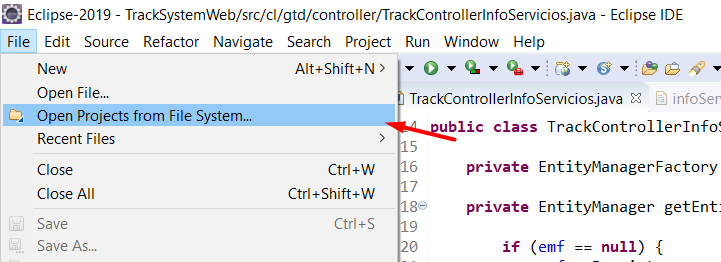
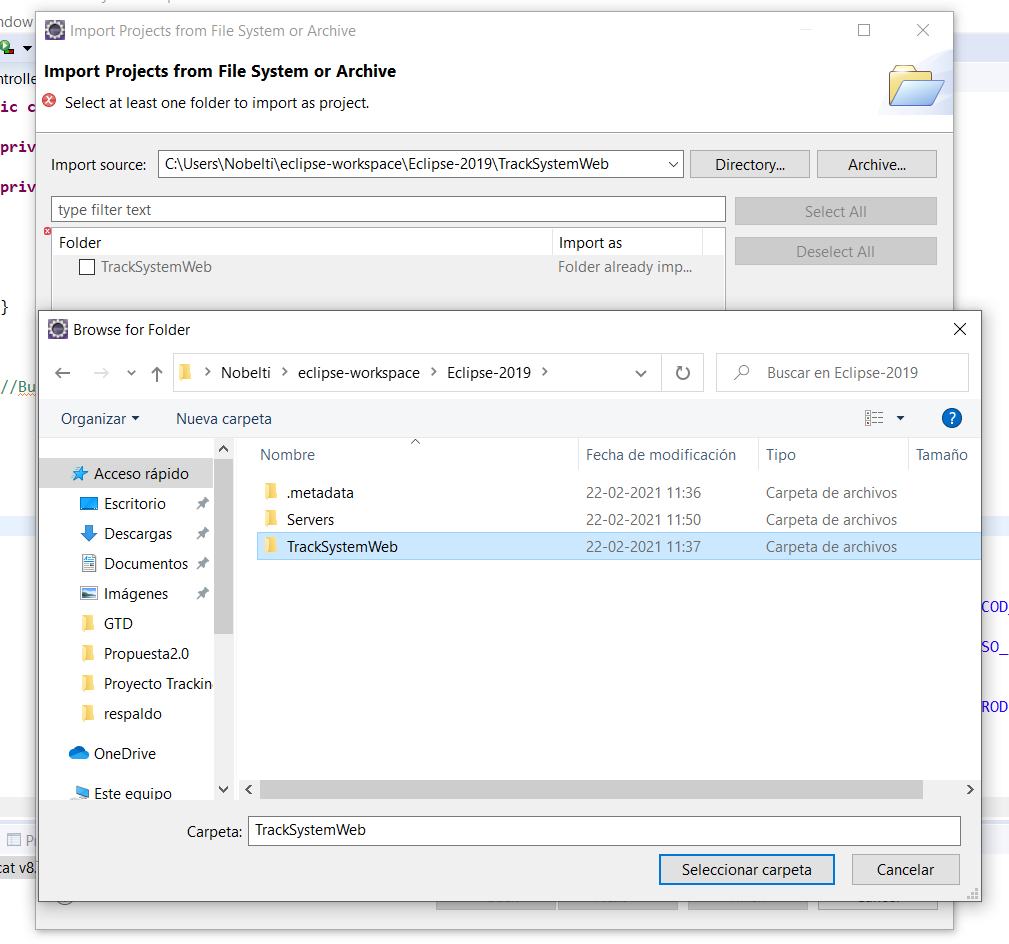
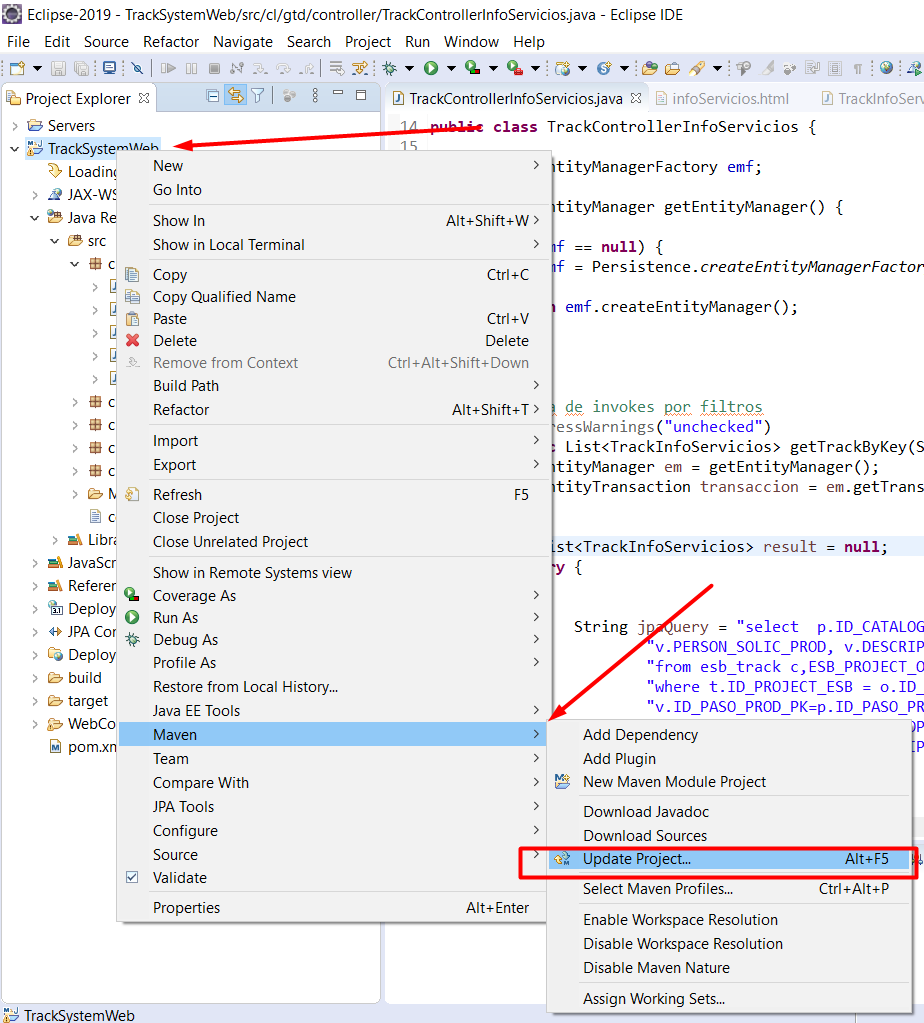
# Importar proyecto de manera local

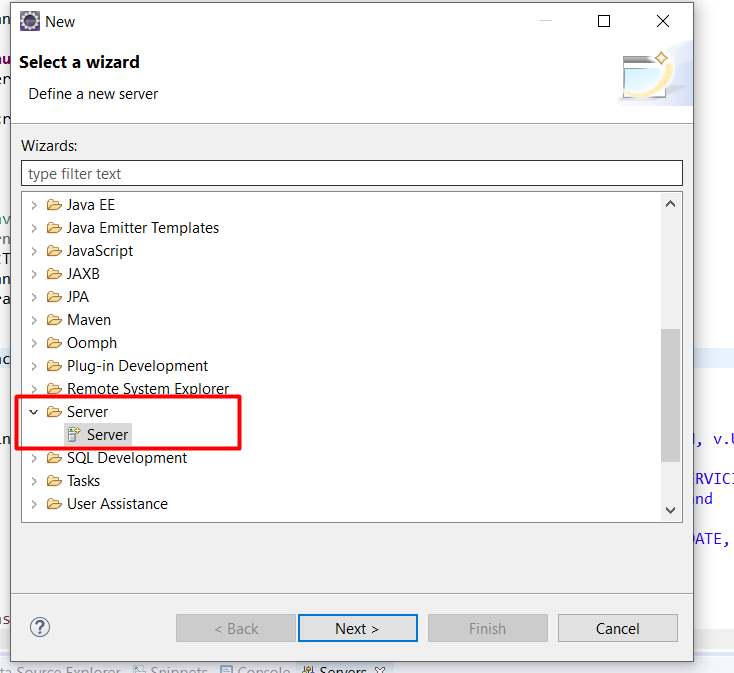




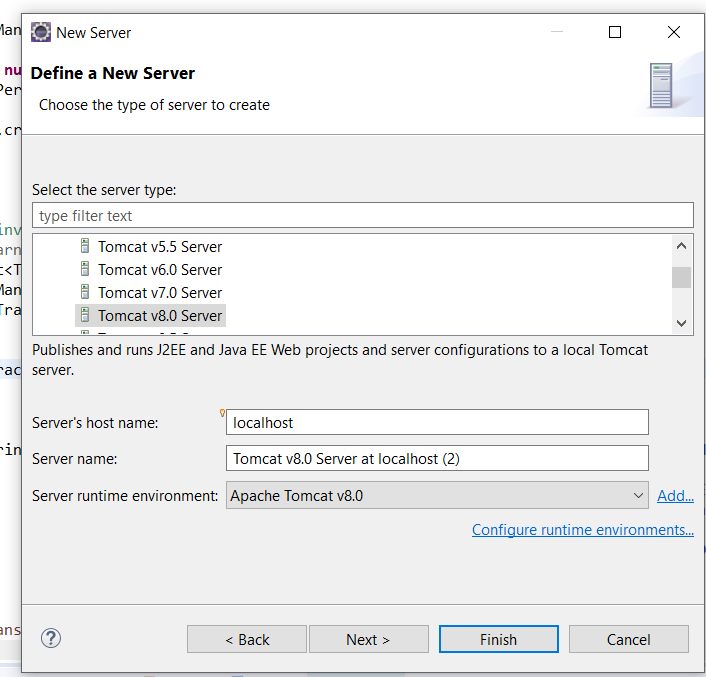
Una vez importado el proyecto se deben actualizar las librerías de maven:



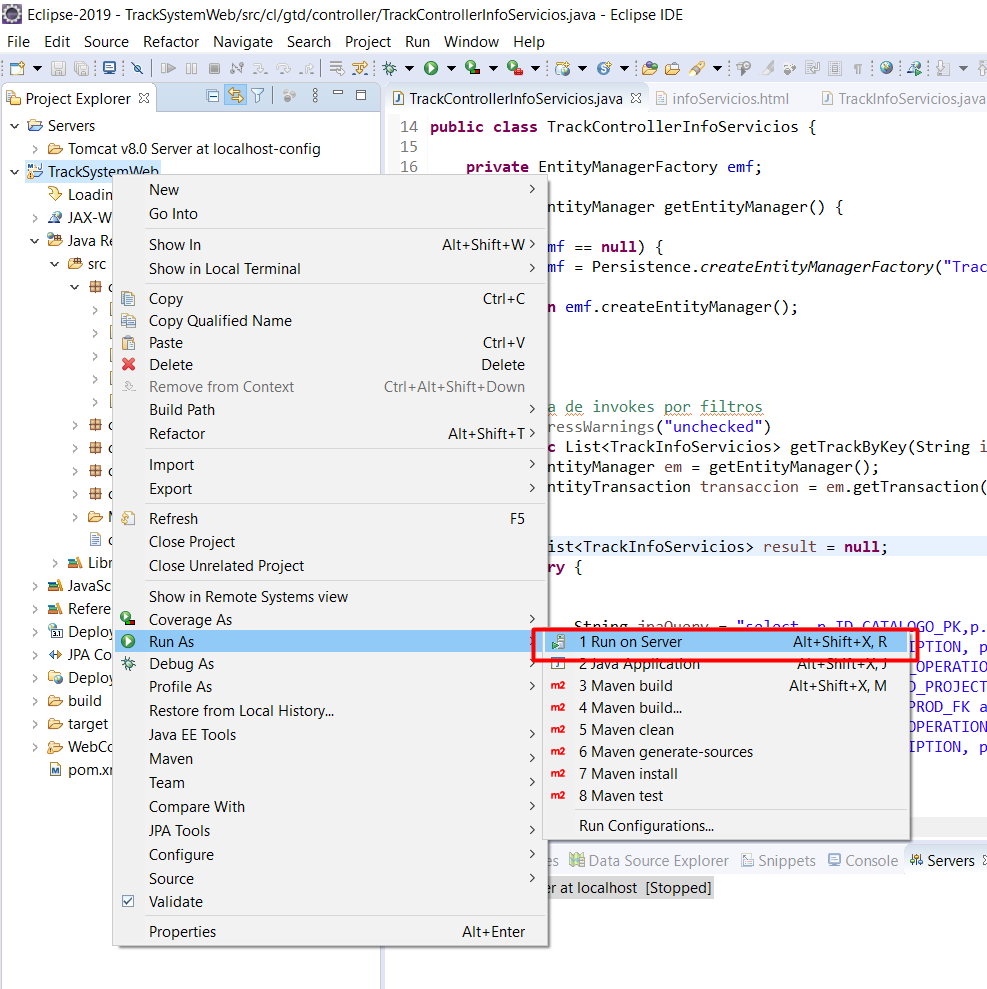
También crear un nuevo servidor para levantar el tracking de manera local:

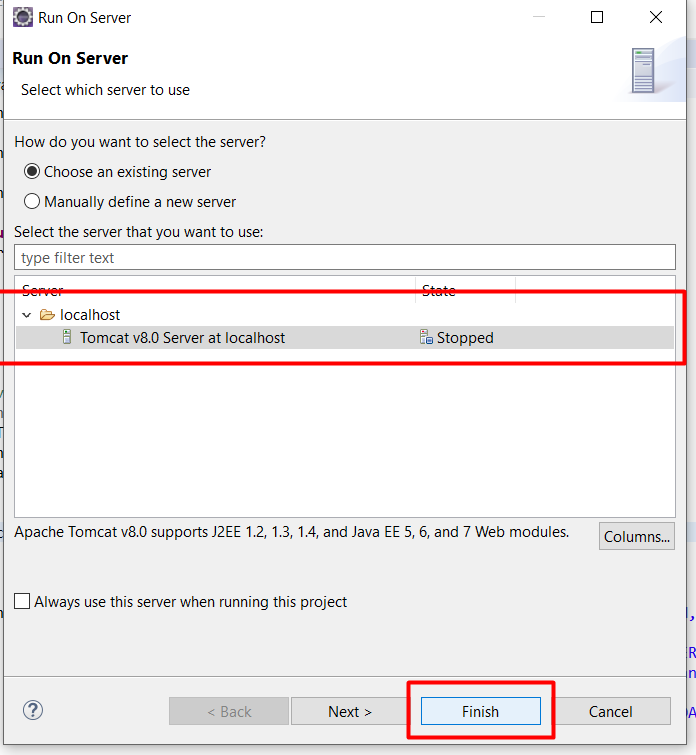


En este caso se utilizó tomcast v8.0



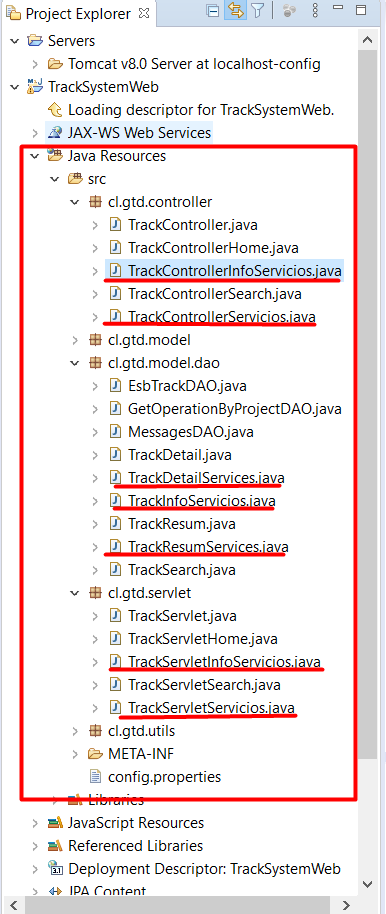
Una vez creado el servidor se levanta el tracking de la siguiente manera:





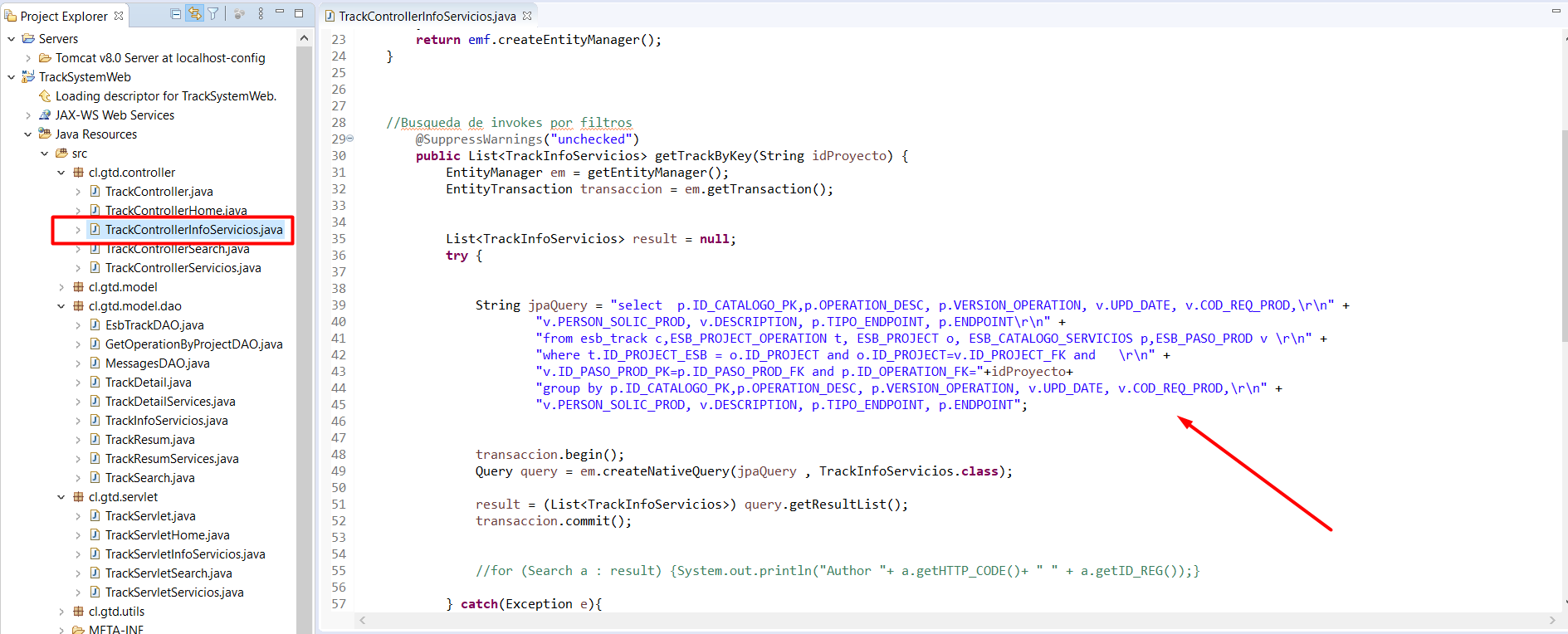
# Componentes modificados en el Backend

Los cambios en el backend son agregados que se adaptan a lo solicitado, por ejemplo en esta imagen se muestra que se crearon clases nuevas para las entidades/datos solicitados por el cliente, luego se mostrará en el frontend.

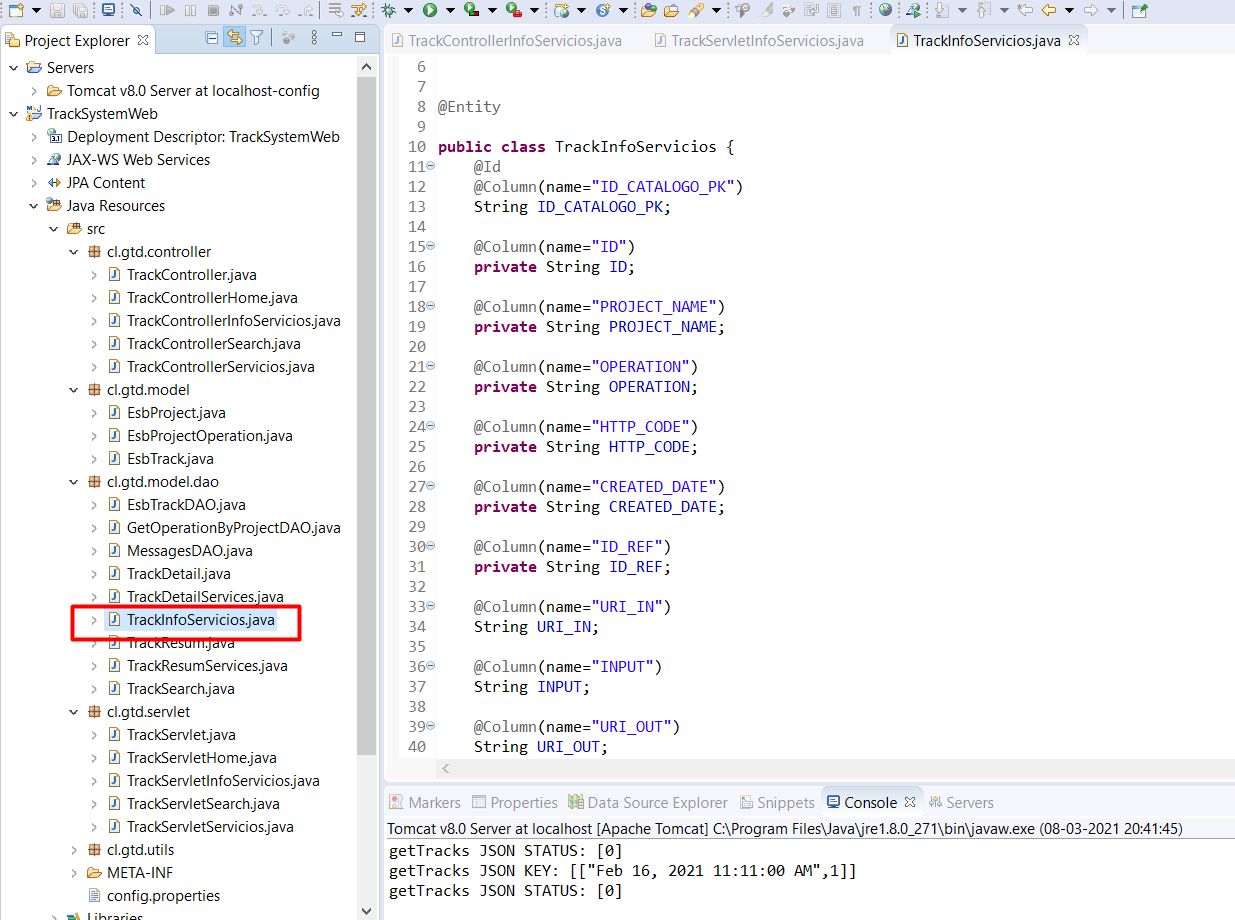


Todo lo subrayado en rojo es lo agregado nuevo al backend. Se utilizó la misma lógica del proyecto base Tracking, pero se crearon nuevas clases para dar referencia a los nuevos datos que el cliente solicitó

