inicial, sugerimos el servicio EC2 con una instancia *m4.xlarge*, que provee:

- Instancia M4 están diseñadas para el uso general de las aplicaciones, proporcionando un equilibrio en los recursos informáticos asegurando su estabilidad.
- No provee capacidades de aumento esporádico (según necesidad) de la capacidad de cómputo.
- 4 CPUs virtuales con velocidad estándar ~ 2,4GHz
- 16 GB RAM
- Ancho de banda de 750 Mbps dedicado
- Almacenamiento de SSD con alta velocidad de I/O
- SO: Linux
- Costo U\$S 131 mensuales

Como servidor de Base de Datos sugerimos el servicio RDS con una instancia *db.m3.large*, que provee:

- 2 CPUs virtuales
- 7,5 GB RAM
- Desempeño Moderado de red
- Costo U\$S 0,245 (suponiendo una disponibilidad 24hs, son U\$S 176,4 mensuales)

Adicionalmente, se debe considerar el servicio *EBS General Purpose SSD* con un costo de \$0,19 al mes por GB de almacenamiento.

#### Requerimientos de Software (generales)

Para la documentación del proyecto utilizaremos Microsoft Word y Google Docs. Ambas herramientas permite adaptarse a diferentes necesidades y se escogerá una u otra según la necesidad de trabajo concurrente que haya.

Las presentaciones se realizarán con Microsoft Power Point o Prezi, también dependiendo de la necesidad particular. Consideramos que una presentación más sencilla en animaciones pero con mayor cantidad de contenido se presenta mejor con Power Point, mientras que una presentación con menor cantidad de información pero donde es necesario hacer enfoque en la relación entre los diferentes tópicos se adapta mejor en Prezi.

En el ámbito del modelado del diseño de la aplicación optamos por un plug-in de Eclipse llamado *ObjectAid* que nos permite generar un diagrama de clases completo a partir del código existente. Dada la necesidad de realizar algún diagrama que no pueda basarse en el código existente (o esquema de base de datos) utilizaremos *Enterprise Architect*, ya que es una herramienta muy completa (reúne todos los elementos del estándar UML) y es de amplio conocimiento para todos los miembros del equipo.

#### Requerimientos de Software (ambiente desarrollo/testing)

Al utilizar una solución cloud del servicio de AWS de Amazon los requerimientos de software necesarios serán:

- Sistema Operativo Linux Debian x64 bits
- Motor de Base de datos PostgreSQL 9.6

#### Requerimientos de Software (ambiente productivo)

Al utilizar una solución cloud del servicio de AWS de Amazon los requerimientos de software necesarios serán:

- Sistema Operativo Linux Debian x64 bits
- Motor de Base de datos PostgreSQL 9.6

En cuanto al software utilizado para el funcionamiento del sistema se deberá contar con tablets que soporten sistema Android 4.4 *KitKat* o superior y que permitan la geolocalización.

En cuanto a los requerimientos de software de los equipos de los ciudadanos que utilizarán la aplicación también deberán soportar Android 4.4 *KitKat* y con posibilidad de geolocalización.

#### Conectividad

Teniendo en cuenta que el sistema a implementar estará basado sobre tecnología Mobile se deberá contar con una conexión mínima a internet de 3G para las tablets de los policías de tránsito. Para la aplicación web (Administrador) la conexión a internet deberá tener un ancho de banda mínimo disponible de 1 MB para bajada y 512 KB para subida.

El sistema proporcionado para los ciudadanos estará implementado sobre sistema Android por lo que cada ciudadano que utilice el sistema deberá contar con un Smartphone o Tablet con conectividad a internet mínima de 3G.

## Configuración

El sistema deberá ser implementado con una configuración inicial que estará basada sobre los requerimientos del Municipio solicitante, tales como perfiles de usuarios, usuarios, entre otros.

Las cuentas de “Policía de Tránsito” y “Punto de carga” deberán ser creadas previamente sobre el sistema, por el “Administrador”.

La aplicación para policías de tránsito debe instalarse manualmente sobre los dispositivos utilizados por estos mismos, ya que no estará disponible en el Play Store de Google. Por el contrario, la aplicación para “Conductor” y “Punto de carga” estarán disponibles en el Play Store.

## Almacenamiento

Los requerimientos de almacenamiento del sistema estarán dados por mecanismos de backups de la totalidad de la información crítica del sistema, como así también el resguardo de información histórica debido a que el sistema será implementado en Municipios que responden a una organización pública y civil proporcionando de esta manera una continua evidencia de la información y disponibilidad de la misma si así se requiere.

## Operativa

El presente proyecto está pensado en base a una problemática muy común y diaria por la cual atraviesa cualquier ciudadano de la ciudad, o cualquier municipio. Ante el gran crecimiento del parque automotriz tanto en la ciudad de Córdoba como así también en las distintas ciudades del interior, el estacionamiento reglamentado y tarifado se ha convertido en una problemática realmente interesante tanto para el ciudadano como para el servicio regulador del mismo. Por otra parte, claramente en los últimos años el avance y crecimiento de la tecnología ha sido exponencial, y como equipo de desarrollo creemos poder conjugar estos puntos.

A continuación, analizamos desde una perspectiva operativa el conjunto de conocimientos y habilidades requeridas por los diferentes usuarios de la aplicación, así como las actividades y artefactos que se entregarán para adaptar estos últimos a los requeridos por el sistema.

### Cliente Conductor

El cliente *Conductor* está apuntado a un público con conocimientos tecnológicos de rango muy amplio; por esto, las necesidades indispensables son: la capacidad de manipular un smartphone y navegar sobre aplicaciones con interfaces táctiles.

La aplicación presentará acciones muy claras y de fácil acceso; en el caso del uso del mapa, aprovecharemos el servicio provisto por Google Maps dado que presenta una interfaz muy conocida.

Por último, estará disponible de manera online una wiki con una explicación breve de las funcionalidades disponibles para este cliente, junto con captura de pantallas. La misma se deberá actualizar con cada versión que incluya cambios en su presentación.

No se considera necesario brindar un servicio de capacitación.



Para el caso de usuarios del sistema de registro de estacionamiento provisto por el municipio que no cuenten con estos conocimientos básicos indispensables, no cuenten con un smartphone compatible, o simplemente no deseen utilizar la aplicación, existe un cliente *Punto de carga* que provee las mismas funcionalidades, pero se accede al servicio mediante un intermediario, por ejemplo: el personal de atención de un kiosco.

#### Cliente Kiosco

El público al cual está apuntado este cliente es el mismo que el del cliente *Conductor*. Así, se contemplan exactamente las mismas consideraciones operativas.

#### Cliente Policía de tránsito

Este cliente constará de un conjunto de manuales de usuarios detallados que permitirán al/los usuarios del sistema familiarizarse rápidamente con él mismo. Se brindará una capacitación de 4hs con un máximo de 15 personas por turno, de manera que el sistema sea probado y el personal sea capacitado correctamente.

La cantidad de capacitaciones que se realizarán dependerá de la cantidad de personal que el municipio desea ubicar para la utilización del nuevo sistema de control de estacionamiento. Las mismas serán realizadas por el Equipo de desarrollo e implementación del sistema.

#### Cliente Administrador

Los usuarios de este cliente requieren de un nivel de conocimiento del mismo óptimo para poder parametrizar el mismo de la mejor manera posible.

Este cliente ofrece funcionalidades, y por ende interfaces, muy diferentes a las de los otros clientes. La funcionalidad principal consiste en configurar parámetros necesarios para alimentar al sistema de estacionamiento, como calles habilitadas para estacionar y el precio del mismo; por esto, la capacitación a estos usuarios se planificará con una duración de 6hs y un máximo de 8 personas por sesión.

#### Personal de mantenimiento

Existen actividades operativas que escapan al uso funcional del sistema, estas son:

- Instalación de aplicación en dispositivos móviles para policías de tránsito.
- Instalación y actualización de cliente web *Administrador*
- Instalación y actualización de servicios consumidos por todos los clientes
- Configuración inicial de algunos clientes

Para que el personal de mantenimiento del software pueda cumplir estas operaciones, se proveerá de una capacitación dividida en 2 sesiones de 4hs cada una con grupos de 8 personas como máximo.

Además, se proveerán manuales de instalación que no cubrirán todas las alternativas, pero sí las más comunes.

## Económica

### Costos del Producto

#### *Costos Generales*

Se determinarán como costos generales a todos los gastos destinados a productos utilizados por el equipo de trabajo en el marco de productos de oficinas.

Ítems	Costo	Costo Mensual
Resma papel impresora Ledesma	\$180	
Cartucho de tinta impresora Epson TX125 (Combo 4 colores)	\$800	
Electricidad		\$150
Internet Fibertel 25 Mb Wifi		\$1200
<b>Total</b>	<b>\$980</b>	<b>\$1350</b>

*Costos Hardware y Software*

En cuanto al hardware y software necesario para el desarrollo del producto no se emitirán gastos algunos ya que el equipo de desarrollo cuenta con PCs propias y los sistemas de softwares utilizados son todos de licencias libres.

El costo de hardware y software del Municipio para la implementación del sistema estará dado por el costo del servicio Amazon AWS y el servicio de internet asociado a la compañía.

El costo del servicio del servidor de aplicaciones puede ser evitado por el Municipio en caso de contar con su propio equipamiento de infraestructura.

Nuestra propuesta estipula los siguientes costos:

Ítems	Costo unitario	Costo Mensual
Servicio procesamiento		U\$S 131
Servicio motor BD		U\$S 176
Servicio almacenamiento (5GB)	U\$S 0,19/GB	U\$S 0,95
<b>Total</b>		<b>U\$S 307,95</b>

#### *Costo de Recursos Humanos*

El proyecto será desarrollado particularmente bajo el marco de Proyecto de la carrera de Analista de Sistemas de la Universidad Tecnológica Nacional Facultad Regional Córdoba por lo que no generará costos. Por ello es que el costo de Recursos Humanos será medido en Horas/Hombre

Concepto	Cantidad
Horas Mensuales por Persona	40
Cantidad de Integrantes	2
Cantidad de Meses	24,5
<b>Total</b>	<b>1960</b>

#### Beneficio

##### *Beneficios Tangibles*

- Ahorro en electricidad para el mantenimiento de parquímetros
- Ahorro en personal para el mantenimiento de parquímetros
- Reducción de costos para el mantenimiento del servicio de estacionamiento público

##### *Beneficios Intangibles*

- Mayor y mejor control de vehículos estacionados en la vía pública.
- Por parte del ciudadano, un mayor y certero conocimiento de posibilidad de estacionamiento en la vía pública.
- Mayor seguridad para el ciudadano al estacionar su vehículo.
- Mayor control de tiempo de estacionamiento.
- Digitalización del precario sistema de estacionamiento en la vía pública.

##### *Beneficio Económico*

Este beneficio va a estar ligado al paquete de implementación deseado por el Municipio teniendo en cuenta la cantidad de licencias de policías de tránsito y administradores deseada para operar.

# Plan de gestión de configuración

## Administración de entregables

Tal como se definió en la sección *metodología* del Estudio inicial, cada funcionalidad es planificada para ser entregada dentro de una iteración; para que esto ocurra, dicha funcionalidad debe ser incluida dentro de una versión y ésta formar parte de un *release*.

### Entregables

Cada entregable consta de un conjunto de componentes de software desplegados (según la definición de entornos), su documentación funcional y técnica asociada y manuales de usuario y despliegue requeridos.

Dado que el producto está distribuido en diferentes componentes cohesivos entre sí, y a la vez evolucionables independientemente, los requerimientos pueden generar cambios sobre uno o más de estos. Por esto, los entregables son “guiados por funcionalidad”; es decir, que se planifican basándose en una funcionalidad, y no por tiempo, componente de software u otra forma de división.

### Clasificación de entregables

Los entregables se clasifican según el grado de confianza que se tiene sobre la calidad de éste. Así, los etiquetamos de la siguiente forma:

- Snapshot (S): generados en etapa de desarrollo. Declara la que la versión no está lista para ser probada y le sirve solamente al desarrollador para realizar validaciones parciales.
- Candidato de release (RC): generados al finalizar el desarrollo de un componente. Declara que el componente incluye cambios que responden a una funcionalidad planificada, y que ésta está completa y lista para ser probada.
- Release (R): generados al finalizar el testing de un componente. Declara que el componente ha sido probado y cumple con el nivel de calidad mínimo requerido para ser entregado a producción.

### Identificación de Artefactos

#### Número de versión

Compuesto por <#Iteración>.<#Funcionalidad>.<#Fix>(Opcional)-<Clasificación del entregable>.

El número de Fix estará presente cuando la versión sea resultado de una corrección de  $n$  defectos encontrados sobre la versión que se corresponde con la misma iteración y funcionalidad. Se incrementará por cada versión correctora que se genere.

Por ejemplo: 2.3.1-R se corresponde con la funcionalidad número 3 dentro de la iteración 2, que corrige  $n$  defectos presentados en la versión 2.3, y que cumple con el nivel de calidad esperado para producción.

