

**FACULTAD DE INGENIERÍA**

**DEPARTAMENTO DE TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIONES**

Luisa Maria Vivas Mayor Git: https://github.com/LuisaVivas/TallerCloud

Curso: Computación en la Nube

Taller Práctico - Google App Engine

Periodo: Enero - Mayo 2017

Profesores: Juan Camilo Sacanamboy - Karen Lorena Lara

Correo: [csacanam@gmail.com](mailto:csacanam@gmail.com), [kllara@icesi.edu.co](mailto:kllara@icesi.edu.co)

Para la realización del siguiente taller se llevó a cabo la implementación de la aplicación de equipos. En la siguiente captura se puede ver el funcionamiento correcto de los servicios en postman:

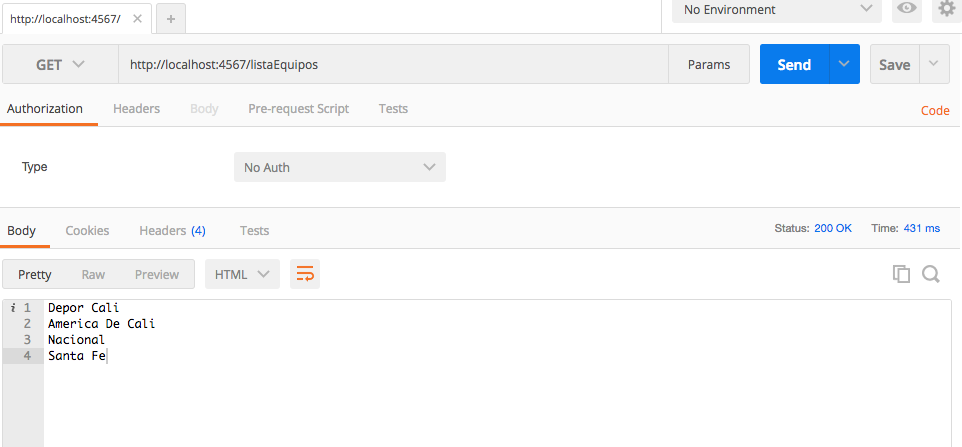


Ilustración Servicio Lista Equipos

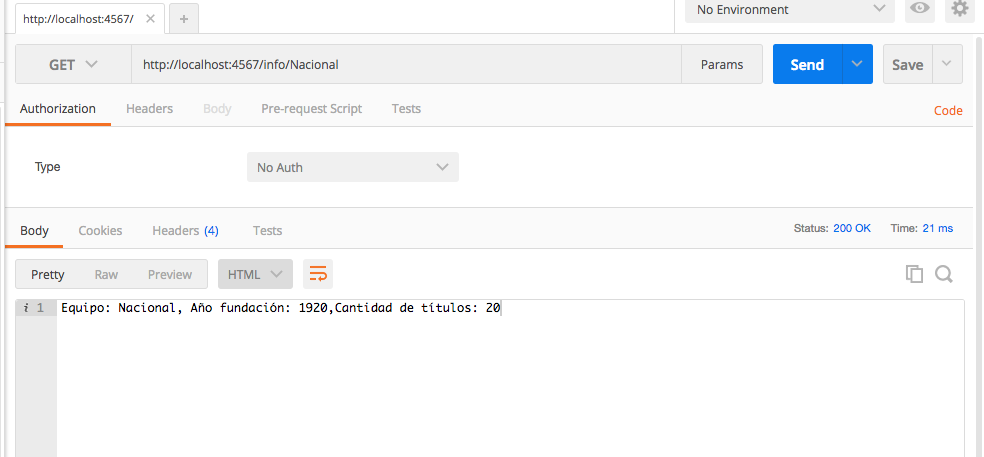


Ilustración Servicio 2 Información Equipo

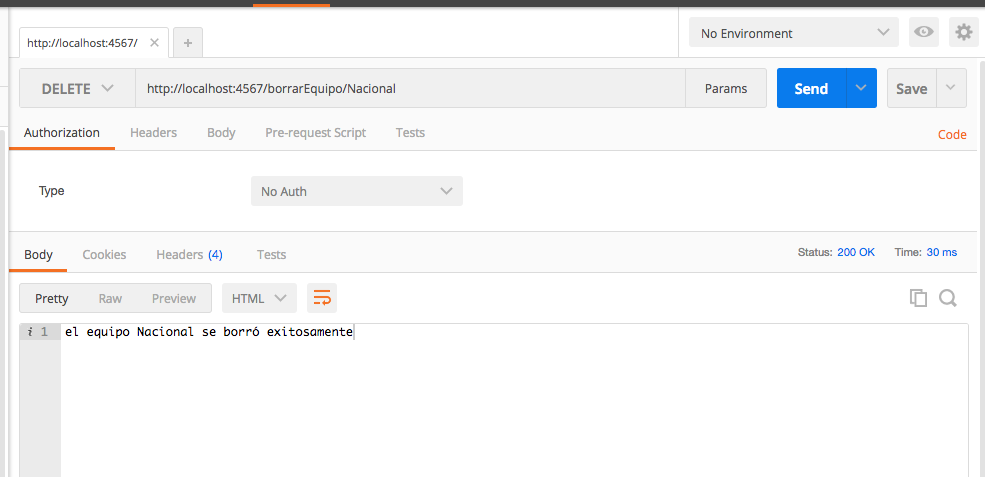


Ilustración Servicio 3 Borrar equipo

Se lista nuevamente para comprobar que el equipo Nacional se borró exitosamente:

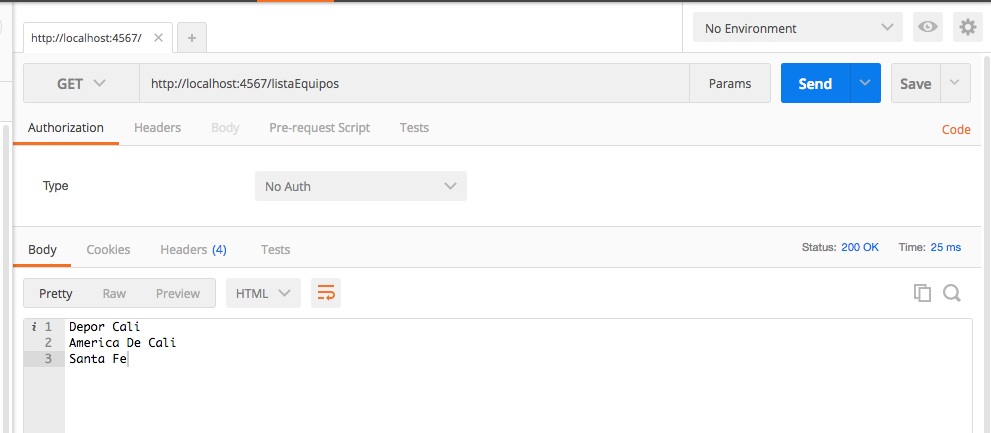


Ilustración Lista de equipos sin el equipo que se ha eliminado

Seguido de esto se comprueba el servicio editar, que permite a partir de un equipo ya creado editarla información de este.

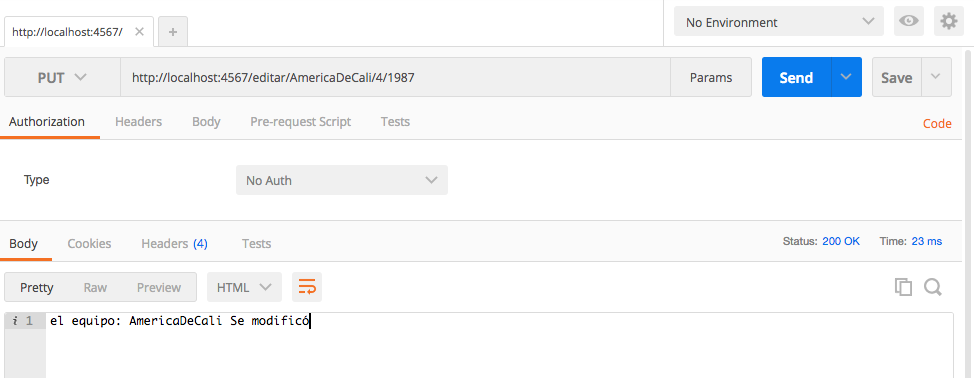


Ilustración Servicio 4 Editar Equipo

Se comprueba que esta información se edite correctamente, por tanto, hacemos una petición de obtener la información del equipo que fue editado. En este caso América De Cali

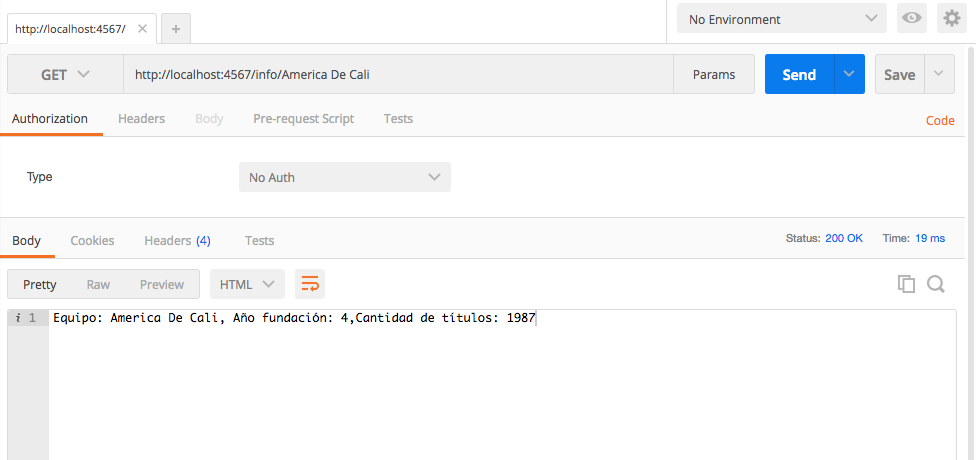


Ilustración Servicio4 funciona correctamente

Por último tenemos un servicio que permite agregar un equipo a la lista de equipos con su respectiva información.