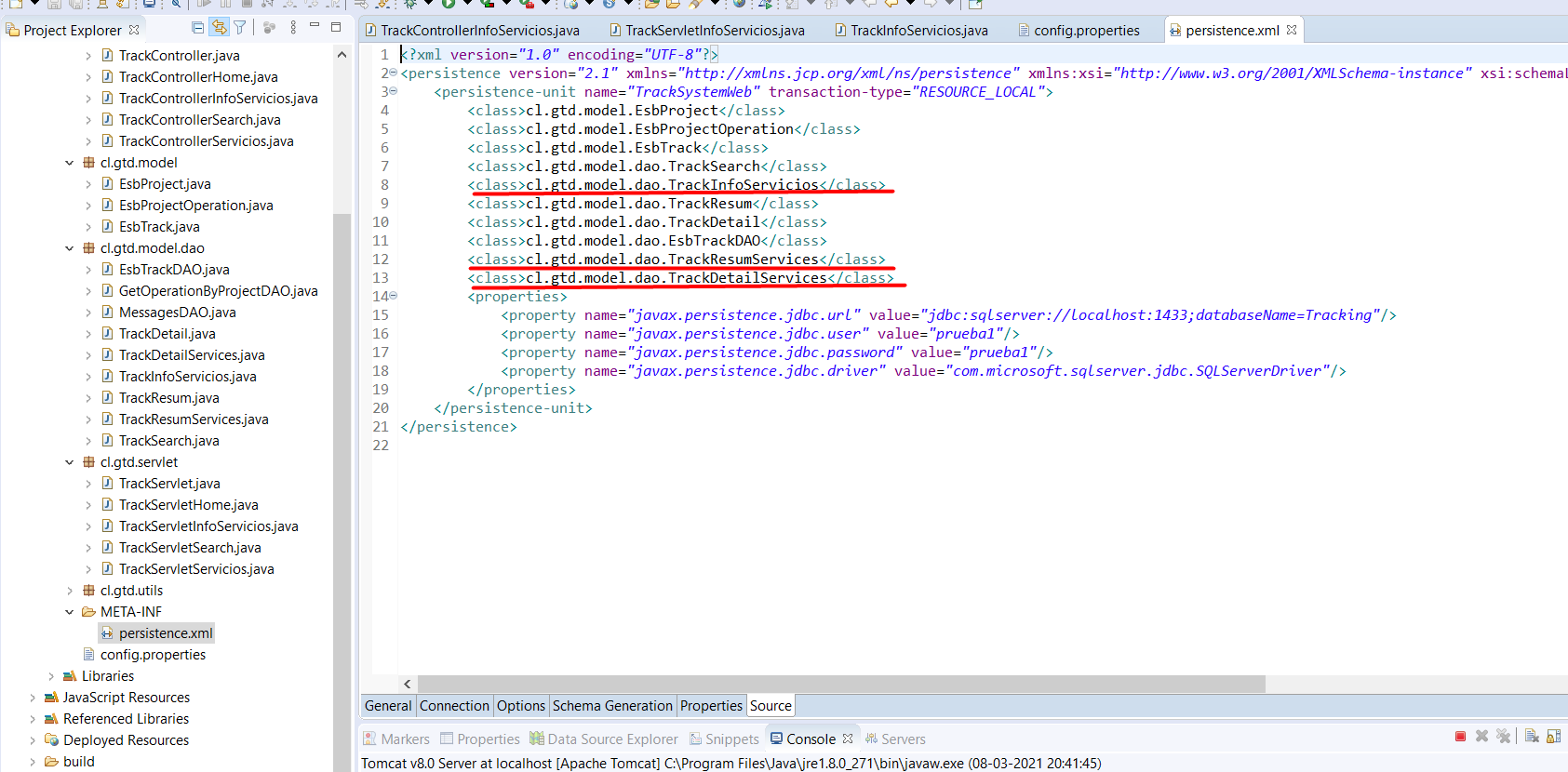
La lógica cambia principalmente en lo que deseamos mostrar y el nombre de los atributos, por ejemplo, aquí aparte de que la clase java tiene otro nombre, la query SQL consulta los datos de los detalles (Imagen 3 del flujo página web)



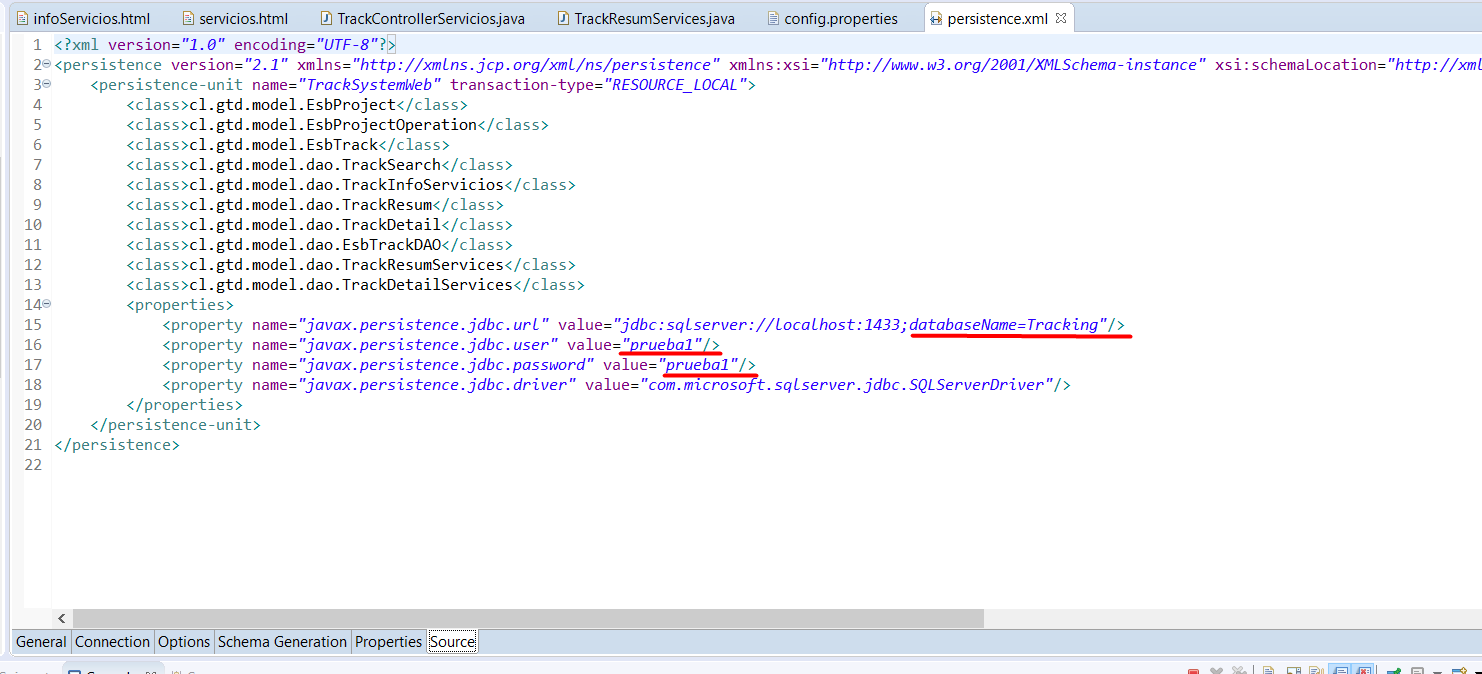
Para que la consulta se realice se debe consultar a través del package DAO, allí se encontraran las mismas entidades de la base de datos, el cual podremos llamarlas mas adelante.



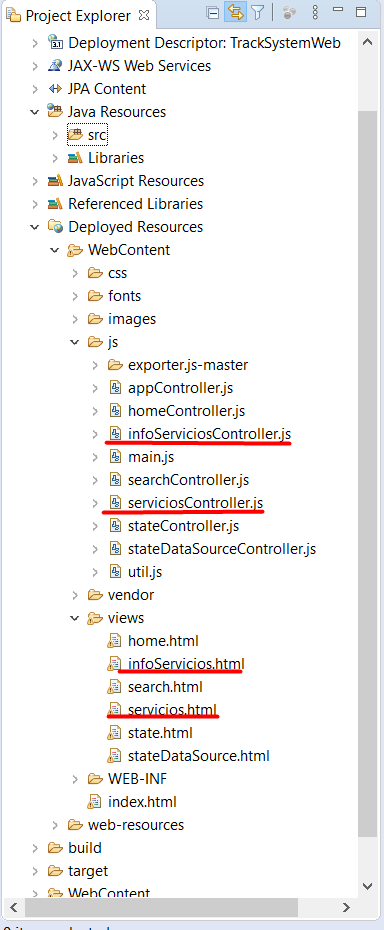
Finalmente, en el source del archivo persistence.xml se agregaron nuevas clases java para la referencia del proyecto modificado



También se debe modificar los datos de la conexión de la base de datos, en este caso se utilizó una de manera local.