## Control de versiones

### Repositorio de código

Se utilizará el sistema de repositorio GIT, el cual estará alojado en BitBucket. Se seguirán las siguientes guías:

- Se tendrá un repositorio por cada proyecto (ws, cliente Android, etc)
- Se utilizarán los siguientes Spaces
  - Master: versión estable, sólo con pull requests.
  - Develop: Versión evolutiva, sobre la cual se trabaja a diario.
  - Feature-Branch <iteración | versión> (Opcional): Branch creada para modificar datos de versión previo a pasar a develop.

### Nombre Repositorio

Compuesto por *parco-<cliente>-<funcionalidad>*. Por ejemplo, *parco-c-estacionamiento* o *parco-a-web*.

### Etiquetas de versiones finales (RELEASE)

Se formar con: <versión>. Por ejemplo, 2.3 para cuando se realice el tag de la versión 2.3.

### Etiquetas de versiones finales de Iteración

Se forman con: *i<#iteración>*. Por ejemplo, i3.

## Dependencias

### Repositorios de dependencias

- Maven Central: <https://repo.maven.apache.org/maven2/>
- JCenter (Opcional): <https://bintray.com/bintray/icenter>
- Google (solo para Android): <https://maven.google.com>

# Plan de pruebas

## Objetivo

El plan de pruebas de Software se elabora con el fin de especificar qué elementos o componentes se van a probar para que el grupo de trabajo pueda realizar el proceso de Validación y Verificación de los requerimientos funcionales y no funcionales del producto.

Este documento tiene como propósito establecer los lineamientos a seguir a la hora de planificar las pruebas, el modo y la forma de reportar errores o fallas, de manera que se realicen las correcciones necesarias en forma más eficiente y eficaz.

El plan de pruebas se aplica sobre el producto, dado que el mismo se desarrollará sobre la metodología de producto mínimo funcional, esto permitirá realizar las pruebas del software funcionales y no funcionales necesarias.

Los resultados de las pruebas son registrados en plantillas previamente definidas, descritas en el desarrollo del presente documento. Las pruebas a implementar son básicas, esto incluye las pruebas unitarias y de integración que son vitales para la validación del producto.

## Estrategia de Pruebas

El proyecto será desarrollado bajo una metodología de “Producto Mínimo Funcional” apoyado por las bases del tablero de Kanban por lo que se definirán las pruebas unitarias a realizar de acuerdo con las historias de usuario desarrolladas en cada iteración, definiendo como una tarea más a realizar las pruebas funcionales y de regresión.

El objetivo de la planificación de las pruebas es asegurar por medio de la validación y la verificación del sistema la integridad del mismo y el correcto funcionamiento de acuerdo a los requerimientos planteados.

Por cada módulo del sistema desarrollado se planificará y estimará la ejecución de pruebas de sistema y regresión correspondientes al módulo.

Dado que el producto cuenta con 4 módulos totalmente diferenciados y los mismos se desarrollarán en forma paralela para otorgar un producto funcionando como entregable de cada iteración, las pruebas de integración serán transversales en cada iteración.

La totalidad de las ejecuciones de los casos de pruebas serán manuales y las mismas serán documentadas en una plantilla de caso prueba y en caso de que estos ameriten el reporte de un defecto o error serán reportados formalmente bajo un documento de bugs previamente definido por el equipo.

### Criterios de inicio

Al comenzar el desarrollo de cada iteración es indispensable disponer de los requerimientos con un alto grado de especificación. Luego, se realizan, por parte de todos los miembros del equipo involucrados en el requerimiento, un análisis que le permite a cada persona interiorizarse y comprender cómo funciona ese requerimiento en el dominio del problema.

### Creación de casos de prueba

Con un suficiente nivel de comprensión del requerimiento, se puede comenzar las tareas necesarias para crear los casos de prueba.

A lo largo del ciclo de desarrollo es posible que se modifique levemente el requerimiento inicial, que se amplíe su grado de especificación o que cambie su análisis, y por ende el entendimiento del mismo. Dadas estas condiciones, los casos de prueba ya creados, o en construcción, deben modificarse consecuentemente respetando el nuevo análisis.

### Ejecución de casos de prueba

Existen dos momentos dentro de cada iteración que disparan el comienzo de la ejecución de los casos de prueba, cada uno con un objetivo diferente: las pruebas automatizadas y las manuales

Las pruebas automatizadas se pueden ejecutar en cualquier momento para proveer feedback rápido al desarrollador. Aquí es necesario hacer algunas aclaraciones:

- Casos de prueba unitarios: suelen ser de ejecución rápida, del orden de los milisegundos o pocos segundos, por eso se pueden ejecutar todas las veces que sea necesario; además, para el feedback rápido que necesita un desarrollador no es necesario la ejecución del mismo en el ambiente de test.



- Casos de prueba de integración o de sistemas: dependen de otros componentes, servicios (base de datos, UI) o sistemas externos; es indispensable su ejecución en entornos de test donde se disponen de la versión correcta de dichas dependencias. Por esto, estos casos se ejecutan todos juntos como un único proceso en un momento donde exista una versión completa del sistema disponible para pruebas.

La ejecución de los casos de prueba manuales, por otro lado, se ejecutan una vez que el/los componentes necesarios estén terminados y disponibles en el repositorio de código para ser construidos y disponibilizados en el ambiente de test. El motivo de esta definición se basa en:

- Costo de ejecución: requiere mucho tiempo (varios minutos) y personas que ejecuten cada paso manualmente.
- Fidelidad: el ambiente de test debe tener sólo versiones release (versiones finales) para dependencias o candidatos release (en caso de no existe una versión release todavía)
- Objetivo: es la verificación de una funcionalidad (o porción) para marcar la versión como release; no es su funcionalidad principal dar feedback rápido a una actividad de desarrollo en curso.

### Criterios de finalización

Para considerar que una funcionalidad entregada a Testing ha sido probada lo suficiente, y se debe versionar el artefacto como release, es necesario tener un alto grado de confianza sobre el mismo. Hay algunos aspectos intrínsecos del proyecto (y de producto) que establecen algunas guías para considerar “suficiente” cierta cantidad de pruebas:

- El producto **no** se va a instalar en una instancia productiva hasta no estar completo, esto implica que:
  - La cantidad de casos de prueba que se incluyen en la regresión a ejecutar en las iteraciones (excepto las finales) serán pocos. El objetivo es asegurar que las funcionalidades principales y los escenarios de éxito comunes se mantengan funcionales a lo largo de toda la fase de construcción. A esta regresión la denominaremos **regresión minimalista**.
  - La última iteración (que incluya la modificación de algún componente) incluirá una suite de regresión mucho más extensa para asegurar que la versión final del sistema sea susceptible de ser desplegada en un ambiente de pruebas de usuario sin defectos de severidad *maj*or en adelante.
- Funcionalidades críticas que requieren un alto grado de confiabilidad para el usuario, como el pago con medios electrónicos o la consulta del estado de estacionamiento de un vehículo (supone la diferencia entre una multa o no), deben funcionar:
  - Se incluye mucho tiempo de Testing exploratorio.
  - Se priorizan para realizar test automatizados a nivel funcional (el servicio completo).
  - Estarán incluidos en la regresión (versión completa y también minimalista).
- Al ser un producto de uso masivo, la usabilidad es un requerimiento muy importante
  - Se deben incluir casos de prueba de usabilidad, sobre todo para clientes “Conductor” y “Kiosco”.

- El Testing exploratorio apoyado en el criterio del tester es muy importante para apreciar características que no se hayan especificado en algún caso de prueba.
- Estas pruebas no serán incluidas en la regresión minimalista por su alto costo.

Con todas estas guías, se confía en la experiencia y capacidad del *colaborador* para tomar la decisión de cuáles casos de prueba comprometer en una iteración, y cuáles no, cuánto tiempo dedicar al Testing exploratorio y qué casos de prueba incluir en regresiones minimalistas.

## Niveles de Pruebas

Las pruebas se estructurarán en diferentes niveles de pruebas que se regirán según objetivos y objetos a probar.

### Pruebas unitarias

Se incluyen pruebas de caja blanca realizadas sobre porciones (normalmente métodos) de código y tienen como objetivo asegurar el correcto funcionamiento de una operación que ejecuta el sistema.

Estas pruebas serán definidas según el criterio del desarrollador y no se especifica una cantidad o cobertura obligatoria. Estas pruebas se ejecutarán independientemente de incluir interfaces/componentes (base de datos, sistemas externos, componentes UI, dispositivos de hardware) y sólo probarán la ejecución de código independiente de su Entrada/Salida.

### Pruebas de integración

Incluimos todos los componentes responsables de llevar a cabo las funcionalidades completas (end-to-end), incluyendo base de datos, componentes UI, sistemas externos, etc.

Estas pruebas estarán respaldadas por casos de prueba basados en las User Stories y se tendrá en cuenta una cobertura completa, al menos para su ejecución.

### Pruebas de sistema

Incluirán una versión del sistema con todos sus componentes y disponible para todas las funcionalidades completadas.

Estas pruebas serán importantes para asegurar la compatibilidad de todas las funciones entregadas en una versión y la cohesión del sistema completo. Pruebas de este tipo serán incluidas (al menos) al final de cada iteración y considerarán pruebas exploratorias.

Si la ejecución es considerada exitosa (existe una cantidad de defectos aceptables según los límites establecidos por criticidad) la versión probada es considerada candidata de release (RC).

### Pruebas de aceptación

Al ser un producto guiado por una oportunidad de mercado, en lugar de guiado por un cliente, la aceptación será a criterio de los mismos integrantes del equipo de trabajo y eventualmente se puede solicitar ayuda de personas externas al grupo para incluir una perspectiva más amplia. Esto no es obligatorio y se realizará según criterio del grupo.

## Ciclos de pruebas

Los ciclos de pruebas constarán de la definición y ejecución de un conjunto de casos de prueba definidos para una versión identificable entregada a testing (se completó al menos una tarea de desarrollo) y el feedback de la totalidad de defectos encontrados al finalizar el ciclo.

Según la definición de *entregables* incluida en el documento de Gestión de configuración, éstos están guiados por funcionalidad, por lo que también lo estarán los ciclos de prueba. Esto significa que, una vez completado el desarrollo de una funcionalidad, se genera una única versión y etiqueta para cada componente que haya sido evolucionado.

Cada ciclo de prueba incluirá entonces, todos los casos de prueba diseñados para comprobar la calidad de un único requerimiento contenido en un único entregable. Lo que significa que, este último no está alineado ni depende de los tiempos de la iteración, sino de la funcionalidad que se entregará.

Una vez que se hayan ejecutado todos los casos de un ciclo de pruebas se registrarán los defectos encontrados. Si se encuentra, al menos un defecto, con severidad mayor a la mínima definida para considerar la funcionalidad como “válida”, será necesario realizar las tareas necesarias para corregirlo. Esto concluirá en un nuevo entregable con una nueva versión, y un nuevo ciclo que contendrá el mismo conjunto de casos de pruebas que el anterior, pero ahora serán ejecutados sobre la nueva versión que, se cree, corrige el defecto encontrado. Este proceso se repite tantas veces como sea necesario para que la versión cumpla con el nivel de calidad definido, tras lo cual la versión pasa a considerarse como versión *Release*.

## Planificación De Pruebas

Las pruebas se realizarán, según la naturaleza de las mismas, en diferentes instancias dentro del proceso de desarrollo del producto.

Las pruebas unitarias serán íntegramente realizadas mediante JUnit, quedando a criterio de los desarrolladores su realización, en función de la criticidad de la funcionalidad. Si bien es deseable su realización, no serán planificadas ni consideradas obligatorias dentro de ninguna iteración.

Las pruebas de integración serán definidas y estructuradas por casos de prueba, y organizadas dentro de Ciclos de pruebas. La creación y ejecución de los casos de prueba definidos para las pruebas de Integración, se realizará de manera deseable durante la iteración en curso, pudiendo en algunos casos adelantarse la creación de los mismos en caso de contar con suficiente Capacidad de Testing y/o en función de la criticidad de la funcionalidad a probar. La ejecución de los casos de prueba asociados a las pruebas de integración podrá realizarse tan pronto como su/s componente/s asociados se hayan desarrollado completamente.

Las pruebas de sistema, por su naturaleza, se ejecutarán al haberse liberado una Versión Candidata a Release al ambiente de testing. Las mismas estarán conformadas por las pruebas de *Regresión minimalista*, sumado a los casos de prueba de Integración considerados más críticos dentro de la iteración en curso, más un componente de Testing exploratorio según capacidad. Para todos los casos, la fecha de ejecución está incluida dentro de la iteración en la que fue desarrollada la funcionalidad a probar. De este modo, al finalizar la iteración la funcionalidad estaría lista para ser usada en producción.

Las pruebas de aceptación serán realizadas a criterio de los integrantes del equipo, con la ayuda de personas externas que pudieran aportar la visión del usuario final, y no estarán planificadas ni calendarizadas. Su realización dependerá de la funcionalidad, su criticidad, y los tiempos con los que se cuente en la iteración.

## Definición de Ambientes de Pruebas

El ambiente donde se ejecuta una prueba es un factor determinante para juzgar la efectividad de la misma. Por otro lado, los procesos de despliegue, la paralelización de ejecución y disponibilidad de ambientes hacen que no todas las pruebas se puedan ejecutar en el mismo ambiente.

Teniendo en cuentas estas aristas, proponemos las siguientes definiciones:

### Ambientes locales

- Los miembros del equipo ejecutan pruebas unitarias y de integración en sus entornos locales para proveer feedback rápido.
- La ejecución de estas pruebas no es considerada como evidencia suficiente para aprobar casos de prueba; por eso, no se registrará su ejecución (aunque sí se registrarán los defectos que se puedan encontrar).

### Ambientes de testing

- Algunas pruebas de integración y las pruebas de sistema se ejecutarán sobre una versión identificable e inmutable del producto en un ambiente de Testing disponible en un entorno accesible por todo el equipo destinado para tal fin (considerado un ambiente de configuración extremadamente similar a un ambiente productivo, pero con menos recursos) y se registrará la ejecución de las mismas.
- El registro de ejecución de las pruebas en el ambiente de Testing son consideradas por el grupo como documentación suficiente de que una versión específica cumple con el criterio de calidad definido para ser entregado.
- De ser necesario, aunque no como configuración inicial, se escalará horizontalmente para disponer de más de uno de estos ambientes (incluyendo independencia de esquema de base de datos e instancia de ejecución) con el objetivo de ejecutar de forma paralela pruebas sobre dos versiones diferentes del sistema.

### Ambiente productivo

No se utilizará para ejecutar ninguna prueba.

### Procedimiento pasaje entre ambientes

Definimos diferentes criterios y condiciones que se deben cumplir para considerar a una funcionalidad (o componente de ésta) lista para realizar la transición de un ambiente a otro, y por ende, avanzar en el ciclo de vida definido para la misma.

#### Transición: Desarrollo → Testing

Sólo se puede considerar que una versión de la aplicación está lista para pasar al ambiente de Testing cuando se termina el desarrollo de una funcionalidad que alcance para completar, al menos, un escenario de la historia de usuario. Es decir que, es posible pasar una versión *snapshot* al ambiente de Testing sin haber completado la totalidad de la user story pero siempre y cuando, estando terminado, se puedan ejecutar todos los casos de prueba considerados dentro de escenario de ejecución del requerimiento.

Al no existir un ambiente centralizado de desarrollo, sino que estará disponible en la computadora personal de cada miembro del equipo, existen altas probabilidades de que al pasar una versión al ambiente de Testing, ésta falle por diferencias de configuración del

servidor, capacidad del entorno, datos disponibles, sistema operativo, dependencias externas. Esto no es un problema, sino todo lo contrario; es un caso de éxito en el Testing, ya que implementar en el ambiente es un caso de prueba de instalación que se ejecutara como prerequisite antes de iniciar cada ciclo de pruebas. Además, es considerada una operación fail-fast; este es el mejor resultado posible de Testing ya que encuentra un defecto/error con un costo de ejecución casi despreciable.

#### Transición: Testing → Producción

Esta transición tiene lugar sólo cuando las siguientes condiciones se cumplen:

- Se ejecutaron pruebas de sistema sobre la versión y se reportan el resultado de las mismas.
- El resultado de las pruebas de sistema con sus defectos reportados es aceptado por el usuario y se acuerda ejecutar la transición con esos defectos (si los hubiese).
- Existe un acuerdo entre el usuario y el equipo encargado de llevar a cabo la transición (implementación) en un intervalo de tiempo determinado y acordado.
- Se ejecutó una regresión y se reportó el resultado de la misma.
- El resultado de la regresión con sus defectos reportados es aceptado por el usuario y se acuerda ejecutar la transición con esos defectos (si los hubiese).
- **No** se ejecutó una regresión y esto fue un pedido explícito del equipo de implementación o de los usuarios, y estos últimos están de acuerdo (su intención es cubrir escenarios donde se requiere implementar rápidamente una corrección).

## Niveles de Severidad de defectos

Se definen los siguientes niveles de severidad:

Blocker	Inhibe la continuidad del desarrollo o de las pruebas.
Critical	Pérdida total de la funcionalidad, pérdida de datos.
Major	Pérdida mayor de la funcionalidad, datos erróneos.
Minor	Pérdida menor de la funcionalidad.
Trivial	Aspectos cosméticos, ortografía, visualización.
Enhancement	Propuestas de mejoras en cuanto a funcionalidades, controles, etc.



## Plantillas

Definimos algunas plantillas útiles para documentar los componentes definidos en este plan.

### Plantilla de casos de prueba

Luego de Analizar templates de casos de prueba existentes, optamos por redefinir uno estándar haciendo algunas modificaciones en base a nuestras necesidades.

Número de Caso de prueba			
Nombre Caso de prueba			
Módulo/s afectado/s			
Componente/s afectado/s			
Descripción			
Pre condiciones			
Fecha de creación			
Resultado esperado			
Flujo del caso de prueba	<b>N° paso</b>	<b>Usuario del sistema</b>	<b>Acción</b>
Post condiciones			

## Plantilla de ejecución de caso de prueba

Número de ejecución			
Identificador de Caso de prueba			
Fecha de ejecución			
Descripción			
Flujo de pasos ejecutados	<b>N° paso</b>	<b>Usuario del sistema</b>	<b>Acción</b>
Resultado obtenido			
Estado			

## Plantilla de defectos

Para la gestión de los defectos, se optó por utilizar la herramienta “Issue tracker”, provista por Bitbucket, e integrada al repositorio de código. La misma ofrece una interfaz simplificada para la creación y seguimiento de los defectos, pudiendo asignar responsables de corrección lo que favorece su control.

ParCo / ParCo / pd-c-mobile

## Issues

### Create issue

Title\*

Description

Assignee

[Assign to me](#)

Kind\*

Priority\*

Attachments

La descripción de los campos es la siguiente:

- *Title* → Nombre del defecto. Debe resumir la pérdida de funcionalidad detectada.
- *Description* → Descripción del defecto. Permite que el colaborador incluya la información que crea necesaria para facilitar identificar las características del defecto, su reproducción y corrección.
- *Assignee* → Permite identificar al colaborador que realizó acciones sobre el defecto identificado. Cuando el defecto se crea este campo queda vacío. Luego, un colaborador se lo auto asigna y lo resuelve (o cancela).
- *Kind* → El tipo de incidencia. Sólo se utilizará la opción **bug**.
- *Priority* → Define el nivel de criticidad del defecto según los niveles definidos en este plan.
- *Attachments* → Archivos adjuntos. Es un campo opcional, útil para incluir elementos complementarios que faciliten su reproducción y corrección (por ejemplo: capturas de pantalla o logs).

# Análisis de Factibilidad

## Técnica

### Requerimientos de Hardware

La implementación del sistema se realizará sobre instancias de *Amazon Web Services (AWS)*. Consideramos que el grado de adopción de los usuarios para con el producto irá creciendo gradualmente en el tiempo. Por esto, es requerido un servicio de hosting que se pueda escalar fácilmente a medida que aumente la carga del sistema, y por ende la necesidad de hardware más potente, crezca.

Para una configuración inicial, sugerimos el servicio EC2 con una instancia *m4.xlarge*, que provee:

- Instancia M4 están diseñadas para el uso general de las aplicaciones, proporcionando un equilibrio en los recursos informáticos asegurando su estabilidad.
- No provee capacidades de aumento esporádico (según necesidad) de la capacidad de cómputo.
- 4 CPUs virtuales con velocidad estándar ~ 2,4GHz
- 16 GB RAM
- Ancho de banda de 750 Mbps dedicado
- Almacenamiento de SSD con alta velocidad de I/O
- SO: Linux
- Costo U\$S 131 mensuales

Como servidor de Base de Datos sugerimos el servicio RDS con una instancia *db.m3.large*, que provee:

- 2 CPUs virtuales
- 7,5 GB RAM
- Desempeño Moderado de red
- Costo U\$S 0,245 (suponiendo una disponibilidad 24hs, son U\$S 176,4 mensuales)

Adicionalmente, se debe considerar el servicio *EBS General Purpose SSD* con un costo de \$0,19 al mes por GB de almacenamiento.

Para las tablets a ser utilizadas por los Policías de tránsito, sugerimos el modelo Samsung Galaxy Tab A6 T285. El mismo cuenta con los requisitos recomendados para correr la aplicación, al poseer conectividad inalámbrica LTE, versión de Android compatible y capacidad de procesamiento suficiente.

- Costo aproximado: U\$S 290 cada una.

## Requerimientos de Software

Al utilizar una solución cloud del servicio de AWS de Amazón los requerimientos de software necesarios serán:

- Sistema Operativo Linux Debian x64 bits
- Motor de Base de datos PostgreSQL 9.6

En cuanto al software utilizado para el funcionamiento del sistema se deberá contar con tablets que soporten sistema Android 4.4 *KitKat* o superior. Las mismas deben contar con un servicio de conectividad a internet de tipo satelital para poder operar desde cualquier ubicación.

En cuanto a los requerimientos de software de los equipos de los ciudadanos que utilizarán la aplicación también deberán soportar Android 4.4 *KitKat* y con posibilidad de geolocalización.

## Conectividad

El sistema está basado sobre tecnología Mobile, por lo que se deberá contar con una conexión mínima a internet de 3G para las tablets de los policías de tránsito. Para la aplicación web (Administrador) la conexión a internet deberá tener un ancho de banda mínimo disponible de 1 MB para bajada y 512 KB para subida.

El sistema proporcionado para los ciudadanos se encuentra desarrollado sobre el sistema Android por lo que cada ciudadano que utilice el sistema deberá contar con un Smartphone o Tablet con conectividad a internet mínima de 3G.

## Configuración

El sistema deberá ser implementado con una configuración inicial que estará basada sobre los requerimientos del Municipio solicitante, tales como perfiles de usuarios, usuarios, entre otros.

Es necesario, para comenzar a operar el sistema, que se parametricen las calles y tramos donde el estacionamiento se encuentra regulado, y donde no se encuentra permitido. Esta tarea deberá ser llevada a cabo por el administrador.

La aplicación para policías de tránsito debe instalarse manualmente sobre los dispositivos utilizados por estos mismos, ya que no estará disponible en el Play Store de Google. Por el contrario, la aplicación para “Conductor” y “Punto de carga” estarán disponibles en el Play Store.

## Almacenamiento

Los requerimientos de almacenamiento del sistema estarán dados por mecanismos de backups de la totalidad de la información crítica del sistema, como así también el resguardo de información histórica debido a que el sistema será implementado en Municipios que responden a una organización pública y civil proporcionando de esta manera una continua evidencia de la información y disponibilidad de la misma si así se requiere.

## Operativa

### Cliente Conductor

El cliente *Conductor* está apuntado a un público con conocimientos tecnológicos de rango muy amplio; por esto, las necesidades indispensables son: la capacidad de manipular un smartphone y navegar sobre aplicaciones con interfaces táctiles.

La aplicación presentará acciones muy claras y de fácil acceso; en el caso del uso del mapa, aprovecharemos el servicio provisto por Google Maps dado que presenta una interfaz muy conocida.

Por último, estará disponible de manera online un manual de usuario con una explicación breve de las funcionalidades disponibles para este cliente, junto con captura de pantallas. La misma se deberá actualizar con cada versión que incluya cambios en su presentación.

No se considera necesario brindar un servicio de capacitación.

Para el caso de usuarios del sistema de registro de estacionamiento provisto por el municipio que no cuenten con estos conocimientos básicos indispensables, no cuenten con un smartphone compatible, o simplemente no deseen utilizar la aplicación, existe un cliente *Punto de carga* que provee las mismas funcionalidades, pero se accede al servicio mediante un intermediario, por ejemplo: el personal de atención de un kiosco.

### Cliente Punto de carga

El público al cual esta apuntado este cliente es el mismo que el del cliente *Conductor*. Así, se contemplan exactamente las mismas consideraciones operativas.

### Cliente Policía de tránsito

Este cliente constara de un conjunto de manuales de usuarios detallados que permitirán al/los usuarios del sistema familiarizarse rápidamente con él mismo. Se brindará una capacitación de 4hs con un máximo de 15 personas por turno, de manera que el sistema sea probado y el personal sea capacitado correctamente.

La cantidad de capacitaciones que se realizarán dependerá de la cantidad de personal que el municipio desea disponer para la utilización del nuevo sistema de control de estacionamiento. Las mismas serán realizadas por el Equipo de desarrollo e implementación del sistema.

### Cliente Administrador

Los usuarios de este cliente requieren de un nivel de conocimiento del mismo óptimo para poder parametrizar el mismo de la mejor manera posible.

Este cliente ofrece funcionalidades, y por ende interfaces, muy diferentes a las de los otros clientes. La funcionalidad principal consiste en configurar parámetros necesarios para alimentar al sistema de estacionamiento, como calles habilitadas para estacionar y el precio del mismo, como así también la correcta interpretación de los reportes ofrecidos; por esto, la capacitación a estos usuarios se planificará con una duración de 6hs y un máximo de 8 personas por sesión.

## Personal de mantenimiento

Existen actividades operativas que escapan al uso funcional del sistema, estas son:

- Instalación de aplicación en dispositivos móviles para policías de tránsito.
- Instalación y actualización de cliente web *Administrador*
- Instalación y actualización de servicios consumidos por todos los clientes
- Configuración inicial de algunos clientes

Para que el personal de mantenimiento del software pueda cumplir estas operaciones, se proveerá de una capacitación dividida en 2 sesiones de 4hs cada una con grupos de 8 personas como máximo.

Además, se proveerán manuales de instalación que no cubrirán todas las alternativas, pero sí las más comunes.

## Económica

### Costos del Producto

#### Costos Hardware y Software

La aplicación tiene requisitos de Hardware mencionados en el punto [Requerimientos de hardware](#) del presente documento. Se incluyen precios estimativos teniendo en cuenta lo descrito, pudiendo utilizar el municipio servidores propios que disminuyan estos costos, siempre que cumplan con los requerimientos necesarios. La cantidad de tablets a utilizar por el Cliente, dependerá de la cantidad de usuarios “Policía de Tránsito” solicitados. Se realizan los cálculos de costos, tomando por defecto el plan más bajo disponible, de 15 cuentas.

Ítems	Costo unitario	Costo Mensual
<b>Servicio procesamiento</b>		U\$S 131
<b>Servicio motor BD</b>		U\$S 176
<b>Servicio almacenamiento (5GB)</b>	U\$S 0,19/GB	U\$S 0,95
<b>Total</b>		<b>U\$S 307,95</b>

Ítems	Costo unitario	Cantidad	Costo total
<b>Tablet 4g Samsung A6 T-285</b>	U\$S 290	15	U\$S 4350
<b>Total</b>			<b>U\$S 4350</b>

#### Costo de Recursos Humanos

Los costos asociados a Recursos Humanos, son los relacionados a las necesidades de capacitación que se necesitan para la correcta utilización y parametrización del sistema, descriptas en el apartado [Operativa](#) del corriente documento.

Concepto	Cantidad
<b>Horas de capacitación necesarias</b>	10
<b>Costo capacitación por hora</b>	U\$S 20
<b>Total</b>	<b>U\$S 200</b>



## Conclusión de factibilidad

La factibilidad de la implementación de un sistema de Información en un ambiente productivo es quizás uno de los eventos más inciertos cuando se comienza un proyecto. La cantidad de variables que pueden afectar y determinar la viabilidad del mismo es muy amplia, por lo que es necesario realizar un correcto análisis en el inicio del proyecto, que permita, al menos, contar con una base de información que permita conocer si las posibilidades reales de despliegue son esperanzadoras.

Consideramos que el estudio de prefactibilidad al comienzo del proyecto fue suficientemente completo y abarcaba gran parte de las variables que pudieran modificar su curso, lo que nos permite, al compararlo con el contexto actual, observar que el mismo continúa siendo válido.

El correcto Análisis de Hardware necesario realizado durante el estudio de Prefactibilidad, nos permitió elegir equipos (Servidores, tablets) con capacidad de procesamiento suficiente para soportar cambios en los requerimientos que pudiesen afectar las capacidades mínimas necesarias, por lo cual siguen siendo perfectamente válidos.

En cuanto a la factibilidad operativa, el producto continúa evolucionando según la idea y planificación inicial, persiguiendo una facilidad de uso y simplicidad de interfaces visuales, que permitan que su distribución y utilización sea masiva y no existan grupos de usuarios que, por sus particularidades, puedan quedar excluidos de su utilización. Se siguen manteniendo los tiempos de capacitación necesarios estimados originalmente, para aquellos usuarios finales de las aplicaciones “ParCo – PDT” y “Parco Web”, ya que, de su correcta utilización e interpretación, depende el beneficio real del producto.

Con la factibilidad económica sucede algo similar. Los precios contemplados durante la prefactibilidad fueron considerados en su totalidad en dólares estadounidenses (U\$S), por lo cual continúan con su vigencia en la actualidad, debiendo simplemente realizar la conversión a moneda local al momento deseado. Es importante destacar que una de las características del producto, que permite destacarlo por sobre productos similares del mercado, es su ligereza y facilidad de implementación. Al cubrir la totalidad de las funcionalidades de manera completamente online, permite que los costos asociados al mismo sean considerablemente más bajos que aquellas soluciones que involucran componentes físicos (sensores, puntos de cobro digitales, tarjetas magnéticas, etc), y amortizables en corto plazo.

Otro punto destacable dentro de la factibilidad es el tiempo estimado de las actividades de despliegue. Las características propias de la arquitectura elegida permiten que las actividades propias del despliegue tomen un tiempo considerablemente corto. Su arquitectura liviana basada en microservicios permite que la configuración de las instancias de servidores utilizadas tome muy poco tiempo (aproximadamente dos semanas) en comparación con tradicionales instalaciones en servidores físicos (que muchas veces no cuentan con las características requeridas). Su facilidad y simplicidad de uso, en conjunto con los manuales de usuario realizados, también contribuye a que las capacitaciones necesarias para su correcta configuración y utilización requieran de poco tiempo.

En conclusión, realizar el estudio de factibilidad en este momento de avance del proyecto, nos permite corroborar las expectativas que teníamos en el inicio del proyecto. El proyecto tiene muchas perspectivas de crecimiento, y altas posibilidades de ser implementado

en entornos productivos. El mayor riesgo que existe consiste en las características propias de los potenciales clientes del mismo. Al estar el mismo apuntado a municipios, sabemos que existen procesos administrativos y legales que podrían dificultar las posibilidades de ofrecer nuestra solución (Por lo general, los entes municipales realizan este tipo de contrataciones a través de procesos de licitación, debiendo poseer características legales y de experiencia previa que podrían suponer un problema para el equipo). Este punto deberá ser reevaluado cuando se tenga la versión final del producto.

# Informe de avances

## Iteración 1

### Alcance

- Definición de Arquitectura
- Investigación frameworks y servicios para Android
- Configuración entorno
- Modelado de dominio
- Gestión de configuración

### Estimación

4 semanas

### Resultados

#### Duración

02/07/2017 al 23/07/2017 (~3 semanas)

### Retrospectiva

La primera iteración nos sirvió para consolidar las ideas que teníamos sobre la arquitectura de la aplicación, las herramientas y frameworks que utilizaremos para el desarrollo. Las definiciones básicas descritas en el estudio inicial probaron no ser todas aceptables (ni las mejores) para encarar el proyecto; gracias a las pruebas e investigaciones que realizamos pudimos comprender mejor el funcionamiento de algunos frameworks y mejorar las decisiones previas.

La actividad de armado de entorno se realizó mucho más rápido de lo esperado una vez que comprendimos mejor la utilización, y sobre todo la instalación, de las herramientas luego de leer la parte de la documentación.

Dentro de las cosas a mejorar identificamos dos oportunidades importantes:

1. No registramos el tiempo insumido. Esto nos limita la posibilidad de obtener feedback sobre la velocidad y capacidad real del equipo. Además, es una buena práctica para comparar contra lo estimado y mejorar el proceso de estimación.
2. Al no haber definido un marco o “espacio” donde escribir y organizar la documentación, la misma fue escasa. Esto no genera inconvenientes hoy, pero en el corto plazo, y más aún en el mediano/largo plazo, puede llevarnos a cometer un error más de una vez o deshacer una decisión que se tomó anteriormente con justificaciones.

Para la siguiente iteración, pretendemos mejorar estas cuestiones planteadas, además de intentar continuar el ritmo de trabajo de tuvimos; el cual nos pareció muy adecuado según la planificación y demanda del proyecto.

## Iteración 2

### Alcance

- [C] Registro estacionamiento
- [P] Consultar estacionamiento

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación (hs)
[C] Registro estacionamiento	Construcción	Construcción	64
[P] Consulta estacionamiento	Construcción	Construcción	32
<b>Total</b>			<b>96</b>

### Resultados

#### Duración

24/07/17 al 04/09/17 (~6 semanas)

#### Esfuerzo

Requerimiento	Tarea	Tipo	Reales (hs)
[C] Registro estacionamiento	Construcción	Construcción	49,5
[P] Consulta estacionamiento	Construcción	Construcción	10
[C] Registro estacionamiento	[pd-c-mobile] # 1 - Validación dirección nula	Bug fixing	1
[C] Registro estacionamiento	[pd-c-mobile] # 2 - Error al ingresar un valor de tiempo negativo	Bug fixing	2
[C] Registro estacionamiento	[pd-c-estacionamiento] # 1 - Error al ingresar un valor de tiempo negativo	Bug fixing	1
[C] Registro estacionamiento	[pd-c-estacionamiento] # 2 - No se almacena en BD la dirección ingresada por el usuario	Bug fixing	2
[C] Registro estacionamiento	[pd-c-estacionamiento] # 3 - La información horaria se guarda en formato UTC	Bug fixing	2
[C] Registro estacionamiento	[pd-c-mobile] # 4 - No envía el texto de dirección al realizar la petición al servicio	Bug fixing	1
[P] Consulta estacionamiento	[pd-p-mobile] # 1 - Luego de consultar el dominio no muestra ningún resultado	Bug fixing	1
No planificado	Entorno testing	No planificado	13
General	Documentación	Doc	19
<b>Total</b>			<b>101,5</b>

## Retrospectiva

La segunda iteración es la primera que tiene dentro de su alcance requerimientos del proyecto. Como tal, nos obligó a formar nuestros hábitos como equipo (formas de trabajo, motivaciones, intereses) y en el uso de las tecnologías seleccionadas.

### *Mejorar estimaciones*

Con el objetivo de poder hacer un análisis basado en información real, y no sólo percepciones subjetivas), armamos una simple tabla con los requerimientos incluidos en la iteración (separados por sus actividades), la cantidad de horas estimadas para cada uno, las horas reales insumidas y el desvío en porcentaje (calculado a partir de ambos últimos campos):

Requerimiento	Hs Estimadas	Hs Reales	Desvío
<b>[C] Registro estacionamiento</b>	<b>64</b>	<b>58,5</b>	<b>-9%</b>
Bug fixing	0	9	0%
Construcción	64	49,5	-23%
<b>[P] Consulta estacionamiento</b>	<b>32</b>	<b>11</b>	<b>-66%</b>
Bug fixing	0	1	0%
Construcción	32	10	-69%
<b>No planificado</b>	<b>0</b>	<b>32</b>	<b>0%</b>
Entorno testing	0	13	0%
Gestión	0	19	0%
<b>Total general</b>	<b>96</b>	<b>101,5</b>	<b>6%</b>

Desde el punto de vista de la planificación, y en comparación a la iteración anterior, comenzamos a incluir una estimación de los requerimientos comprometidos. Sin embargo, luego de construir un reporte con las estimaciones y horas reales que tomó cada ítem, identificamos la necesidad de desagregar las mismas en tareas más chicas.

Además, encontramos varias actividades no planificadas (“Gestión” y “Entorno Testing”) que surgieron luego de comenzada la iteración. Consideramos que esta situación es normal, pero es necesario tener en cuenta estas actividades no identificadas para la estimación de la próxima iteración.

### *Análisis de desvíos*

Consideramos que un desvío del 6% (5,5hs) es un número muy bueno para ser la primera iteración con estimaciones. Consideramos que se debe mayoritariamente a tareas no estimadas:

1. Entorno Testing: Sin bien, ya habíamos investigado y realizado una prueba de concepto sobre los entornos en la nube (AWS), no habíamos configurado ninguno basado en la estructura de nuestro proyecto. Crear la base de datos fue lo más sencillo. Pero luego, para cada aplicación que se conecta a la misma hay que tener en cuenta un grupo de seguridad, entre otras cosas.

2. Gestión: El armado de la documentación del proyecto y del producto requirió de un tiempo que nunca tuvimos en cuenta y que debemos empezar a considerar a partir de la próxima iteración.

#### *Estimar bug fixing*

Para estimar correctamente el bug fixing analizamos los resultados de la iteración:

Requerimiento	Hs reales	Ratio bug fixing - Construcción
<b>[C] Registro estacionamiento</b>		
Bug fixing	9	18,18%
<b>[P] Consulta estacionamiento</b>		
Bug fixing	1	10,00%
<b>Total general</b>	<b>69,5</b>	

Esto nos permite determinar qué porcentaje de bug fixing deberíamos aplicar según el tiempo de estimación de desarrollo. Podemos promediar un 15% del tiempo estimado a construcción (incluye Testing) que deberíamos aplicar a bug fixing a partir de la siguiente iteración.

## Iteración 3

### Alcance

- [C] Login
- [C] Menú
- [C] Consulta de tiempo disponible

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación (hs)
[C] Login	Diseño UI	Construcción	1
[C] Login	Desarrollo	Construcción	1
[C] Login	Testing	Testing	1
[C] Login	Bug fixing	Bug fixing	0,5
[C] Menú	Investigación	Construcción	3
[C] Menú	Diseño UI	Construcción	1
[C] Menú	Desarrollo	Construcción	5
[C] Menú	Testing	Testing	1
[C] Menú	Bug fixing	Bug fixing	2
[C] Consulta de tiempo disponible	Diseño UI	Construcción	2
[C] Consulta de tiempo disponible	Desarrollo Front end	Construcción	5
[C] Consulta de tiempo disponible	Desarrollo Back end	Construcción	10
[C] Consulta de tiempo disponible	API	Construcción	2
[C] Consulta de tiempo disponible	Testing	Testing	2
[C] Consulta de tiempo disponible	Bug fixing	Bug fixing	3
General	Documentación	Doc	6
<b>Total</b>			<b>45,5</b>

## Resultados

## Duración

12/09/17 al 20/10/17 (~5 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales (hs)
[C] Login	Diseño UI	Construcción	1
[C] Login	Desarrollo	Construcción	3
[C] Login	Testing	Testing	2
[C] Login	[pd-c-mobile] # 5 - Al ingresar un usuario inexistente, el sistema muestra el mensaje "Contraseña incorrecta"	Bug fixing	0,5
[C] Menú	Investigación	Construcción	2
[C] Menú	Diseño UI	Construcción	3
[C] Menú	Desarrollo	Construcción	4
[C] Menú	Testing	Testing	2
[C] Menú	Bug fixing	Bug fixing	0
[C] Consulta de tiempo disponible	Diseño UI	Construcción	4
[C] Consulta de tiempo disponible	Desarrollo Front end	Construcción	15
[C] Consulta de tiempo disponible	Desarrollo Back end	Construcción	8,5
[C] Consulta de tiempo disponible	API	Construcción	1
[C] Consulta de tiempo disponible	Testing	Testing	3,5
[C] Consulta de tiempo disponible	Bug fixing	Bug fixing	0
General	Documentación	Doc	10
No planificado	Presentación Avances	No planificado	8
<b>Total</b>			<b>67,5</b>



## Análisis de desvíos

Tarea	Hs Estimadas	Hs Reales	Desvío
[C] Login	3,5	6,5	86%
[C] Menú	12	11	-8%
[C] Consulta de tiempo disponible	24	32	33%
General	6	10	67%
Documentación	6	10	67%
<b>Total general</b>	<b>45,5</b>	<b>59,5</b>	<b>31%</b>

## Actividades No planificadas

Requerimiento	Hs
No planificado	8
Presentación Avances	8
<b>Total general</b>	<b>8</b>

## Retrospectiva

*Corregir los desvíos*

En la tercera iteración tuvimos un desvío mayor en relación con la anterior, alcanzando un 31% de lo planificado. No identificamos un problema o situación particular como responsable de este desvío, simplemente la estimación no reflejó el esfuerzo requerido.

Sin embargo, logramos identificar que las actividades que generaron los mayores desvíos son las de Diseño UI y codificación Front End (en el caso de Login y Menú se llaman simplemente "Codificación"). Esto guarda relación con nuestro bajo conocimiento de la tecnología Android, lo cual es uno de los desafíos más importantes desde el punto de vista técnicos asumidos en este proyecto.

Para la próxima iteración intentaremos mejorar las estimaciones, intentado no subestimar los componentes de presentación desarrollados bajo la plataforma Android.

## Iteración 4

### Alcance

- [C] Determinar ubicación
- [A] Reporte de Estacionamientos por hora
- [C] Manual de Usuario
- [P] Manual de Usuario
- Análisis de Factibilidad

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación (hs)
[C] Determinar ubicación	Diseño UI	Construcción	5
[C] Determinar ubicación	Desarrollo	Construcción	15
[C] Determinar ubicación	Testing	Testing	3
[C] Determinar ubicación	Bug fixing	Bug fixing	2
[A] Reporte de Estacionamientos por hora	Diseño template	Construcción	5
[A] Reporte de Estacionamientos por hora	Generación	Construcción	10
[A] Reporte de Estacionamientos por hora	Descarga web	Construcción	2
[A] Reporte de Estacionamientos por hora	Testing	Testing	2
[A] Reporte de Estacionamientos por hora	Bug fixing	Bug fixing	1
[C] Manual de usuario	Manual de usuario	Doc	6
[P] Manual de usuario	Manual de usuario	Doc	4
General	Documentación	Doc	10
General	Análisis de factibilidad	Doc	6
<b>Total</b>			<b>71</b>

## Resultados

## Duración

22/10/17 al 28/11/17 (~5 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales (hs)
[C] Determinar ubicación	Diseño UI	Construcción	4
[C] Determinar ubicación	Desarrollo	Construcción	14
[C] Determinar ubicación	Testing	Testing	2
[C] Determinar ubicación	Bug fixing	Bug fixing	0
[A] Reporte de Estacionamientos por hora	Diseño template	Construcción	6
[A] Reporte de Estacionamientos por hora	Generación	Construcción	10
[A] Reporte de Estacionamientos por hora	Descarga web	Construcción	2
[A] Reporte de Estacionamientos por hora	Testing	Testing	3
[A] Reporte de Estacionamientos por hora	Bug fixing	Bug fixing	1
[C] Manual de usuario	Manual de usuario	Doc	4
[P] Manual de usuario	Manual de usuario	Doc	1
[P] Login	Desarrollo	No planificado	4
General	Análisis de factibilidad	Doc	4
General	Documentación	Doc	6
Cierre regularidad	Documentación técnica	No planificado	8
Cierre regularidad	Documentación de gestión	No planificado	4
Cierre regularidad	Documentación de proyecto	No planificado	4
Cierre regularidad	Presentación	No planificado	16
<b>Total</b>			<b>93</b>

### Estimado vs real

La siguiente tabla muestra una relación entre las horas estimadas y las horas consumidas por cada tarea

Tarea	Hs Estimadas	Hs Reales	Desvío
[C] Determinar ubicación	25	20	-20%
[A] Reporte de Estacionamientos por hora	20	22	10%
[C] Manual de usuario	6	4	-33%
[P] Manual de usuario	4	1	-75%
[P] Login	0	4	0%
General	16	10	-38%
Cierre regularidad	0	32	0%
<b>Total general</b>	<b>71</b>	<b>93</b>	<b>31%</b>

Esta tabla nos permite inferir que:

- Las estimaciones fueron muy acertadas, habiendo tomado el total de actividades un 20% menos de lo previsto.
- El único requerimiento que tuvo desvío positivo es *[A] Reporte de Estacionamientos por hora* (un 10%), correspondiendo éste con el primer reporte desarrollado por el equipo.
- Se logró revertir la constante de que la actividad de Gestión de cada iteración toma más de lo estimado (se desvía).

### Porcentaje por tipo de actividad

El siguiente cuadro muestra el porcentaje de tiempo que tomo cada tipo de actividad del total del esfuerzo de la iteración:

Tipo de actividad	% por tipo
Construcción	38,71%
No planificado	38,71%
Doc	16,13%
Testing	5,38%
Bug fixing	1,08%

Analizando el cuadro Porcentaje por tipo de actividad apreciamos que en esta iteración se dedicó mucho tiempo a actividades no planificadas (cerca del 40%).

## Retrospectiva

### *Reducir las tareas no planificadas*

Analizamos las causas que llevaron a que se agreguen estas tareas sin estimación a una iteración ya comenzada. Son tareas relacionadas al cierre de la regularidad; como no van a volver a ocurrir, no creemos conveniente tomar ninguna medida para prevenirlas en el futuro.

### *Alcance de la iteración*

Según el *Estudio inicial*, esta iteración debería haber incluido los siguientes items:

- [C] Pago (integración con Google Wallet)
- [P] Login
- [A] Login

Sin embargo, debido a los requerimientos de la cátedra para regularizar la asignatura, debimos cambiarlos por los items que realmente se entregaron.

Consideramos que es un logro poder flexibilizar el alcance para cumplir los objetivos. Es una muestra de agilidad.

### *Revisión del backlog del producto*

Una mejora que queremos introducir es, antes de iniciar cada iteración, revisar y refinar el backlog del producto para tener la oportunidad de agregar, cancelar o modificar items.

Al concluir esta iteración, se agregó al producto deuda técnica que nos puede dificultar avanzar con la velocidad esperada; vamos a corregir esa deuda y necesitamos que este en el backlog del producto.

## Iteración 5

### Product Backlog

- [C] Reutilizar componente de menú
- [C] Manejar respuesta de servicios
- [C] Mejorar la precisión de la geolocalización
- [C] Manejo de excepciones
- [C] Autenticación de usuario
- [C] Pago
- [C] Registro de vehículos
- [A] Carga de información de calles
- [C] Consulta información de calles
- [C] Agregar tiempo de estacionamiento
- [C] Ubicar vehículo estacionado
- [K] Ingreso manual de ubicación
- [K] Login
- [A] Definición tarifa única

### Alcance

- [C] Reutilizar componente de menú
- [C] Manejar respuesta de servicios
- [C] Mejorar la precisión de la geolocalización
- [C] Manejo de excepciones
- [C] Autenticación de usuario

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación (hs)
[C] Reutilizar componente de menú	Investigación	Construcción	6
[C] Reutilizar componente de menú	Desarrollo	Construcción	6
[C] Reutilizar componente de menú	Testing	Testing	1
[C] Reutilizar componente de menú	Bug fixing	Bug fixing	0,5
[C] Manejar respuesta de servicios	Desarrollo	Construcción	6
[C] Manejar respuesta de servicios	Testing	Testing	1
[C] Manejar respuesta de servicios	Bug fixing	Bug fixing	0,5
[C] Mejorar la precisión de la geolocalización	Investigación	Construcción	4
[C] Mejorar la precisión de la geolocalización	Desarrollo	Construcción	6
[C] Mejorar la precisión de la geolocalización	Testing	Testing	2
[C] Mejorar la precisión de la geolocalización	Bug fixing	Bug fixing	1
[C] Manejo de excepciones	Investigación	Construcción	3
[C] Manejo de excepciones	Desarrollo	Construcción	2

## ParCo | Proyecto

[C] Autenticación de usuario	Desarrollo Front end	Construcción	2
[C] Autenticación de usuario	Desarrollo Back end	Construcción	4
[C] Autenticación de usuario	Testing	Testing	2
[C] Autenticación de usuario	Bug fixing	Bug fixing	1
General	Gestion	Gestion	8
<b>Total</b>			<b>56</b>

## Objetivo

Con el objetivo de completar los compromisos de la iteración anterior a tiempo, decidimos aplazar algunas cuestiones técnicas, las cuales identificamos ahora. Por ende, esta iteración va a estar compuesta en su totalidad por deuda técnica, la cual nos va a permitir seguir construyendo sobre una base más robusta; consideramos esto como una inversión de esfuerzo ahora, que nos va a permitir avanzar más rápidamente y con menor defectos en el futuro.

## Resultados

## Duración

28/12/17 al 26/12/17 (~4 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales (hs)
[C] Reutilizar componente de menú	Investigación	Construcción	6
[C] Reutilizar componente de menú	Desarrollo	Construcción	8
[C] Reutilizar componente de menú	Testing	Testing	2
[C] Reutilizar componente de menú	Bug fixing	Bug fixing	1
[C] Manejar respuesta de servicios	Desarrollo	Construcción	6
[C] Manejar respuesta de servicios	Testing	Testing	2
[C] Manejar respuesta de servicios	Bug fixing	Bug fixing	3
[C] Mejorar la precisión de la geolocalización	Investigación	Construcción	4
[C] Mejorar la precisión de la geolocalización	Desarrollo	Construcción	6
[C] Mejorar la precisión de la geolocalización	Testing	Testing	2
[C] Mejorar la precisión de la geolocalización	Bug fixing	Bug fixing	0
[C] Manejo de excepciones	Investigación	Construcción	1
[C] Manejo de excepciones	Desarrollo	Construcción	2
[C] Login	Desarrollo Front end	Construcción	3
[C] Login	Desarrollo Back end	Construcción	3
[C] Login	Testing	Testing	2
[C] Login	Bug fixing	Bug fixing	0
General	Documentación	Doc	8
<b>Total</b>			<b>59</b>



## Estimado vs real

Tarea	Hs Estimadas	Hs Reales	Desvío
[C] Reutilizar componente de menú	13,5	17	26%
[C] Manejar respuesta de servicios	7,5	11	47%
[C] Mejorar la precisión de la geolocalización	13	12	-8%
[C] Manejo de excepciones	5	3	-40%
[C] Login	9	8	-11%
General	8	8	0%
<b>Total general</b>	<b>56</b>	<b>59</b>	<b>5%</b>

## Retrospectiva

*Mejora de la calidad*

En esta iteración corregimos gran parte de la deuda técnica identificada del producto. Si bien, puede parecer que pasaron casi 4 semanas en las cuales no hubieron muchos avances funcionales, estamos convencidos que la deuda técnica nos puede traer complicaciones a futuro. Creemos que fue la decisión correcta.

*Unidad de medida de las estimaciones*

Vamos a cambiar la unidad de medida de las estimaciones de *horas* a *días*. La razón de esta decisión es que nos va a facilitar registrar los avances. Y también, las estimaciones; lo más común en este procedimiento, es que intentamos pensar cuántos días nos puede tomar cada tarea, y luego lo traducimos a horas. Ahora nos vamos a ahorrar este último paso.

## Iteración 6

### Product Backlog

- [C] Pago
- [A] Carga de información de calles
- [C] Consulta información de calles
- [C] Registro de vehículos
- [C] Agregar tiempo de estacionamiento
- [C] Ubicar vehículo estacionado
- [K] Ingreso manual de ubicación
- [K] Login
- [A] Definición tarifa única

## Alcance

- [C] Pago
- [A] Carga de información de calles

## Estimaciones

Requerimiento	Tarea	Tipo	Estimación (d)
[C] Pago	Integración Google Pay	Investigación	3
[C] Pago	Integración Google Pay	Construcción	5
[C] Pago	Pagos de prueba	Investigación	1
[C] Pago	Pago	Testing	2
[C] Pago	[Regresión] Estacionamiento	Testing	1
[C] Pago	Bug fixing	Bug fixing	2
[A] Carga de información de calles	Integración Google Maps	Investigación	1
[A] Carga de información de calles	Formularios Angular	Investigación	2
[A] Carga de información de calles	Integración Google Maps	Construcción	2
[A] Carga de información de calles	Formulario de información de calle	Construcción	5
[A] Carga de información de calles	Listado de calles	Construcción	2
[A] Carga de información de calles	Nueva calle	Testing	1
[A] Carga de información de calles	Modificación de calle	Testing	1
[A] Carga de información de calles	Eliminar calle	Testing	1
[A] Carga de información de calles	Listar calles	Testing	1
[A] Carga de información de calles	ABMC calles	Bug fixing	2
General	Documentación iteración	Doc	2
<b>Total</b>			<b>34</b>

## Resultados

## Duración

04/12/17 al 22/04/18 (~20 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Real (d)
[C] Pago	Integración Google Pay	Investigación	5
[C] Pago	Integración Google Pay	Construcción	1
[C] Pago	Pagos de prueba	Investigación	6
[C] Pago	Pago	Testing	4
[C] Pago	[Regresión] Estacionamiento	Testing	1
[C] Pago	Bug fixing	Bug fixing	3
[C] Pago	Integración MercadoPago	Investigación	2
[C] Pago	Integración MercadoPago	Construcción	13
[A] Carga de información de calles	Integración Google Maps	Investigación	2
[A] Carga de información de calles	Formularios Angular	Investigación	2
[A] Carga de información de calles	Integración Google Maps	Construcción	5
[A] Carga de información de calles	Formulario de información de calle	Construcción	14
[A] Carga de información de calles	Listado de calles	Construcción	3
[A] Carga de información de calles	Nueva calle	Testing	1
[A] Carga de información de calles	Modificación de calle	Testing	2
[A] Carga de información de calles	Eliminar calle	Testing	1
[A] Carga de información de calles	Listar calles	Testing	1
[A] Carga de información de calles	ABMC calles	Bug fixing	3
General	Documentación iteración	Doc	2
<b>Total</b>			<b>71</b>

## Cumplimiento

Para comparar la estimación del esfuerzo con el tiempo real que tomó la iteración registramos ambos datos de cada tarea y luego calculamos el desvío en porcentaje:

Tarea	Días Estimados	Días Reales	Desvío
General	2	2	0%
[C] Pago	14	35	150%
[A] Carga de información de calles	18	34	89%
Total general	34	71	109%

Como la tabla indica, el esfuerzo real es más del doble del estimado. Vamos a analizar cada requerimiento por separado; comenzamos con **[C] Pago**:

Tarea	Tipo	Días Estimados	Días Reales	Desvío
<b>Integración Google Pay</b>				
	Investigación	3	5	67%
	Construcción	5	1	-80%
<b>Integración MercadoPago</b>				
	Investigación	0	2	0%
	Construcción	0	13	0%
<b>Pagos de prueba</b>				
	Investigación	1	6	500%
<b>Pago</b>				
	Testing	2	4	100%
<b>[Regresión] Estacionamiento</b>				
	Testing	1	1	0%
<b>Bug fixing</b>				
	Bug fixing	2	3	50%
<b>Total general</b>		<b>14</b>	<b>35</b>	<b>150%</b>

La principal causa del desvío es el retrabajo en las tareas de investigación y construcción. La plataforma candidata para integrar era *Google Pay*, pero tuvimos dos dificultades fundamentales para hacer la integración. En primer lugar, los entornos de desarrollo y prueba (llamados *Sandbox*) requieren muchas configuraciones previas dependientes del dispositivo, e inclusive registrar los datos de una tarjeta real para habilitar la API. En segundo lugar, la plataforma brinda una SDK (la forma de integración recomendada por el proveedor) que se integra con mayor facilidad que la API REST, pero solo para versiones de Android 5.0 o superior (recordemos que Parco fue desarrollada para ser compatible con Android 4.4, para dar soporte a una mayor cantidad de usuarios). La integración con la API es más compleja y cuenta con menos documentación.

Entonces, tomamos la decisión de buscar otra alternativa como plataforma de pago, descartando los avances en las tareas **Integración Google Pay** (investigación y construcción) y **Pagos de prueba**.

Analizando alternativas, finalmente optamos por *MercadoPago*, la cual cumple con todas nuestras necesidades identificadas. Al desconocer la complejidad preferimos no realizar una

nueva estimación de las tareas a rehacer; consideramos que una nueva estimación no nos iba a brindar información sobre la cual accionar. En su lugar, preferimos dedicar más tiempo en el análisis técnico para asegurarnos que las decisiones tomadas sean mejores.

Por último, analizamos las tareas de **[A] Carga de información de calles:**

Tarea	Tipo	Días Estimados	Días Reales	Desvío
<b>Integración Google Maps</b>				
	Investigación	1	2	100%
	Construcción	2	5	150%
<b>Formularios Angular</b>				
	Investigación	2	2	0%
<b>Formulario de información de calle</b>				
	Construcción	5	14	180%
<b>Listado de calles</b>				
	Construcción	2	3	50%
<b>Nueva calle</b>				
	Testing	1	1	0%
<b>Modificación de calle</b>				
	Testing	1	2	100%
<b>Eliminar calle</b>				
	Testing	1	1	0%
<b>Listar calles</b>				
	Testing	1	1	0%
<b>ABMC calles</b>				
	Bug fixing	2	3	50%
<b>Total general</b>		<b>18</b>	<b>34</b>	<b>89%</b>

En este caso, la única causa de los desvíos que pudimos identificar es la subestimación debido al desconocimiento del framework usado, Angular 7. Si bien esto se tuvo en cuenta a la hora de estimar, la integración con *Google Maps* hizo que gran parte del trabajo finalizado se tenga que reescribir para que pueda ser integrado. La tabla analizada ratifica que el grueso del desvío está asociado a tareas de construcción.

## Retrospectiva

Concluyó una iteración que se prolongó por más del doble del tiempo estimado. Las causas ya fueron descritas y analizadas en cada requerimiento, así que ahora tenemos que razonar decisiones que nos permitan mejorar nuestro rendimiento y evitar repetir los errores cometidos.

### *Identificar y comprometer requerimientos más chicos*

En primer lugar, cuando se analiza y se diseña una solución técnica para un requerimiento *grande* y luego, mientras el desarrollo está en curso, se comprueba que este diseño no es viable, debemos revisar y, quizás, descartar todo el desarrollo. Esto claramente va en contra de la agilidad que queremos lograr. El primer cambio que vamos a intentar es partir los próximos requerimientos en bloques más pequeños y priorizarlos de tal forma que los que contengan tareas con mayor riesgo estén primero. Así, evitaríamos el efecto dominó que causa un problema.

## Iteración 7

### Product Backlog

- [A] Clasificación de segmentos de calles
- [C] Historial de estacionamientos
- [C] Vehículos favoritos
- [C] Validar estacionamiento tarifado
- [C] Consulta información de calles
- [C] Agregar tiempo de estacionamiento
- [C] Ubicar vehículo estacionado
- [K] Búsqueda de ubicación manual
- [K] Login
- [A] Definición tarifa única



## Alcance

- [A] Clasificación de segmentos de calles
- [C] Historial estacionamientos
- [C] Vehículos favoritos
- [C] Validar estacionamiento tarifado

## Estimaciones

Requerimiento	Tarea	Tipo	Estimación (d)
[A] Clasificación de segmentos de calles	Registrar tipo de sección de calle	Construcción	3
[A] Clasificación de segmentos de calles	Registro de secciones de calle	Testing	1
[A] Clasificación de segmentos de calles	Bug fixing	Bug fixing	1
[C] Validar estacionamiento tarifado	Validar ubicación habilitada	Construcción	1
[C] Validar estacionamiento tarifado	Validación de registro de estacionamiento	Testing	1
[C] Validar estacionamiento tarifado	Bug fixing	Bug fixing	1
[C] Historial estacionamientos	Android cards	Investigación	1
[C] Historial estacionamientos	Consultar datos de historial	Construcción	2
[C] Historial estacionamientos	Diseñar las "cards"	Construcción	2
[C] Historial estacionamientos	Mostrar historial en "cards"	Construcción	2
[C] Historial estacionamientos	Consultar historial de estacionamientos	Testing	1
[C] Historial estacionamientos	Bug fixing	Bug fixing	1
[C] Vehículos favoritos	Consultar dominios favoritos	Construcción	1
[C] Vehículos favoritos	Proponer dominio favorito	Construcción	1
[C] Vehículos favoritos	Proponer dominios favoritos (filtrado)	Construcción	1
[C] Vehículos favoritos	Proponer dominio favorito	Testing	0,5
[C] Vehículos favoritos	Proponer dominios favoritos (filtrado)	Testing	0,5
[C] Vehículos favoritos	Bug fixing	Bug fixing	1
General	Documentación	Doc	3
<b>Total</b>			<b>25</b>

## Resultados

## Duración

21/05/18 al 22/07/18 (~9 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales (d)
[A] Clasificación de segmentos de calles	Registrar tipo de sección de calle	Construcción	4
[A] Clasificación de segmentos de calles	Registro de secciones de calle	Testing	1
[A] Clasificación de segmentos de calles	Bug fixing	Bug fixing	1
[C] Validar estacionamiento tarifado	Validar ubicación habilitada	Construcción	2
[C] Validar estacionamiento tarifado	Validación de registro de estacionamiento	Testing	1
[C] Validar estacionamiento tarifado	Bug fixing	Bug fixing	0
[C] Historial estacionamientos	Android cards	Investigación	1
[C] Historial estacionamientos	Consultar datos de historial	Construcción	2
[C] Historial estacionamientos	Diseñar las "cards"	Construcción	1
[C] Historial estacionamientos	Mostrar historial en "cards"	Construcción	4
[C] Historial estacionamientos	Consultar historial de estacionamientos	Testing	1
[C] Historial estacionamientos	Bug fixing	Bug fixing	1
[C] Vehículos favoritos	Consultar dominios favoritos	Construcción	1
[C] Vehículos favoritos	Proponer dominio favorito	Construcción	2
[C] Vehículos favoritos	Proponer dominios favoritos (filtrado)	Construcción	1
[C] Vehículos favoritos	Proponer dominio favorito	Testing	0,5
[C] Vehículos favoritos	Proponer dominios favoritos (filtrado)	Testing	0,5
[C] Vehículos favoritos	Bug fixing	Bug fixing	0
General	Documentación	Doc	3
<b>Total</b>			<b>27</b>

## Retrospectiva

*Mejoramos las estimaciones*

Logramos corregir desvíos en las estimaciones que tuvimos la iteración pasada. Principalmente creemos que fue correcta la definición de partir los requerimientos en tareas más pequeñas, fáciles de estimar y de detectar desvíos. También, separamos en tareas las integraciones de algún componente que, inherentemente tienen mayor incertidumbre.

## Iteración 8

### Product backlog

- [A] Dashboard de administración del sistema
- [C] Mapa de calles
- [A] [C] Definición tarifa única
- [C] Agregar tiempo de estacionamiento
- [C] Ubicar vehículo estacionado
- [K] Búsqueda de ubicación manual
- [K] Login

## Alcance

- [A] Dashboard de administración del sistema
- [C] Mapa de calles
- [A] [C] Definición tarifa única

## Estimaciones

Requerimiento	Tarea	Tipo	Estimación (d)
[A] Dashboard de administración del sistema	Menú y navegación	Investigación	1
[A] Dashboard de administración del sistema	Template dashboard	Investigación	1
[A] Dashboard de administración del sistema	Menú y navegación	Construcción	1
[A] Dashboard de administración del sistema	Maquetado de tarjetas de indicadores	Construcción	5
[C] Mapa de calles	Google Map en Android	Investigación	3
[C] Mapa de calles	Consulta de calles	Construcción	2
[C] Mapa de calles	Dibujado de calles	Construcción	5
[C] Mapa de calles	Información de calles tarifadas y prohibidas	Testing	1
[C] Mapa de calles	Bug fixing	Bug fixing	1
[A] [C] Definición Tarifa Única	[A] Registrar nueva tarifa	Construcción	1
[A] [C] Definición Tarifa Única	[C] Calcular precio estacionamiento	Construcción	1
[A] [C] Definición Tarifa Única	[C] Carga de detalles de MercadoPago	Construcción	3
[A] [C] Definición Tarifa Única	[A] Registro de nueva tarifa	Testing	1
[A] [C] Definición Tarifa Única	[C] Precio del estacionamiento	Testing	1
[A] [C] Definición Tarifa Única	[A] Bug fixing	Bug fixing	0,5
[A] [C] Definición Tarifa Única	[C] Bug fixing	Bug fixing	0,5
General	Documentación	Doc	2
<b>Total</b>			<b>30</b>

## Resultados

## Duración

23/07/18 al 07/10/18 (~11 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales (d)
[A] Dashboard de administración del sistema	Menu y navegación	Investigación	2
[A] Dashboard de administración del sistema	Template dashboard	Investigación	2
[A] Dashboard de administración del sistema	Menu y navegación	Construcción	0,5
[A] Dashboard de administración del sistema	Maquetado de tarjetas de indicadores	Construcción	7
[C] Mapa de calles	Google Map en Android	Investigación	1
[C] Mapa de calles	Consulta de calles	Construcción	3
[C] Mapa de calles	Dibujado de calles	Construcción	7
[C] Mapa de calles	Información de calles tarifadas y prohibidas	Testing	1
[C] Mapa de calles	Bug fixing	Bug fixing	0
[A] [C] Definición Tarifa Única	[A] Registrar nueva tarifa	Construcción	3
[A] [C] Definición Tarifa Única	[C] Calcular precio estacionamiento	Construcción	1
[A] [C] Definición Tarifa Única	[C] Carga de detalles de MercadoPago	Construcción	2
[A] [C] Definición Tarifa Única	[A] Registro de nueva tarifa	Testing	0,5
[A] [C] Definición Tarifa Única	[C] Precio del estacionamiento	Testing	1
[A] [C] Definición Tarifa Única	[A] Bug fixing	Bug fixing	0,5
[A] [C] Definición Tarifa Única	[C] Bug fixing	Bug fixing	0,5
General	Documentación	Doc	2
<b>Total</b>			<b>34</b>

## Análisis de desvíos

Requerimiento	Días Estimados	Días Reales	Desvío
[A] Dashboard de administración del sistema	8	11,5	44%
[C] Mapa de calles	12	12	0%
[A] [C] Definición Tarifa Única	8	8,5	6%
General	2	2	0%
Total general	30	34	13%

El requerimiento con mayor desvío claramente es **[A] Dashboard de administración del sistema**. En este caso, simplemente subestimamos las tareas de construcción debido a que contamos con una maqueta para descargar gratuitamente; lo que llevó a pensar que no quedaban muchas cosas por hacer. Sin embargo, tomar la maqueta y adaptarla para nuestro proyecto con Angular fue más complicado de lo que parecía inicialmente.

## Retrospectiva

*Nuevos requerimientos*

Desde la *iteración 5* hemos incluido varios requerimientos nuevos (funcionales y deudas técnicas). Si bien, está demorando la entrega final del producto, estamos convencidos de que es algo bueno ya que estamos construyendo un producto más amplio, que abarca más necesidades identificadas.

## Iteración 9

### Product Backlog

- [C] Agregar tiempo de estacionamiento
- [C] Redirección a estacionamiento vigente
- [C] Ubicar vehículo estacionado
- [A] Mapa de calor
- [K] Búsqueda de ubicación manual
- [K] Login

## Alcance

- [C] Agregar tiempo de estacionamiento
- [C] Redirección a estacionamiento vigente
- [A] Mapa de calor

## Estimaciones

Requerimiento	Tarea	Tipo	Estimación
[C] Agregar tiempo de estacionamiento	Modales en Android	Investigación	1
[C] Agregar tiempo de estacionamiento	Actualizar estacionamiento	Construcción	3
[C] Agregar tiempo de estacionamiento	Modal con tiempo a agregar	Construcción	4
[C] Agregar tiempo de estacionamiento	Actualizar estacionamiento activo	Testing	1
[C] Agregar tiempo de estacionamiento	Bug fixing	Bug fixing	1
[C] Redirección a estacionamiento vigente	Abrir Google Maps	Investigación	2
[C] Redirección a estacionamiento vigente	Consulta estacionamiento vigente	Construcción	1
[C] Redirección a estacionamiento vigente	Redirección a pantalla de estacionamiento	Construcción	4
[C] Redirección a estacionamiento vigente	Redirección con datos de estacionamiento vigente	Testing	1
[C] Redirección a estacionamiento vigente	Bug fixing	Bug fixing	1
[A] Mapa de calor	Google Heatmap	Investigación	3
[A] Mapa de calor	Consultar estadísticas de estacionamientos	Construcción	3
[A] Mapa de calor	Dibujar capa heatmap sobre el mapa	Construcción	6
[A] Mapa de calor	Mapa de calor para fecha actual	Testing	2
[A] Mapa de calor	Bug fixing	Bug fixing	2
General	Documentación	Doc	2
<b>Total</b>			<b>37</b>



## Resultados

## Duración

08/10/18 al 20/01/19 (~13 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales (d)
[C] Agregar tiempo de estacionamiento	Modales en Android	Investigación	2
[C] Agregar tiempo de estacionamiento	Actualizar estacionamiento	Construcción	3
[C] Agregar tiempo de estacionamiento	Modal con tiempo a agregar	Construcción	3
[C] Agregar tiempo de estacionamiento	Actualizar estacionamiento activo	Testing	1
[C] Agregar tiempo de estacionamiento	Bug fixing	Bug fixing	0
[C] Redirección a estacionamiento vigente	Abrir Google Maps	Investigación	2
[C] Redirección a estacionamiento vigente	Consulta estacionamiento vigente	Construcción	2
[C] Redirección a estacionamiento vigente	Redirección a pantalla de estacionamiento	Construcción	2
[C] Redirección a estacionamiento vigente	Redirección con datos de estacionamiento vigente	Testing	1
[C] Redirección a estacionamiento vigente	Bug fixing	Bug fixing	2
[A] Mapa de calor	Google Heatmap	Investigación	3
[A] Mapa de calor	Consultar estadísticas de estacionamientos	Construcción	4
[A] Mapa de calor	Dibujar capa heatmap sobre el mapa	Construcción	3
[A] Mapa de calor	Mapa de calor para fecha actual	Testing	2
[A] Mapa de calor	Bug fixing	Bug fixing	1
General	Documentación	Doc	2
<b>Total</b>			<b>33</b>

## Análisis de desvíos

Requerimiento	Días Estimados	Días Reales	Desvío
[A] Mapa de calor	16	13	-19%
[C] Redirección a estacionamiento vigente	9	9	0%
[C] Agregar tiempo de estacionamiento	10	9	-10%
General	2	2	0%
Total general	37	33	-11%

En esta iteración no tuvimos desvío positivo en ningún requerimiento. Quizás, pensamos que el requerimiento **[A] Mapa de calor** llevaría más esfuerzo debido a que es una integración nueva; también, nos preocupaba la velocidad de respuesta por todos los datos que hay que procesar eventualmente. Afortunadamente, la documentación que encontramos es clara y el prerequisite más complejo ya estaba cumplido al tener integrado el mapa de Google en la aplicación.

## Retrospectiva

*Alcance acotado*

Estamos muy conformes con la estrategia que usamos para encarar el requerimiento **[A] Mapa de calor**. Acotamos el alcance del mismo a la integración del servicio y al procesamiento de los datos para enfocarnos en esos problemas. Entendemos que se debe seguir trabajando sobre filtros o parametrizaciones pero, así podemos estimar mejor la siguiente iteración.

## Iteración 10

### Product Backlog

- [A] Indicadores online
- [A] Filtrar fechas en mapa de calor
- [C] Restricción estacionamiento mismo dominio
- [C] Ubicar vehículo estacionado
- [K] Búsqueda de ubicación manual
- [K] Login

### Alcance

- [A] Indicadores online
- [A] Filtrar fechas en mapa de calor

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación
[A] Filtrar fechas en mapa de calor	Filtrar datos	Construcción	2
[A] Filtrar fechas en mapa de calor	Aplicar filtros	Construcción	2
[A] Filtrar fechas en mapa de calor	Testing	Testing	2
[A] Filtrar fechas en mapa de calor	Bug fixing	Bug fixing	1
[A] Indicadores online	Cantidad de estacionamientos activos	Construcción	1
[A] Indicadores online	Cant. Estacionamientos registrados por día	Construcción	3
[A] Indicadores online	Historial de tarifas	Construcción	3
[A] Indicadores online	Descarga de reporte "Estacionamientos por hora"	Construcción	1
[A] Indicadores online	Cantidad de estacionamientos activos	Testing	0,5
[A] Indicadores online	Cant. Estacionamientos registrados por día	Testing	2
[A] Indicadores online	Historial de tarifas	Testing	1
[A] Indicadores online	Descarga de reporte "Estacionamientos por hora"	Testing	0,5
[A] Indicadores online	Bug fixing	Bug fixing	2
General	Documentación	Doc	2
<b>Total</b>			<b>23</b>

## Resultados

## Duración

04/03/19 al 07/04/19 (~5 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales
[A] Filtrar fechas en mapa de calor	Filtrar datos	Construcción	2
[A] Filtrar fechas en mapa de calor	Aplicar filtros	Construcción	4
[A] Filtrar fechas en mapa de calor	Testing	Testing	2
[A] Filtrar fechas en mapa de calor	Bug fixing	Bug fixing	4
[A] Indicadores online	Cantidad de estacionamientos activos	Construcción	1
[A] Indicadores online	Cant. Estacionamientos registrados por día	Construcción	3
[A] Indicadores online	Historial de tarifas	Construcción	2
[A] Indicadores online	Descarga de reporte "Estacionamientos por hora"	Construcción	3
[A] Indicadores online	Cantidad de estacionamientos activos	Testing	0,5
[A] Indicadores online	Cant. Estacionamientos registrados por día	Testing	1
[A] Indicadores online	Historial de tarifas	Testing	0,5
[A] Indicadores online	Descarga de reporte "Estacionamientos por hora"	Testing	0,5
[A] Indicadores online	Bug fixing	Bug fixing	2
General	Documentación	Doc	2
<b>Total</b>			<b>27,5</b>

## Análisis de desvíos

Requerimiento	Días Estimados	Días Reales	Desvío
[A] Filtrar fechas en mapa de calor	7	12	71%
[A] Indicadores online	14	13,5	-4%
General	2	2	0%
Total general	23	27,5	20%

El requerimiento **[A] Filtrar fechas en mapa de calor** es el causante de todo el desvío.

El motivo principal es el esfuerzo de bug fixing que tuvo. Se encontraron varios defectos que luego resultaron ser un problema de datos, debido a que se registraban estacionamientos fuera del ejido municipal que disminuyen la intensidad de los colores del mapa. Todos esos defectos requirieron tiempo de análisis y se probaron distintas correcciones técnicas (sin resultados) hasta que se detectó el verdadero causante.

## Retrospectiva

*Mejora del producto*

Nos gustó mucho poder pensar en las necesidades de información para la toma de decisiones de los usuarios que administran el sistema. Es un desafío complicado, y probablemente diferentes municipios tengan diferentes necesidades, por lo que nos enfocamos en las más básicas y obvias. Aún así, creemos que el producto dió un salto de calidad brindando beneficios más allá de la transaccionalidad.

## Iteración 11

### Product Backlog

- [K] Creación app
- [K] Registro estacionamiento
- [A] Filtro de secciones por nombre
- [C] Restricción estacionamiento mismo dominio
- [C] Ubicar vehículo estacionado

### Alcance

- [K] Creación app
- [K] Registro estacionamiento
- [A] Filtro de secciones por nombre

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación
[K] Creación app	Fork cliente conductor	Construcción	1
[K] Creación app	Menu y navegación	Construcción	2
[K] Registro estacionamiento	Ingreso de direccion manual	Construcción	2
[K] Registro estacionamiento	Registrar estacionamiento	Construcción	2
[K] Registro estacionamiento	Pago del estacionamiento	Construcción	1
[K] Registro estacionamiento	Registrar estacionamiento	Testing	1
[K] Registro estacionamiento	Bug fixing	Bug fixing	1
[A] Filtro de secciones por nombre	Filtrar listas en Angular	Investigación	1
[A] Filtro de secciones por nombre	Filtro por nombre	Construcción	1
[A] Filtro de secciones por nombre	Filtrar secciones por nombre	Testing	1
[A] Filtro de secciones por nombre	Bug fixing	Bug fixing	1
General	Documentación	Doc	2
General	Presentación Estudio Inicial	Doc	4
<b>Total</b>			<b>20</b>

## Resultados

## Duración

15/04/19 al 19/05/19 (~5 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales
[K] Creación app	Fork cliente conductor	Construcción	2
[K] Creación app	Menu y navegación	Construcción	2
[K] Registro estacionamiento	Ingreso de direccion manual	Construcción	3
[K] Registro estacionamiento	Registrar estacionamiento	Construcción	1
[K] Registro estacionamiento	Pago del estacionamiento	Construcción	2
[K] Registro estacionamiento	Registrar estacionamiento	Testing	3
[K] Registro estacionamiento	Bug fixing	Bug fixing	1
[A] Filtro de secciones por nombre	Filtrar listas en Angular	Investigación	1
[A] Filtro de secciones por nombre	Filtro por nombre	Construcción	1
[A] Filtro de secciones por nombre	Filtrar secciones por nombre	Testing	1
[A] Filtro de secciones por nombre	Bug fixing	Bug fixing	0,5
General	Documentación	Doc	2
General	Presentación Estudio Inicial	Doc	6
<b>Total</b>			<b>25,5</b>

## Análisis de desvíos

Requerimiento	Días Estimados	Días Reales	Desvío
[K] Creación app	3	4	33%
[K] Registro estacionamiento	7	10	43%
[A] Filtro de secciones por nombre	4	3,5	-13%
General	6	8	33%
Total general	20	25,5	28%

El requerimiento **[K] Registro estacionamiento** es el que tuvo la mayor cantidad de tiempo de desvío y, de hecho tomó 3 días adicionales.

La causa principal que identificamos es el testing. El proceso de registración de estacionamiento es el más complejo del sistema. Siendo que no tuvimos inconvenientes, ni demoras, entendemos que sólo subestimamos el esfuerzo que requería, teniendo en cuenta que las pruebas eran regresiones completas.

## Retrospectiva

### *Tamaño de la iteración*

Esta iteración, al momento de planificarla, la hicimos con un tamaño más chico al que veníamos acostumbrados. Venimos achicando poco a poco el tamaño de las iteraciones y creemos que hemos llegado a un punto justo. Vamos a intentar seguir manteniendo un tamaño similar que nos permita enfocarnos en algún aspecto particular del sistema y nos de mayor visibilidad del avance de la iteración completa.



## Iteración 12

### Product Backlog

- [C] Restricción estacionamiento mismo dominio
- [C] Ubicar vehículo estacionado
- [C] Finalizar estacionamiento

### Alcance

- [C] Restricción estacionamiento mismo dominio
- [C] Ubicar vehículo estacionado
- [C] Finalizar estacionamiento

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación
[C] Restricción estacionamiento mismo dominio	Validar dominio estacionado	Construcción	1
[C] Restricción estacionamiento mismo dominio	Validar registro de estacionamiento	Construcción	1
[C] Restricción estacionamiento mismo dominio	Restringir estacionamiento	Testing	1
[C] Restricción estacionamiento mismo dominio	Bug fixing	Bug fixing	1
[C] Ubicar vehículo estacionado	Servicios de Google Maps	Investigación	2
[C] Ubicar vehículo estacionado	Link en pantalla de estacionamiento	Construcción	1
[C] Ubicar vehículo estacionado	Integración con servicios de Google Maps	Construcción	3
[C] Ubicar vehículo estacionado	Ubicar vehículo en Google Maps app	Testing	1
[C] Ubicar vehículo estacionado	Bug fixing	Bug fixing	1
[C] Finalizar estacionamiento	Permitir finalizar en pantalla de estacionamiento activo	Construcción	2
[C] Finalizar estacionamiento	Finalizar estacionamiento	Construcción	2
[C] Finalizar estacionamiento	Finalizar estacionamiento	Testing	1
[C] Finalizar estacionamiento	Bug fixing	Bug fixing	1
General	Documentación	Doc	2
<b>Total</b>			<b>20</b>

## Resultados

## Duración

19/05/19 al 07/07/19 (~7 semanas)

## Esfuerzo

Requerimiento	Tarea	Tipo	Reales
[C] Restricción estacionamiento mismo dominio	Validar dominio estacionado	Construcción	3
[C] Restricción estacionamiento mismo dominio	Validar registro de estacionamiento	Construcción	2
[C] Restricción estacionamiento mismo dominio	Restringir estacionamiento	Testing	1
[C] Restricción estacionamiento mismo dominio	Bug fixing	Bug fixing	2
[C] Ubicar vehículo estacionado	Servicios de Google Maps	Investigación	3
[C] Ubicar vehículo estacionado	Link en pantalla de estacionamiento	Construcción	1
[C] Ubicar vehículo estacionado	Integración con servicios de Google Maps	Construcción	2
[C] Ubicar vehículo estacionado	Ubicar vehículo en Google Maps app	Testing	1
[C] Ubicar vehículo estacionado	Bug fixing	Bug fixing	0
[C] Finalizar estacionamiento	Permitir estacionar en pantalla de estacionamiento activo	Construcción	1
[C] Finalizar estacionamiento	Finalizar estacionamiento	Construcción	3
[C] Finalizar estacionamiento	Finalizar estacionamiento	Testing	1
[C] Finalizar estacionamiento	Bug fixing	Bug fixing	0
General	Documentación	Doc	2
<b>Total</b>			<b>22</b>

## Análisis de desvíos

Requerimiento	Días Estimados	Días Reales	Desvío
[C] Restricción estacionamiento mismo dominio	4	8	100%
[C] Ubicar vehículo estacionado	8	7	-13%
[C] Finalizar estacionamiento	6	5	-17%
General	2	2	0%
Total general	20	22	10%

## Retrospectiva

*Estabilidad y funcionalidades finales*

Como se venía dando en las últimas iteraciones, el equipo logra alcanzar una buena precisión con las estimaciones lo que permite que no se produzcan desvíos importantes en las fechas previstas. En esta iteración nos centramos en validaciones (Restricción estacionamiento mismo dominio) y funcionalidades que nos restaban del cliente *Conductor*, pudiendo así completar la totalidad del desarrollo del módulo. El mismo fue entonces tagueado y listo para una potencial implementación.

## Iteración 13

### Product Backlog

- [P] Re-styling
- [P] Consulta de estacionamiento activo

### Alcance

- [P] Re-styling
- [P] Consulta de estacionamiento activo

### Estimaciones

Requerimiento	Tarea	Tipo	Estimación
[P] Re-styling	Layout de tablet	Investigación	1
[P] Re-styling	Adaptar el cliente a tablet	Construcción	2
[P] Re-styling	Regresión	Testing	1
[P] Re-styling	Bug fixing	Bug fixing	1
[P] Consulta de estacionamiento activo	Consultar estacionamiento	Construcción	1
[P] Consulta de estacionamiento activo	Consultar estacionamiento	Testing	1
[P] Consulta de estacionamiento activo	Bug fixing	Bug fixing	1
General	Documentación	Doc	2
<b>Total</b>			<b>10</b>

### Resultados

#### Duración

15/07/19 al 11/08/19 (~4 semanas)

#### Esfuerzo

Requerimiento	Tarea	Tipo	Reales
[P] Re-styling	Layout de tablet	Investigación	0,5
[P] Re-styling	Adaptar el cliente a tablet	Construcción	1
[P] Re-styling	Regresión	Testing	1
[P] Re-styling	Bug fixing	Bug fixing	0
[P] Consulta de estacionamiento activo	Consultar estacionamiento	Construcción	2
[P] Consulta de estacionamiento activo	Consultar estacionamiento	Testing	0,5
[P] Consulta de estacionamiento activo	Bug fixing	Bug fixing	0
General	Documentación	Doc	2
<b>Total</b>			<b>7</b>

## Análisis de desvíos

Requerimiento	Días Estimados	Días Reales	Desvío
[P] Re-styling	5	2,5	-50%
[P] Consulta de estacionamiento activo	3	2,5	-17%
General	2	2	0%
Total general	10	7	-30%

## Retrospectiva

*Cierre*

En esta iteración, nos propusimos finalizar lo único que nos restaba para poder completar la primera versión del sistema: Adaptar el estilo del módulo Policía de Tránsito, y adaptar la consulta de estacionamiento activo. Ambas funcionalidades terminaron siendo menos costosas de lo estimado, lo que nos permitió cerrar el desarrollo con algunas semanas de anterioridad. La iteración 13 representó el hito más importante de todo el proyecto, que fue la finalización del código de todos los módulos que componen ParCo.

## Anexo

Marco legal

**Ley nacional de tránsito 24449:**

[http://www.vialidad.gba.gov.ar/datos/educacion\\_vial/publicaciones/ley%2024449.pdf](http://www.vialidad.gba.gov.ar/datos/educacion_vial/publicaciones/ley%2024449.pdf)

**Ley provincial de tránsito N° 8560:**

<http://web2.cba.gov.ar/web/leyes.nsf/483754e228b7388903257bfd004c085c/e3a9034bb531ddb503257bfe006257ce?OpenDocument>

**Normativa general:**

<http://www.concejaldomina.com.ar/files/pdf/ordenanzas/Ord-12195.pdf>

**Ordenanza municipal 12626 (pliego "Sistema medido de estacionamiento en la via pública):**

<https://www.concejaldomina.com.ar/files/pdf/ordenanzas/Ord-12626.pdf>

**Ordenanza municipal 9981 (Código de tránsito):**

<http://servicios.cordoba.gov.ar/DigestoWeb/Page/Documento.aspx?Nro=7945>

**Licitación pública Nacional e Internacional N° 16/17:**

[http://servicios.cordoba.gov.ar/licitaciones/default.aspx?ir=36\\_1\\_1566](http://servicios.cordoba.gov.ar/licitaciones/default.aspx?ir=36_1_1566)

Links a Aplicaciones similares

**Apparkb:**

<https://play.google.com/store/apps/details?id=cat.bcn.apparkB&hl=es>

**Bahía Parquímetros:**

<https://play.google.com/store/apps/details?id=ar.com.unixonon.BahiaParquimetros>

**Movypark:**

[https://play.google.com/store/apps/details?id=com.movypark.cordoba&hl=es\\_AR](https://play.google.com/store/apps/details?id=com.movypark.cordoba&hl=es_AR)

Integrantes del Equipo

## Mariano Mezi

Desarrollador Java en eSolutions SA

[marianomezi@hotmail.com](mailto:marianomezi@hotmail.com)

### Summary

Estudiante de Ingeniería en Sistemas en la Universidad Tecnológica, Facultad Regional Córdoba. De gran interés por la tecnología y el desarrollo. Actualmente me encuentro como Desarrollador Java en la empresa ESolutions

### Experience

#### **Desarrollador Java at eSolutions SA**

noviembre de 2014 - Present

Análisis, diseño, desarrollo y mantenimiento de sistema de Ingeniería de Red para Claro Argentina, Uruguay y Paraguay.

#### **Asistente PMO at Grupo Prominente**

abril de 2013 - octubre de 2014 (1 año 6 meses)

# Confección y actualización de inventarios de Proyectos y Servicios Activos en la empresa.

# Confección de Informes de Seguimiento de Proyectos. # Elaboración de informes de desvíos

presupuestarios # Monitoreo y control de información.

# Elaboración de Informes de Análisis de Estimación de Hs/Recursos vs. Consumo Real.

# Elaboración de informes de Distribución de horas mensuales cargadas por los recursos

# Administración de la herramienta Team System:

o Gestión de Permisos sobre la herramienta.

o Alta/Baja/Modificación de Proyectos activos

o Definición y Configuración de Campos y Workflows sobre cada WorkItem.

#### **Empleado administrativo**

enero de 2012 - abril de 2013 (1 año 3 meses)

### Education

**UTN**

2010 – 2017

## Martin Pianello

Technical Lead / Java Developer en eSolutions SA

[martin\\_p\\_02@hotmail.com](mailto:martin_p_02@hotmail.com)

### Experience

#### **Technical Lead / Java Developer at eSolutions SA**

agosto de 2015 - Present

#### **Scrum Master / Technical Lead / Java Developer at eSolutions SA**

enero de 2015 - agosto de 2015 (7 meses)

+ Coordinación del trabajo de un equipo.

+ Referente técnico del proyecto y consultor en materia de diseño.

Proyecto: LedZite, utilizado de la gestión del proceso de ingeniería de red de Claro

Argentina-Uruguay-

Paraguay.

#### **Desarrollador Java at eSolutions SA**

agosto de 2013 - enero de 2015 (1 año 5 meses)

### Education

#### **Universidad Tecnológica Nacional FRC**

Ingeniería en Sistemas de Información, 2010 - 2016

#### **Colegio San José**

Bachiller en Humanidades y Ciencias Sociales, Humanidades y Ciencias Sociales, 2003

- 2009