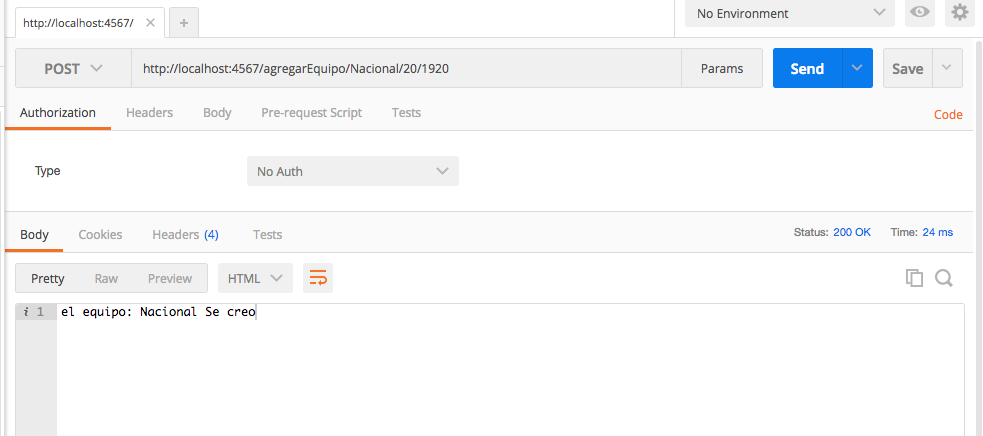


Ilustración Servicio 5 Agregar equipo

Y del mismo modo como hemos estado realizando lo anterior, comprobamos que se agregue exitosamente a la lista de Equipos.

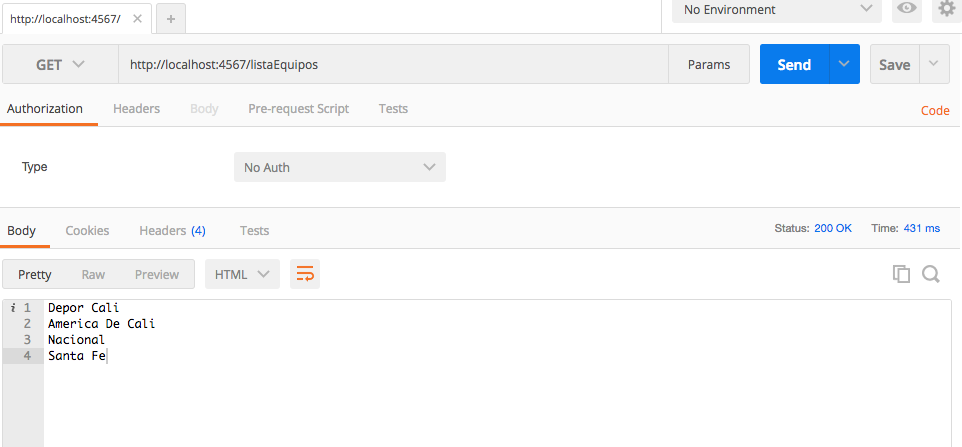


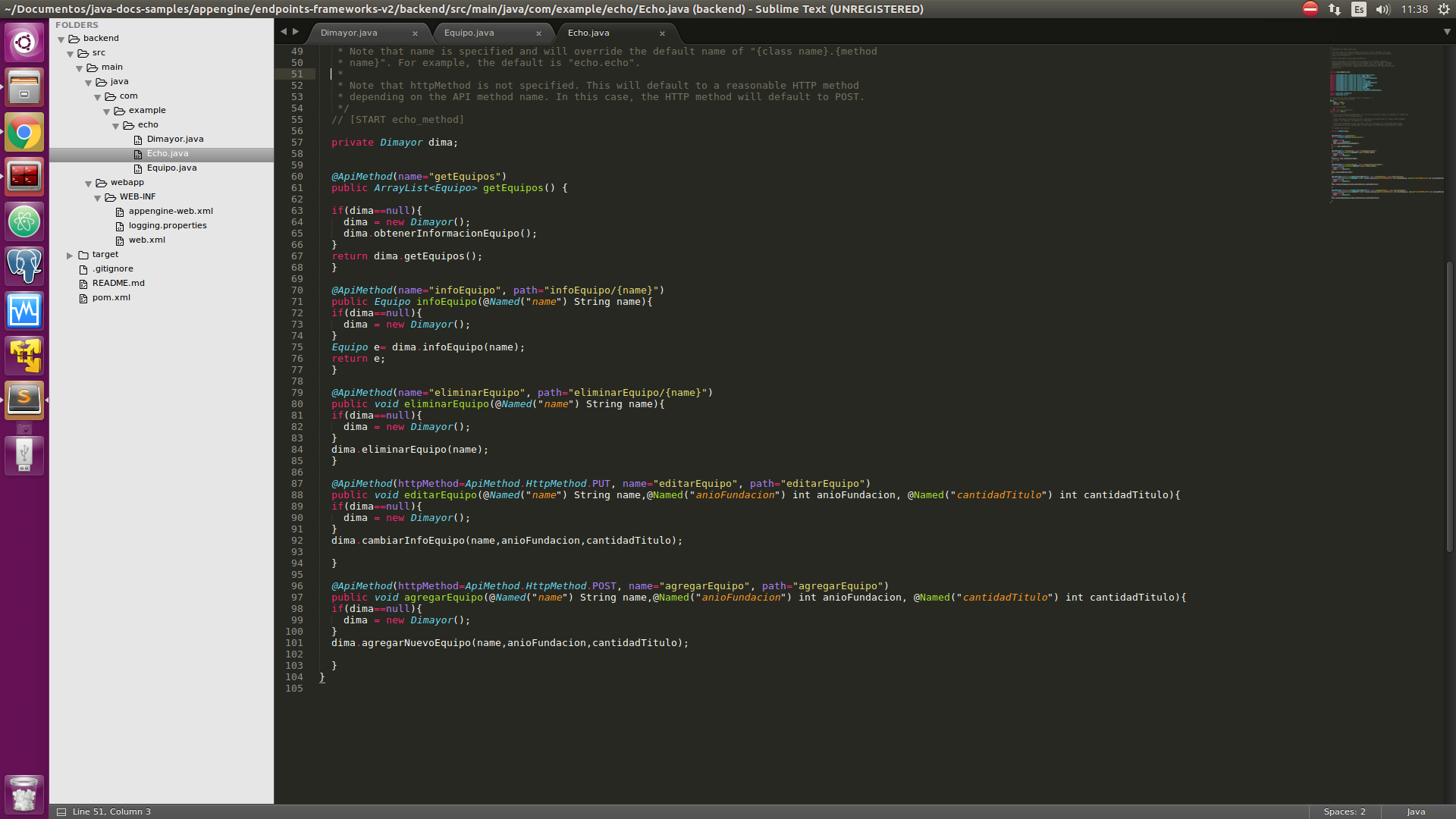
Ilustración Lista de equipos ctualizada

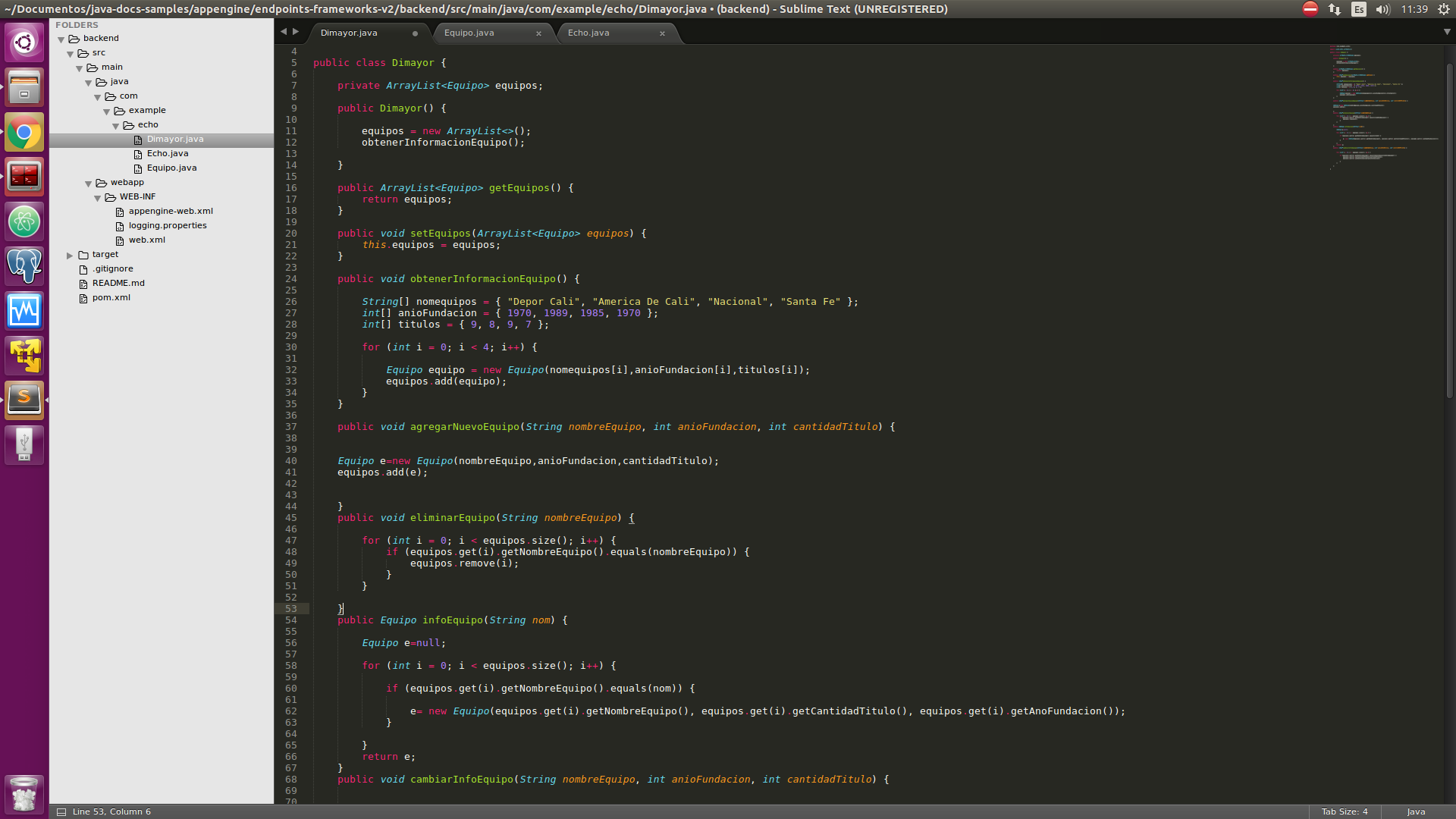
El servicio que provee Google para facilitar la integración con múltiples clientes son los Google Cloud Endpoints.