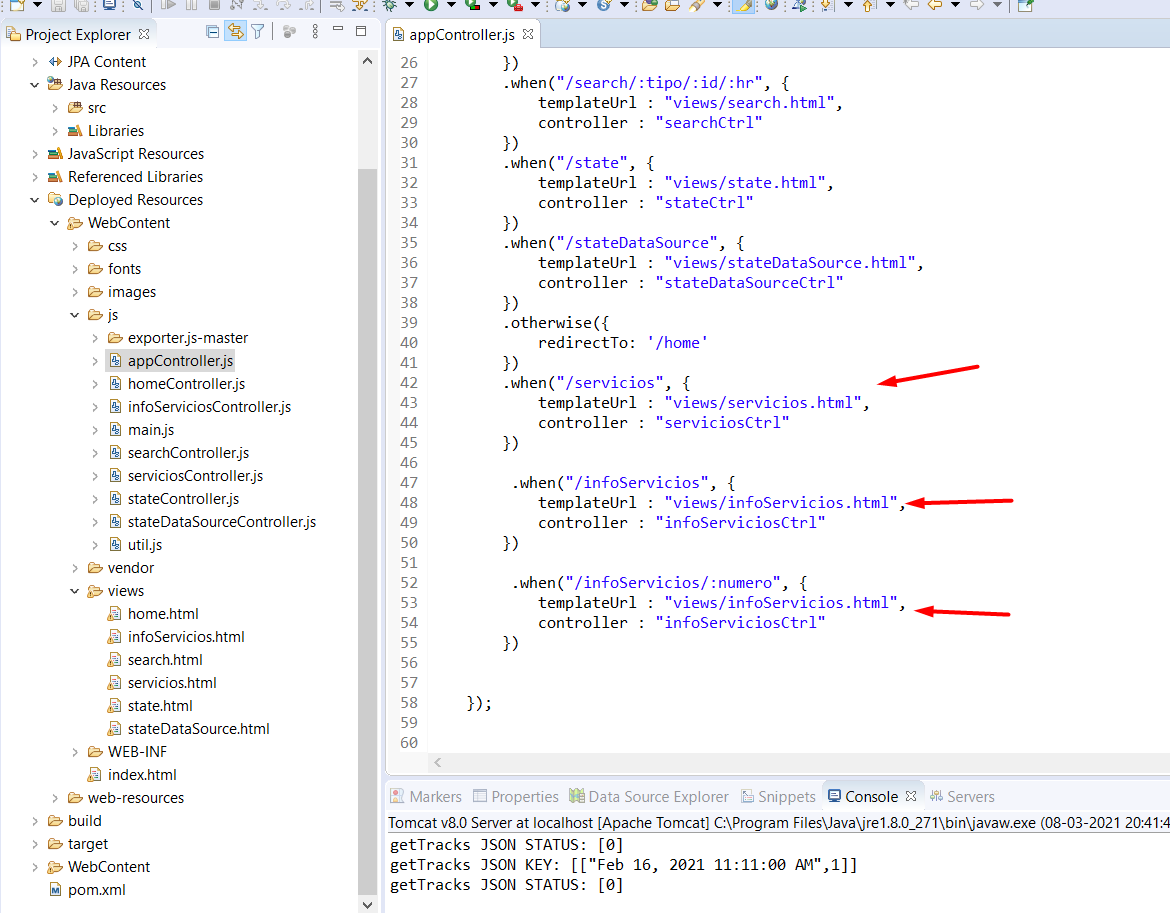


# Componentes modificados en el Frontend

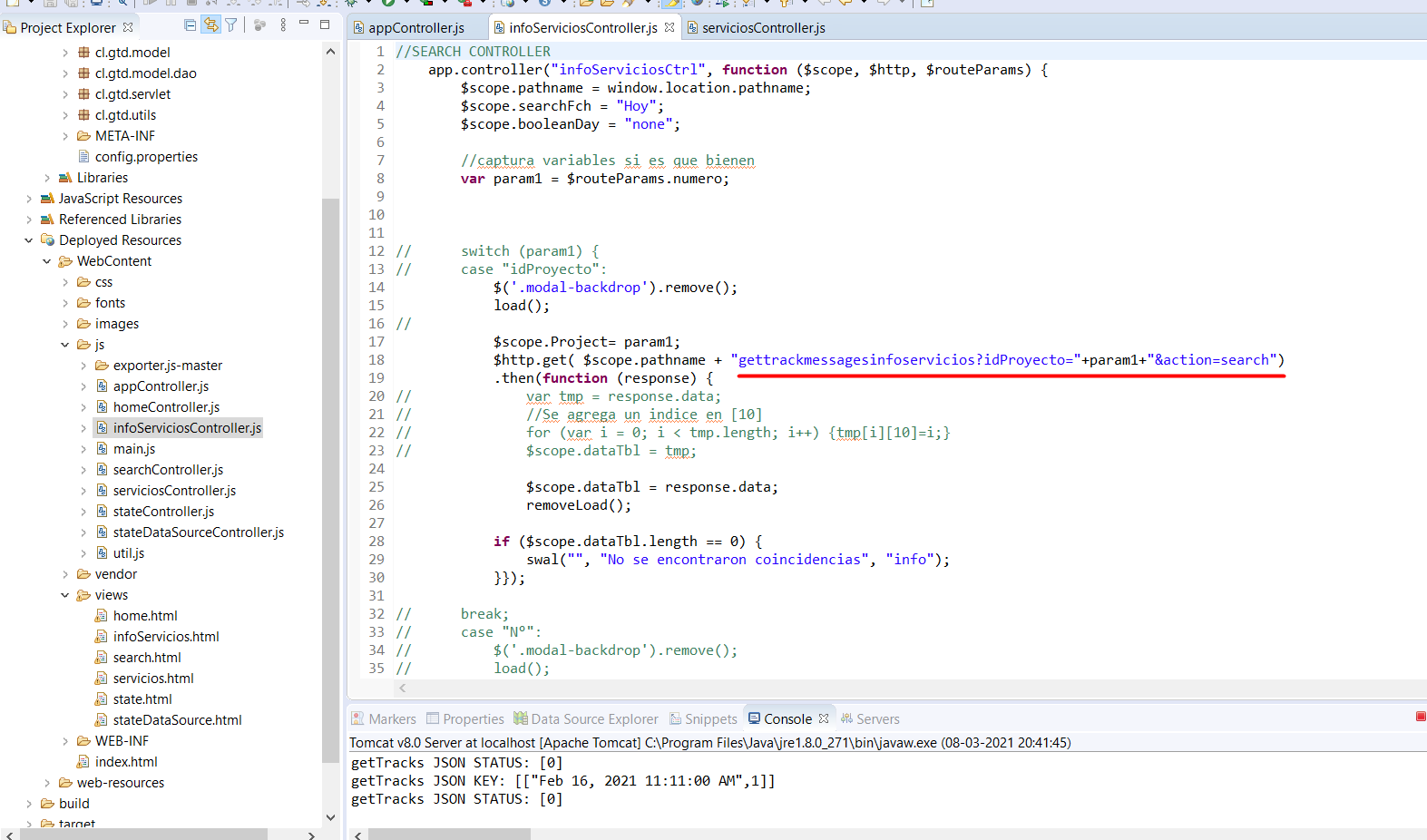


Todo lo subrayado en rojo es lo agregado nuevo al backend. Se utilizó la misma lógica del proyecto base Tracking, pero se crearon nuevas clases para dar referencia a los nuevos datos que el cliente solicitó

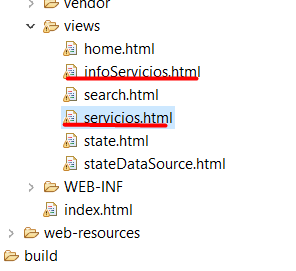
Lo principal fue definir las rutas de las views(html) que utilizaremos, referenciándolas hacia el servlet(Backend)



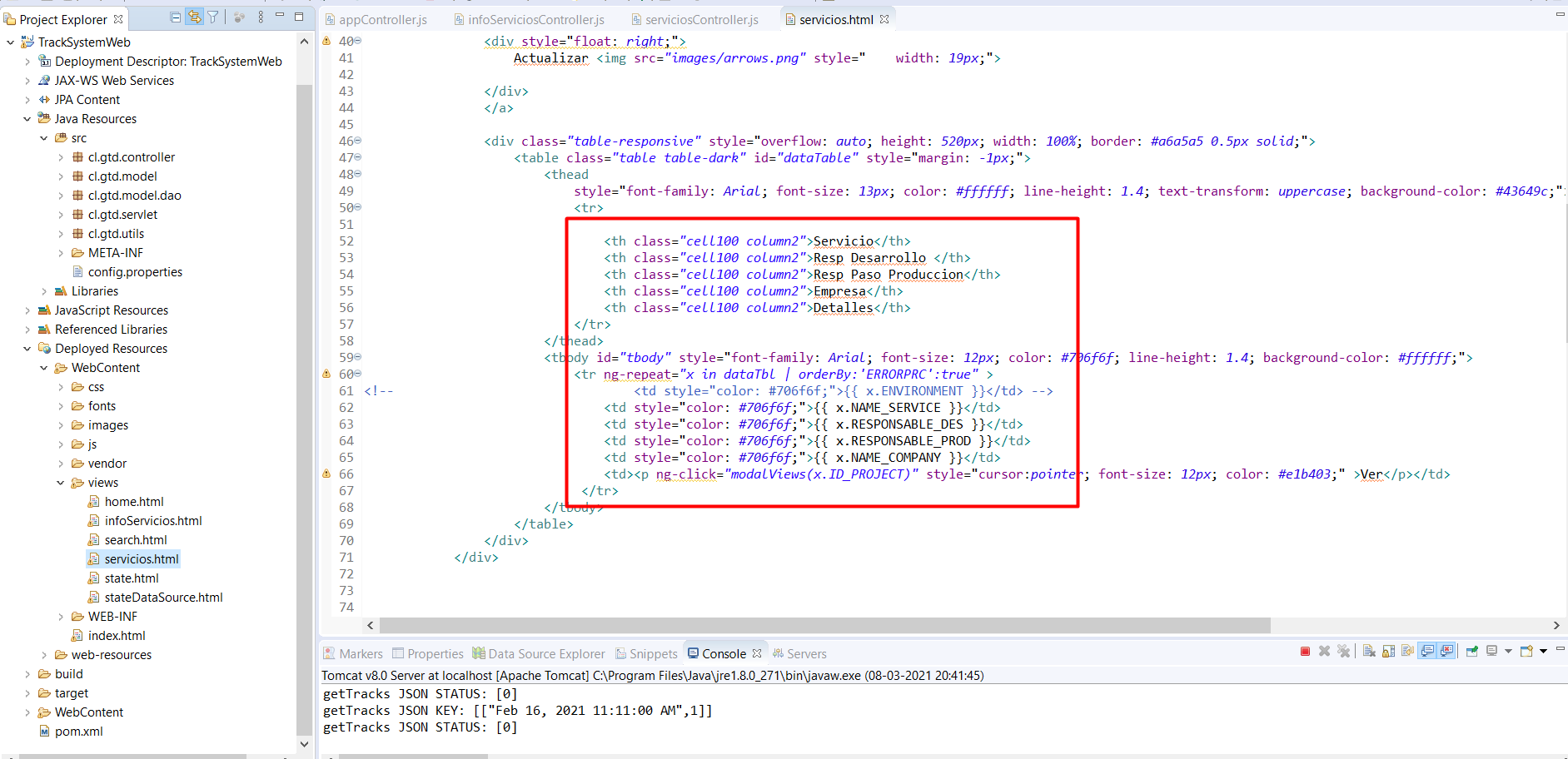
La lógica viene siendo similar a las otras plantillas, se comentó mucho código debido a que era incensario para este caso. Esta funciona js captura el id del proyecto que nos ayuda a mostrar las operaciones según su id, esto es en la consulta sql del backend



Finalmente, en las views se agregaron unas nuevas identificándolas por el nombre de **service**

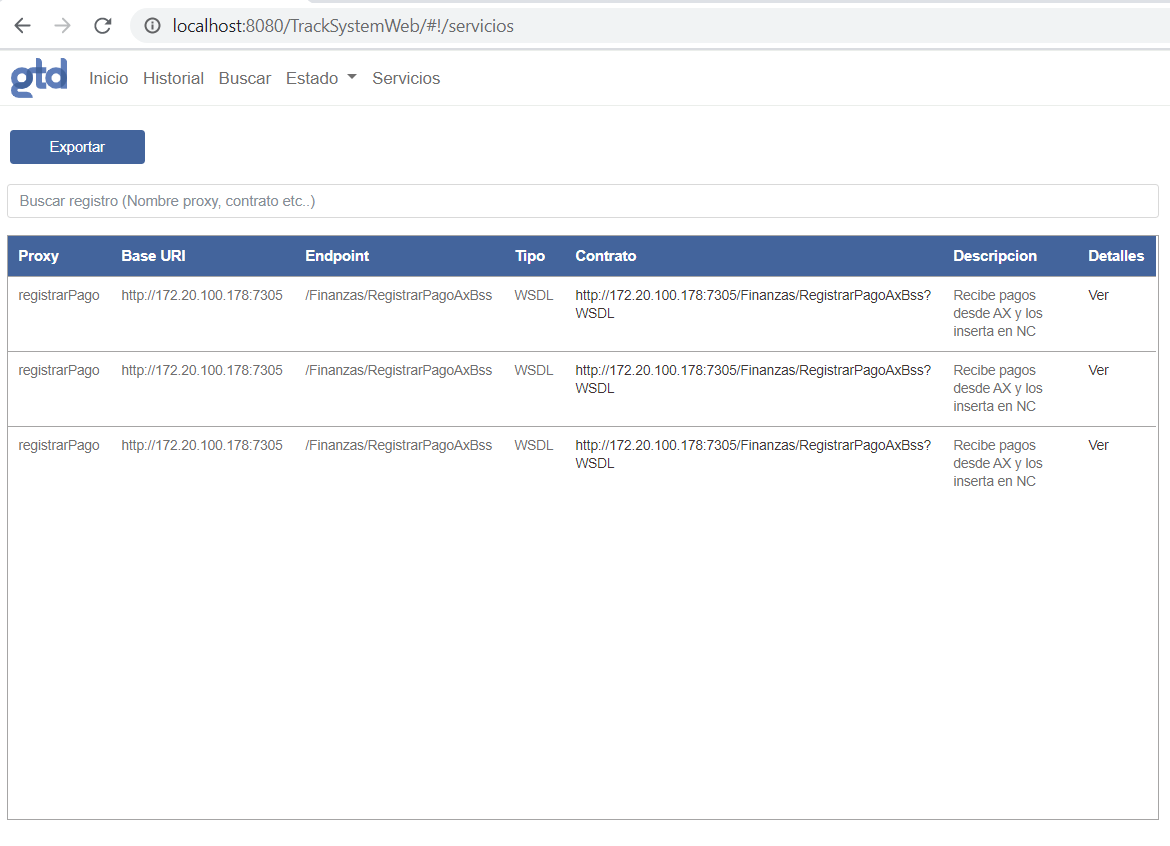


También modificando los datos a mostrar en la tabla por los datos solicitados por el cliente



# Flujo página web

Muestra la tabla principal listando los servicios.



Una vez seleccionado el proyecto la página sigue su flujo mostrando la tabla final, listando las operaciones asociadas al proyecto.

