

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

Documentación Técnica	4
Arquitectura	4
Motivación	4
Solución	4
Componentes arquitectónicos	5
Stack Tecnológico	6
Cliente	6
Servidor	6
Diagramas Arquitectónicos	7
Diagrama de Clases	7
Diagrama de Estados: Estacionamiento	8
Diagrama de componentes	9
Ingeniería de Requerimientos	12
Product Backlog	12
UserStories	14
Ejecución de Pruebas	30
Casos de prueba	30
Ejecuciones de casos de prueba	37
Ciclo 1	37
Ciclo 2	48
Resumen de ejecuciones de casos de pruebas	48
Defectos encontrados	48
Plan de despliegue del sistema	57
Objetivos	57
Vista de despliegue	57
Referencias	57
Requisitos de Hardware	61
Requisitos de Software	62
Instancias de cómputo	62
Instancias de base de datos	62
Estrategia de despliegue	63
Despliegue de componentes Cloud	63
Despliegue de clientes Android	63
Actividades de despliegue	63
Responsabilidad del equipo ParCo	64

ParCo | Sistema

Responsabilidad del cliente	64
Manuales de Usuario	65
ParCo Conductor	65
ParCo Policía de Tránsito	74
ParCo Administrador	77
Administrar usuarios Kiosco	82
ParCo Kiosco	84

Documentación Técnica

Arquitectura

Motivación

La naturaleza del producto define las siguientes restricciones a considerar para elegir el modelo arquitectónico del mismo:

- 1) La interacción con el usuario (componente UI) deben ejecutarse sobre un cliente móvil de propiedad de cada usuario, es decir, un smartphone.
- 2) La información producida por las transacciones del sistema debe almacenarse de forma centralizada debido a dos necesidades: el resguardo de la información personal y la disponibilidad de la misma para el municipio dueño de ella.
- 3) Existen parámetros, tales como el precio del estacionamiento por unidad de tiempo, que deben poder ser parametrizados y aplicados de forma instantánea. Por ello, dichos cambios no pueden requerir actualización ni modificación de componentes instalados en los dispositivos móviles de los usuarios finales.
- 4) Es requerido acceder a información de usuario final brindada por el sistema operativo Android, tal como la ubicación del mismo.

Solución

Con estas restricciones en mente, se escogió un modelo arquitectónico *cliente-servidor* orientado a servicios SOA, el cual nos brinda los siguientes beneficios:

- La cantidad e identidad de los *clientes* no es un dato necesario para operatividad del *servidor*. Esto nos permite cumplir con la restricción 1) y 4); los componentes *cliente* se instalan, desinstalan y evolucionan independientemente del *servidor*.
- El subsistema de almacenamiento de información se instala en el *servidor*. Esto es necesario para satisfacer la restricción 2) y 3); los servicios provistos por el *servidor* son consumidos por los clientes cada vez que se requieren (sin interacción de caches), así se pueden actualizar datos y disponibilizarlos inmediatamente.
- La presentación del sistema (cliente liviano), junto con el acceso a funcionalidades provistas por el sistema operativo Android se instalan en el *cliente*, permitiendo al *servidor* funcionar independientemente de las diferentes versiones Android.

Riesgos

Las desventajas que se pueden encontrar en este modelo son:

- Cambios grandes que implican modificaciones en varias capas de la aplicación (negocio, presentación y acceso a datos) son costosas de aplicar debido al alto grado de distribución de los componentes entre diferentes nodos. La forma de minimizar este problema es creando interfaces claras entre ambos módulos que mantengan siempre compatibilidad con versiones anteriores; así, las ventajas de los cambios grandes se obtienen solo al actualizar, pero no se convierte en una acción obligatoria.
- El cliente liviano hace que, ante un aumento muy grande en la cantidad de transacciones o el costo computacional de las mismas, los requerimientos de hardware del *servidor* crezcan demasiado. La forma de minimizar esta

ParCo | Sistema

consecuencia es separando los componentes del *servidoren* artefactos administrables y escalables individualmente (micro servicios) para poder escalar sólo los necesarios.

Componentes arquitectónicos

Cliente

El módulo *clientes* tipo liviano (*lightweight*) ya que concentra sólo funcionalidades de presentación. Sus características fundamentales son:

- Bajos requerimientos de hardware, para que pueda funcionar en dispositivos móviles de gama baja.
- No requiere actualización ante modificaciones en reglas de negocio que no impacten en la presentación.

Servidor

El módulo *servidor* se desplegará como una serie de micro servicios web. Estos tienen como características más destacadas:

- La independencia entre sí, permitiendo que cada uno modifique independientemente con el objetivo de reducir el acoplamiento entre componentes y disminuir el impacto en un cambio o evolución de un requerimiento particular.
- Cada servicio se puede escalar de manera independiente, permitiendo crecer en recursos servicio diferentes según la demanda aumente o disminuye.

Stack Tecnológico

Cliente

Presentación móvil

Del lado *cliente móvil* la aplicación están escrita en lenguaje *Java* versión 7, el cual es compilado y ejecutado por un **Sistema Operativo** residente en los dispositivos móviles.

Sistema operativo

El desarrollo utiliza la *Android SDK API 19* para poder ser ejecutado en *Android 4.4 (Kitkat)*.

Presentación web

Del lado *cliente web* la aplicación corre sobre cualquier navegador web que soporte *HTML 5* y *CSS 3*. Se utiliza la librería *JQuery* versión 3.2, escrita en JavaScript.

Servidor

Del lado del servidor la aplicación está formada por una serie de **servicios web**, alojados en un **servidor web** (interpreta y responde peticiones HTTP), que responde cada uno a un “recurso del sistema”. Cada servicio persiste y obtiene los datos de una **base de datos**.

Base de datos

El RDBMS elegido es *PostgreSQL* versión 9.6.3.

Web Services

El lenguaje de desarrollo es *Java* versión 1.8.

Los servicios web, utilizan un **ORM** (Object-RelationalMapping) para establecer la relación entre registros de la base de datos y objetos de Java.

Para inyectar en el código dependencias, y así reducir el acoplamiento que genera crear las mismas, se utiliza un **framework de DI** (DependencyInjection).

Por último, para poder presentar el resultado en formato JSON bajo el protocolo HTTP y siguiente la guía arquitectónica web REST, se utiliza un **framework web**.

ORM

Se eligió *Hibernate* versión 5.0; este nos permite usar la especificación JPA 2.1.

Framework de DI

Delegamos la responsabilidad de crear los objetos del sistema al *Spring* versión 4.3.

Framework web

Delegamos la responsabilidad de generar las respuestas web que requiere el servidor web a *Spring-wb-mvc* versión 4.3.

Servidor web

Aprovechamos la implementación del protocolo HTTP utilizada por *Apache Tomcat* versión 8.

Diagramas Arquitectónicos

Diagrama de Clases

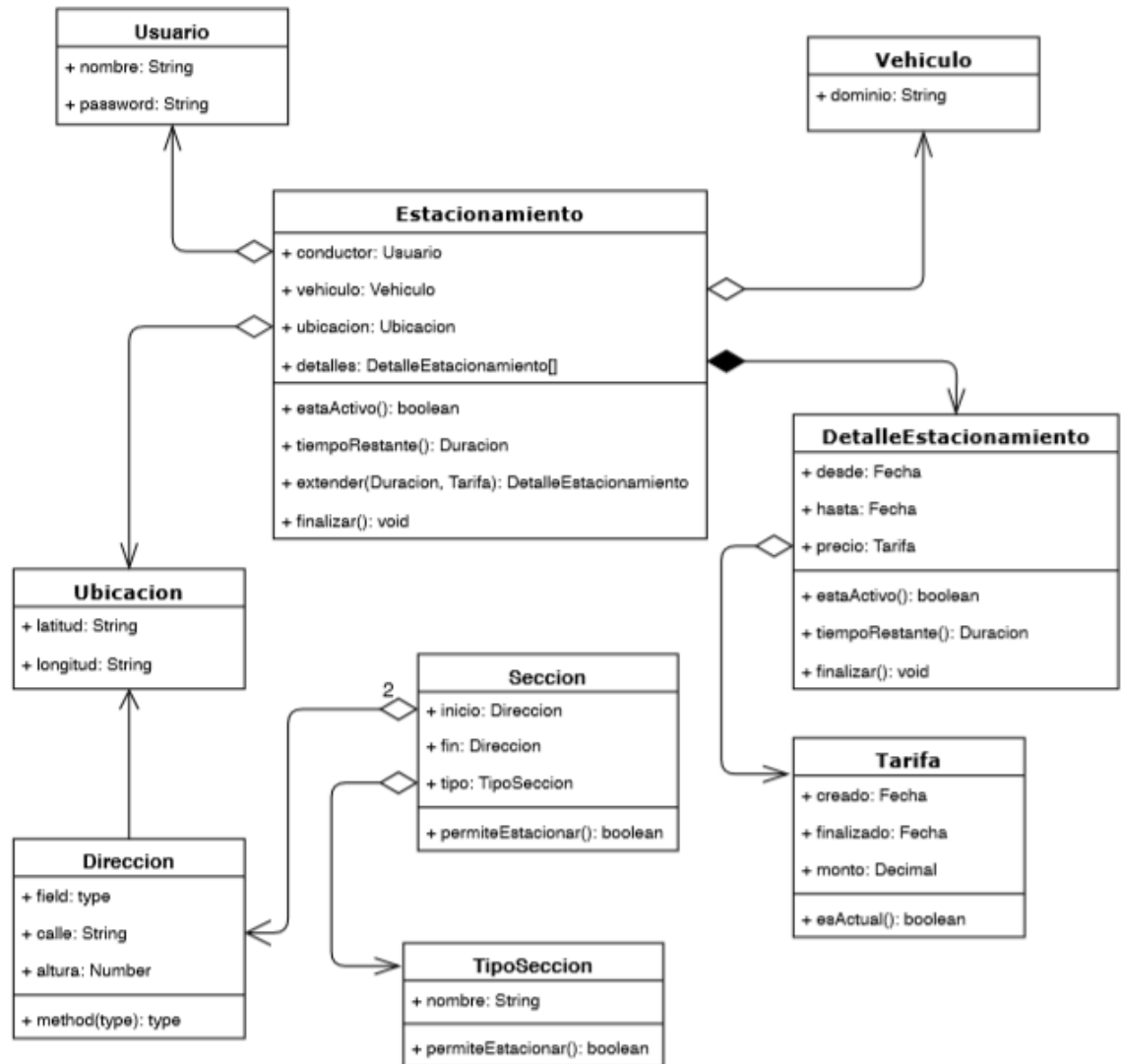


Diagrama de Estados: Estacionamiento

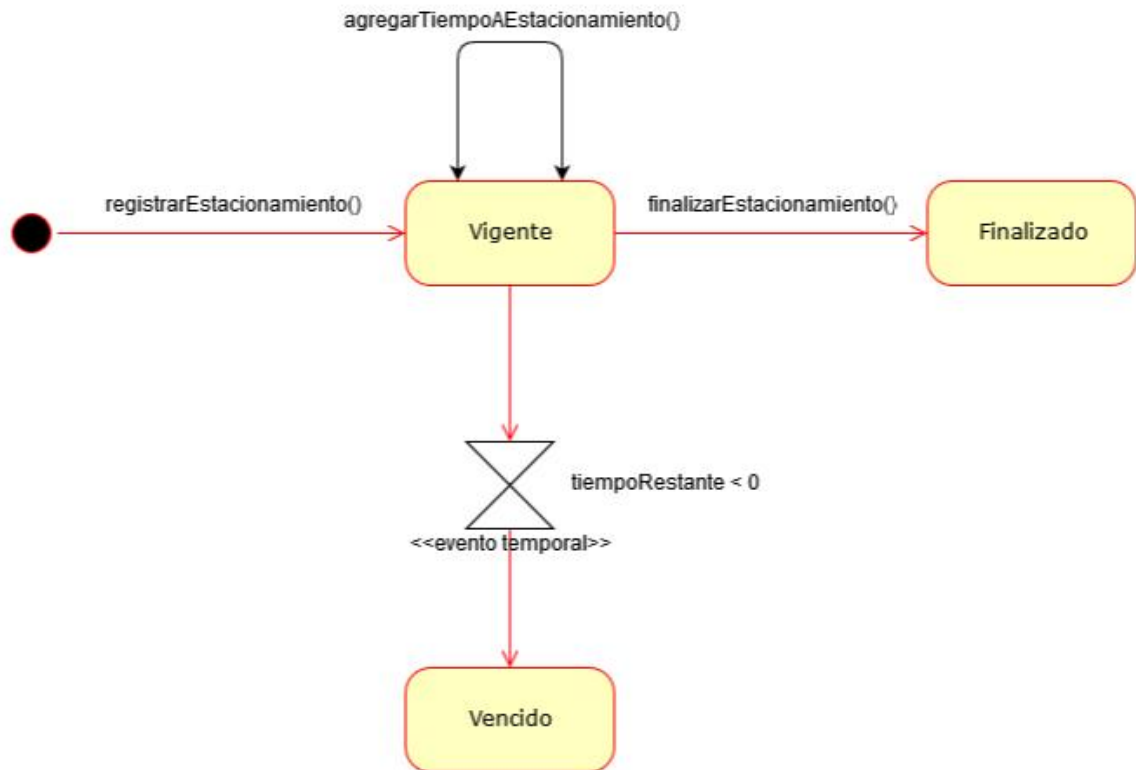
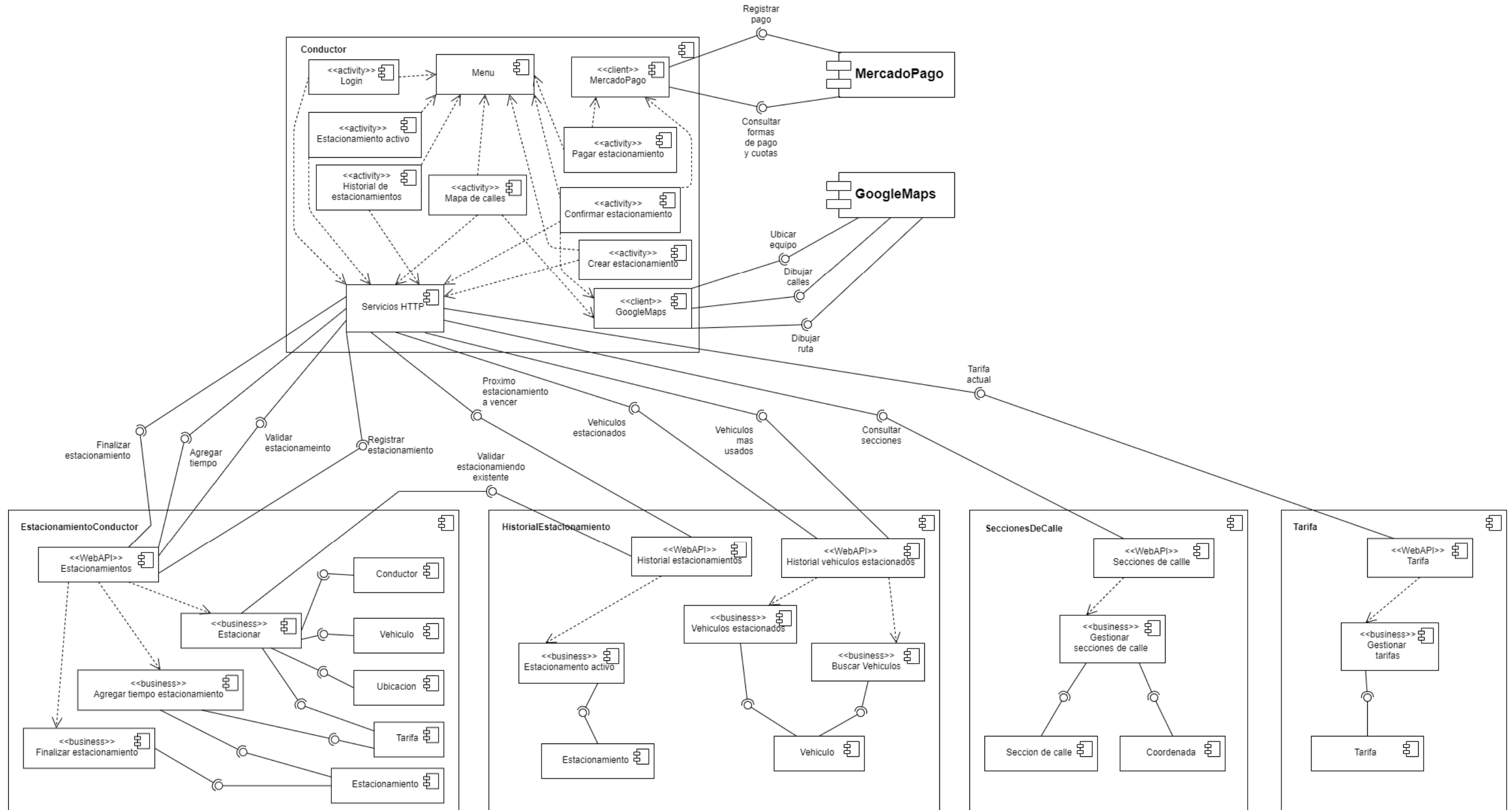
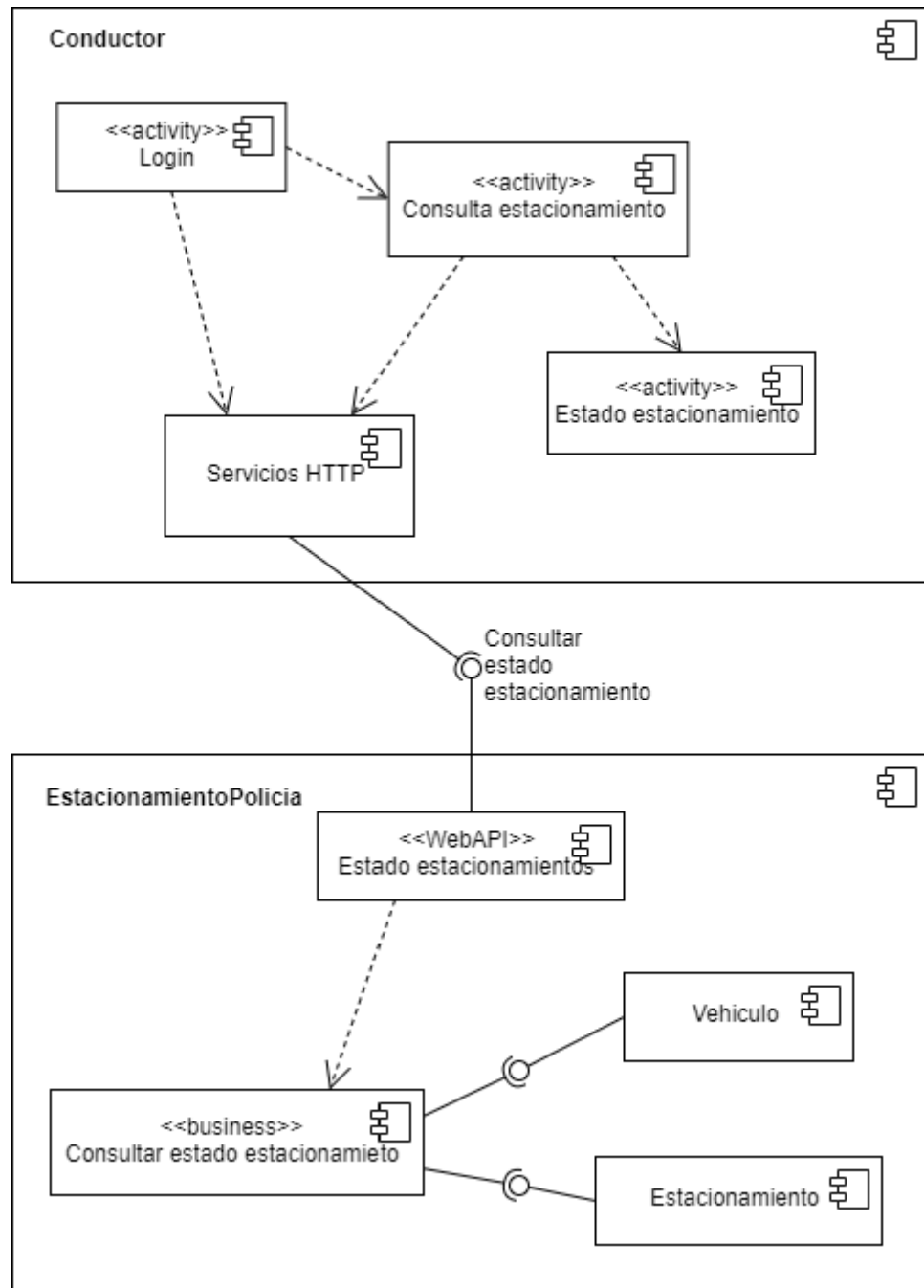


Diagrama de componentes

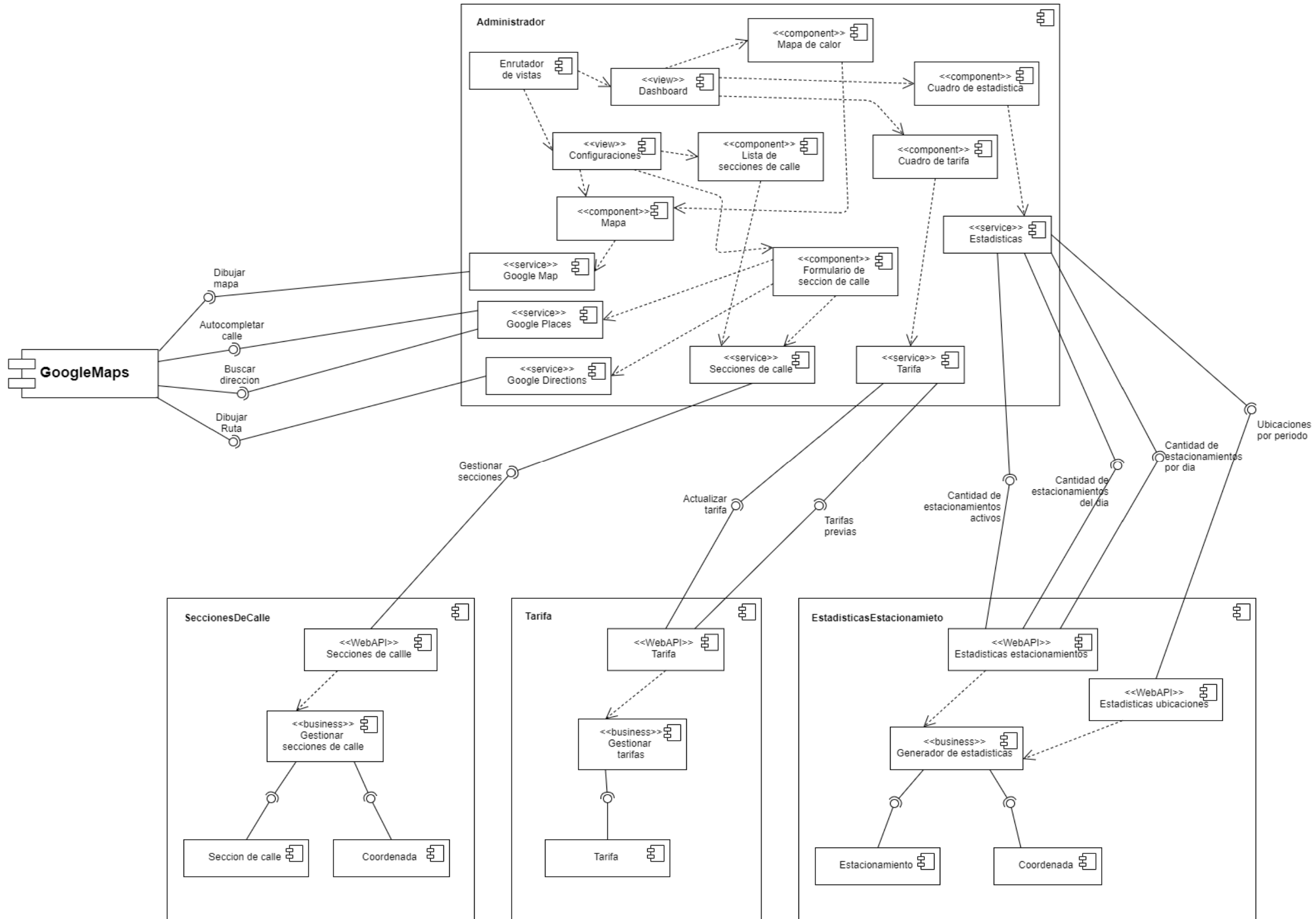
Vista Conductor/Kiosco



Vista Policía de Tránsito



Vista Administrador



Ingeniería de Requerimientos

Product Backlog

Nombre	Módulo	Creado (it)	Realizado (it)	Trazabilidad (repo[etiqueta])
[C] Registro estacionamiento	Conductor	Inicio	2	parco-db: 2.2.0-R parco-domain: 2.2.0.R parco-c-parking: 2.2.1.R parco-c-mobile: 2.1.1.R
[P] Consultar estacionamiento	PDT	Inicio	2	parco-p-parking: 2.2.0.R parco-p-mobile: 2.2.0.R
[C] Login	Conductor	Inicio	3	parco-c-mobile: 3.1.1.R
[C] Menú	Conductor	3	3	parco-c-mobile: 3.2.0.R
[C] Consulta de tiempo disponible	Conductor	Inicio	3	parco-domain: 2.2.0.R parco-c-mobile: 3.3.1.R
[C] Determinar ubicación	Conductor	Inicio	4	parco-c-mobile: 4.1.0.R
[A] Reporte de Estacionamientos por hora	Administrador	Inicio	4	parco-stats: 4.2.0.R
[C] Reutilizar componente de menú	Conductor	5	5	parco-c-mobile: 5.1.0.R
[C] Manejar respuesta de servicios	Conductor	5	5	parco-c-mobile: 5.2.0.R
[C] Mejorar la precisión de la geolocalización	Conductor	5	5	parco-c-mobile: 5.3.0.R
[C] Manejo de excepciones	Conductor	5	5	parco-c-mobile: 5.4.0.R
[C] Autenticación de usuario	Conductor	5	5	parco-domain: 5.5.0.R parco-c-parking 5.5.0.R parco-c-mobile 5.5.1.R
[C] Registro de vehículos	Conductor	Inicio	CANCELADO	
[C] Pago	Conductor	Inicio	6	parco-db: 6.1.0.R parco-domain 6.1.0.R parco-c-parking 6.1.0.R parco-c-mobile 6.1.1.R
[A] Carga de información de calles	Administrador	Inicio	6	parco-db: 6.2.0.R parco-domain: 6.2.0.R parco-street-section: 6.2.1.R parco-a-web: 6.2.1.R
[A] Clasificación de segmentos de calles	Administrador	Inicio	7	parco-db: 7.1.0.R parco-domain: 7.1.0.R parco-street-section: 7.1.0.R parco-a-web: 7.1.0.R
[C] Historial estacionamientos	Conductor	7	7	parco-c-history: 7.2.1.R parco-c-mobile: 7.2.1.R
[C] Vehículos favoritos	Conductor	7	7	parco-c-history: 7.3.1.R parco-c-mobile: 7.3.1.R
[C] Validar estacionamiento tarifado	Conductor	7	7	parco-street-section: 7.4.0.R parco-c-parking: 7.4.0.R parco-c-mobile: 7.4.0.R
[A] Dashboard de administración del sistema	Administrador	8	8	parco-a-web: 8.1.0.R
[C] Mapa de calles	Conductor	Inicio	8	parco-street-section: 8.2.0.R parco-c-mobile: 8.2.1.R
[A] [C] Definición tarifa única	Conductor, Administrador	Inicio	8	parco-db: 8.3.0.R parco-domain: 8.3.0.R parco-a-pricing: 8.3.0.R parco-c-parking: 8.3.0.R parco-a-web: 8.3.0.R parco-c-mobile: 8.3.0.R
[C] Agregar tiempo de estacionamiento	Conductor	Inicio	9	parco-domain: 9.1.0.R parco-c-parking: 9.1.1.R parco-c-mobile: 9.1.1.R
[C] Redirección a estacionamiento vigente	Conductor	9	9	parco-domain: 9.2.0.R parco-c-history: 9.2.0.R parco-c-mobile: 9.2.0.R
[A] Mapa de calor	Administrador	9	9	parco-stats: 9.3.0.R parco-a-web: 9.3.1.R
[A] Indicadores online	Administrador	10	10	parco-stats: 10.1.0.R parco-a-web:10.1.0.R

ParCo | Sistema

[K] Registro estacionamiento	Kiosco	Inicio	11	parco-c-parking: 11.1.0.R parco-k-mobile: 11.1.1.R
[A] Filtro de secciones por nombre	Administrador	11	11	parco-a-web: 11.2.1.R
[C] Restricción estacionamiento mismo dominio	Conductor	10	12	parco-c-parking: 12.1.0.R parco-c-mobile: 12.1.0.R parco-k-mobile: 12.1.0.R
[K] Login	Kiosco	Inicio	CANCELADO	
[C] Ubicar vehículo estacionado	Conductor	Inicio	12	parco-c-history: 12.2.0.R parco-c-mobile: 12.2.1.R
[C] Finalizar estacionamiento	Conductor	12	12	parco-domain: 12.3.0.R parco-c-parking: 12.3.0.R parco-c-mobile: 12.3.0.R
[P] Re-styling	PDT	13	13	parco-p-mobile: 13.1.0.R
[P] Consulta de estacionamiento activo	PDT	13	13	parco-p-parking: 13.2.0.R parco-p-mobile: 13.2.0.R

UserStories

[C] Registro estacionamiento

Requerimiento

El registro de un estacionamiento requiere los siguientes datos:

- Dirección: texto (en formato calle número, Ciudad, País) validado por Google Places API (referencia sólo ubicaciones válidas y existentes).
- Patente (dominio del vehículo): texto alfanumérico validado según los formatos de patente admitidos en Argentina (ABC123 o AB123CD¹).
- Tiempo: texto numérico que representa la cantidad de horas por las que se registrará y cobrará el estacionamiento. No admite fracciones ni números menores a 1.

El sistema valida que el vehículo ingresado no cuente con un estacionamiento vigente y registra un estacionamiento con los datos ingresados, la fecha y hora actual del sistema, asociado al vehículo.

Resultados posibles

1. Se registra el estacionamiento
 - a. Todos los datos son validados correctamente.
2. Error en datos ingresados
 - a. Dirección está vacía o no fue validada por Google Places API.
 - b. Patente vacía o no respeta el formato requerido.
 - c. Tiempo vacío o menor a 1.
3. Error de validación de negocio
 - a. El vehículo ingresado cuenta con un estacionamiento vigente.
4. Error en aplicación
 - a. Error en Base de datos
 - b. Error en algún servicio.
 - c. Error en aplicación móvil.

[P] Consultar estacionamiento

Requerimiento

La consulta de un estacionamiento requiere los siguientes datos:

- Patente (dominio del vehículo): texto alfanumérico validado según los formatos de patente admitidos en Argentina (ABC123 o AB123CD).

El sistema busca si el vehículo ingresado cuenta con un estacionamiento vigente e informa el resultado.

Resultados posibles

1. Se encuentra un estacionamiento vigente para el vehículo con el dominio ingresado
 - a. El dominio se validó correctamente y existe un estacionamiento vigente asociado.
2. No se encuentra un estacionamiento vigente para el vehículo con el dominio ingresado

¹ Según DNRPA. http://www.dnrpa.gov.ar/portal_dnrpa/chapa-mercosur.php

ParCo | Sistema

- a. El dominio se validó correctamente y no existe un estacionamiento vigente asociado.
- 3. Error de validación de negocio
 - a. El vehículo ingresado cuenta con un estacionamiento vigente.
- 4. Error en aplicación
 - a. Error en Base de datos
 - b. Error en algún servicio.
 - c. Error en aplicación móvil.

[C] Consulta de tiempo disponible*Requerimiento*

Una vez autenticado en la aplicación, en cualquier momento, se puede consultar por los estacionamientos activos² desde el menú, si hubiese alguno.




Para realizar la consulta no es necesario introducir ningún dato.

El sistema obtiene la cantidad de estacionamientos activos del usuario autenticado y, luego genera la respuesta adecuada en base a este número.

Resultados posibles

1. Si el usuario no cuenta con ningún estacionamiento activo, el sistema informa la situación mediante el mensaje “No se encuentra ningún estacionamiento activo”.
2. Si el usuario cuenta con un único estacionamiento activo, el sistema muestra los datos del mismo.
3. Si el usuario cuenta con más de un estacionamiento activo, el sistema muestra los datos de uno sólo, el más próximo a vencer³.

Información sobre el estacionamiento

1. Dominio del vehículo estacionado. Este dato se identifica con el ícono 
2. Ubicación del estacionado, expresada en el formato calle n°, código postal, ciudad, país, siendo estos datos opcionales según la disponibilidad de la ubicación en la API de Google Maps. Este dato se identifica con el ícono 
3. Tiempo restante del estacionamiento, expresado en el formato hora:minutos:segundos. Este dato se identifica con el ícono 

[C] Determinar ubicación*Requerimiento*

Estando el usuario logueado en la aplicación, tras haber ingresado los datos referidos al dominio del vehículo, y la cantidad de tiempo que desea estacionar, el sistema debe

² *Estacionamiento activo*: el cual tiene fecha y hora de vencimiento posterior a la fecha y hora en el momento de la consulta.

³ *Estacionamiento más próximo a vencer*: el cual posee la fecha y hora de vencimiento más cercana, pero mayor a la fecha y hora en el momento de la consulta.

ParCo | Sistema

determinar la ubicación del dispositivo, y ofrecer al usuario una pantalla de confirmación que permita validar los datos visualizados, o volver a ingresarlos.

La geolocalización debe poder establecer la ubicación del dispositivo con una precisión de, al menos, 30 metros, pudiendo aumentar la precisión a 15 metros si el usuario lo precisa.



El sistema debe ofrecer una interfaz clara e intuitiva, que permita reintentar la detección de ubicación en caso de que la misma no haya podido ser detectada.

Resultados posibles

1. Si el sistema no puede determinar la ubicación del dispositivo con la precisión necesaria, debe informar un mensaje informando esta situación, permitiendo reintentar la geolocalización, o volver a la pantalla anterior.
2. Si el sistema puede determinar la ubicación del dispositivo, el sistema debe ofrecer una interfaz de confirmación, contando la misma con un mapa que identifique el lugar detectado, la dirección del mismo, el dominio del vehículo y el tiempo a estacionar, permitiendo al usuario reintentar la detección de ubicación (aumentando la precisión de la misma de 30 a 15 metros).

Información sobre la ubicación


1. Ubicación del vehículo estacionado, expresada en su conjunto mediante un mapa provisto por GoogleMaps, y un texto en formato NombreDeCalle + n°. Este dato se

identifica con el ícono . Contiguo a los datos de ubicación, se encuentra el ícono , que permite relocalizar el dispositivo, aumentando la precisión necesaria.

2. Vehículo que registrará el estacionamiento, identificado a través de su dominio. Este

dato se identifica con el ícono .

3. Tiempo a estacionar requerido, expresado en el formato hora:minutos:segundos. Este

dato se identifica con el ícono .

[A] Reporte cantidad de estacionamientos por hora*Requerimiento*

Estando el usuario logueado en el sistema Parco Web, y habiendo ingresado el mismo en la subsección "Reportes", deberá aparecer el reporte "Cantidad de estacionamientos por hora", para seleccionar. Tras seleccionarlo, deberán poder seleccionarse como datos de entrada, la fecha desde, y fecha hasta del reporte, siendo las mismas obligatorias. La fecha desde y fecha hasta, deberán seleccionarse de un calendario que se desplegará al hacer click sobre las entradas de texto correspondientes. El sistema exportará, en formato .xlsx, el reporte solicitado.

El reporte debe contar con una tabla donde se encuentre la siguiente información referida a los estacionamientos comprendidos en el rango de fechas solicitado:

- Dominio de los vehículos estacionados.
- Dirección de estacionamiento (En formato Calle + nro)

ParCo | Sistema

- Fecha y hora inicio estacionamiento (En formato dd/mm/aaaa HH:MM)
- Fecha y hora fin estacionamiento (En formato dd/mm/aaaa HH:MM)

El reporte debe contar con un gráfico de líneas, que cumpla con las siguientes características:

- En el eje horizontal, deberán expresarse de forma discreta las horas del día.
- En el eje vertical, números que representen cantidad de vehículos estacionados
- El reporte deberá contar con dos líneas que representen lo siguiente:
 - o Una línea representará la cantidad de vehículos que comenzaron su estacionamiento entre las horas fijadas en el reporte.
 - o La otra línea, deberá representar la cantidad de vehículos estacionados en ese momento, determinándose a través de la cantidad de estacionamientos que se encontraban vigentes en ese momento.

Nota: Requerimiento acotado

Al ser el reporte un requerimiento que no se encontraba planificado originalmente para esta iteración, se acotó su alcance para poder adelantarlo, debiendo el mismo retomarse en iteraciones posteriores para lograr su completitud.

El requerimiento acotado, no contemplará la interfaz visual del ingreso de datos para la generación del reporte, debiendo el mismo generarse a través de una URL con los parámetros de fechas contenidos en ella. La generación y visualización del reporte no sufren cambios.

[C] Reutilizar componente de menú

Motivación

Actualmente se repite el código que le da la funcionalidad al menú en cada pantalla; esto produce inconsistencias entre diferentes pantallas.

Requerimiento

La funcionalidad de menú debe actuar diferente según la pantalla en la que se encuentra el usuario. Esto hace que el código que le da el comportamiento al menú sea diferente según la situación. Sin embargo, para cada opción se puede codificar una acción genérica y luego, especificar la diferencia dentro de cada pantalla.

El objetivo es tomar el código que brinda la funcionalidad de cada botón, y la estructura, del menú y separarlo como un componente que se consuma en cada pantalla, parametrizando la una porción chica del código que difiere de la implementación por defecto, según cada interfaz.

[C] Manejar respuesta de servicios

Motivación

En este momento contamos con 2 clientes de servicios web (registro de estacionamiento e historial de estacionamientos). Ambos clientes utilizan la misma funcionalidad de conexión, liberación de recursos, manejo de excepciones, detección de timeout y lectura del mensaje (o cuerpo) de la respuesta.

El objetivo es generar un mecanismo para consumir servicios web y manejar las diferentes respuestas.

Requerimiento

Todas las aplicaciones móviles disparan la ejecución de funcionalidad de negocio, y consumen datos de la misma, a través de servicios web basados en la arquitectura RESTful. Estos

ParCo | Sistema

servicios, sin importar el resultado de la operación ejecutada, devuelven un código HTTP y el cuerpo de la respuesta. Esta última varía según la operación.

El cliente que consume los servicios devuelve el JSON proporcionado por el servidor como una cadena de texto y el código HTTP como un número entero.

El objetivo es Disponer de un mecanismo (encapsulado en un componente independiente) para traducir el cuerpo de la respuesta de una forma dinámica que sirva para diferentes usos, y parametrizar el comportamiento a partir de la respuesta HTTP (código).

Así, la lógica de presentación deja de depender de una implementación concreta (servicio WEB con código particulares) y se hace más flexible a los cambios.

[C] Mejorar la precisión de la geolocalización

Motivación

El proceso actual espera indefinidamente hasta que el servicio de detección de ubicación devuelva un resultado con la precisión esperada. Como primera implementación fue útil para validar la factibilidad del requerimiento. Ahora es necesario mejorar el proceso para que espere cierto tiempo máximo, tras el cual libere los recursos del sistema.

Requerimiento

La funcionalidad de geolocalización provista por la API de Google Maps requiere que se inicie un servicio de detección de ubicación mediante GPS y devuelve resultados periódicamente, con diferentes parámetros de precisión según los servicios con los que cuente.

Nuestra aplicación requiere un nivel mínimo de precisión, lo que nos obliga a evaluar cada respuesta provista por el hilo de ejecución del servicio de Google Maps.

Objetivo

Investigar un mecanismo para que la aplicación no quede dependiente de la respuesta del servicio de Google Maps de forma síncrona.

También, intentar evitar la necesidad de solicitar la ubicación múltiples veces (incurre en demoras) y evaluarla, haciendo que la API de Google Maps resuelva esto por nosotros.

[C] Manejo de excepciones

Cada vez que hay que revisar el motivo de falla de la aplicación se necesita instalarla en modo depuración conectada a una computadora para recibir los logs. Se requiere persistir los logs de la aplicación, al menos los de error, en el dispositivo para facilitar el análisis de errores.

Aún no hemos encontrado la maneja de persistir un archivo con logs de la ejecución de la aplicación móvil para revisar posibles fallos.

Requerimiento

Identificar un mecanismo para persistir en algún medio no volátil la información de logs dejada por la aplicación, de forma separada de los logs de servicios de Android.

[C] Autenticación de usuario

Motivación

Debido a los tiempos apremiantes de las iteraciones anteriores decidimos posponer el desarrollo de un mecanismo de autenticación de usuarios que realmente los controle con los persistidos en la base de datos.

Esto nos obliga a atarnos a credenciales “bobas” estáticas que se declaran en la aplicación móvil y nos quitan la posibilidad de manejar nuevos o distintos usuarios.

Requerimiento

Completar el proceso de autenticación mediante el control de las credenciales proporcionadas por el usuario en contraste por las que están almacenadas en la base de datos centralizada de la aplicación.

[C] Registro de vehículos

Motivo de cancelación

Luego de concluir con el requerimiento **[C] Pago** entendemos que no aporta valor al usuario registrar los datos de sus vehículos. Al contrario, es una potencial demora para el usuario tener que registrar el vehículo para poder registrar el estacionamiento (aunque sea por única vez).

Al cambiar el componente de un selector a un campo de texto, se le permite por al usuario registrar el estacionamiento para cualquier dominio válido.

[C] Pago

Requerimiento

Una vez que el usuario haya establecido la cantidad de horas para registrar el estacionamiento, se debe calcular el monto del pago teniendo en cuenta la tarifa única por hora; el usuario no debe poder modificar dicho monto. Antes de hacer efectivo el pago, el usuario debe tener una instancia de confirmación donde se vea el monto a pagar y la moneda.

Para procesar el pago electrónico se escoge una plataforma de pago existente con el objetivo de aprovechar componentes existentes y probados (para evitar errores en acciones de criticidad tan alta, como lo son los pagos); también, es un beneficio su popularidad, lo que reducirá la posibilidad de transacciones fallidas por desconfianza del usuario.

La plataforma debe aceptar tarjetas de crédito y débito como medio de pago para, al menos los dos proveedores más conocidos en Argentina (Visa y Mastercard).

Definiciones técnicas adicionales

Los lineamientos generales de integración con Mercado Pago deben consultarse en el siguiente link: <https://www.mercadopago.com.ar/developers/es/guides/payments/mobile-checkout/introduction/>

Es necesario realizar una descarga de las SDKs brindadas por Mercado Pago, las versiones utilizadas por el producto son:

- com.mercadopago:sdk:3.8.1: Brinda una SDK completa con clases útiles para el manejo de los componentes de mercadopago.
- com.mercadopago.android.px:checkout:4.0.1: Brinda una utilidad completa para el proceso de pago simplificado, otorgando las interfaces visuales necesarias para el ingreso de las tarjetas de crédito/débito.

Es necesario contar con un TOKEN de autenticación que se deberá generar mediante una petición de tipo **POST**al siguiente link: <https://api.mercadopago.com/oauth/token>.

Para realizar el pago es necesario enviar una petición de tipo **POST**al link: https://api.mercadopago.com/checkout/preferences?access_token=TOKEN_GENERADO_PASO_ANTERIOR

anexándole a la URL el token generado previamente.

[A] Carga de información de calles

Requerimiento

El cliente administrador debe parametrizar las *secciones de calle* en las que se habilita el estacionamiento. Esta información sirve como entrada para que el usuario pueda ubicar un tramo de calle dónde estacionar de forma legal, y también para que el sistema valide y evite estacionamientos en secciones no permitidas.

La información relevante de cada tramo de calle es *nombre de calle*, *altura desde* y *altura hasta*. Se debe visualizar en el mapa los puntos para proveer una confirmación visual al usuario que está realizando la carga de datos.

Todos los campos deben ser validados y se debe comprobar la existencia de los puntos desde y hasta para que la sección se pueda registrar. Dado que existen calles que tiene cortes o “saltos” de numeración en algún tramo, la tarea de validar que el tramo sea el correcto queda a cargo del usuario, quien cuenta con el mapa como apoyo visual de los datos ingresados.

Todas las secciones cargadas se deben mostrar en una tabla (todos los campos pertinentes) que, además, permita la selección de un registro para eliminarlo y modificarlo.

Definiciones técnicas adicionales

Las especificaciones de integración a la API de Google Place Search están en el siguiente link: <https://developers.google.com/places/web-service/search?hl=es-419>

Todas las peticiones efectuadas desde la aplicación deben contener como límites de búsqueda dos parámetros:

1. *Bounds*, que representa los límites geográficos usados para buscar los resultados dentro de la ciudad de Córdoba, definidos como:
 - a. Límite izquierdo inferior: -31.464186,-64.287061
 - b. Límite derecho superior: -31.356719, -64.119723
2. *Query*, que representa el texto de búsqueda ingresado por el usuario. No se aplican filtros ni restricciones.

La respuesta de Google contiene un campo llamado *results*, el cual es un arreglo con las coincidentes de la búsqueda que se muestran al usuario.

[A] Clasificación de segmentos de calles

Motivación

El requerimiento **[A] Carga de información de calles** nos permite registrar tramos tarifados de calles. Esto es suficiente para permitir y no el registro de un estacionamiento, ya que solo sería válido en un tramo tarifado. Pero para el requerimiento **[C] Mapa de calles** es necesario distinguir entre tramos como tarifados, gratuitos y prohibidos.

Requerimiento

El usuario *Administrador* actualmente define tramos de calles donde el estacionamiento está habilitado. Todo tramo de calle que no esté cargado se considera gratuito de estacionamiento.

A esto, es necesario adicionarle la posibilidad de definir un tramo como prohibido, con el principal objetivo de informar al ciudadano (funcionalidad fuera del alcance de esta actividad); y como objetivo adicional, restringir el estacionamiento para evitar multas.

Todos los estacionamientos generados por los clientes deben ubicarse en un tramo habilitado, de lo contrario la aplicación no debe permitirlo.

[C] Historial de estacionamientos

Motivación

Con el objetivo de que un usuario pueda ver sus estacionamientos, extender el tiempo registrado según el requerimiento planificado **[C] Agregar tiempo de estacionamiento** y, dado que el requerimiento **[C] Vehículos favoritos** permite a los usuarios tener más de un estacionamiento vigente a la vez, es necesario proveerle al usuario una forma de consultar sus estacionamientos. En este requerimiento no incluiremos ningún filtro de fecha ni estado.

Requerimiento

Solo los usuarios *Conductores* deben poder ver el historial de sus estacionamientos.

Los datos que deben visualizarse son:

- Ubicación (calles y altura)
- Estado (Finalizado o Vigente)
- Fecha y horario (desde - hasta)
- Dominio

Los estacionamientos se deben ordenar por fecha, de forma descendente, mostrando primero el más reciente.

[C] Vehículos favoritos

Motivación

En reemplazo de **[C] Registro de vehículos**, agregamos la posibilidad de que la aplicación sugiera los vehículos (dominios) más usados por el usuario para evitar el tipeo del mismo al momento de registrar el estacionamiento.

Requerimiento

El objetivo, es facilitar el registro del estacionamiento aprovechando el historial de dominios que carga un usuario.

Solo para los usuarios *Conductores*, al momento de registrar el estacionamiento vigente, y cuando el campo donde se carga el dominio del vehículo está completamente vacío, la aplicación debe sugerir el dominio del “vehículo favorito”. Así se le llama al dominio con mayor cantidad de estacionamientos registrados por el usuario logueado.

En el caso donde el campo tenga al menos un carácter tipeado por el usuario, la aplicación mostrará, de forma ordenada, los dominios más usados con contengan dichos caracteres.

Esta funcionalidad sólo estará disponible si se cuenta con, al menos, un estacionamiento previamente registrado.

[C] Validar estacionamiento tarifado

Motivación

Luego de contar con la información precisa sobre la clasificación de las calles según **[A] Clasificación de segmentos de calles**, es necesario validar y prohibir el registro de estacionamiento en cualquier ubicación que no esté tarifada.

Requerimiento

Ningún usuario *Conductor* podrá registrar un estacionamiento cuando se encuentre en una ubicación cuya clasificación sea diferente a “tarifado”.

Cuando el usuario quiera registrar un estacionamiento, el sistema obtiene la ubicación de este geolocalizándolo. Luego, deberá comprobar que la dirección esté clasificada como tarifada; en caso de que así sea, el proceso para registrar el estacionamiento continúa normalmente.

En caso contrario, el sistema debe bloquear al usuario la posibilidad de continuar. Es importante que muestre la dirección geolocalizada para permitirle recalcularla. También, deberá informar el motivo del bloqueo, pudiendo ser: “ubicación gratuita” o “ubicación prohibida”.

[A] Dashboard de administración del sistema

Motivación

El administrador web, además de configurar parámetros como las secciones de calles, necesita contar con información que le permita tomar decisiones. Para poder incluir a futuro información, es necesario un dashboard o tablero de control con un diseño que dé lugar a agregar indicadores y gráficos.

Además, se usará como base para que uno de los datos de la pantalla sea la tarifa, según el requerimiento **[A] [C] Definición tarifa única**.

Requerimiento

El objetivo es disponer de una pantalla donde presentar en forma de tarjetas o “cards” los indicadores y reportes disponibles a los administradores del sistema.

Se usará como template inicial a *CoreUI*(<https://coreui.io/>).

Debe contar con un menú que permita navegar entre las pantallas disponibles:

- Item “Dashboard”: pantalla con indicadores y reportes.
- Item “Configuraciones”: pantalla de listado y carga de secciones de calles.

[C] Mapa de calles

Requerimiento

El objetivo es brindarles a los usuarios conductores la posibilidad de consultar en la aplicación información sobre los tramos de calle tarifados, prohibidos y gratuitos.

Para esto se define el uso de un mapa que mejora la percepción de la ubicación del usuario y diferenciar los diferentes tramos con colores, de la misma forma que se definieron para el cliente administrador al momento de su carga:

- Color rojo para tramos prohibidos
- Color verde para tramos tarifados
- Sin color para tramos gratuitos

Definiciones técnicas adicionales

Para dibujar rutas sobre el mapa se utiliza la el módulo *Polylines and Polygons* de la SDK de Google Maps para Android, siguiendo la siguiente referencia: <https://developers.google.com/maps/documentation/android-sdk/polygon-tutorial>

Obtenemos los puntos de ruta de cada sección cargada en el sistema a través de la interfaz de carga de sección de calle del Administrador. Luego, es necesario crear un objeto *PolylineOptions* para cada sección usando los siguientes datos:

1. A través del método *add*, agregar todos los puntos asociados.
2. A través del método *width*, establecer el valor 12.
3. A través del método *color*, utilizar la instancia de color verde o rojo según si la sección de calles es tarifada o prohibida, respectivamente.
4. A través del método *geodisic*, establecer el varlortrue.

Por último, agregar cada *PolylineOptions* (uno por cada sección de calle independiente) al mapa mediante el método *addPolyline*.

[A] [C] Definición Tarifa Única

Requerimiento

El administrador del sistema debe contar con la posibilidad de definir, y actualizar, la tarifa única por hora. Se le llama única, porque aplica sin restricción de franja horaria ni zona geográfica.

Una vez que el usuario establece un nuevo valor de tarifa, esta rige como la tarifa vigente. Es decir que, el próximo estacionamiento que se registre debe calcular su costo en función del nuevo precio.

El usuario debe poder ver como indicador separado y claro la tarifa vigente. También, se debe poder consultar las últimas 10 tarifas con su fecha de definición.

El proceso de pago del usuario Conductor y Kiosko, deben consultar de forma online el valor de la tarifa única vigente para calcular el costo del estacionamiento al momento de la registración. La fórmula de cálculo es *cantidad de horas x tarifa única*, sin excepción para todos los casos.

[C] Agregar tiempo de estacionamiento

Requerimiento

El objetivo es permitirle al usuario *Conductor* ampliar el tiempo de un estacionamiento vigente. Si luego se consulta el estacionamiento debe modificarse la fecha de finalización para extenderlo la cantidad de horas necesarias; es decir, que no debe registrarse un nuevo estacionamiento.

Para esta funcionalidad existe un workaround, que consiste en registrar un nuevo estacionamiento en el momento en el que finaliza el vigente. Sin embargo, en el lapso de tiempo entre que se vence el vigente y se registra el nuevo, el usuario pasa a estar expuesto a recibir una multa por parte de un *Policía de tránsito* si realizara la consulta en ese interín.

[C] Redirección a estacionamiento vigente

Motivación

Es un requerimiento que nos va a permitir darle una mejor experiencia al usuario. Luego de registrar un estacionamiento, si cerramos la aplicación y la volvemos a abrir vemos la pantalla de registro de estacionamiento y para acceder a los datos del estacionamiento vigente es necesario ir hasta la pantalla de historial de estacionamiento y seleccionar el correcto (el primero de la lista). Interpretamos que en la mayoría de los casos, los usuarios particulares, van a tener un único estacionamiento vigente a la vez.

Requerimiento

Cuando un usuario *Conductor* cuente con un estacionamiento vigente, al momento de abrir la aplicación, y luego de loguearse exitosamente, la primer pantalla que deben ver es la pantalla de consulta de estacionamiento con los datos del estacionamiento vigente (en curso).

El objetivo es evitar que el usuario se equivoque al abrir la aplicación e intente registrar un estacionamiento sobre un dominio con estacionamiento vigente; hay que tener en cuenta que actualmente, al ingresar a la aplicación, se abre la pantalla de registro de estacionamiento.

[A] Mapa de calor

Motivación

Dado que al registrar un estacionamiento almacenamos su ubicación identificamos la oportunidad de explotar esa información para que los administradores de un municipio lo usen para tomar decisiones sobre zonas geográficas. La idea es contar con un mapa donde podamos “pintar” zonas o calles, que denoten el grado o medida de convergencia de estacionamientos. Sabemos que se podrían agregar filtro y configuraciones, pero vamos a comenzar con una versión donde el foco sea la integración con los componentes de mapa y “dibujado” de las zonas de calor.

Requerimiento

En el dashboard del cliente *Administrador* añadir una tarjeta con un mapa, aprovechando la integración con Google Maps. Sobre el mapa, dibujar un punto por cada estacionamiento, de tal forma que los puntos más cercanos se junten y vayan cambiando el color. No hay una restricción de radio de los puntos, gama de colores ni intensidad de estos para este requerimiento.

Para esta iteración, sólo se necesitan la ubicación de los estacionamientos registrados en el día de la fecha. No siendo esto variable ni configurable.

Definiciones técnicas adicionales

Para dibujar los puntos sobre el mapa es necesario usar la API de Google llamada *HeatmapLayer*, siguiendo la guía descrita en <https://developers.google.com/maps/documentation/javascript/heatmaplayer>.

Para crear el objeto *heatmapLayer* es necesario pasarle tres argumentos:

1. *map*: mapa a usar (debe estar inicializado previamente).
2. *data*: puede ser nulo inicialmente, en cuyo caso se utiliza el método *setData* para actualizarlo cada vez que sea necesario. Debe contener un arreglo de objetos *LatLng* que contienen las coordenadas de cada estacionamiento.
3. *opacity*: siempre el valor 1.

[A] Indicadores online

Motivación

En la iteración 8 detectamos la necesidad de proveer al usuario *Administrador* información relevante para la toma de decisiones; por ese motivo, comenzamos con el requerimiento **[A] Dashboard de administración del sistema**, a partir del cual agregamos una pantalla con una estructura flexible para agregar “tarjetas” con información. Con este nuevo requerimiento vamos a agregar información en forma de indicadores y gráficos simples que muestren los números más relevantes sobre los estacionamientos de forma online.

Requerimiento

El objetivo fundamental es procesar datos transaccionales del sistema para generar información que el usuario *Administrador* pueda usar para tomar decisiones estratégicas relacionadas al dominio de los estacionamientos tarifados.

Definimos los siguientes indicadores y gráficos, a implementar cada uno en una tarjeta en el dashboard del administrador:

- Cantidad de estacionamientos vigentes.
 - Indicador. Número que se actualiza periódicamente con la cantidad de estacionamientos vigentes; es decir, la cantidad de vehículos estacionados en el momento de la consulta.
- Cantidad de estacionamientos registrados.
 - Indicador. Número que se actualiza periódicamente con la cantidad de estacionamientos registrados en el día independientemente de su vigencia.
 - Gráfico de barras. Cada barra representa un día. Se muestra un total de 7 días hacia atrás (incluyendo el día actual) para mostrar una vista “semanal”. No cuenta con escala pero deben indicar de alguna forma el número de cada barra (día).
- Historial de tarifas
 - Indicador. Número que representa la tarifa actual. Debe aceptar decimales.
 - Gráfico de línea (dispersión). El eje horizontal representa la variable temporal (sin escala) y el eje vertical representa el monto de la tarifa en pesos (sin escala). Cada punto sobre el gráfico representa un precio y debe indicarlo de alguna forma. Todos los puntos se deben conectar con una línea que muestre la tendencia. Se define en 10 el máximo de puntos a mostrar.
- Descarga de reporte “Estacionamientos por hora” según se definió en el requerimiento **[A] Reporte de Estacionamientos por hora**.
 - Filtro de fecha desde. Obligatorio.
 - Filtro de fecha hasta. Opcional. Si no se especifica un valor, asume el día actual.
 - Botón de descarga. Sólo se debe habilitar cuando los filtros aplicados son correctos. No se especifica un límite de cantidad de descargas.

[K] Registro estacionamiento

Requerimiento

La ubicación del estacionamiento no debe ser determinada por el servicio de geolocalización, sino por una entrada manual. Para esto se tomará como base el código de la aplicación al finalizar la Iteración 3, donde ésta era única forma de registrar un estacionamiento.

Las calles que aparezcan en el componente de búsqueda deben estar limitadas a la ciudad de Córdoba. El proceso debe continuar realizando la validación de ubicación habilitada para estacionar. El proceso de pago debe permanecer igual al existente, dejando fuera del alcance la forma en la que el punto de carga le cobra al cliente final.

[A] Filtrar fechas en mapa de calor

Motivación

Como mencionamos en la iteración 10, vamos a agregarle iterativamente comportamiento al mapa de calor creado según el requerimiento **[A] Mapa de calor**. Con este nuevo requerimiento queremos permitirle al usuario filtrar los estacionamientos que se dibujan en el mapa según las fechas que necesite.

Requerimiento

Agregar dos filtros de fecha, desde y hasta, ambos opcionales que asuman como valor por defecto el día de hoy.

Estos filtros afectarán el comportamiento del mapa de calor. Éste debe adecuar el dibujo de los puntos según el rango de días especificados por las fechas seleccionadas, acumulando los estacionamientos registrados.

En ninguna ocasión, el valor del filtro desde puede ser mayor al del filtro hasta; en cuyo caso el sistema debe informar visualmente algún error sobre los componentes del filtro y limpiar el mapa (borrar todos los puntos). No se mostrará ningún mensaje ni leyenda.

[C] Restricción estacionamiento mismo dominio

Motivación

Al completar, en la iteración 9, el requerimiento **[C] Redirección a estacionamiento vigente** descubrimos que, si bien al tener un estacionamiento vigente, la aplicación te muestra el mismo por defecto, aún es posible registrar un estacionamiento con el mismo dominio de uno vigente. Con este nuevo requerimiento vamos a restringir esta posibilidad.

Requerimiento

Ningún usuario podrá registrar un estacionamiento sobre un vehículo que tenga un estacionamiento vigente.

Cuando el usuario ingresa un dominio y presiona “Iniciar estacionamiento”, en la pantalla de registro de estacionamiento, antes de validar los datos y realizar el pago, el sistema deberá validar que el dominio ingresado no cuente con un estacionamiento vigente.

Si el dominio ya está estacionado, el sistema mostrará un mensaje informando la situación, y, sólo en caso de que el estacionamiento vigente este registrado con el usuario que está intentando registrar el nuevo estacionamiento, agrega un link que redirecciona a la pantalla de consulta del mismo.

[K] Login

Motivo de cancelación

La estrategia para generar el cliente Kiosco es reutilizar el código disponible para el conductor. Identificamos que la funcionalidad de login es la misma, por lo que podemos cancelar este requerimiento.

[K] Creación app

Motivación

El cliente Kiosco no existe todavía como tal. Sin embargo, se puede tomar como base el cliente Conductor.

Requerimiento

El objetivo de este requerimiento es contar con una aplicación cliente que llamamos Kiosco para poder instalarla en teléfonos smartphones o tablets en los puntos de carga designados por el municipio. Sobre el cliente *Conductor* se deben tener en cuenta las siguientes modificaciones:

- Cambio de logo en la pantalla de login por uno que identifique al cliente.
- La ubicación del estacionamiento no debe ser determinada por el servicio de geolocalización, sino por una entrada manual. Se separa esta funcionalidad en otro requerimiento, **[K] Registro estacionamiento**.
- Al momento de ingresar un dominio para registrar un estacionamiento, no debe sugerir ningún dominio.
- Al ingresar a la aplicación no debe redirigir a la pantalla de consulta de estacionamiento, sino a la pantalla de registro de estacionamiento.

[A] Filtro de secciones por nombre

Motivación

Al registrar cierta cantidad de secciones de calles en la pantalla de configuración del cliente Administrador identificamos que se hace muy complicado encontrar una existente para editarla o borrarla. Por esto, vamos a agregar un filtro de búsqueda por calle, que afecto los resultados que se muestran del listado.

Requerimiento

Con el objetivo de facilitarle al usuario *Administrador* encontrar una sección de calle creada, se añade un buscador de calles. Al tipear sobre el campo, los registros en la tabla se deben filtrar, sin la necesidad de usar un botón “buscar” o similar.

En caso de que el filtro no encuentre ningún resultado, la tabla debe mostrar el mensaje “No existen secciones para esta búsqueda”.

El texto se podrá borrar del campo en cualquier momento y el filtro debe dejar de aplicarse instantáneamente.

[C] Finalizar estacionamiento

Motivación

Entendemos que el usuario hoy no tendría un beneficio real, finalizando el estacionamiento. Pero, en caso de que se quiera cambiar el esquema de pago a una modalidad “pago por uso”, en lugar del pago fijo por adelantado que se maneja ahora, pasa a ser indispensable.

ParCo | Sistema

Por otro lado, el *Administrador* sí tiene un beneficio; la información en sus indicadores y reportes se va a actualizar y podrá contar con datos más reales.

Requerimiento

Dado un estacionamiento vigente, el usuario debe contar con la posibilidad de registrar que finaliza el mismo en cualquier momento.

La funcionalidad debe estar disponible mediante un botón o link siempre activo, en la pantalla de consulta de estacionamiento. No es relevante, desde donde haya navegado el usuario hasta ésta.

Para todos los casos, se debe requerir al usuario que confirme la acción para evitar errores involuntarios.

El sistema, al recibir la petición de finalización de estacionamiento, debe modificar el estado del estacionamiento y la fecha y hora de fin del mismo.

[C] Ubicar vehículo estacionado*Requerimiento*

En la pantalla de consulta de estacionamiento del usuario Conductor, cuando hay un estacionamiento vigente, añadir un botón o link que abra la aplicación Google Maps del teléfono seleccionando Direcciones. La ubicación desde debe ser la ubicación actual detectada por el servicio de ubicación del dispositivo, y la ubicación hasta debe ser la ubicación donde se registró el estacionamiento. De ser posible, seleccionar como transporte “caminando”.

De esta forma, el usuario podrá guiarse hasta dónde estacionó el vehículo si no le es suficiente la dirección (calle y numeración) que aparece en la pantalla.

[C] Finalizar estacionamiento*Requerimiento*

Dado un estacionamiento vigente, al ingresar en la pantalla de consulta de estacionamiento, se agrega un botón similar al botón agregado por el requerimiento **[C] Agregar tiempo de estacionamiento** que permita al usuario finalizar el estacionamiento vigente.

El sistema debe modificar la fecha de fin establecida, y reemplazarla por la fecha de finalización. El estacionamiento pasa a estar terminado, y no se podrá agregarle más tiempo. El dominio queda libre de estacionamientos vigentes y se podrá registrar un nuevo estacionamiento con el mismo dominio.

[P] Re-styling*Motivación*

La aplicación para el cliente *Policía de tránsito* no tuvo modificaciones desde la iteración 4. A lo largo de este tiempo, los clientes móviles tuvieron un cambio de estilo. Nos proponemos reemplazar los estilos actuales por los nuevos para unificar la estética de la aplicación.

Requerimiento

La línea de diseño que tienen todos los clientes móviles deben ser similares. Lo mismo debe ocurrir con la experiencia de usuario. El alcance de este requerimiento abarca todas las modificaciones, sin afectar la funcionalidad, para lograr que las animaciones y estilos del cliente *Policía de tránsito* sean similares a los usados en el cliente *Conductor*.

[P] Consulta de estacionamiento activo

Motivación

Tenemos que adaptar la consulta que realiza el cliente *Policía de tránsito* para que funcione con el nuevo esquema. Luego de los requerimientos **[C] Agregar tiempo de estacionamiento** y **[C] Finalizar estacionamiento**, se introdujeron cambios que hacen que se deba adaptar la forma en que este cliente consulta la vigencia del estacionamiento.

Requerimiento

El cliente *Policía de tránsito* debe poder consultar si un dominio tiene estacionamiento vigente en cualquier momento, de forma online (realizar la consulta sin asumir que el estado es el mismo, sin importar cuándo se haya consultado por última vez) y sin restricciones.

La única validación requerida es que el dominio ingresado tenga un formato válido.

Las respuestas esperadas sólo pueden ser dos:

- El estacionamiento está vigente, sin especificar cuánto tiempo lleva ni cuánto le queda.
- El estacionamiento **no** está vigente, sin especificar nada más.

Para ambos casos, se mostrará **la fecha y hora** en que el servidor respondió la petición, para tener como referencia en caso de que haya latencia entre el servidor y el cliente; también, cubre el caso en que el usuario vea el resultado de la consulta más tarde del momento en que realizó la consulta, por ejemplo, si bloquea el dispositivo por un tiempo.

Ejecución de Pruebas

Casos de prueba

CP – Login fallido con usuario vacío

Cliente

Conductor

Módulo

Login

Descripción

Al intentar autenticarse con el campo usuario vacío la aplicación no inicia el proceso de autenticación e informa al usuario sobre los datos faltantes.

Precondiciones

No aplica

Pasos de ejecución

1. Ingresar a la aplicación ParCo
2. Ingresar una contraseña (al menos 1 caracter) en campo “Contraseña”
3. Presionar el botón “Ingresar”

Resultado esperado

El login falla. La aplicación muestra el mensaje “Ingrese su nombre de usuario”

CP - Login fallido con contraseña vacía

Cliente

Conductor

Módulo

Login

Descripción

Al intentar autenticarse con el campo contraseña vacía la aplicación no inicia el proceso de autenticación e informa al usuario sobre los datos faltantes.

Precondiciones

No aplica

Pasos de ejecución

1. Ingresar a la aplicación ParCo
2. Ingresar un usuario (al menos 1 caracter) en campo “Nombre de usuario”
3. Presionar el botón “Ingresar”

Resultado esperado

El login falla. La aplicación muestra el mensaje “Ingrese su contraseña”

CP - Login fallido con usuario y contraseña vacíos

Cliente

Conductor

Módulo

Login

Descripción

Al intentar autenticarse con el campo usuario y contraseña vacíos la aplicación no inicia el proceso de autenticación e informa al usuario sobre los datos faltantes.

Precondiciones

No aplica

Pasos de ejecución

1. Ingresar a la aplicación ParCo
2. Ingresar un usuario (al menos 1 caracter) en campo "Nombre de usuario"
3. Ingresar una contraseña (al menos 1 caracter) en campo "Contraseña"
4. Presionar el botón "Ingresar"

Resultado esperado

El login falla. La aplicación muestra el mensaje "Ingrese su nombre de usuario" sobre el campo usuario e "Ingrese su contraseña" sobre el campo contraseña

CP - Login exitoso

Cliente

Conductor

Módulo

Login

Descripción

Al intentar autenticarse con datos de usuario correctos la aplicación autentica al usuario permitiéndole ingresar al sistema.

Precondiciones

- Existe un usuario con contraseña válido.

Pasos de ejecución

1. Ingresar a la aplicación ParCo
2. Ingresar el nombre de usuario en campo "Nombre de usuario"
3. Ingresar la contraseña en campo "Contraseña"
4. Presionar el botón "Ingresar"

Resultado esperado

El usuario se autentica. La aplicación no muestra ningún mensaje y redirige a la pantalla principal del sistema.

CP -Login fallido con usuario no válido

Cliente

Conductor

Módulo

Login

Descripción

Al intentar autenticarse con datos de usuario no válidos (usuario no registrado) la aplicación no autentica al usuario.

Precondiciones

- El usuario que se intenta autenticar no existe

Pasos de ejecución

1. Ingresar a la aplicación ParCo
2. Ingresar un usuario no existente en el campo “Nombre de usuario”
3. Ingresar una contraseña (al menos 1 caracter) en campo “Contraseña”
4. Presionar el botón “Ingresar”

Resultado esperado

El login falla. La aplicación muestra el mensaje “El usuario ingresado no existe”.

CP - Login fallido con usuario válido y contraseña incorrecta

Cliente

Conductor

Módulo

Login

Descripción

Al intentar autenticarse con datos de usuario válidos (usuario registrado) pero contraseña incorrecta la aplicación no autentica al usuario.

Precondiciones

- Existe un usuario con contraseña válido.

Pasos de ejecución

1. Ingresar a la aplicación ParCo
2. Ingresar un usuario válido en el campo “Nombre de usuario”
3. Ingresar una contraseña diferente a la asociada al usuario en el campo “Contraseña”
4. Presionar el botón “Ingresar”

Resultado esperado

El login falla. La aplicación muestra el mensaje “Contraseña incorrecta”.

CP - Abrir el menú con botón

Cliente

Conductor

Módulo

Menú


Descripción

Al apretar el botón de menú, se despliega un menú con las funcionalidades habilitadas para el usuario autenticado

Precondiciones

- Usuario autenticado
- Menú cerrado

Pasos de ejecución

1. Presionar el botón  que se encuentra en la esquina superior izquierda de la pantalla

Resultado esperado

Se despliega un menú a la izquierda de la pantalla que muestra todas las funciones habilitadas para el usuario autenticado y el nombre del mismo.

CP - Abrir el menú deslizando la pantalla

Cliente

Conductor

Módulo

Menú

Descripción

Al deslizar la pantalla, se despliega un menú con las funcionalidades habilitadas para el usuario autenticado.

Precondiciones

- Usuario autenticado
- Menú cerrado

Pasos de ejecución

1. Deslizar la pantalla desde el extremo izquierdo hacia el derecho

Resultado esperado

Se despliega un menú a la izquierda de la pantalla que muestra todas las funciones habilitadas para el usuario autenticado y el nombre del mismo.

CP - Cerrar el menú deslizando la pantalla

Cliente

Conductor

Módulo

Menú

Descripción

Al deslizar la pantalla, se repliega el menú que estaba abierto.

Precondiciones

- Usuario autenticado

ParCo | Sistema

- Menú abierto

Pasos de ejecución

1. Deslizar la pantalla desde el extremo derecho hacia el izquierdo

Resultado esperado

Se repliega el menú hacia la izquierda de la pantalla, hasta cerrarse por completo.

CP - Cerrar el menú tocando fuera del mismo*Cliente*

Conductor

Módulo

Menú

Descripción

Al tocar la pantalla donde no se muestra el menú, se repliega el mismo.

Precondiciones

- Usuario autenticado
- Menú abierto

Pasos de ejecución

1. Tocar la pantalla en una zona donde no se muestre el menú

Resultado esperado

Se repliega el menú hacia la izquierda de la pantalla, hasta cerrarse por completo.

CP - Consultar estacionamiento sin ninguno activo*Cliente*

Conductor

Módulo

Consulta estacionamiento

Descripción

Al consultar los estacionamientos activos, y no poseer ninguno, se informa ésto al usuario.

Precondiciones

- Usuario autenticado
- Menú abierto

Pasos de ejecución

1. Tocar el botón  con leyenda "Mis Estacionamientos"

Resultado esperado

La aplicación no cambia de pantalla y aparece un mensaje informativo con el texto "No se encuentra ningún estacionamiento activo".

CP - Consultar estacionamiento con uno activo

Cliente

Conductor

Módulo

Consulta estacionamiento

Descripción

Al consultar los estacionamientos activos, y poseer uno solo, se muestra la pantalla de consulta de estacionamiento que visualiza los datos del mismo.

Precondiciones

- Usuario autenticado
- Menú abierto
- Existe un único estacionamiento activo para el usuario

Pasos de ejecución

1. Tocar el botón  con leyenda "Mis Estacionamientos"

Resultado esperado

La aplicación redirecciona a la pantalla de consulta de estacionamiento y muestra los siguientes datos sobre el estacionamiento activo: dominio del vehículo, dirección del estacionamiento y tiempo restante.

CP - Consultar estacionamiento con más de uno activo

Cliente

Conductor

Módulo

Consulta estacionamiento


Descripción

Al consultar los estacionamientos activos, y poseer dos o más, se muestra la pantalla de consulta de estacionamiento que visualiza los datos del estacionamiento más próximo al vencimiento, independientemente del vehículo.

Precondiciones

- Usuario autenticado
- Menú abierto
- Existen dos estacionamientos activos, de diferentes vehículos, para el usuario

Pasos de ejecución

1. Tocar el botón  con leyenda "Mis Estacionamientos"

Resultado esperado

La aplicación redirecciona a la pantalla de consulta de estacionamiento y obtiene el estacionamiento activo **más próximo a vencer** (el estacionamiento debe ser el que posea la fecha y hora de fin más cercana, pero mayor, a la fecha y hora del sistema en el momento de la consulta); sobre éste se muestran los siguientes datos: dominio del vehículo, dirección del estacionamiento y tiempo restante.

CP - Consultar estacionamiento sin conectividad a internet

Cliente

Conductor

Módulo

Consulta estacionamiento

Descripción

Al consultar los estacionamientos activos, y sin disponer de conexión a internet, se avisa al usuario sobre la situación.

Precondiciones

- Usuario autenticado
- Menú abierto

Pasos de ejecución

1. Tocar el botón  con leyenda "Mis Estacionamientos"

Resultado esperado

La aplicación no cambia de pantalla y muestra el mensaje "No pudimos contactar al servidor. Por favor, revise su conectividad a internet".

CP - Consultar estacionamiento con el servidor no disponible

Cliente

Conductor

Módulo

Consulta estacionamiento

Estado

Invalidado (19/10/2017 19:30)

Descripción

Al consultar los estacionamientos activos, y estando el servidor no disponible, se avisa al usuario sobre la situación.

Precondiciones

- Usuario autenticado
- Menú abierto

Pasos de ejecución

1. Tocar el botón  con leyenda "Mis Estacionamientos"

Resultado esperado

La aplicación no cambia de pantalla y muestra el mensaje "Hubo un problema con el Servidor. Por favor, intente nuevamente más tarde".

Ejecuciones de casos de prueba

Ciclo 1

Inicio

18/10/2017 16:30

Fin

19/10/2017 18:00

#1 - CP - Login fallido con usuario vacío

Fecha ejecución

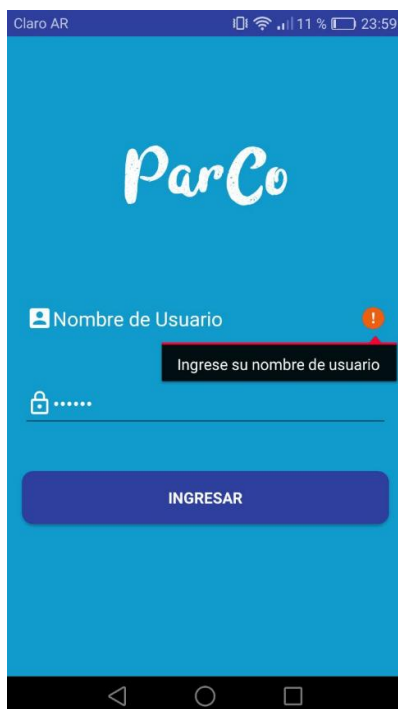
18/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación.
2. Se deja vacío el campo nombre de usuario
3. Se ingresa una contraseña de al menos un carácter.
4. Se Presiona el botón "Ingresar"

Resultado obtenido

El caso de Prueba se ejecuta correctamente.



#1 - CP - Login fallido con contraseña vacía

Fecha ejecución

18/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación.
2. Se ingresa el nombre de usuario "mezi".
3. Se deja vacío el campo "contraseña".
4. Se Presiona el botón "Ingresar".

Resultado obtenido

El caso de Prueba se ejecuta correctamente



#1 - CP - Login fallido con usuario y contraseña vacíos

Fecha ejecución

18/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación.
2. Se deja vacío el campo "nombre de usuario".
3. Se deja vacío el campo "contraseña".
4. Se Presiona el botón "Ingresar".

Resultado obtenido

El caso de Prueba se ejecuta correctamente

ParCo | Sistema



#1 - CP - Login exitoso

Fecha ejecución

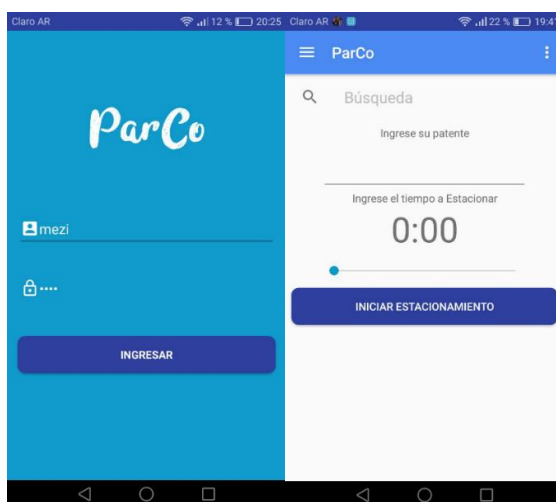
18/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación.
2. Se ingresa el nombre de usuario “mezi” en el campo “Nombre de usuario”.
3. Se ingresa la contraseña “mezi” en el campo “Contraseña”.
4. Se presiona el botón “Ingresar”.

Resultado obtenido

El caso de Prueba se ejecuta correctamente



#1 - CP - Login fallido con usuario no válido

Fecha ejecución

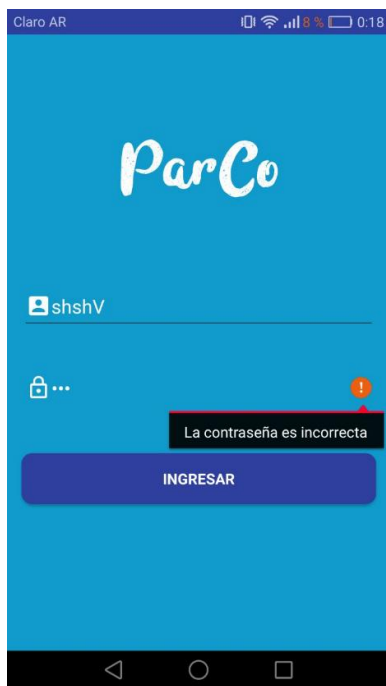
18/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación ParCo.
2. Se Ingresa el usuario "shshV" (no existente) en el campo "Nombre de usuario".
3. Se ingresa la contraseña "123456" en el campo "Contraseña".
4. Se presiona el botón "Ingresar"

Resultado obtenido

El caso de prueba Falla. La aplicación muestra el mensaje "La contraseña es incorrecta"



#1 - CP - Login fallido con usuario válido y contraseña incorrecta

Fecha ejecución

18/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación ParCo.
2. Se Ingresa el usuario "mezi" en el campo "Nombre de usuario".
3. Se ingresa la contraseña "1234" (diferente a la asociada al usuario) en el campo "Contraseña".
4. Se Presiona el botón "Ingresar".

Resultado obtenido

El caso de Prueba se ejecuta correctamente




#1 - CP - Abrir el menú con botón

Fecha ejecución

18/10/2017

Pasos ejecutados

1. Presionar el botón  que se encuentra en la esquina superior izquierda de la pantalla.

Resultado obtenido

El caso de prueba se ejecuta correctamente.

#1 - CP - Abrir el menú deslizando la pantalla

Fecha ejecución

18/10/2017

Pasos ejecutados

1. Se desliza la pantalla desde el extremo izquierdo hacia el derecho.

Resultado obtenido

El caso de prueba se ejecuta correctamente.

*#1 - CP - Cerrar el menú deslizando la pantalla***Fecha ejecución**

18/10/2017

Pasos ejecutados

1. Se desliza la pantalla desde el extremo derecho hacia el izquierdo.

Resultado obtenido

Se repliega el menú hacia la izquierda de la pantalla, hasta cerrarse por completo.

*#1 - CP - Cerrar el menú tocando fuera del mismo***Fecha ejecución**

18/10/2017

Pasos ejecutados

1. Se presiona la pantalla en una zona donde no se muestre el menú.


Resultado obtenido

Se repliega el menú hacia la izquierda de la pantalla, hasta cerrarse por completo.

*#1 - CP - Consultar estacionamiento sin ninguno activo***Fecha ejecución**

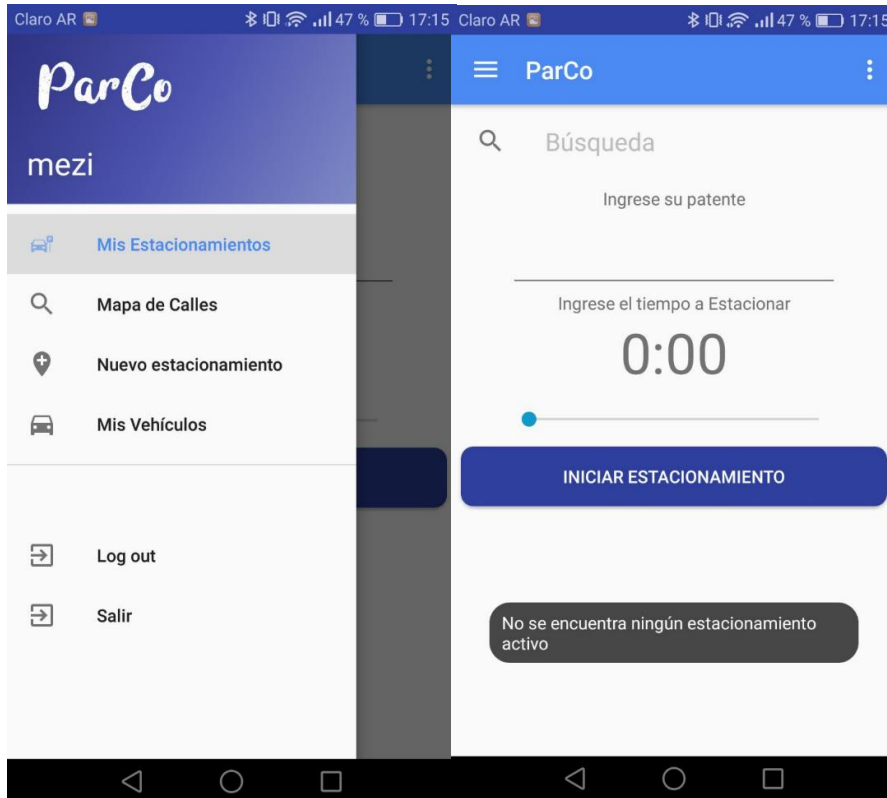
19/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación Parco.
2. Se ingresa el nombre de usuario "mezi" y la contraseña "mezi".
3. Se presiona el botón "Ingresar".
4. Se presiona el botón  del menú con leyenda "Mis Estacionamientos".

Resultado obtenido

El caso de prueba se ejecuta correctamente.




#1 - CP - Consultar estacionamiento con uno activo

Fecha ejecución

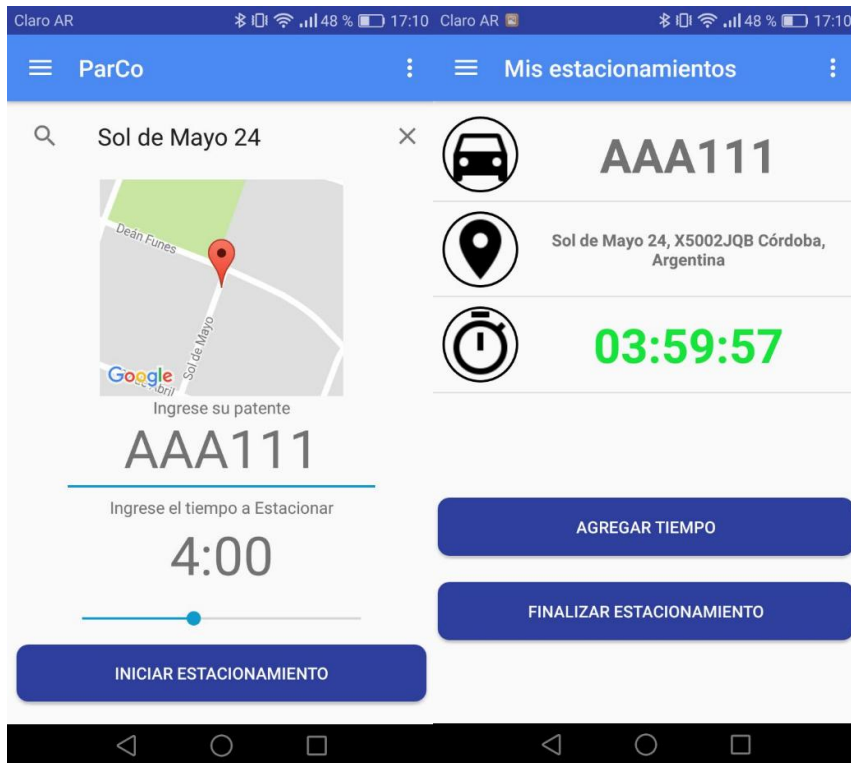
19/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación ParCo.
2. Se ingresa el nombre de usuario "mezi" y la contraseña "mezi".
3. Se presiona el botón "Ingresar".
4. Se registra un estacionamiento.
5. Se presiona el botón  con leyenda "Mis Estacionamientos".

Resultado obtenido

El caso de prueba se ejecuta correctamente.




#1 - CP - Consultar estacionamiento con más de uno activo

Fecha ejecución

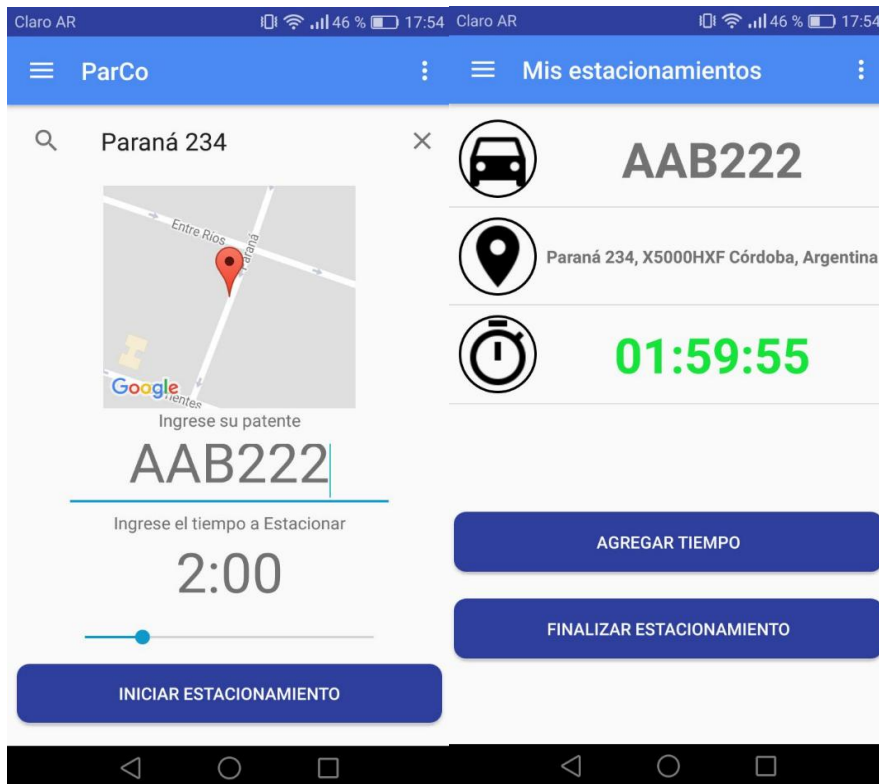
19/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación ParCo.
2. Se ingresa el nombre de usuario "mezi" y la contraseña "mezi".
3. Se presiona el botón "Ingresar".
4. Se registra un estacionamiento para el vehículo dominio "AAA111" con duración de 4 horas
5. Se registra un estacionamiento para el vehículo dominio "AAB222" con duración de 2 horas.
6. Se presiona el botón  con leyenda "Mis Estacionamientos"

Resultado obtenido

La aplicación sólo muestra un estacionamiento activo, el más próximo a vencer, con un tiempo restante aproximado de 2 horas.




#1 - CP - Consultar estacionamiento sin conectividad a internet

Fecha ejecución

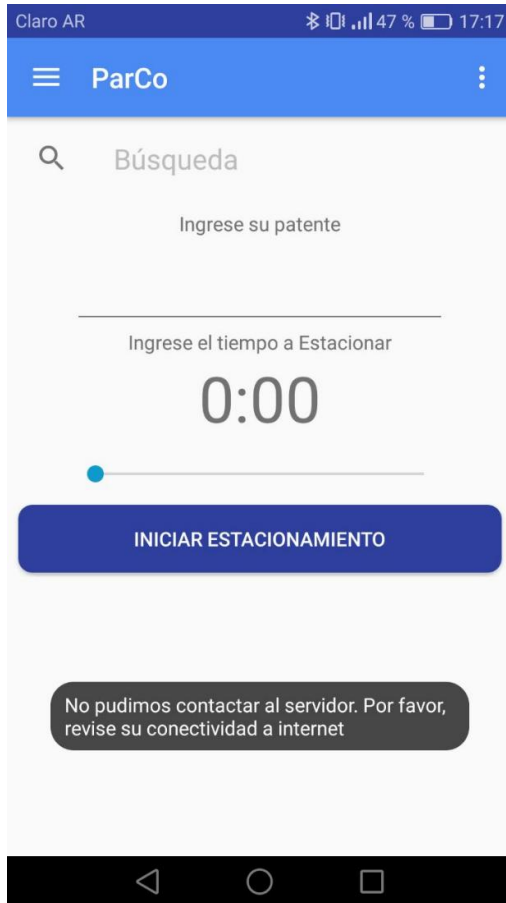
19/10/2017

Pasos de ejecutados

1. Se ingresa a la aplicación ParCo.
2. Se ingresa el nombre de usuario "mezi" y la contraseña "mezi".
3. Se apagan todas las conectividades a internet que el teléfono posea (Wifi , datos).
4. Se presiona el botón  con leyenda "Mis Estacionamientos".

Resultado obtenido

El caso de prueba se ejecuta correctamente




#1 - CP - Consultar estacionamiento con el servidor no disponible

Fecha ejecución

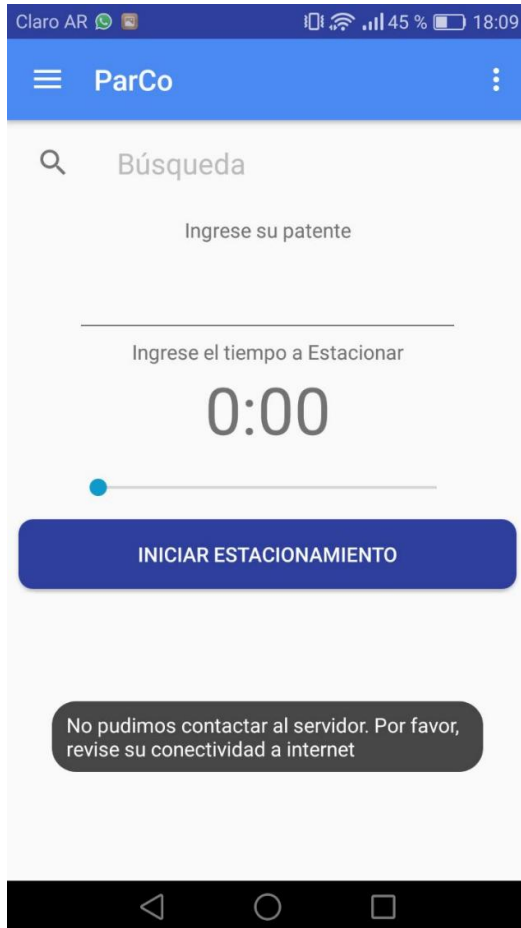
19/10/2017

Pasos de ejecutados

1. Se ingresa a la aplicación Parco.
2. Se ingresa el nombre de usuario "mezi" y la contraseña "mezi".
3. Se presiona el botón  con leyenda "Mis Estacionamientos".

Resultado obtenido

El caso de prueba falla. La aplicación devuelve el mensaje “No pudimos contactar al servidor. Por favor, revise su conectividad a Internet”.



Ciclo 2

Inicio

19/10/2017 20:00

Fin

19/10/2017 20:30

#2 - CP - Login fallido con usuario no válido

Fecha ejecución

19/10/2017

Pasos ejecutados

1. Se ingresa a la aplicación ParCo.
2. Se Ingresa el usuario "POluyt" (no existente) en el campo "Nombre de usuario".
3. Se ingresa la contraseña "9876" en el campo "Contraseña".
4. Se presiona el botón "Ingresar".

Resultado obtenido

El caso de Prueba se ejecuta correctamente. La aplicación muestra el mensaje "El usuario ingresado no existe".

Resumen de ejecuciones de casos de pruebas

	CP CREADOS	CP EJECUTADOS	CP PASARON	CP FALLARON	DEFECTOS
CICLO 1	15	15	14	1	1
CICLO 2	0	1	1	0	0
TOTAL	15	16			1

CP: Caso de prueba

Defectos encontrados

Título	Validación dirección Nula
Descripción	Al intentar registrar un estacionamiento sin ingresar ninguna ubicación, la aplicación deja de responder, en los logs se evidencia excepción de tipo "NullPointerException"
Componente	parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Major
Iteración de creación	2
Iteración de Resolución	2

ParCo | Sistema

Título	Error al ingresar un valor de tiempo negativo
Descripción	Al registrar un estacionamiento utilizando un valor negativo para el campo "Cantidad de Horas a Estacionar", la aplicación deja de responder. El backend loguea el error lanzado por la base de datos de tipo "ERR 23514 – Checkconstraint"
Componente	parco-c-parking
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Major
Iteración de creación	2
Iteración de Resolución	2

Título	Error al ingresar un valor de tiempo negativo.
Descripción	Al registrar un estacionamiento utilizando un valor negativo para el campo "Cantidad de Horas a Estacionar", la aplicación deja de responder. El backend lanza el mensaje "La cantidad de horas a estacionar debe ser mayor o igual a 1"
Componente	parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Major
Iteración de creación	2
Iteración de Resolución	2

Título	La dirección ingresada por el usuario no se persiste en la base de datos
Descripción	Tras ingresar la dirección a estacionar, el sistema geolocaliza de forma correcta pero no persiste la información en la base de datos.
Componente	Parco-c-parking
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Critical
Iteración de creación	2
Iteración de Resolución	2

ParCo | Sistema

Título	La información horaria se guarda en formato UTC
Descripción	La información referida a fechas y horas se está guardando en formato UTC. Es necesario que se guarde según zona horaria
Componente	Parco-c-parking
Responsable	Pianello, Martín
Tipo	Bug
Prioridad	Minor
Iteración de creación	2
Iteración de Resolución	2

Título	Usuario inexistente muestra mensaje incorrecto
Descripción	En el login, al ingresar un usuario que no existe en el sistema, la aplicación muestra el mensaje "Contraseña incorrecta" debiendo mostrar el mensaje "El usuario es incorrecto"
Componente	parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Minor
Iteración de creación	3
Iteración de Resolución	3

Título	"Mis estacionamientos" muestra mensaje incorrecto con servidor no disponible
Descripción	Al consultar la funcionalidad "mis estacionamientos", estando el servidor no disponible, el sistema muestra el mensaje incorrectamente el mensaje "No pudimos contactar al servidor. Por favor, revise su conectividad a Internet" en lugar de "Hubo un problema con el Servidor. Por favor, intente nuevamente más tarde"
Componente	parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Minor
Iteración de creación	3
Iteración de Resolución	3

ParCo | Sistema

Título	Error en solicitud de permisos
Descripción	La aplicación no solicita todos los permisos necesarios al instalar la aplicación. Esto ocasiona que, de no tener el permiso de obtener ubicación, la aplicación entre en un ciclo infinito al intentar obtenerla.
Componente	Parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Major
Iteración de creación	5
Iteración de Resolución	5

Título	Manejo de vencimiento de token MercadoPago
Descripción	El token de autenticación de la API de MercadoPago tiene un vencimiento de 24horas. Actualmente Parco utiliza un token fijo para autenticar por lo que necesita contemplarse un mecanismo de actualización de este
Componente	Parco-c-parking
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Blocker
Iteración de creación	6
Iteración de Resolución	6

Título	Reubicación en mapa tras cambiar ubicación
Descripción	Tras ingresar una dirección, se realiza la validación de esta y se ubica dentro del mapa. Tras corregir la dirección, el componente no realiza una nueva validación y por consiguiente no dibuja nuevamente en el mapa.
Componente	Parco-a-web
Responsable	Pianello Martín
Tipo	Bug
Prioridad	Minor
Iteración de creación	6
Iteración de Resolución	6

ParCo | Sistema

Título	Loader de validaciones en componente WEB
Descripción	La barra de estado de validaciones del componente web (Loader) se actualiza correctamente mientras valida, aunque no se oculta una vez finalizadas las mismas
Componente	Parco-a-web
Responsable	Pianello, Martin
Tipo	Bug
Prioridad	Minor
Iteración de creación	6
Iteración de Resolución	6

Título	Mensaje incorrecto al validar el campo "Hasta"
Descripción	Tras ingresar un valor en campo "Hasta" menor al campo "Desde" la aplicación muestra el mensaje "Altura inválida", debiendo ser el mismo "La altura Hasta no puede ser menor a la altura Desde"
Componente	Parco-a-web
Responsable	Pianello, Martin
Tipo	Bug
Prioridad	Minor
Iteración de creación	6
Iteración de Resolución	6

Título	Error en validaciones al realizar la carga rápidamente
Descripción	Si se cargan todos los campos tabulando antes de que termine de realizarse la validación, el componente deja de validar y permite guardar cualquier valor
Componente	Parco-a-web
Responsable	Pianello, Martin
Tipo	Bug
Prioridad	Minor
Iteración de creación	6
Iteración de Resolución	6

Título	Listado de calles no se actualiza al editar sección
Descripción	Tras editar una sección, el cambio no se ve reflejado en el listado de secciones
Componente	Parco-a-web
Responsable	Pianello, Martin
Tipo	Bug
Prioridad	Minor
Iteración de creación	6

ParCo | Sistema

Iteración de Resolución	6
Título	Permiten cargarse calles solapadas
Descripción	No existe ninguna validación que restrinja el solapamiento de calles. Es decir, el sistema permite cargar una sección prohibida "Calle A 100 al 200" y una sección permitida "Calle A 120 al 180"
Componente	Parco-street-sections
Responsable	Pianello, Martín
Tipo	Bug
Prioridad	Mayor
Iteración de creación	6
Iteración de Resolución	6

Título	Ambigüedad de nombres de calles con ciudades
Descripción	Determinadas calles, cuyo nombre coincide con nombre de ciudades, provocan error en el componente validador. Ejemplo "Santa Fé", "Buenos Aires"
Componente	Parco-a-web
Responsable	Pianello, Martín
Tipo	Bug
Prioridad	Mayor
Iteración de creación	6
Iteración de Resolución	6

Título	Falla en componente web cuando se edita durante la validación
Descripción	Mientras se están corriendo las validaciones, tras cambiar el texto el componente deja de validar permitiendo ingresar cualquier valor.
Componente	Parco-a-web
Responsable	Pianello, Martín
Tipo	Bug
Prioridad	Minor
Iteración de creación	6
Iteración de Resolución	6

ParCo | Sistema

Título	Fecha incorrecta en historial de estacionamientos
Descripción	La fecha que se muestra en el historial de estacionamientos es incorrecta. Se está convirtiendo de manera incorrecta el LocalDateTime.
Componente	Parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Minor
Iteración de creación	7
Iteración de Resolución	7

Título	Vehículos favoritos – No se oculta componente
Descripción	El desplegable “Vehículos favoritos” no se oculta tras seleccionar una opción del desplegable.
Componente	Parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Minor
Iteración de creación	7
Iteración de Resolución	7

Título	Mapa de calles – Baja performance con mucha cantidad de calles
Descripción	Al cargar más de 100 secciones de calles, el mapa presenta una lentitud extrema al dibujar las mismas en el mapa
Componente	Parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Minor
Iteración de creación	8
Iteración de Resolución	8

Título	Agregar tiempo de estacionamiento – tiempo negativo
Descripción	Al agregar tiempo de estacionamiento el componente permite valores negativos. Al ingresar un valor negativo, la aplicación deja de responder
Componente	Parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Major
Iteración de creación	9
Iteración de Resolución	9

ParCo | Sistema

Título	Mapa de calor – Acumulado de varios días no funciona al cambiar de mes
Descripción	Al seleccionar mediante el filtro, un acumulado de días que continúe de un mes a otro (por ejemplo 27/07 al 03/08) el mapa de calor solo contempla las fechas contempladas en el último mes (en este ejemplo, contempla del 01 al 03 de agosto)
Componente	Parco-c-mobile
Responsable	Pianello, Martin
Tipo	Bug
Prioridad	Minor
Iteración de creación	10
Iteración de Resolución	10

Título	Filtro de Secciones de calle – Filtro contempla columnas
Descripción	Al filtrar para encontrar una sección, el componente realiza una búsqueda sobre toda la información cargada en la tabla, incluidas las columnas. Por lo que al filtrar por una cadena que coincida con el nombre de una columna (ejemplo, al introducir la cadena de caracteres “tip” para buscar la calle “Tipayante”, el filtro muestra todos los resultados, ya que coincide con la columna “Tipo”)
Componente	Parco-a-web
Responsable	Pianello, Martin
Tipo	Bug
Prioridad	Minor
Iteración de creación	11
Iteración de Resolución	11

Título	Creación estacionamiento – Ubicación Manual permite “sitios”
Descripción	Al introducir de forma manual la ubicación, el componente permite seleccionar “Sitios” en lugar de solo calles. Ejemplo, al introducir “Mons” para que la aplicación sugiera la calle “Monseñor Pablo Cabrera”, la aplicación sugiere también “Montserrat”
Componente	Parco-k-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Major
Iteración de creación	11
Iteración de Resolución	11

ParCo | Sistema

Título	Ubicar vehículo estacionado
Descripción	Al seleccionar la opción “¿Cómo llegar?”, se muestra una ruta que respeta las vías de circulación de las calles, asumiendo una ruta en vehículo. Es necesario proveer la ruta más corta asumiendo que el usuario se desplaza a pie
Componente	Parco-c-mobile
Responsable	Mezi, Mariano
Tipo	Bug
Prioridad	Minor
Iteración de creación	12
Iteración de Resolución	12

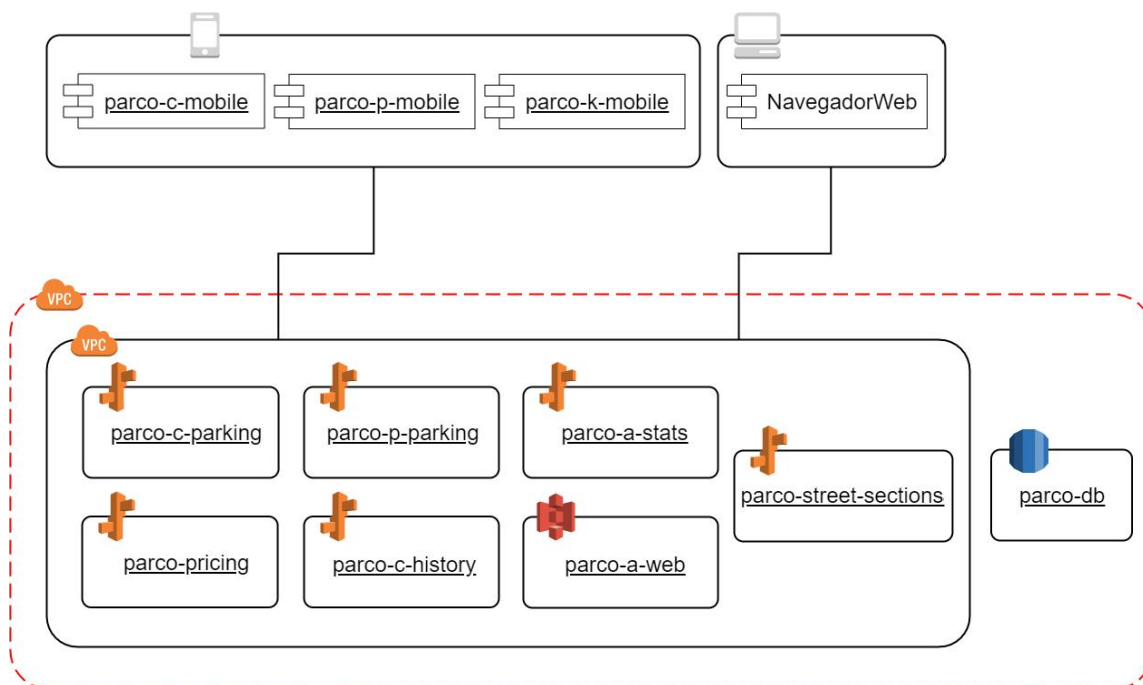
Plan de despliegue del sistema

Objetivos

Definir y documentar los requerimientos, de Software y Hardware, y la secuencia de actividades requeridos para desplegar el producto completo en un entorno productivo capaz de ser utilizado por los usuarios finales. También, definir las tareas de seguimiento y mantenimiento necesarias para que éste continúe operando.

Vista de despliegue

Vista que permite ver la organización lógica de los componentes de la solución



Referencias

Significado y aclaraciones sobre la simbología y alcance de los componentes utilizados en la vista de despliegue.

Simbología

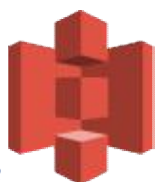
Descripción de la simbología utilizada para definir los contenedores de instancias ejecutables en entorno Cloud. Se utiliza la simbología de Amazon Web Services (AWS), por ser el proveedor recomendado, pero podría utilizarse otro.



AWS Compute - ElasticBeanstalk

Servicio de AWS(sección *Compute*) que proporciona plataformas preconfiguradas para ejecutar aplicaciones web. Este servicio se encarga de aprovisionar los recursos necesarios para el despliegue y ejecución de una aplicación según la plataforma seleccionada.

En todos los casos, la plataforma seleccionada es “Java 8 running on 64bit Amazon Linux”, la cual provee un entorno con sistema operativo basado en Linux y una JRE 8. Una vez creado y configurado el entorno, la plantilla detecta que el binario proporcionada es una aplicación *Spring Boot* y la ejecuta como instancia autocontenida.



AWS Storage - S3

Servicio de AWS (sección *Storage*) que provee medios de almacenamiento. Opcionalmente, permite configurarse para funcionar como un servidor de contenido estático mediante la opción “Amazon Simple Storage Service”.

En nuestro caso, la opción *Amazon Simple Storage Service* está habilitada para servir la aplicación Angular.



AWS Database - RDS

Servicio de AWS (sección *Database*) que provee plantillas para crear, configurar y mantener instancias de servidores de base de datos. Este servicio se encarga de aprovisionar los componentes de almacenamiento, cómputo y el software de base de datos automáticamente según la selección.

En nuestro caso, seleccionamos el motor *PostgreSQL*.



AWS Network – VPC

Servicio de AWS(sección *Network*) que provee redes virtuales privadas asociadas a la cuenta de AWS. El conjunto de componentes de AWS que estén dentro de una VPC tienen visibilidad entre sí a nivel de red mediante una dirección IPv4 privada que se asigna a cada uno. Al tratarse de una red privada, los componentes pueden interrelacionarse entre ellos, por ejemplo, una instancia de computo con una aplicación se puede conectar a la una instancia de base de datos. Opcionalmente, se puede configurar un Gateway para darle a la VPC acceso a internet, mediante una dirección pública.

En nuestro caso, hay dos VPCs:

ParCo | Sistema

- La primera, más amplia, es una red privada que contiene todos los nodos administrables, permitiendo visibilidad entre ellos. No se requieren subredes.
- La segunda, mas reducida, es una red con acceso a internet mediante un gateway que contiene el subconjunto de nodos que exponen servicios o datos.

*Clientes móviles*

Representa los componentes que están disponibles en el App Store de Google para instalar en dispositivos Android que cumplan con los requerimientos especificados para cada componente.

*Clientes web*

Representa cualquier cliente web capaz de ejecutar los componentes genéricos especificados en el diagrama.

Instancias

Descripción de instancias ejecutables de Software que forman parte del producto.

parco-c-parking

Componente de software encargado de las siguientes funcionalidades relacionadas al cliente *Conductor*:

- Registro de estacionamiento
- Extender tiempo a estacionamiento vigente
- Validar ubicación de estacionamiento
- Finalizar estacionamiento

parco-p-parking

Componente de software encargado de las siguientes funcionalidades relacionadas al cliente *Policía de tránsito*:

- Consulta de estacionamiento vigente

parco-pricing

Componente de software encargado de las siguientes funcionalidades relacionadas al módulo de tarifa:

- Consulta de tarifa vigente
- Consulta de historial de tarifas
- Definición de tarifa única

parco-c-history

Componente de software encargado de las siguientes funcionalidades relacionadas al módulo consulta de datos de estacionamientos:

- Consulta de estacionamiento próximo a vencer de un usuario
- Consulta de estacionamientos activos de un usuario
- Consulta de dominios frecuentemente usados de un usuario

ParCo | Sistema

- Consulta de todos los estacionamientos de un dominio
- Consulta de todos los estacionamientos de un usuario

parco-a-stats

Componente de software encargado de las siguientes funcionalidades relacionadas a la consulta de estadísticas para el usuario *Administrador*:

- Consulta de cantidad de estacionamientos activos
- Consulta de cantidad de estacionamientos registrados en un día
- Consulta de estacionamientos registrados en un período
- Consulta de ubicaciones de estacionamientos en un período.
- Generación de reporte “Estacionamientos por hora”

parco-street-sections

Componente de software encargado de las siguientes funcionalidades relacionadas a las secciones de calles:

- Consulta de secciones de calle según límites de coordenadas
- Alta de una sección de calle
- Baja de una sección de calle
- Modificación de una sección de calle
- Consulta de una sección de calle

parco-a-web

Componente de software encargado de manejar la presentación web del cliente *Administrador*.

parco-db

Componente de software encargado de manejar la persistencia de la aplicación.

parco-c-mobile

Componente de software encargado de manejar la presentación Android del cliente *Conductor*.

parco-p-mobile

Componente de software encargado de manejar la presentación Android del cliente *Policía de tránsito*.

parco-k-mobile

Componente de software encargado de manejar la presentación Android del cliente *Kiosco*.

Requisitos de Hardware

Los requisitos óptimos de hardware van a depender del poder de cómputo del proveedor cloud contratado, pero más aun de la cantidad de clientes concurrentes que se necesite. Nosotros propondremos una configuración mínima para cada instancia.

Instancia parco-c-parking

Al ser la instancia que realiza la mayoría de las transacciones tiene requisitos de hardware mayores al resto.

- 4 CPUs virtuales
- 8 GB RAM

Resto de las instancias

- 2 CPUs virtuales
- 4 GB RAM

Instancia de Base de datos

- 8 CPUs virtuales
- 32 GB RAM

Todas las instancias

Todas las instancias deben estar configuradas con escalamiento automática, con un mínimo de 2 instancias, para tener alta disponibilidad, a la vez que redimensionar el cluster cuando la aplicación tenga un pico de peticiones.

Además de esta configuración básica, la configuración óptima va a depender de la cantidad de usuarios concurrentes que estime el municipio.

Requisitos de Software

Al utilizar un proveedor de infraestructura cloud de tipo PaaS (Platform as a Service), las instancias ya cuentan con el sistema operativo y middleware necesario para la ejecución de la aplicación.

Instancias de cómputo

- SO Linux 64 bits
- JRE 8 o superior

Instancias de base de datos

- SO Linux 64 bits
- Motor de base de datos PostgreSQL 9.6.1 o superior

Estrategia de despliegue

Despliegue de componentes Cloud

La configuración inicial de los entornos, así como la instalación de cada nueva actualización queda a cargo del equipo ParCo. Una vez instalado el sistema, cada municipio debe manejar la configuración de la VPC pública para controlar el acceso productivo a la aplicación.

Si luego de la instalación inicial, el equipo de IT del municipio desea modificar la cantidad de instancias, la escalabilidad y el tipo de instancia usadas, podrá hacerlo; ninguna configuración debe quedar por debajo de los requisitos mínimos de hardware detallados anteriormente.

Despliegue de clientes Android

El equipo de ParCo será el encargado de realizar la instalación del aplicativo “Parco – Policía de tránsito” en los dispositivos provistos por el municipio (el mismo no se encontrará subido dentro del Google Play Store para evitar confusión o usos indebidos de usuarios conductores).

El equipo de Parco se encargará también de realizar el despliegue de la aplicación “ParCo – Conductor” y “ParCo – Kiosco” dentro del Google Play Store, disponible para ser descargada por todos los usuarios Android.

Si bien las aplicaciones para los clientes *Conductor* y *Kiosco* estarán disponibles para todos los usuarios, al requerir un inicio de sesión, sólo los usuarios que cuenten con una cuenta Kiosco válida podrán ingresar a la aplicación. Para esto se provee un procedimiento de alta de usuario Kiosco.

Actividades de despliegue

1. Definición de Instancias de Servidores. Cantidad / tipo
 - En función de las necesidades particulares del cliente, cantidad de usuarios estimada, disponibilidad, etc. se estudian y determinan en conjunto, la cantidad y características de las instancias de servidores a contratar.
 - Deberán evaluarse la cantidad de usuarios de tipo “Policía de tránsito” que operarán el sistema para proveer los archivos .apk necesarios.
Duración estimada: 3 semanas de trabajo
2. Configuración de instancias
 - Se deben configurar las instancias contratadas, y realizar pruebas que aseguren que las mismas se encuentran listas para responder a todas las funcionalidades de la aplicación.
 - Se realizará la instalación de la aplicación “Parco – Policía de tránsito” en todos los dispositivos provistos por el municipio.
 - Realizar la configuración y despliegue de la aplicación “Parco – Conductor” y “Parco – Kiosco” en el Google Play Store.
Duración estimada: 3 semanas de trabajo
3. Capacitación a usuarios
 - Los tiempos de capacitación para los usuarios de *Policía de tránsito* y *Administrador* dependerán de la cantidad de usuarios necesarios, ya que las mismas tienen un límite de usuarios por capacitación. Suponiendo un mínimo de 15 cuentas *Policía de tránsito* y *Administrador*, se requerirían 10 horas de capacitación.
Duración estimada: 1 semana de trabajo

Responsabilidad del equipo ParCo

El equipo de ParCo no se responsabiliza por pérdida de disponibilidad de la aplicación causada por el proveedor de servicios de infraestructura, ni por cambios en la configuración que no respeten los requisitos detallados en este documento.

Cualquier defecto o comportamiento inesperado de la aplicación debe ser reportado al equipo para evaluar su criticidad y prioridad. No se comprometen plazos de entrega de actualizaciones o correcciones.

Responsabilidad del cliente

El equipo de IT del municipio es responsable de crear y administrar la cuenta del proveedor de infraestructura seleccionado. Luego, debe proveerle al equipo de ParCo un usuario con privilegios suficientes para la creación, configuración y modificación de instancias.

Manuales de Usuario

ParCo Conductor

Instalación

Requisitos de aplicación

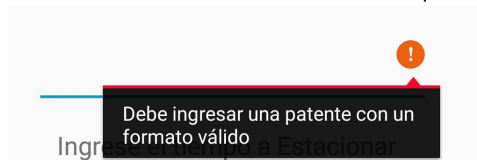
- Smartphone Android 4.4 KitKat o superior.
- Contar con un plan de datos móviles.
- Contar con Conectividad GPS habilitada (recomendable)

Generalidades

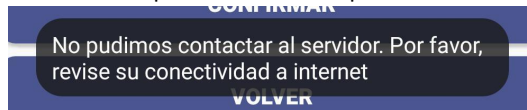
Errores

La aplicación informará errores de dos maneras distintas:

- 1- Error en información ingresada por el usuario:
 - o El sistema informará el error mediante un mensaje en el/los campo/s que se encuentren con información requerida faltante o errónea.

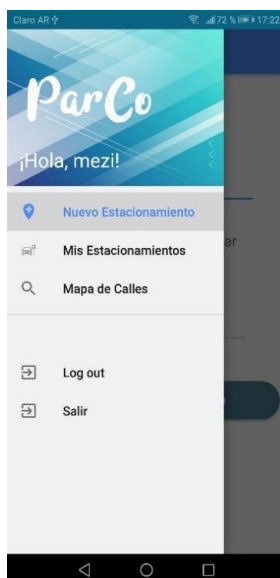


- 2- Error de conectividad con el servidor (Falta de internet, servidor no disponible, etc)
 - o El sistema informará el error mediante un mensaje informando la raíz del problema, en la parte inferior de la pantalla durante 4 segundos.



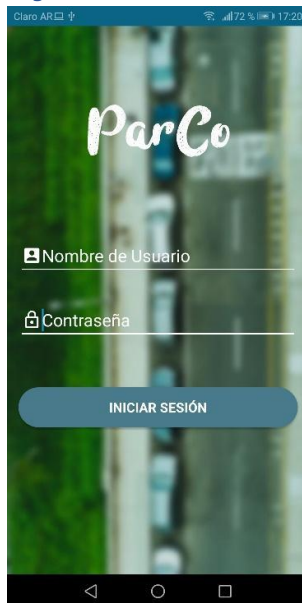
Navegabilidad

La navegabilidad del sistema estará guiada por un menú lateral desde el cual se podrán acceder a las distintas funcionalidades del sistema



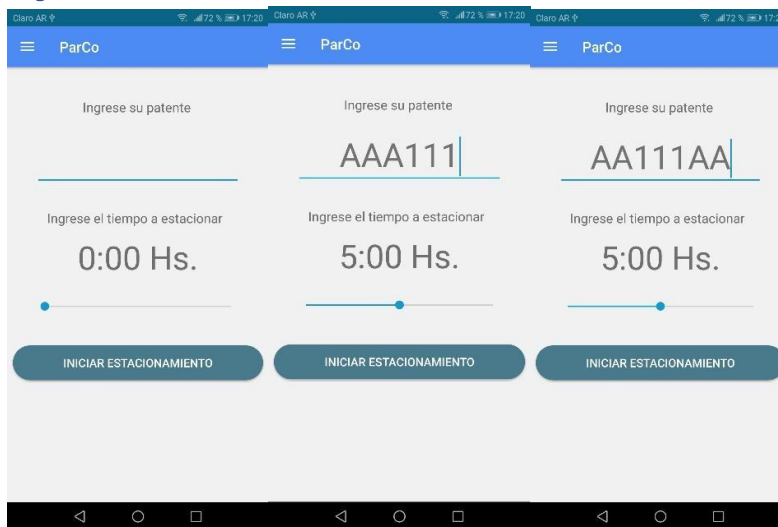
Módulos Funcionales

Login



Punto de acceso al sistema. Se deberán ingresar de manera obligatoria “Nombre de Usuario” y “Contraseña”, debiendo los mismos ser válidos. Al ingresar de manera exitosa, se mostrará la pantalla de “Nuevo Estacionamiento”

Registro De Estacionamiento



Se deberá ingresar una patente con los formatos válidos para Argentina, siendo los mismos del tipo “AAA111” o “AA111AA”.

El tiempo a estacionar debe ser ingresado deslizando la barra en el espacio inferior de la pantalla, pudiendo estacionar de 1 a 10 horas.

La aplicación mostrará un “loader” mientras intenta establecer la ubicación (Se recomienda fuertemente contar con la conectividad GPS activada, para facilitar la detección de la misma) hasta lograr detectarla de manera correcta, procediendo a mostrar una pantalla de

ParCo | Sistema


confirmación, dependiendo la misma de la ubicación en la que nos encontremos, pudiendo ser:

Ubicación Tarifada

Ubicación Gratuita

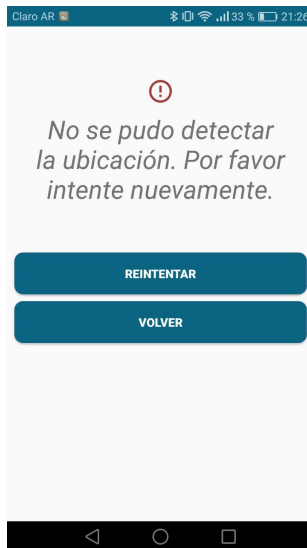
Ubicación Prohibida



- Ubicación actual, permitiendo actualizar la misma a través del ícono  (obtiene resultados de ubicación con un error de no más de 15 metros, **requiere gps activado**)
- Dominio del vehículo a estacionar
- Confirmación del tiempo Requerido
- Precio total del estacionamiento.
- El botón "Confirmar" solo se habilitará en tramos habilitados.
- El botón "VOLVER" Permitirá volver a la pantalla anterior.

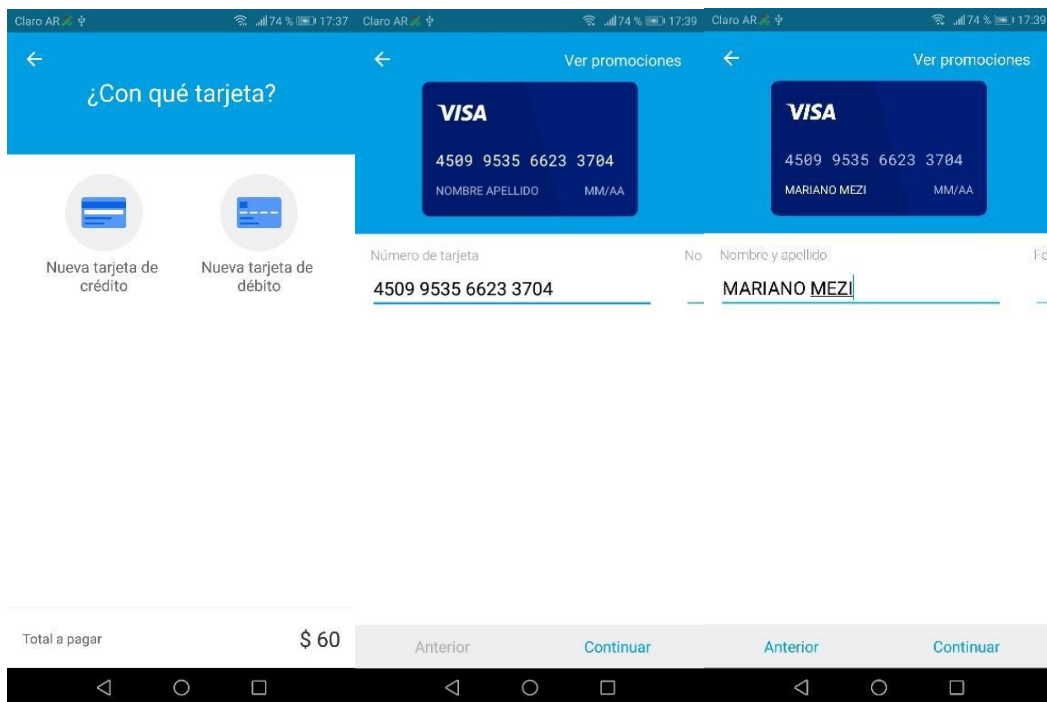
ParCo | Sistema

En caso de no poder detectar la ubicación, el sistema informará al usuario de esta situación a través de la siguiente pantalla, permitiendo reintentar la detección:



Módulo de Pago.

La API de pago utilizada por ParCo es MercadoPago. El diseño de su interfaz es intuitivo y guiado paso por paso:

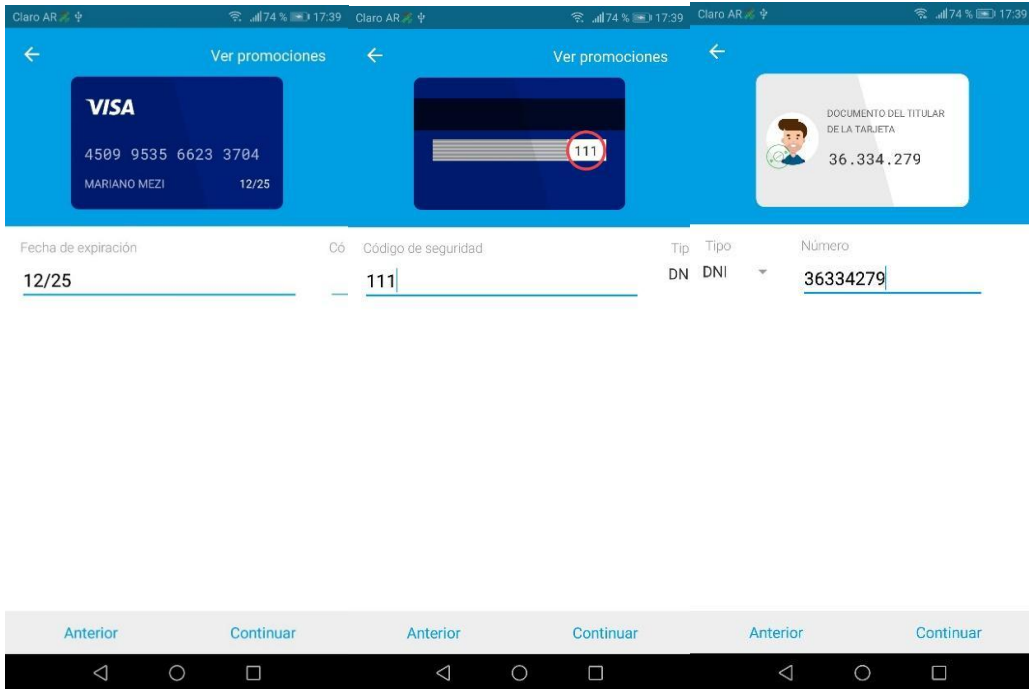


Elección de Medio de Pago

Ingreso número de Tarjeta

Ingreso Nombre de Titular

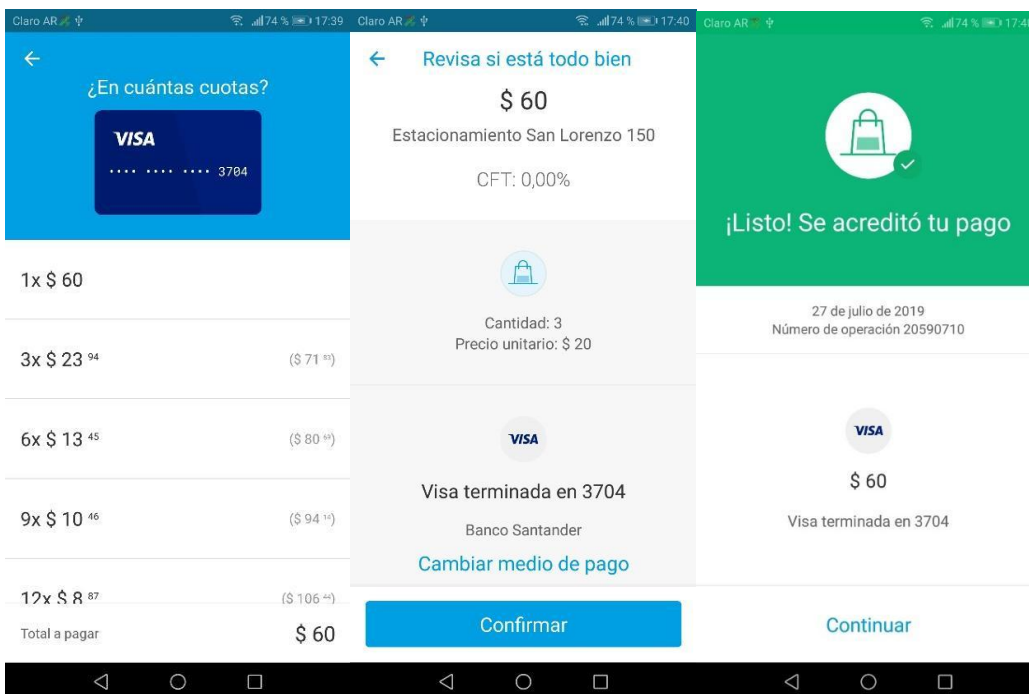
ParCo | Sistema



Ingreso de fecha de vencimiento

Ingreso código de Seguridad

Ingreso DNI de titular



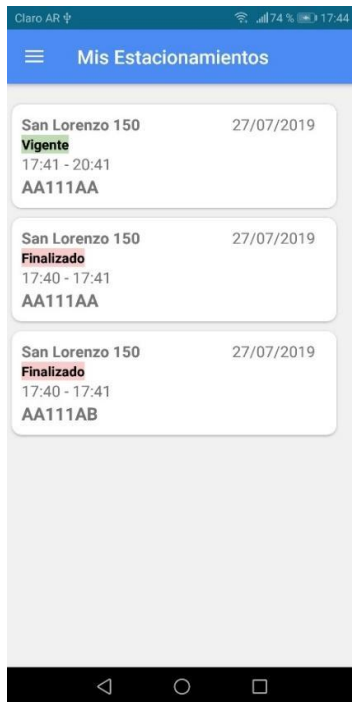
Elección cantidad de cuotas

Confirmación de Pago

Información de pago confirmado

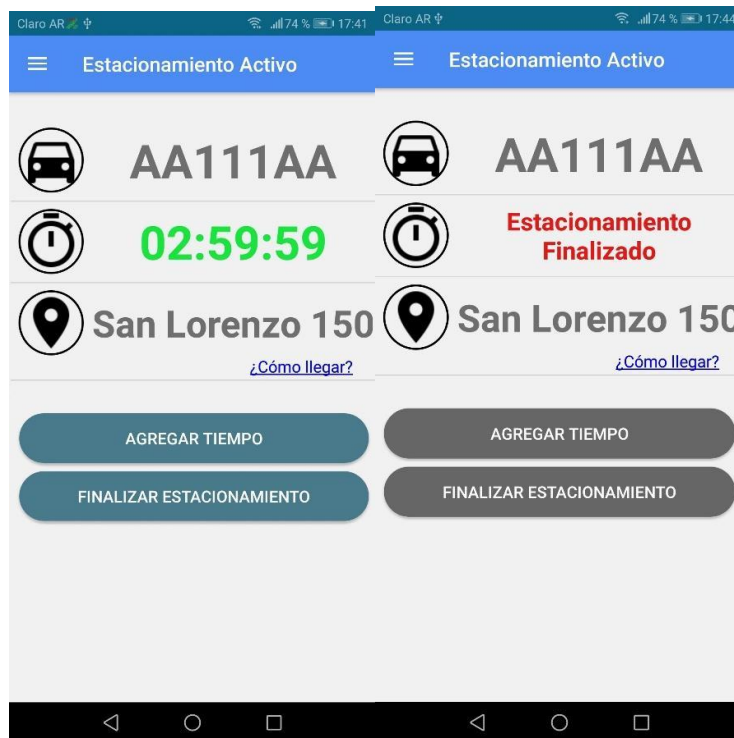
Mis estacionamientos

A través de la barra de navegación lateral, se puede acceder a “Mis estacionamientos”



Cada “Card” representa un estacionamiento, vigente o finalizado, con su correspondiente información asociada. Al seleccionar un registro, el sistema nos lleva a la pantalla de visualización de estacionamiento.

Visualización de Estacionamiento



Estacionamiento vigente

Estacionamiento Finalizado

En caso de que el estacionamiento se encuentre en vigencia, el tiempo disponible se actualiza de manera regresiva en la pantalla, para informar el tiempo exacto que queda hasta que el mismo se venza. El color del mismo cambiará según el tiempo disponible restante, de la siguiente manera:

Verde: El estacionamiento cuenta con, al menos, 30:01 Minutos de Vigencia.



Amarillo: La vigencia del estacionamiento se encuentra entre los 15:01 minutos, y los 30:00 minutos.

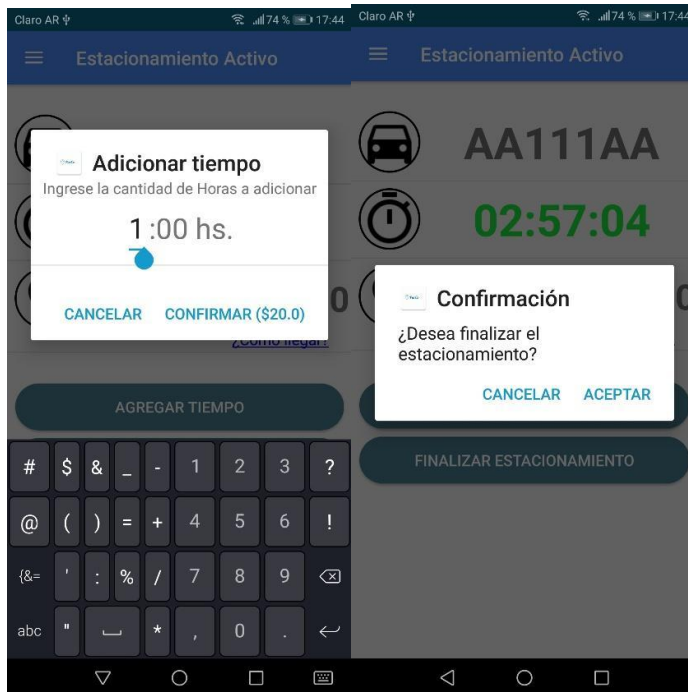


Rojo: El estacionamiento cuenta con menos de 15:00 minutos de vigencia restante. En caso de Agotarse el tiempo restante, se muestra el mensaje "Sin tiempo disponible"



Acciones sobre Estacionamiento Vigente

Parco permite adicionar tiempo a un estacionamiento vigente, o cancelar el mismo, a través de los botones “Agregar Tiempo” y “Cancelar Estacionamiento”.

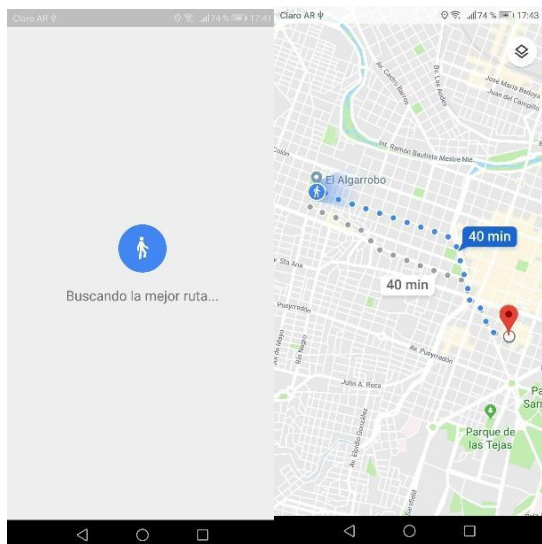


Agregar Tiempo

Cancelar Estacionamiento

¿Cómo llegar?

Al presionar sobre la leyenda [¿Cómo llegar?](#), ParCo utilizará Google Maps para ofrecernos el mejor camino para llegar hasta nuestro estacionamiento:



Búsqueda de ruta

Navegación hasta estacionamiento

Mapa de Calles

Parco brinda una funcionalidad que permite al usuario la obtención de información referida a la permisibilidad de estacionar en los diferentes tramos de calles en la ciudad, siendo:

Rojos, aquellos tramos en donde el estacionamiento se encuentra **prohibido**

Verde, aquellos tramos en donde rige el estacionamiento **tarifado**.

Aquellos tramos en donde no se indique color, el estacionamiento es **gratuito**.



ParCo Policía de Tránsito

Instalación

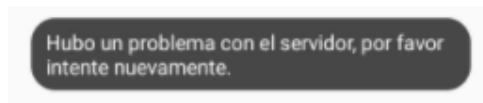
Requisitos de aplicación

- Tablet o Smartphone Android 4.4 KitKat o superior /
- Contar con un plan de datos móviles.

Generalidades

Errores

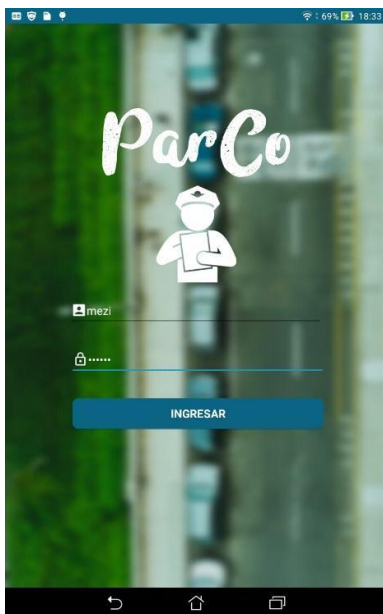
La aplicación informará errores mediante un mensaje comunicando la raíz del problema, en la parte inferior de la pantalla durante 4 segundos.



Módulos Funcionales

Login

Punto de acceso al sistema. Se deberán ingresar de manera obligatoria "Nombre de Usuario" y "Contraseña", debiendo los mismos ser válidos. Al ingresar de manera exitosa, se mostrará la pantalla de "Consulta de Estacionamiento"



Consulta de Estacionamiento

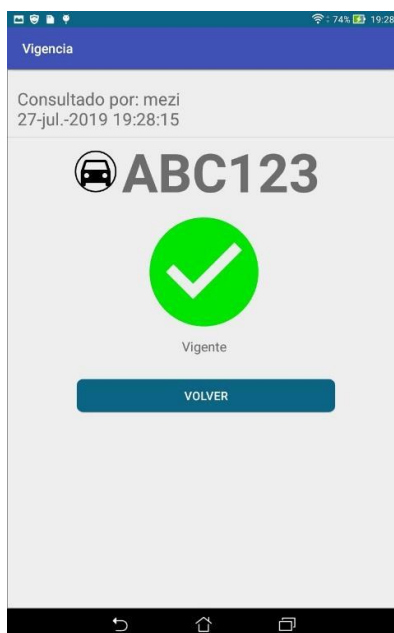
Pantalla principal del sistema. Permite consultar un estacionamiento a través de una patente, debiendo ingresarse la misma de manera obligatoria para realizar la consulta.



Tras presionar el botón "Consultar" se pueden producir dos resultados:

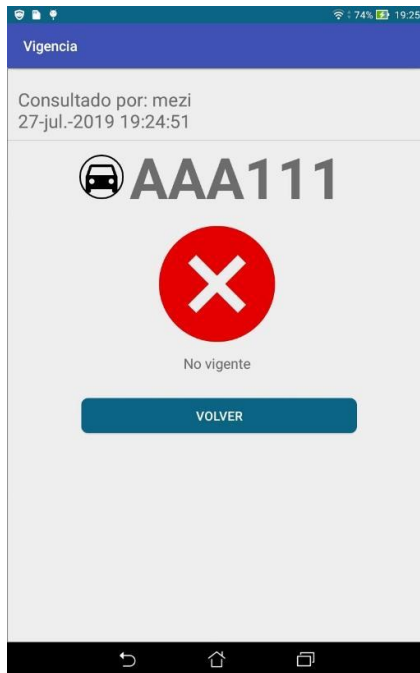
Consulta de estacionamiento – Estacionamiento Vigente

El Sistema mostrará una ventana emergente con un mensaje que informa que el estacionamiento se encuentra vigente:



Consulta de estacionamiento – Estacionamiento no vigente

El sistema mostrará una ventana emergente con un mensaje informando que el estacionamiento NO se encuentra vigente



ParCo Administrador

Instalación

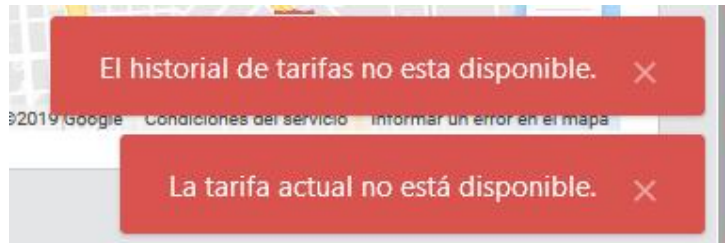
Requisitos de aplicación:

- Pc o dispositivo con internet que cuente con un navegador web.

Generalidades

Errores

La aplicación indicará los errores a través de mensajes de error en su pantalla principal, indicando cual es el componente que presenta el inconveniente



Módulos Funcionales

Configuración


La aplicación permite listar las secciones de calles que se encuentren tarifadas y prohibidas, permitiendo filtrar, modificar y borrar las mismas.

Secciones						
						<input type="text" value="Buscar por nombre"/>
Calle	Desde	Hasta	Tipo	Acciones		
Angelo de Peredo	1	200	Tarifado	Modificar	Borrar	
Avenida Pueyrredón	1	199	Prohibido	Modificar	Borrar	
Avenida Vélez Sarsfield	410	1200	Tarifado	Modificar	Borrar	
Avenida Vélez Sarsfield	1300	1520	Tarifado	Modificar	Borrar	
Balcarce	402	600	Tarifado	Modificar	Borrar	

Nuevo

A través del botón “Nuevo” permite definir una sección nueva.

Nueva sección de calle



Calle:

Desde:

Hasta:

Tipo: Prohibido **Tarifado**

El sistema automáticamente valida que las ubicaciones pertenezcan a la ciudad de Córdoba, y ofrece sugerencias relacionadas a los caracteres ingresados

Calle:

Desde:

Hasta:

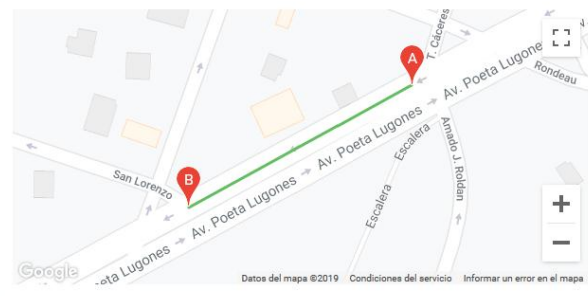
Tipo:

- Avenida Poeta Lugones Córdoba, Argentina
- Pablo Pizzurno Córdoba, Argentina
- Paso de los Andes Córdoba, Argentina
- Padre Luis Monti Córdoba, Argentina
- Paraná Córdoba, Argentina

powered by Google

El sistema valida que las alturas de la calle seleccionada sean válidas y existan y que la altura “desde” no sea superior a la “hasta”, y ofrece un componente visual para confirmar los datos ingresados, y un selector para elegir el tipo de tramo (Tarifado/prohibido) y la posibilidad de guardar la sección.

Nueva sección de calle



Calle:

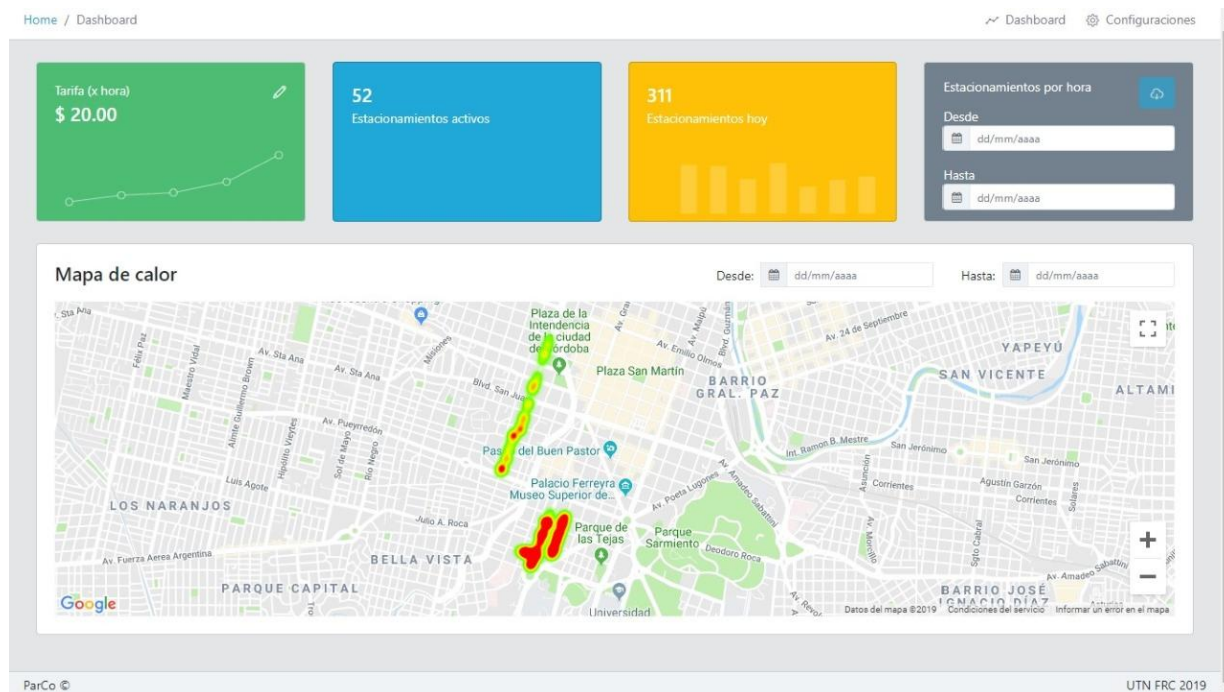
Desde:

Hasta:

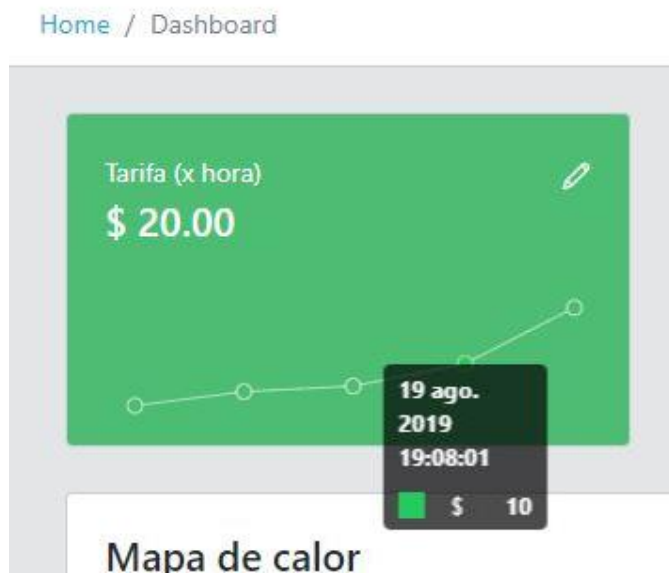
Tipo: Prohibido **Tarifado**

Dashboard

La pantalla de “Dashboard” del sistema, ofrece indicadores e información en tiempo real.



Indicador “Tarifa”



Ofrece información histórica sobre la fluctuación de la tarifa por hora de estacionamiento.

Permite también la edición de la tarifa actual a través del ícono 

Indicador “Estacionamientos activos”

Informa la cantidad de vehículos que cuentan con un estacionamiento vigente.

Indicador “Estacionamientos hoy”

Contabiliza la cantidad de estacionamientos que fueron registrados durante el día. El gráfico de barras en la sección inferior de la tarjeta permite obtener los contadores de los últimos 7 días.

Reporte “Estacionamientos por franja horaria”

Estacionamientos por hora

Desde

Hasta

Un formulario de reporte con un título 'Estacionamientos por hora' y un ícono de exportación. Incluye dos campos de fecha con íconos de calendario, uno etiquetado como 'Desde' y otro como 'Hasta'.

Permite filtrar las fechas que se quieren consultar, y exportar el reporte en Excel. El reporte brinda información acerca de los estacionamientos registrados y acumulados por franja horaria.

El reporte se compone por:

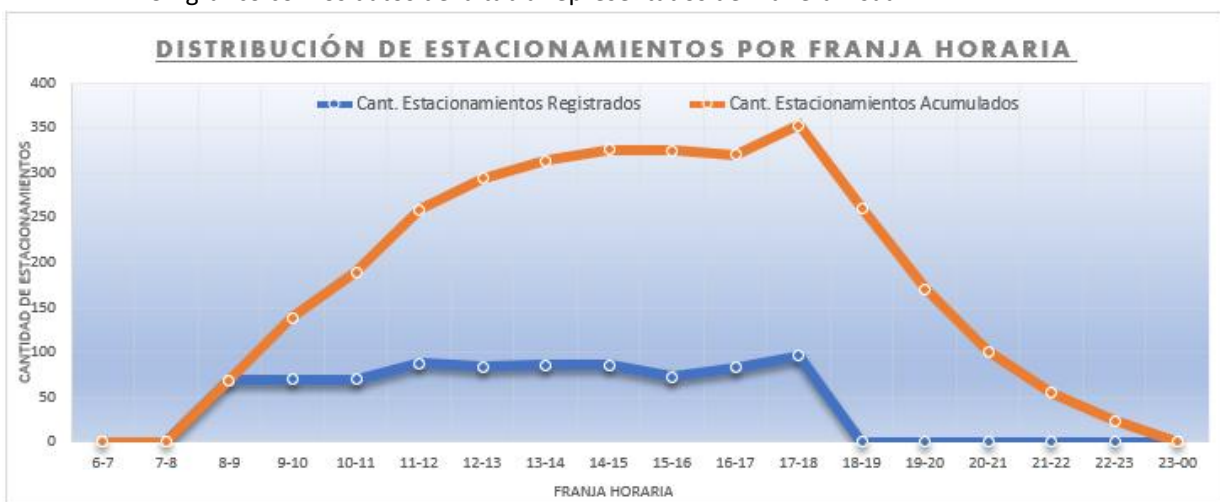
- una cabecera, con información de la fecha filtrada:

	Reporte : Distribución de Estacionamientos por franja Horaria
	Periodo Filtrado: 05/08/2019 - 14/08/2019
	Generado Por: Administrador Fecha: 19/08/2019 19:27:17

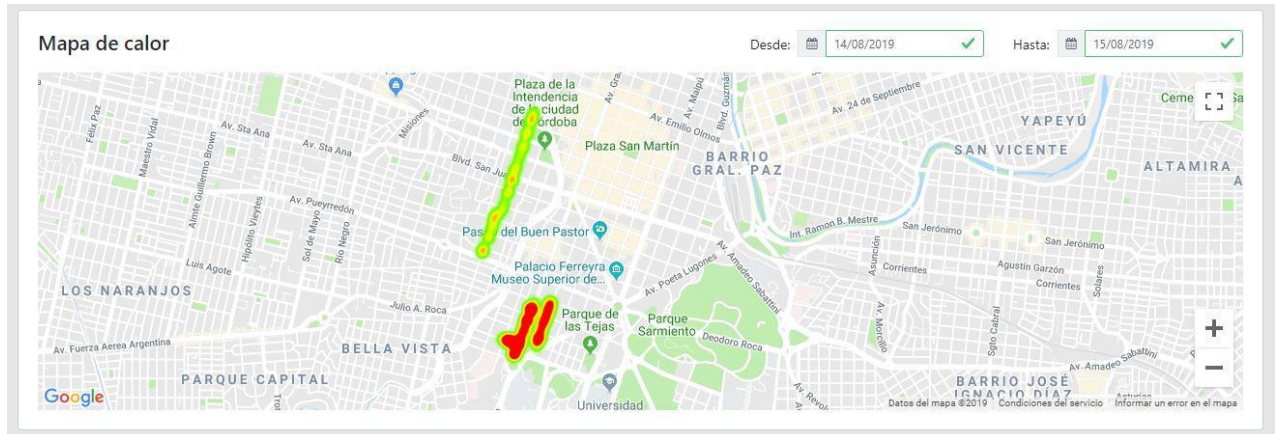
- Una tabla con los datos de estacionamientos por intervalo horario, donde “Estacionamientos registrados”, representa la cantidad de estacionamientos que se registraron en ese horario, y “Estacionamientos acumulados” representa la cantidad de estacionamientos que se encuentran en vigencia en ese intervalo horario.

Intervalos Horario	Cant. Estacionamientos Registrados	Cant. Estacionamientos Acumulados
6-7	0	0
7-8	0	0
8-9	69	69
9-10	70	139
10-11	70	189
11-12	88	259
12-13	84	294
13-14	86	314
14-15	86	326
15-16	73	325
16-17	83	320
17-18	96	353
18-19	0	261
19-20	0	171
20-21	0	101
21-22	0	55
22-23	0	24
23-00	0	0

- Un gráfico con los datos de la tabla representados de manera visual.



Mapa de calor



El mapa de calor brinda información relativa a la concentración y distribución de los estacionamientos en las distintas secciones de calles habilitadas para estacionar, permitiendo filtrar fechas para analizar el comportamiento histórico de la distribución.

Administrar usuarios Kiosco

Motivación

El alcance del producto definido no cuenta con la posibilidad de administrar los usuarios Kiosco desde una interfaz usuario dentro del módulo *Administrador*. Es por esto, que se proveen una serie de procedimientos que personal, con un básico conocimiento de SQL, puede ejecutar para tal fin.

Objetivo

Las licencias para los usuarios Kiosco están a cargo del municipio. La creación de este usuario no puede ser libre cómo lo es para los usuarios Conductor.

El objetivo es permitir al municipio administrar los usuarios Kiosco, incluyendo alta, baja y modificación de datos.

Generalidades

Estructura de la tabla USERS

COLUMNAS	TIPO DE DATO	RESTRICCIONES	REFERENCIA
ID	INTEGER	PK	Identificador único auto-generado
USERNAME	VARCHAR(12)	NO NULO UNICO	Nombre de usuario
PASSWORD	VARCHAR(12)	NO NULO	Constraseña
KIND	VARCHAR(10)	NO NULO	Tipo (ADMIN/DRIVER/KIOSK/POLICE)
END_DATE	TIMESTAMP		Fecha de baja

Conexión con la instancia de Base de Datos

Para poder realizar cualquier procedimiento que agregue o modifique los usuarios de aplicación es necesario conectarse con la instancia de motor de base de datos productiva.

Para esto se debe utilizar un cliente de base de datos compatible con PostgreSQL y obtener los siguientes datos:

ParCo | Sistema

- URL: el proveedor de infraestructura generará una IP o dirección diferente para cada instancia; la denominamos <dir>. Utilizar la siguiente url generada **jdbc:postgresql://<dir>:5432/parco**
- Usuario: support-user
- Contraseña: tiene un valor generado por el equipo de ParCo, se recomienda cambiar. Utilizar la siguiente contraseña para el primer inicio de sesión **jd7h1wkhd09!#**

Manejo de transacciones

Antes de iniciar cada procedimiento se debe iniciar una transacción. Esto nos va a permitir, al finalizar, decidir si queremos impactar el cambio o revertirlo.

Comenzar la transacción ejecutando la siguiente instrucción: **START TRANSACTION.**

Una vez concluida todas las operaciones, el operador tiene la posibilidad de confirmar o revertir los cambios realizados. Tener en cuenta que hasta que no se ejecute alguna de estas instrucciones, los cambios no son visibles para el sistema ni para otros usuarios.

Confirmar la transacción ejecutando la siguiente instrucción: **COMMIT.**

O revertir la transacción ejecutando la siguiente instrucción: **ROLLBACK.**

Procedimiento de Alta

Elegir un nombre **único** de usuario (<user>) y una contraseña segura (<pwd>) con un máximo de 12 caracteres y ejecutar el siguiente comando:

```
INSERT INTO USERS (USERNAME, PASSWORD, KIND, END_DATE)
VALUES ('<user>', '<pwd>', 'KIOSK', null);
```

Procedimiento de Modificación

Los únicos campos que se pueden modificar son:

- Nombre de usuario. Debe ser único: <new-user>
- Contraseña: <new-pwd>

Identificar el nombre de usuario (<user>) único (respetando mayúsculas y minúsculas) y ejecutar el siguiente comando:

```
UPDATE USERS
SET USERNAME = '<new-user>', PASSWORD = '<new-pwd>'
WHERE USERNAME = '<user>;
```

Procedimiento de Baja

Identificar el nombre de usuario (<user>) único (respetando mayúsculas y minúsculas) y ejecutar el siguiente comando:

```
UPDATE USERS
SET END_DATE = current_timestamp()
WHERE USERNAME = '<user>;
```

ParCo Kiosco

Instalación

Requisitos de aplicación

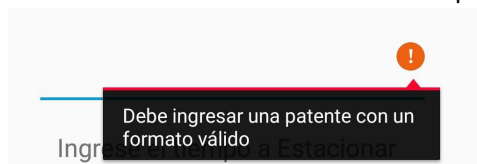
- Smartphone Android 4.4 KitKat o superior.
- Conectividad a Internet.

Generalidades

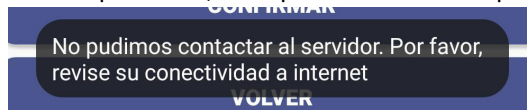
Errores

La aplicación informará errores de dos maneras distintas:

- 3- Error en información ingresada por el usuario:
 - o El sistema informará el error mediante un mensaje en el/los campo/s que se encuentren con información requerida faltante o errónea.

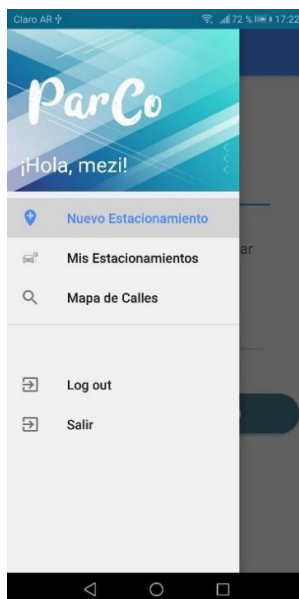


- 4- Error de conectividad con el servidor (Falta de internet, servidor no disponible, etc)
 - o El sistema informará el error mediante un mensaje informando la raíz del problema, en la parte inferior de la pantalla durante 4 segundos.



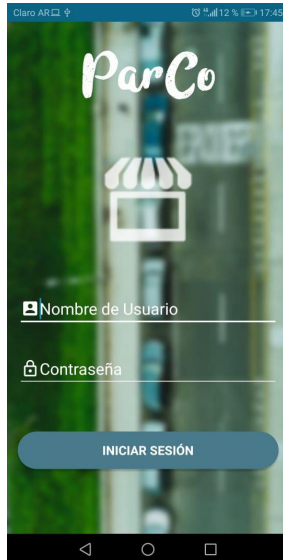
Navegabilidad

La navegabilidad del sistema estará guiada por un menú lateral desde el cual se podrán acceder a las distintas funcionalidades del sistema



Módulos Funcionales

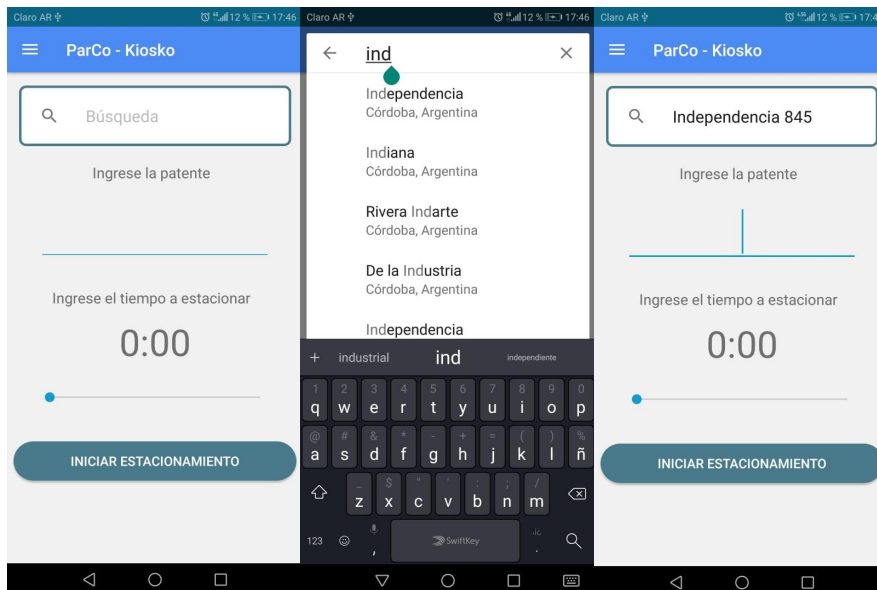
Login



Punto de acceso al sistema. Se deberán ingresar de manera obligatoria “Nombre de Usuario” y “Contraseña”, debiendo los mismos ser válidos. Al ingresar de manera exitosa, se mostrará la pantalla de “Nuevo Estacionamiento”

Registro De Estacionamiento

El sistema ofrecerá una interfaz que permite realizar una búsqueda de ubicación, validando que la misma se encuentre dentro de los límites de la ciudad de Córdoba, y sugiriendo las calles que coincidan con el patrón ingresado en el cuadro de búsqueda



Se deberá ingresar una patente con los formatos válidos para Argentina, siendo los mismos del tipo “AAA111” o “AA111AA”.

El tiempo a estacionar debe ser ingresado deslizando la barra en el espacio inferior de la pantalla, pudiendo estacionar de 1 a 10 horas.

ParCo | Sistema

La aplicación mostrará un “loader” mientras intenta establecer la ubicación (Se recomienda fuertemente contar con la conectividad GPS activada, para facilitar la detección de la misma) hasta lograr detectarla de manera correcta, procediendo a mostrar una pantalla de confirmación, dependiendo la misma de la ubicación en la que nos encontremos, pudiendo ser:

Ubicación Tarifada:

Ubicación Gratuita

Ubicación Prohibida



En todos los casos la pantalla nos brinda la siguiente información:

- Ubicación seleccionada.
- Dominio del vehículo a estacionar
- Confirmación del tiempo Requerido
- Precio total del estacionamiento.
- El botón “Confirmar” solo se habilitará en tramos habilitados.
- El botón “VOLVER” Permitirá volver a la pantalla anterior.

ParCo | Sistema

Módulo de Pago.

La API de pago utilizada por ParCo Kiosk es MercadoPago. El diseño de su interfaz es intuitivo y guiado paso por paso:

Claro AR 74% 17:37 Claro AR 74% 17:39 Claro AR 74% 17:39

← ¿Con qué tarjeta? Ver promociones Ver promociones

Nueva tarjeta de crédito Nueva tarjeta de débito

VISA 4509 9535 6623 3704 NOMBRE APELLIDO MM/AA

VISA 4509 9535 6623 3704 MARIANO MEZI MM/AA

Número de tarjeta No Nombre y apellido

4509 9535 6623 3704 MARIANO MEZI

Total a pagar \$ 60 Anterior Continuar Anterior Continuar

*Elección de Medio de Pago**Ingreso número de Tarjeta**Ingreso Nombre de Titular*

Claro AR 74% 17:39 Claro AR 74% 17:39 Claro AR 74% 17:39

← Ver promociones Ver promociones Ver promociones

VISA 4509 9535 6623 3704 MARIANO MEZI 12/25

DOCUMENTO DEL TITULAR DE LA TARJETA 36.334.279

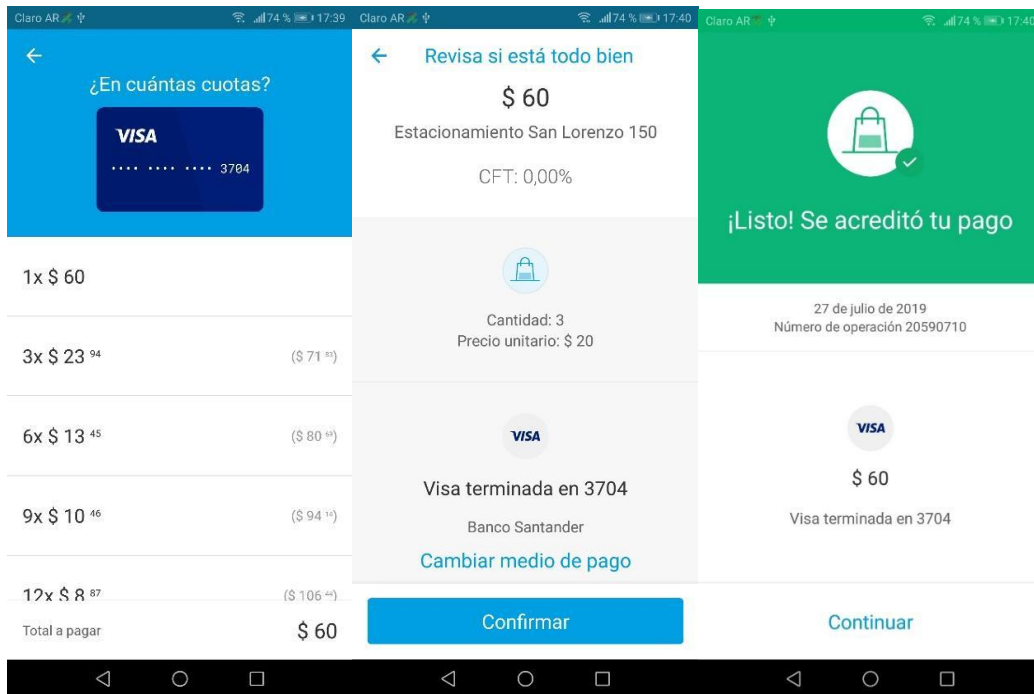
Fecha de expiración Cód Código de seguridad Tip Tipo Número

12/25 111 DN DNI 36334279

Anterior Continuar Anterior Continuar Anterior Continuar

*Ingreso de fecha de vencimiento**Ingreso código de Seguridad**Ingreso DNI de titular*

ParCo | Sistema



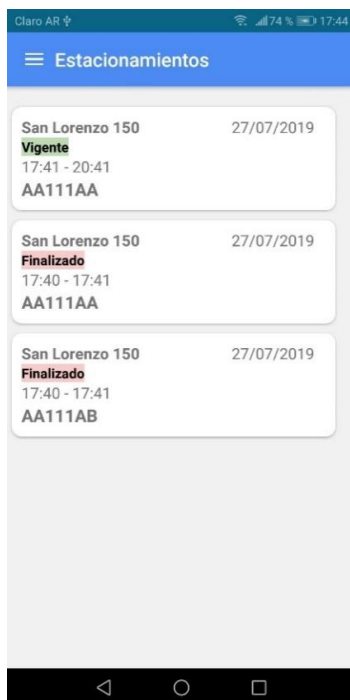
Elección cantidad de cuotas

Confirmación de Pago

Información de pago confirmado

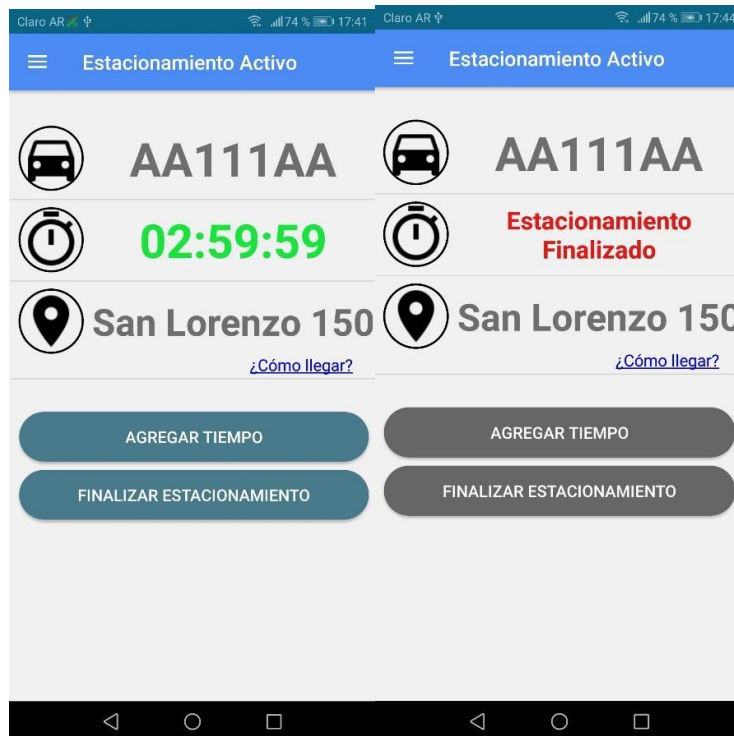
Estacionamientos

A través de la barra de navegación lateral, se puede acceder a “Estacionamientos hoy”



ParCo | Sistema

Cada “Card” representa un estacionamiento registrado en el día de hoy, vigente o finalizado, con su correspondiente información asociada. Al seleccionar un registro, el sistema nos lleva a la pantalla de visualización de estacionamiento.

Visualización de Estacionamiento*Estacionamiento vigente**Estacionamiento Finalizado*

En caso de que el estacionamiento se encuentre en vigencia, el tiempo disponible se actualiza de manera regresiva en la pantalla, para informar el tiempo exacto que queda hasta que el mismo se venza. El color de este cambiará según el tiempo disponible restante, de la siguiente manera:

Verde: El estacionamiento cuenta con, al menos, 30:01 Minutos de Vigencia.



Amarillo: La vigencia del estacionamiento se encuentra entre los 15:01 minutos, y los 30:00 minutos.

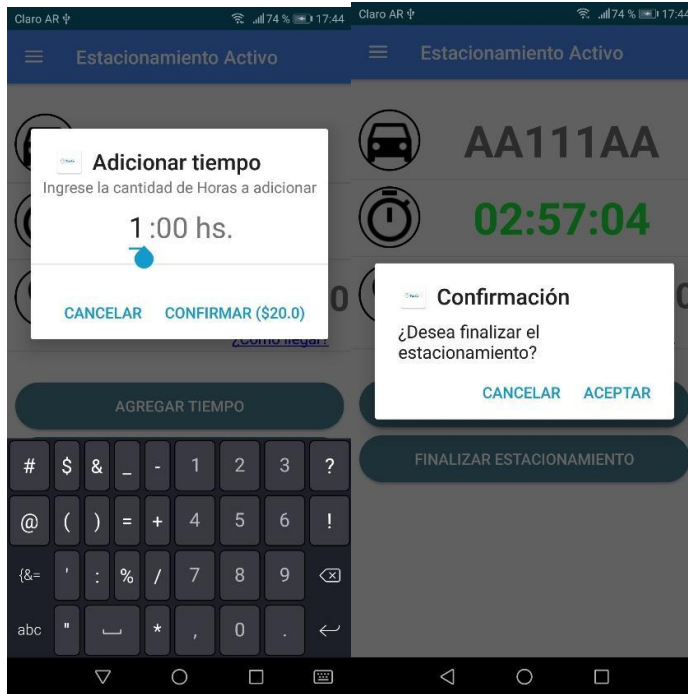


Rojo: El estacionamiento cuenta con menos de 15:00 minutos de vigencia restante. En caso de Agotarse el tiempo restante, se muestra el mensaje “Sin tiempo disponible”



Acciones sobre Estacionamiento Vigente:

Parco permite adicionar tiempo a un estacionamiento vigente, o cancelar el mismo, a través de los botones “Agregar Tiempo” y “Cancelar Estacionamiento”.

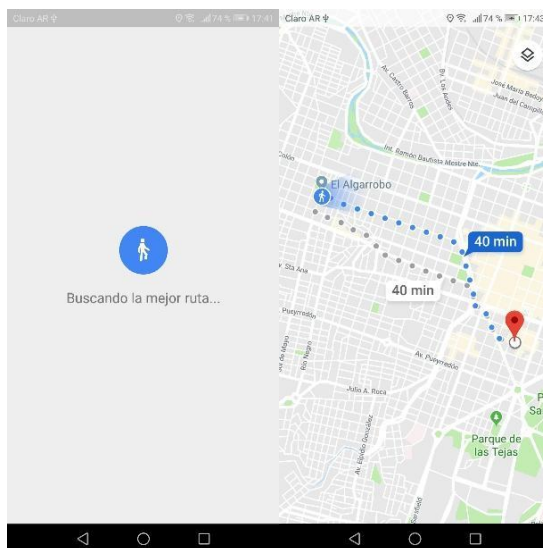


Agregar Tiempo

Cancelar Estacionamiento

¿Cómo llegar?

Al presionar sobre la leyenda [¿Cómo llegar?](#), ParCo utilizará Google Maps para ofrecernos el mejor camino para llegar hasta nuestro estacionamiento:



Búsqueda de ruta

Navegación hasta estacionamiento

ParCo | Sistema

Mapa de Calles

Parco kiosk brinda una funcionalidad que permite al usuario la obtención de información referida a la permisibilidad de estacionar en los diferentes tramos de calles en la ciudad, siendo:

Rojos, aquellos tramos en donde el estacionamiento se encuentra **prohibido**

Verde, aquellos tramos en donde rige el estacionamiento **tarifado**.

Aquellos tramos en donde no se indique color, el estacionamiento es **gratuito**.