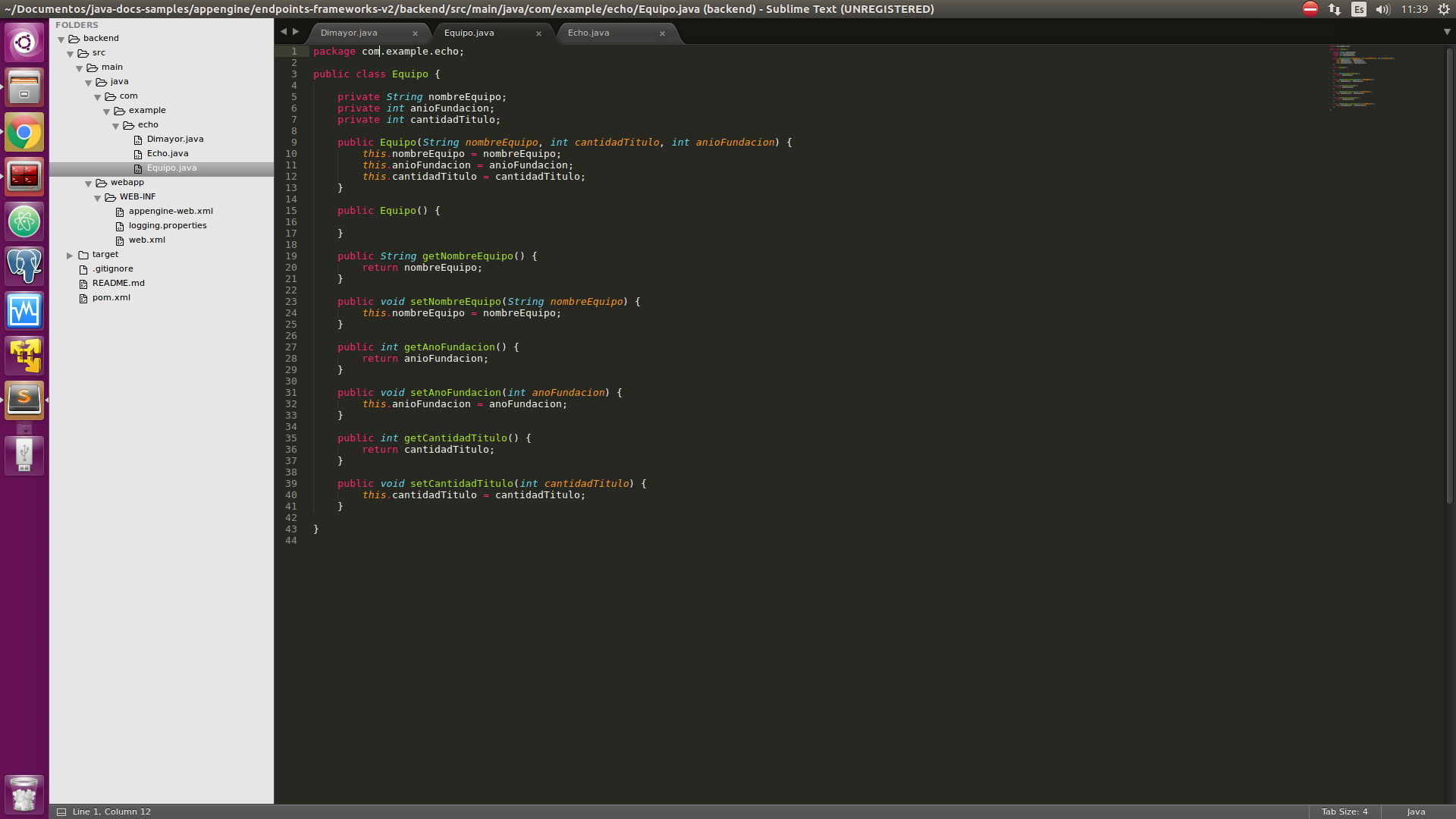
Una vez que hemos configurado el Quickstart para java, debemos migrar los web services creados anteriormente.

Google Cloud Endpoints son un conjunto de herramientas que nos permiten, de forma sencilla e incluso automáticamente, generar las APIs para poder comunicar nuestras aplicaciones clientes (webs y móviles) con una web backend.

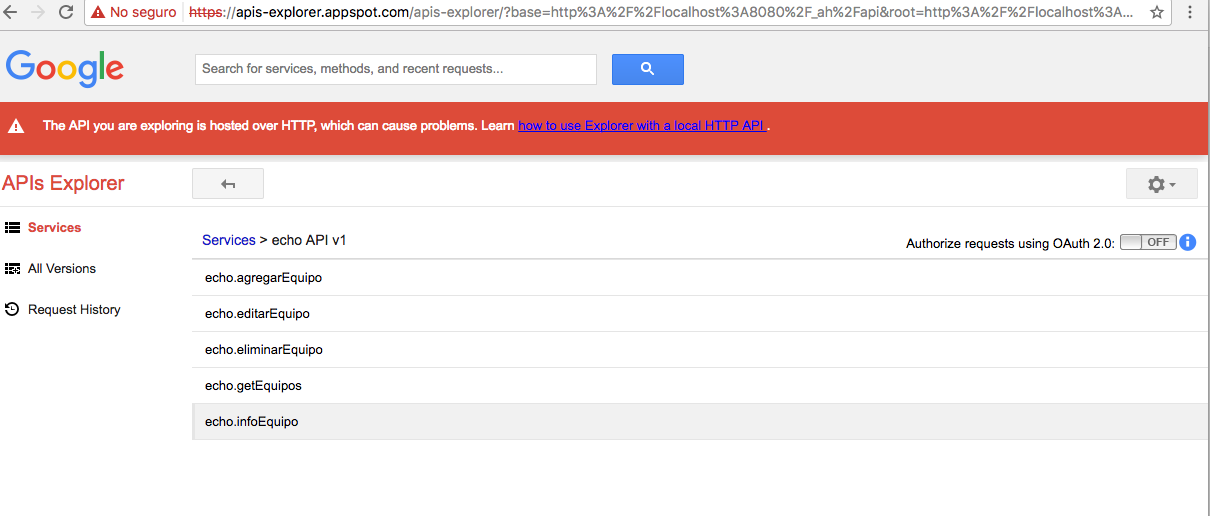
Dentro del directorio /Users/luisavivas/Downloads/java-docs-samples/appengine/endpoints-frameworks-v2/backend ubicamos las clases de nuestra aplicacion y en la clase echo.java llevamos a cabo todo lo necesario para migrar los servicios.



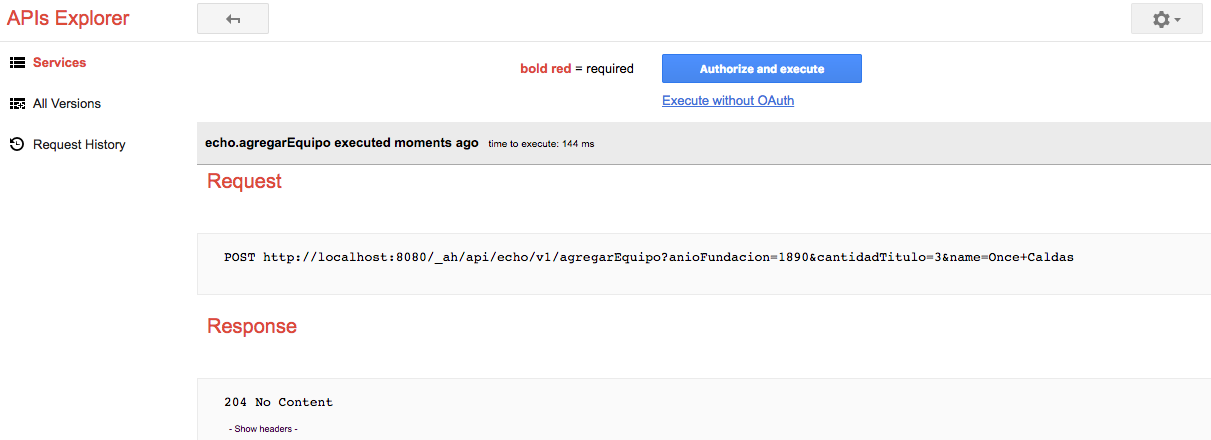




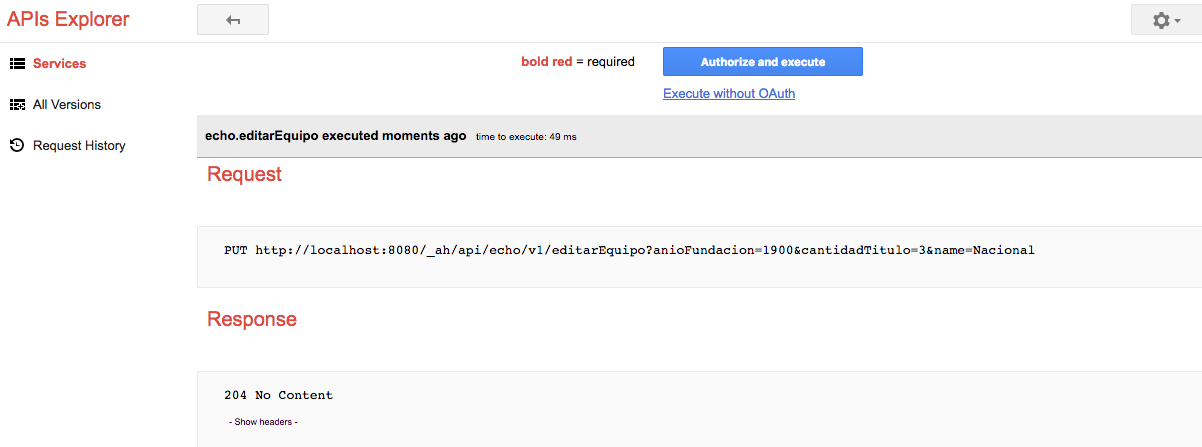
Comprobación de los webs services usando el API Explorer de Google.

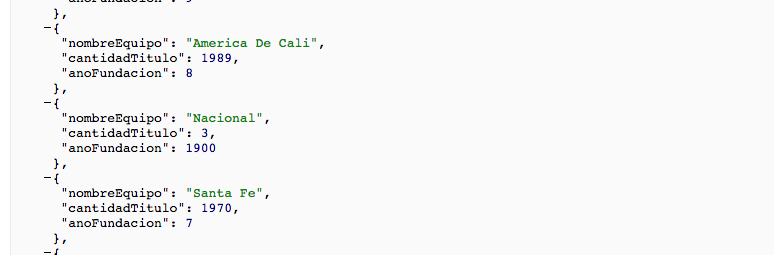


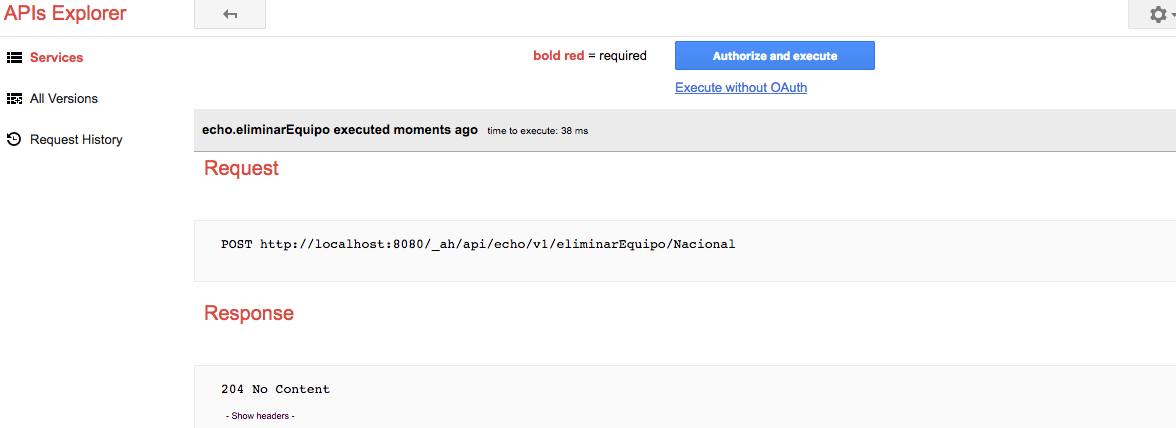






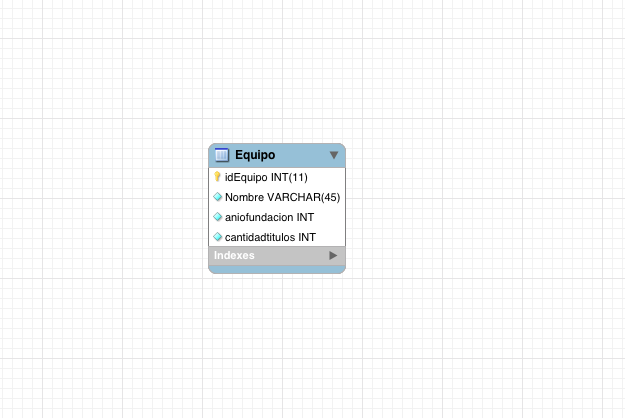


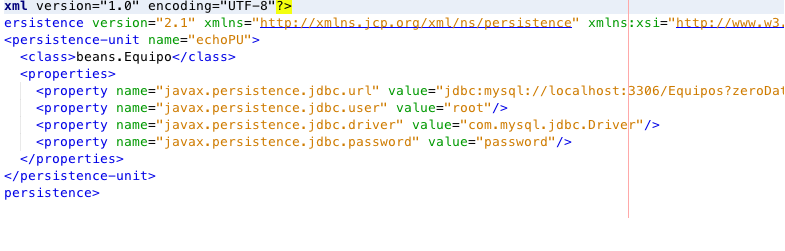




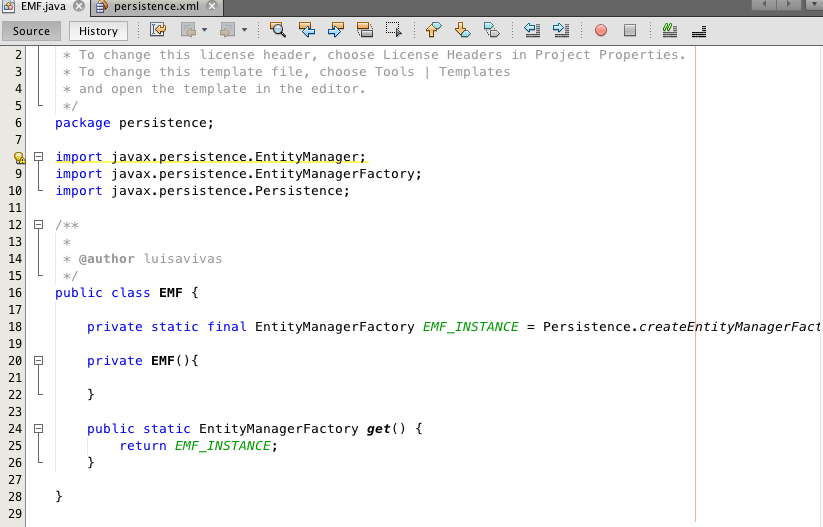
Integración de la base de datos:

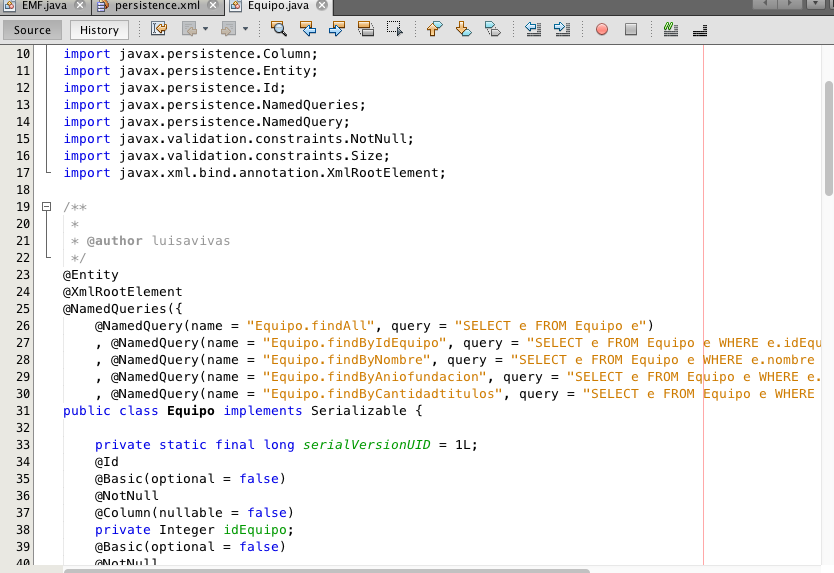
Para este paso lo primero que realizamos fue el modelo de datos de nuestro caso, para poder de esta manera integrar la base de datos:



Persistence.xml generado, el cual debe de estar dentro de los paquetes Other Sources -> src/main/resurces 🡪META-INF 🡪 persistence.xml

Segudo de esto, dentro del proyecto creamos un paquete persistence y este creamos la clase EMF.java el cual contiene la declaración del EntityManager para de esta manera identificar beans como una entidad a guardar en la base de datos y un idEquipo que es diferentes para todos lo equipos.

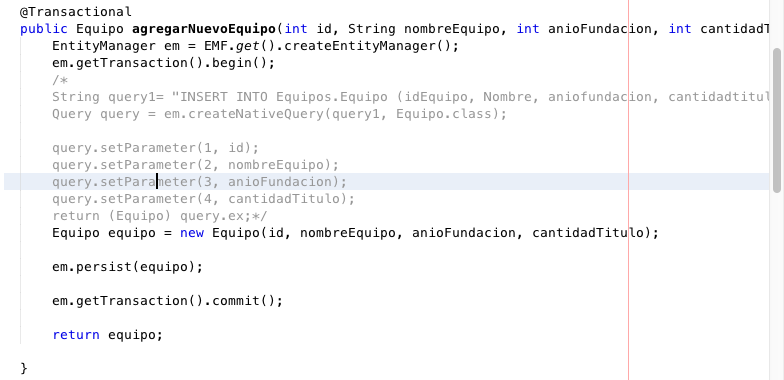




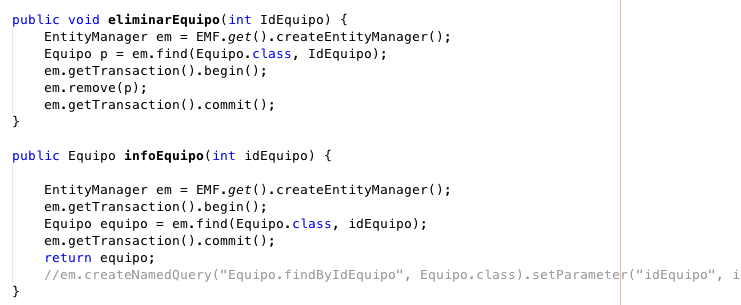
Clase Dimayor() 🡪 método: dar lista de equipos



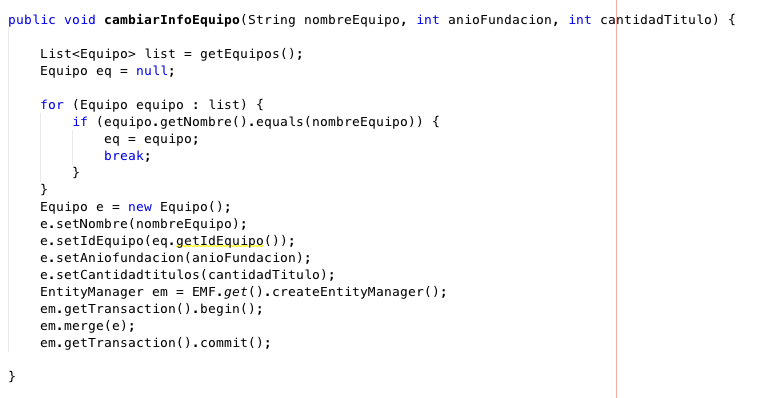
Metodo: Agregar Nuevo equipo



Métodos: Eliminar equipo y dar información equipo



Método editar informacion equipo



Clase Echo metodos:



