

# Proyecto Desarrollo

I. S. basada en servicios



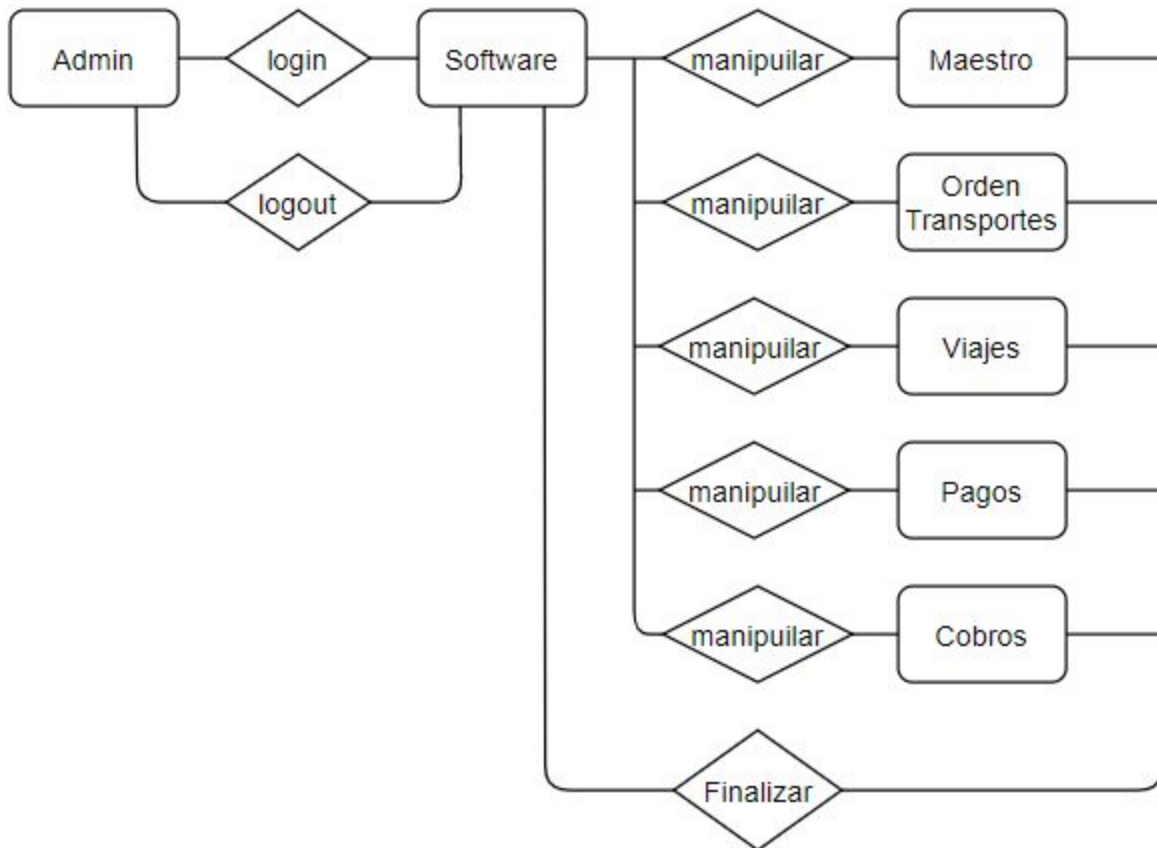
Gastón Bustamante  
Felipe Lagos  
Sebastián Sanhueza

# Contenidos

<b>UML</b>	<b>2</b>
Admin	2
Camionero	3
Mecánico	4
<b>Reutilización, componentes y servicios</b>	<b>5</b>
<b>Diagramas de Colaboración</b>	<b>6</b>
<b>Requerimientos</b>	<b>8</b>
Requerimientos Funcionales	8
Requerimientos Funcionales del Administrador	8
Requerimientos no Funcionales	9
<b>Arquitectura</b>	<b>10</b>

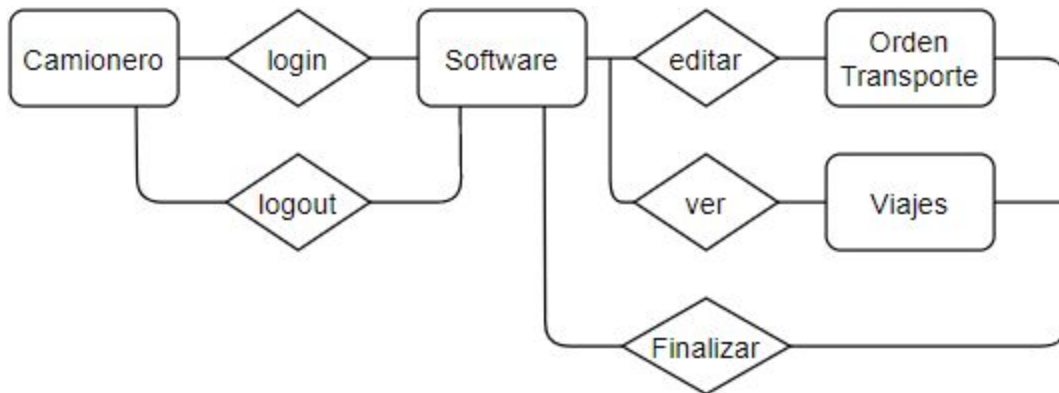
# UML

## Admin



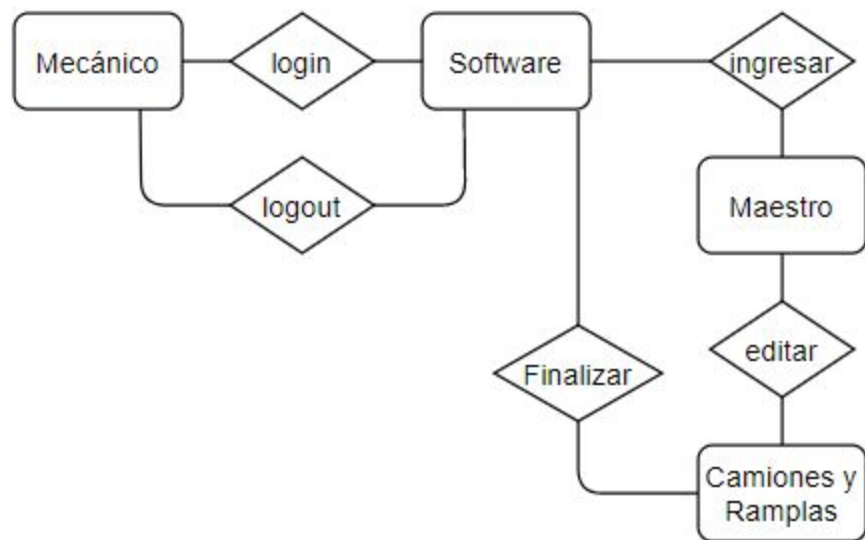
El perfil de Administrador requiere, en primera instancia, ingresar a su cuenta mediante el *login* del software (TTrack). Una vez dentro, la interfaz base le ofrece todas las opciones disponibles de *Ver*, *Editar*, *Añadir* y *Quitar* en cada uno de las secciones del software: *Maestro*, *Orden Transportes*, *Viajes*, *Pagos*, *Cobros*, por separado, además de la opción de *logout*, para cerrar su sesión y salir de la interfaz. Entonces, una vez realizada una de las acciones previamente descritas, vuelve a la interfaz inicial del software, donde se le ofrecen las mismas opciones anteriores.

## Camionero



El perfil de Camionero, al igual que el de administrador, requiere, en primera instancia, el login del usuario, enviándolo luego a la misma interfaz base que la de administrador, pero con sus permisos correspondientes, que son: *Editar*, asociado a *Orden Transporte*, *Ver*, asociado a *Viajes* y *logout*, para salir de la interfaz del software. Como se mencionó anteriormente, al llevar a cabo cualquier acción distinta de *logout*, una vez terminada la tarea, se redirige a la interfaz base, con las mismas opciones anteriores.

## Mecánico



El perfil de Mecánico también requiere de un *login* inicial, que luego lleva a la interfaz base, donde le muestra sus permisos que son: *logout*, para cerrar sesión y salir de la interfaz, *Ingresar a Maestro*, donde se le ofrece un nuevo permiso de *Editar*, asociado a *Camiones y Ramplas*. Una vez terminada esta línea de ejecución, se vuelve a la interfaz base, donde cuenta con las mismas opciones ya mencionadas.

# Reutilización, componentes y servicios

Tuvimos un enfoque en tecnologías que, a fecha de este proyecto, son consideradas actuales y, por lo tanto, con gran versatilidad y documentación para poder ser reutilizadas de diversas maneras, según el problema lo amerite.

Las tecnologías usadas son:

- ◆ Bootstrap es la encargada de proporcionarnos las herramientas necesarias para desarrollar una *Interfaz de Usuario*, la cual es fácilmente adaptable a los múltiples dispositivos a los que apunta nuestro software, aportando, además, una experiencia muy amigable para el usuario.
- ◆ MongoDB es un sistema de base de datos, la cual aprovecha su naturaleza *no relacional* para crear bases de datos de manera mucho más rápido, siendo capaz de manipular una alta gama de datos, y gestionar de manera eficiente aplicaciones de cualquier escala.
- ◆ Node.js posee múltiples cualidades que nos permiten crear aplicaciones en *Javascript*, *trabajando a la par* con el servidor. La propiedad de Node de poder ejecutarse fuera de un explorador web (su ambiente por defecto), es ideal para nuestro objetivo de que el software sea *multiplataforma*.
- ◆ Express.js es un framework de Node, el cual presta mecanismos para:
  - Escritura de manejadores de peticiones con diferentes verbos HTTP en diferentes caminos URL (rutas).
  - Integración con motores de renderización de "vistas" para generar respuestas mediante la introducción de datos en plantillas.
  - Añadir procesamiento de peticiones "middleware" adicional en cualquier punto dentro de la tubería de manejo de la petición.

# Diagramas de Colaboración

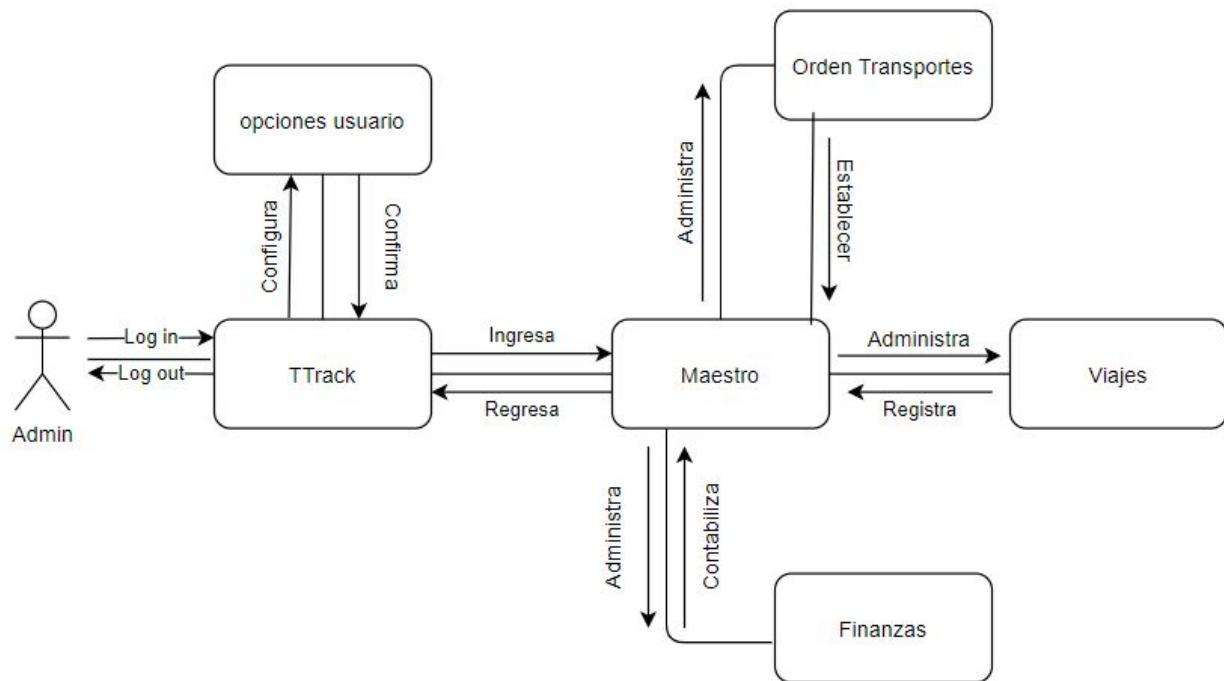


Diagrama Colaboración del Administrador

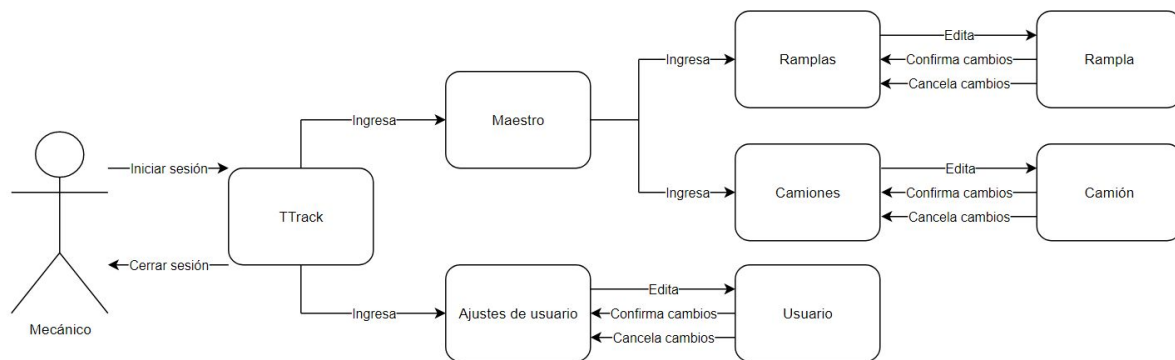


Diagrama de Colaboración del Mecánico

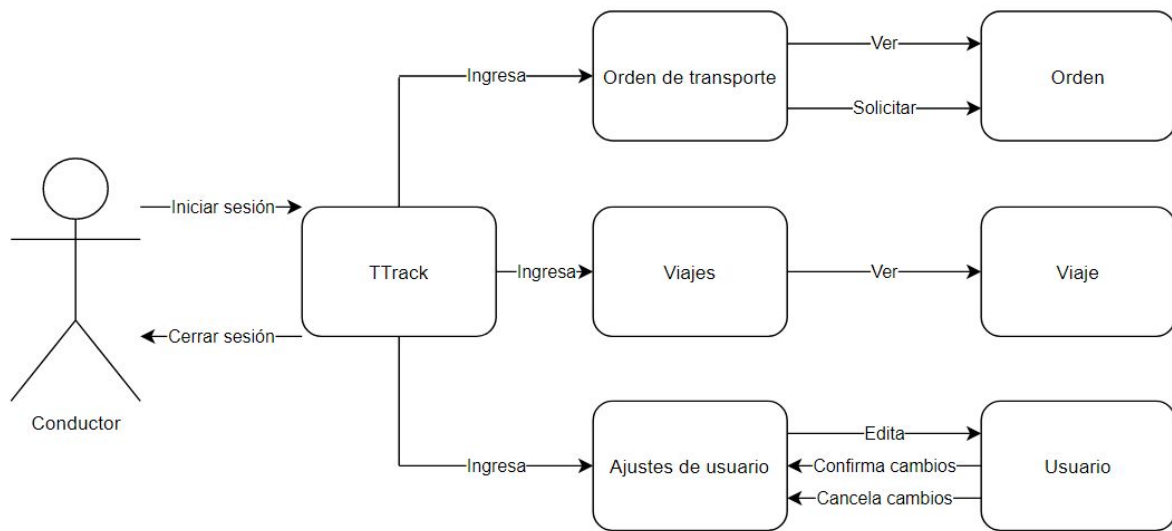


Diagrama de Colaboración del Camionero



# Requerimientos

## Requerimientos Funcionales

- Inicio de Sesión
  - ◆ El sistema debe tener una pantalla de inicio de sesión, con el cual el usuario deja constancia del acceso al sistema.
- Ajustes de Usuario
  - ◆ El sistema debe contar con una sección de ajustes, en donde el usuario pueda configurar su foto de perfil, datos personales, editar la información que tiene permitido modificar y cerrar su sesión.
- Acceso restringido a distintos modales
  - ◆ El sistema tiene que restringir parcialmente el acceso a distintas categorías del mismo, en función de las jerarquías solicitadas.
- Registro de viajes
  - ◆ El sistema otorgará una sección en donde cada trabajador pueda ver el historial de sus viajes.

## Requerimientos Funcionales del Administrador

- Control total sobre la base de datos
  - ◆ El Administrador tiene que tener acceso total a la base de datos, con el fin de poder controlar los datos de la misma.
- Control sobre la información de los otros usuarios
  - ◆ Se debe contar con una sección especial, única para el Administrador de la Empresa, en donde podrá monitorear la información de sus subordinados, clientes y de los procesos de la empresa.

## Requerimientos no Funcionales

- Rendimiento
  - ◆ Permitir que se use en múltiples dispositivos (PC, celulares, tablets).
  - ◆ Optimización de los recursos del dispositivo para mayor fluidez del sistema
- Seguridad
  - ◆ Confiabilidad para el control de las cuentas por pagar.
  - ◆ Confiabilidad para el control de las cuentas por cobrar.
  - ◆ Garantizar privacidad entre trabajadores de poder ver sus propios registros de viajes.
  - ◆ Asegurar que los subordinados tendrán restringido a las secciones que no correspondan a su puesto en la empresa.
- Usabilidad
  - ◆ Un manejo intuitivo para los usuario por parte del sistema.
- Mantenibilidad
  - ◆ Facilidad de mantención por parte de los Trabajadores.

# Arquitectura

Sea el siguiente diagrama una representación gráfica muy sencilla de nuestra arquitectura de TTrack, basado en todo lo anteriormente expuesto:

