ADOPTA.ME

# Manual de instalación

Manual de instalación 1

Introducción 2

Tecnologías utilizadas 2

Instalación modulo web 3

Base de Datos local 3

Entorno local 6

Entorno remoto 14

Base de datos Remoto 15

# Introducción

La aplicación adoptame esta dividida en dos módulos uno web y otro móvil, la aplicación web corre sobre un servidor web y es el modulo encargado de administrar los distintos aspectos de la aplicación así como de exponer una pagina publica para ofrecer información al usuario, además aloja los distintos servicios REST que se consumen desde el modulo móvil.

Este último consiste en una aplicación móvil para android desde donde los usuarios van a poder crear una cuenta y loguearse para utilizar la aplicación.

# Tecnologías utilizadas

Para el modulo web las tecnologías y herramientas que se usaron son:

* Java 8 <https://java.com/en/download/>
* Spring boot 2.1.3 RELEASE <https://spring.io/projects/spring-boot>
* Base de datos Postgres SQL 2.3.5 <https://postgresapp.com/>
* Cliente postgres SQL <https://www.pgadmin.org/download/pgadmin-4-macos/>
* Hosting remoto <https://heroku.com>
* Git <https://git-scm.com/downloads> .
* Github <https://github.com/>
* Maven <https://git-scm.com/downloads>
* Intellij Idea Ultimate 2017.2 <https://www.jetbrains.com/idea/download/>

Para el modulo móvil las tecnologías y herramientas utilizadas son:

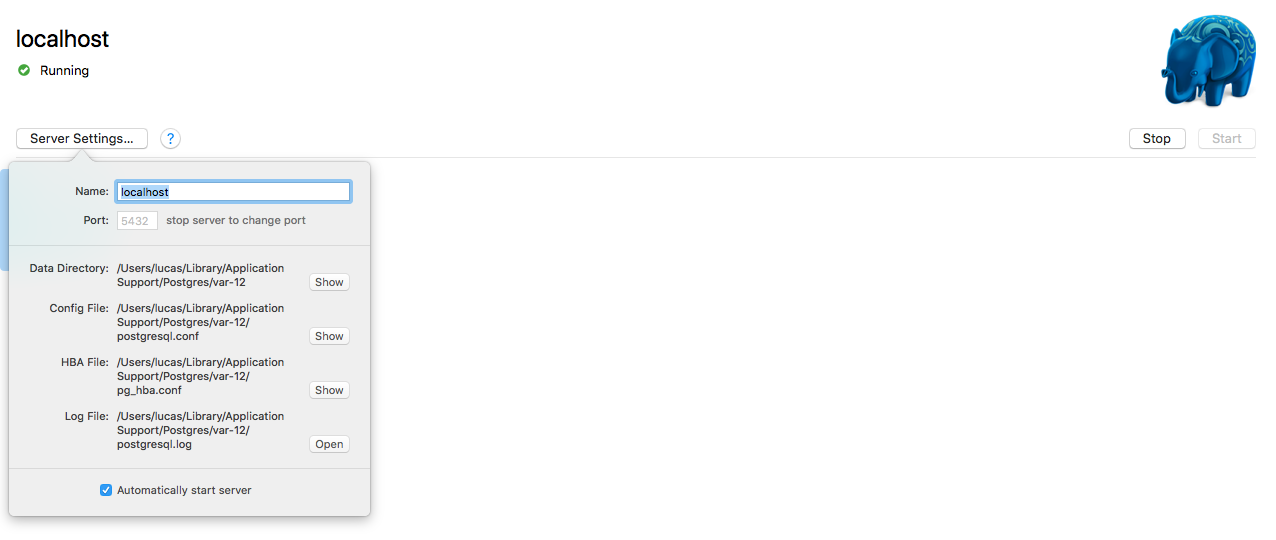
* React Native
* Android SDK 10
* Android Studio (emulador)
* Github para control de versiones

# Instalación modulo web

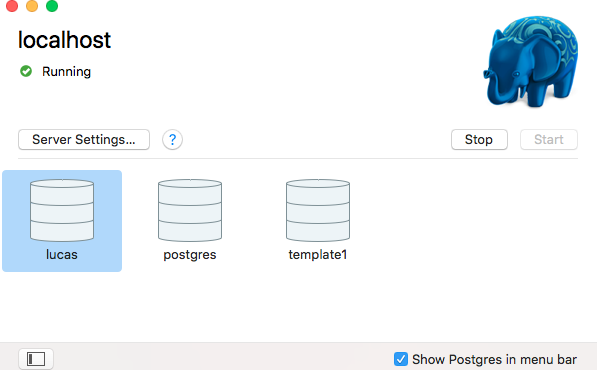
Una vez que tenemos instaladas todas las herramientas mencionadas en el apartado anterior empezamos a configurar la base de datos

## Base de Datos local

Una vez que instalamos postgress.app lo ejecutamos , vamos a server Setting y en “name” le ponemos localhost

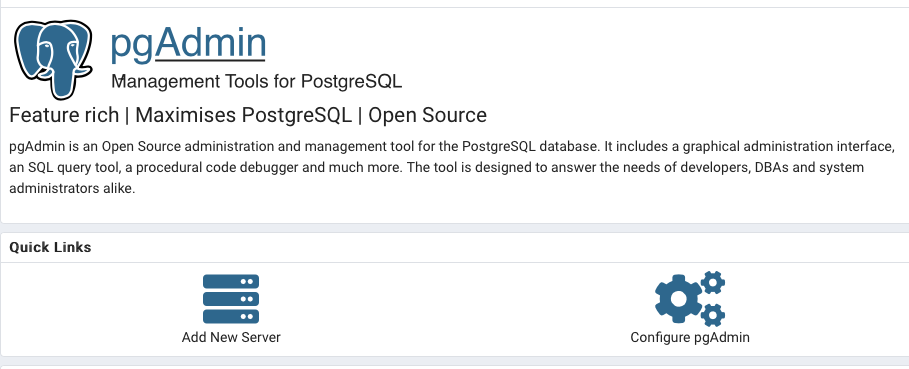


por default nos va a crear una base de datos con nuestro usuario de sistema

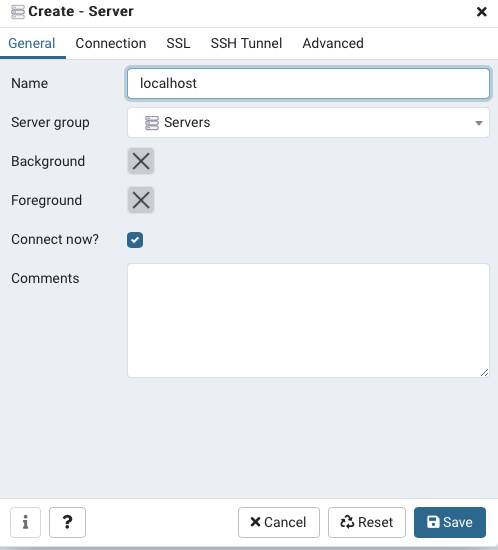


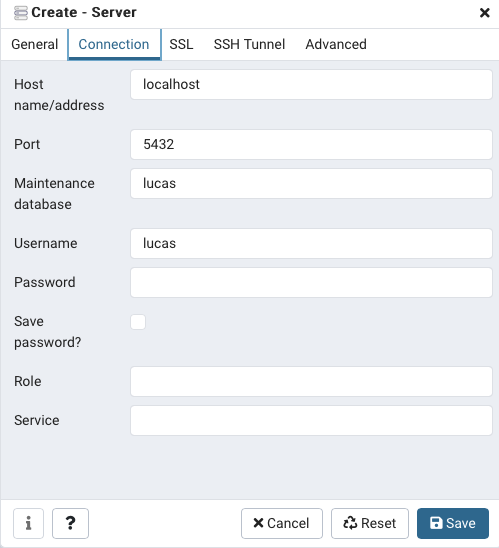
desde acá podemos parar y arrancar la base de datos cuando querramos

para poder conectarnos a la base de datos podemos utilizar el cliente pgAdmin o cualquier otro que querramos, simplemente poniendo la base de datos y el usuario con el que nos conectamos que por default es el usuario del sistema, en caso de usar el cliente pgadmin , creamos un nuevos server

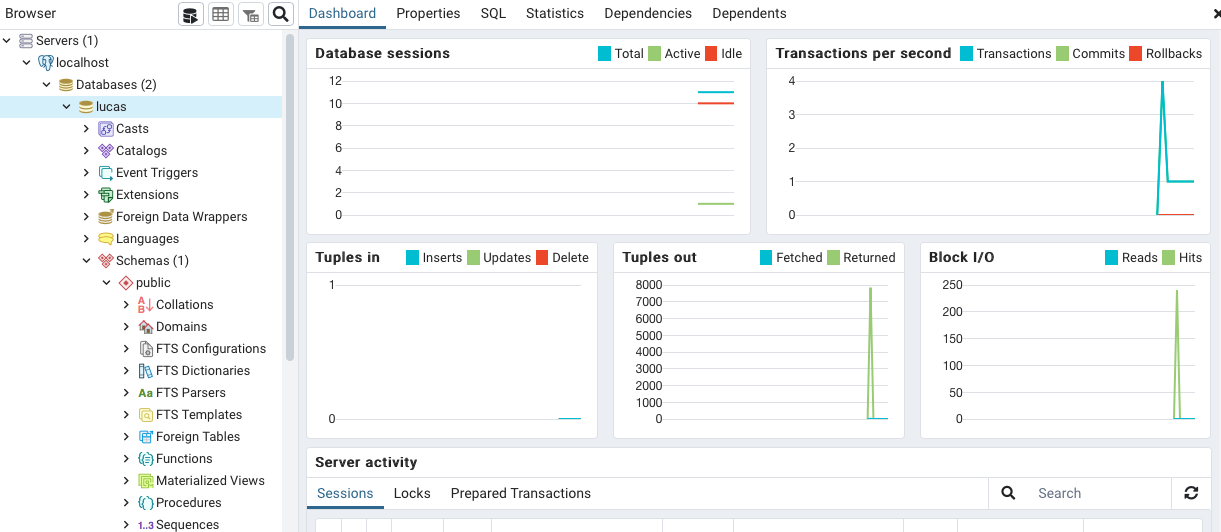


completamos los datos de conexión, en la solapa general y en la de Conection





y listo ya estamos conectados a la base de datos

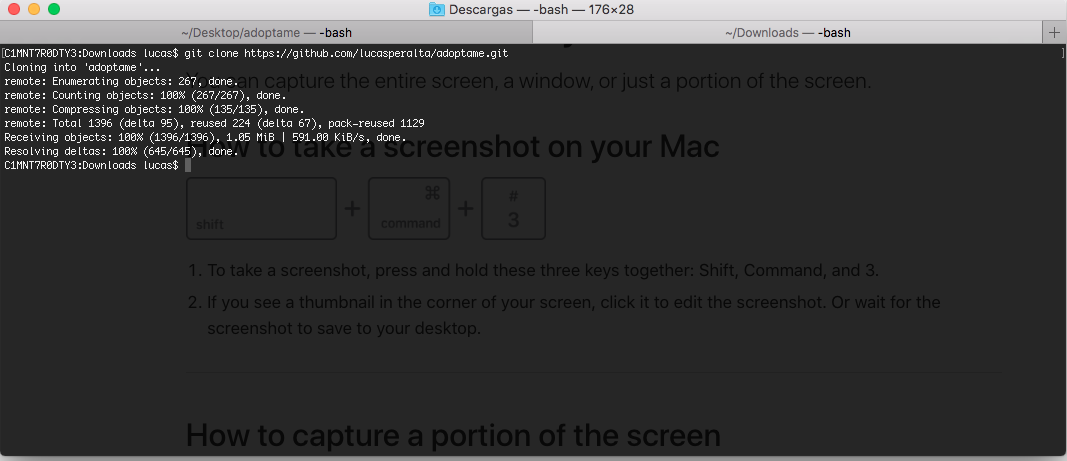


## Entorno local

Para continuar vamos a bajar el código fuente desde github.

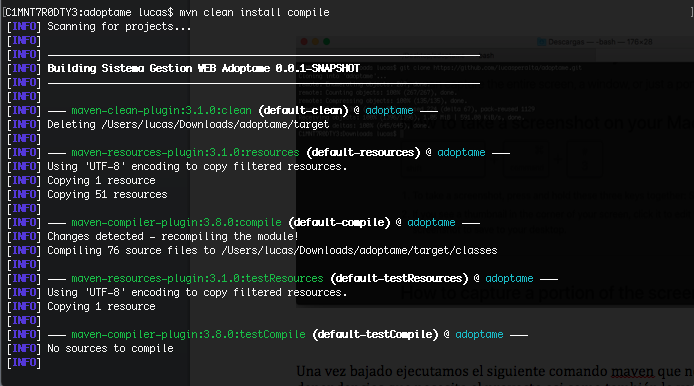
Abrimos una terminal y ejecutamos el siguiente comando en un directorio donde querramos bajar la aplicación

*git clone https://github.com/lucasperalta/adoptame.git*

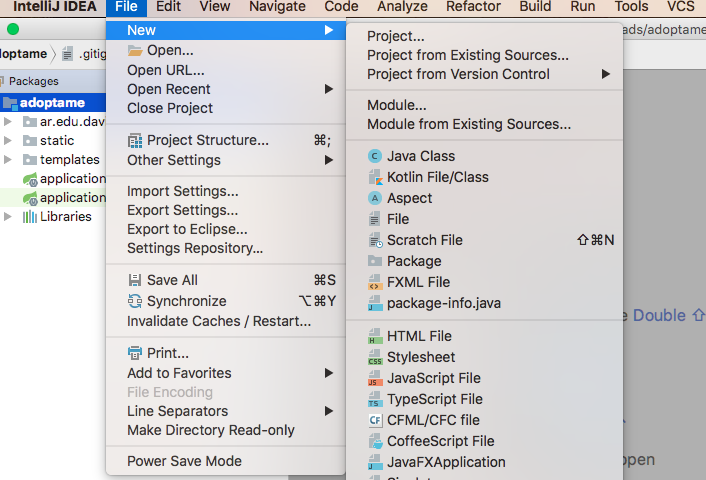


Una vez bajado ejecutamos el siguiente comando maven que nos va a bajar todas las dependencias que necesita el proyecto asi como también lo va a compilar e instalar en el el repositorio de maven, de esta forma nos aseguramos que no haya errores en el código

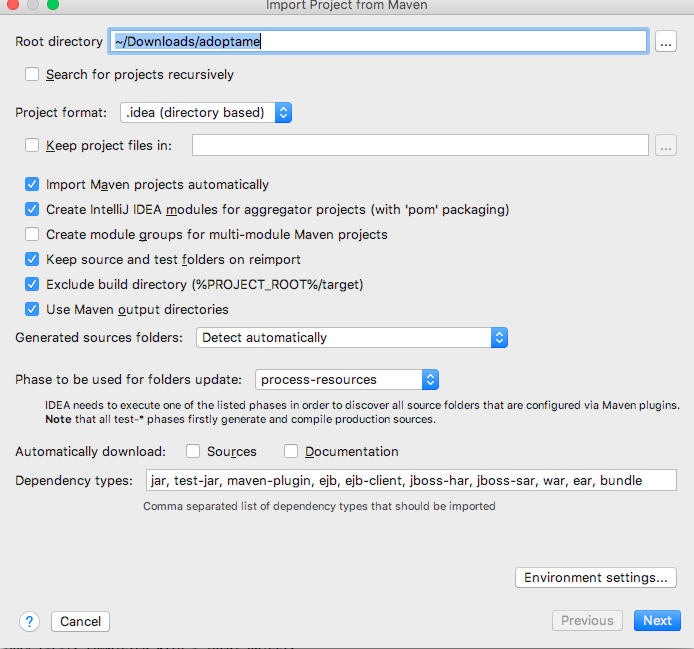
*mvn clean compile install*

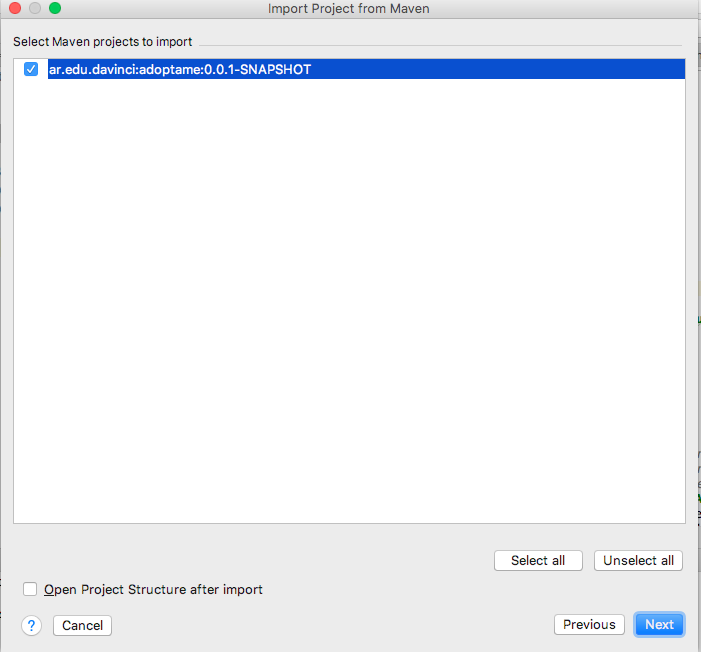


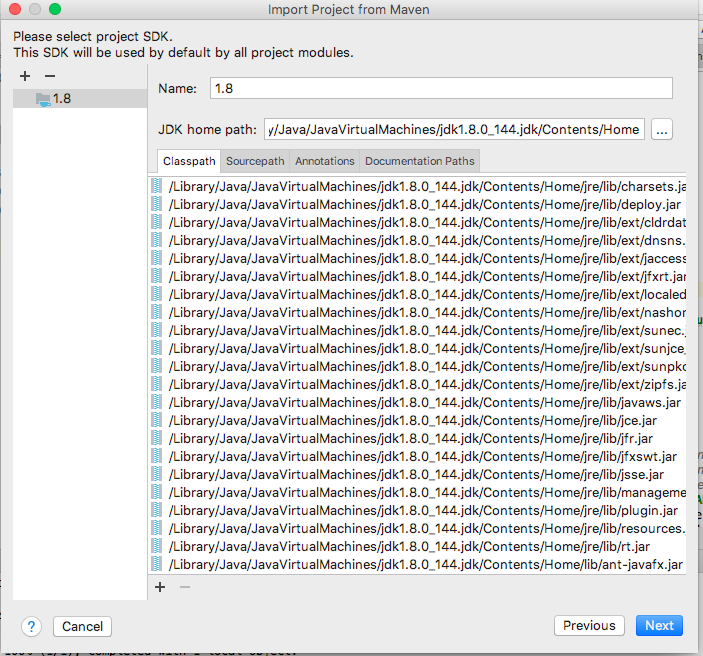
una vez hecho esto abrimos el intelliJ Idea y vamos a File->new->Projects from existing sources y elegimos el archivo POM.xml dentro del directorio del proyecto que bajamos de github

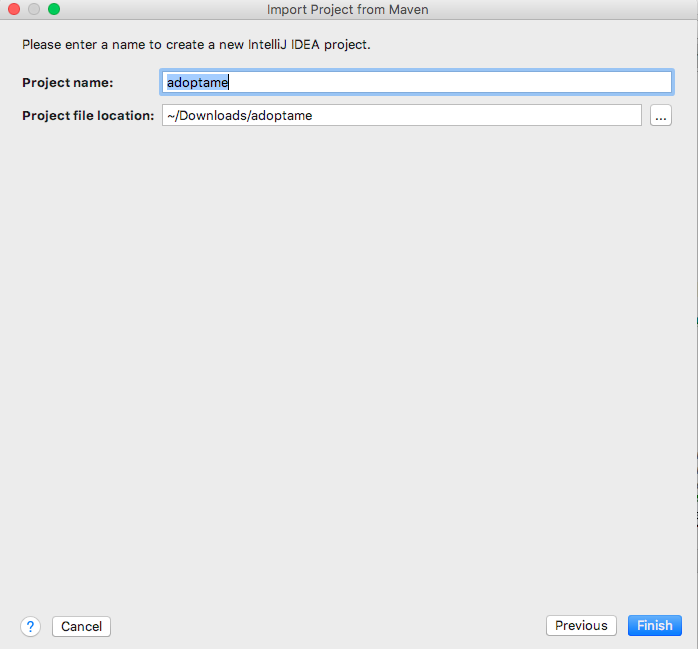


Le damos siguiente hasta terminar

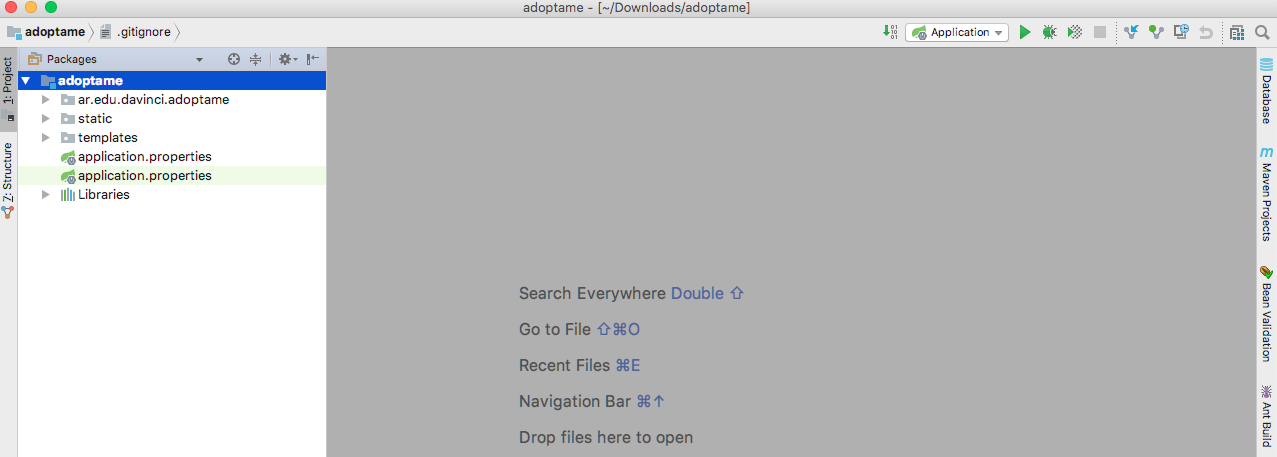






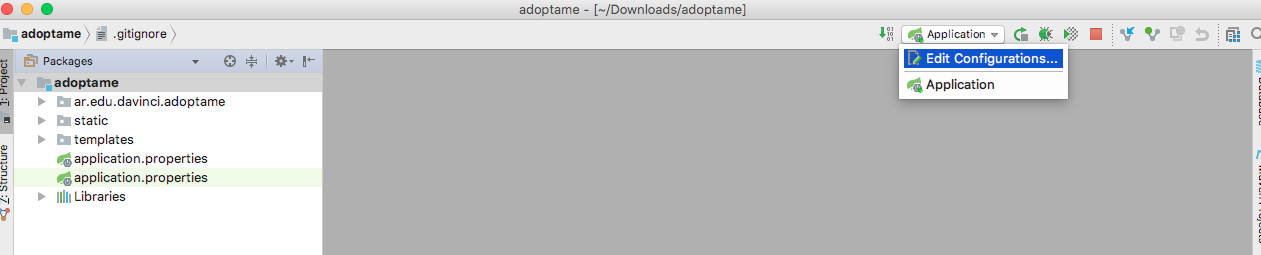


Finalmente después que termine de configurarse el proyecto nos debería quedar asi



Una vez que tenemos la aplicación ya montada en intellij nos falta un paso para poder ejecutarla localmente.

Vamos a Edit configurations…

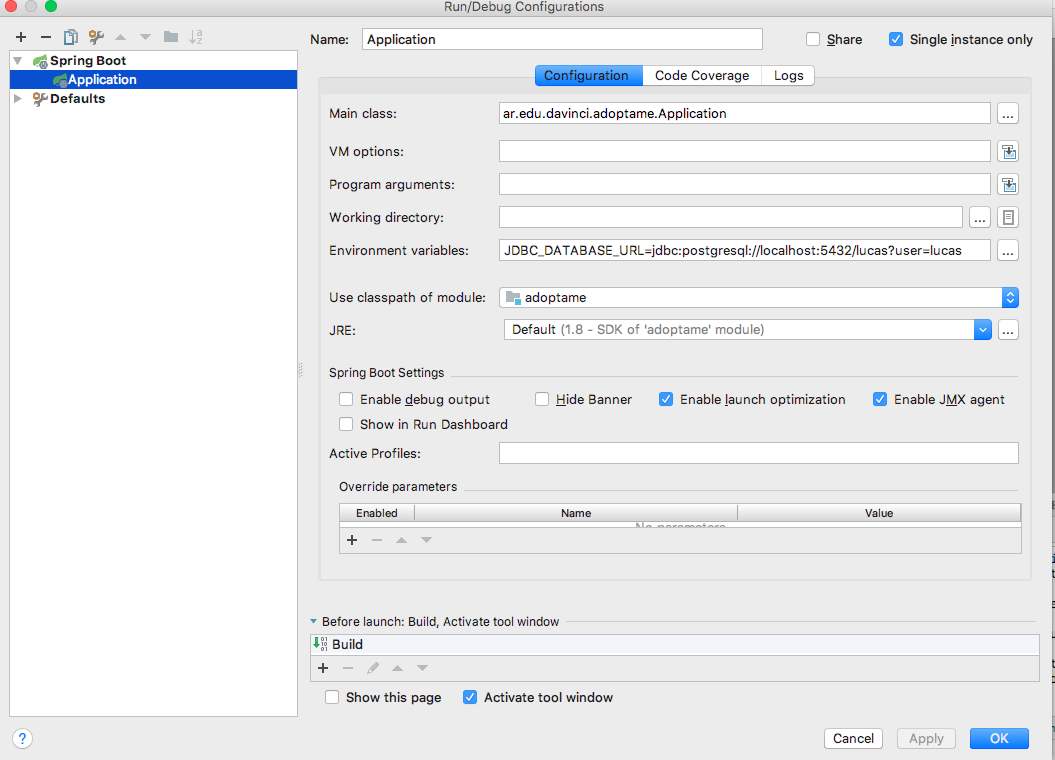


una vez aquí ,tenemos con configurar la variable de entorno JDBC\_DATABASE\_URL, con el string de conexión correspondiente a nuestra base de datos. Esto se hizo así para tener mas flexibilidad a la hora de cambiar la conexión a la base de datos ya que no es lo mismo tenerlo en local que en remoto, configurando esta variable no tenemos que hacer cambios en el código si cambiamos el nombre de la base de datos o su ubicación.

Ejemplo

*JDBC\_DATABASE\_URL=jdbc:postgresql://localhost:5432/lucas?user=lucas*

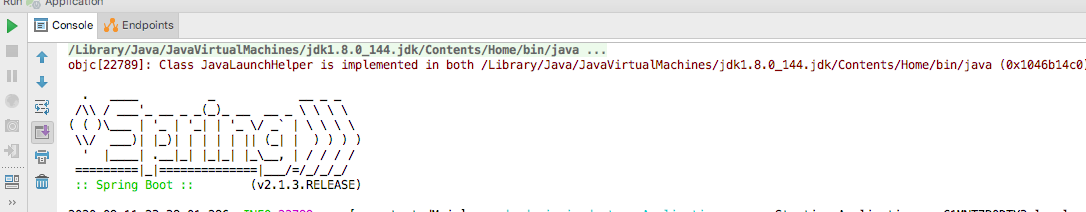
.



clickeamos en apply y ok , y corremos la aplicación dándole al botón de play o debug si queremos debugear la aplicación



Una vez hecho esto vemos como levanta la aplicación y también vemos como las tablas de la base de datos se crean automáticamente cuando levanta la aplicación por primera vez, por lo que no hace falta crearlas a mano.



para comprobar que levanto correctamente vamos a la siguiente url

<http://localhost:8090/adoptame/>

Ahora para poder ingresar a la parte administrativa vamos a necesitar ejecutar los siguientes inserts en la base de datos. Desde el cliente pgadmin ejecutamos las siguientes querys:

Cambiar por el usuario y mail que se quiera

*INSERT INTO personas (tipo\_persona, id, apellido, email, nombre, telefono, ubicacion)*

*VALUES ('Usuario',1,'peralta','lucas.peralta@gmail.com','lucas',NULL,NULL),*

*INSERT INTO estados ( id\_estado , estado)*

*VALUES (1,'ACTIVO'),(2,'INACTIVO'),(3,'SUSPENDIDO');*

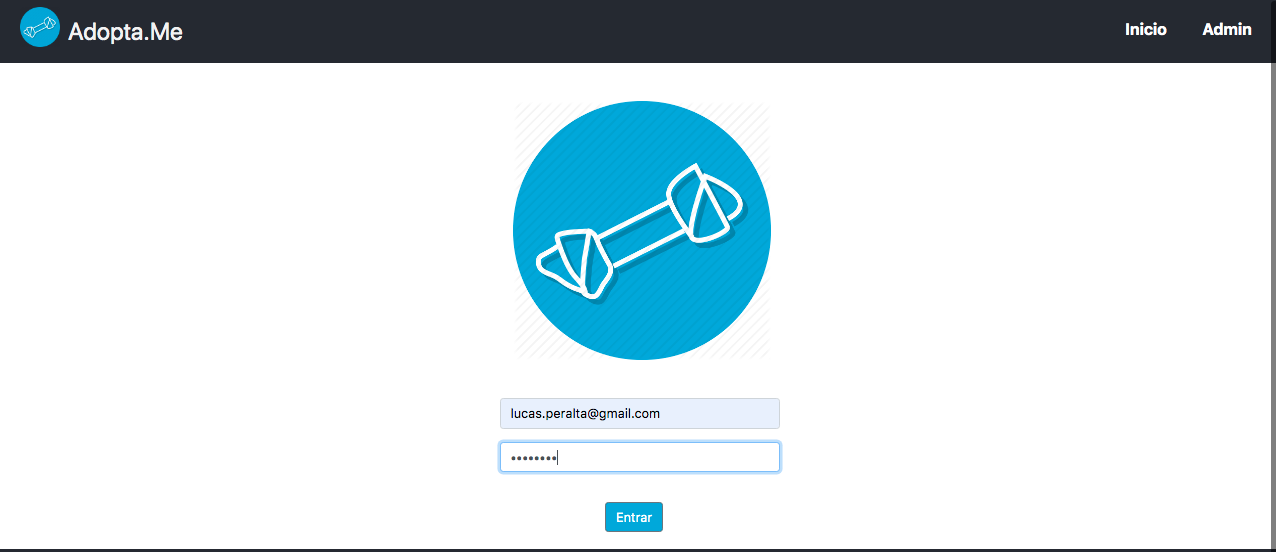
*INSERT INTO roles (id\_rol, nombre\_rol)*

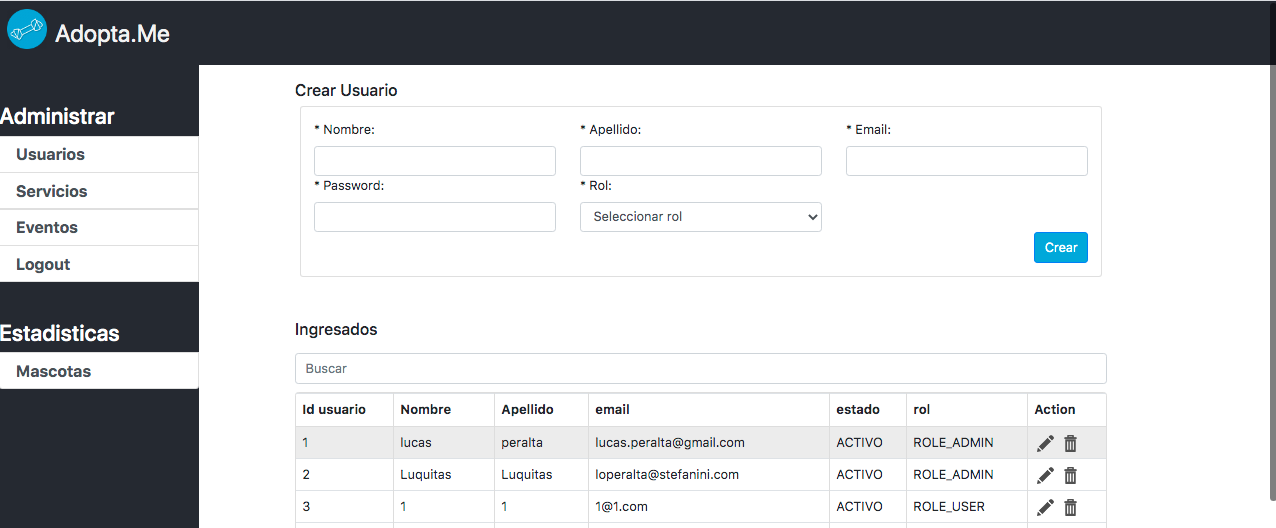
*VALUES (1,'ROLE\_ADMIN'), (2,'ROLE\_USER');*

*INSERT INTO usuarios (password, id,id\_estado, id\_rol)*

*VALUES ('$2a$10$u6i5mjYoeN7RJdifc3dw9e3G81IA1mNlqbDYfx9KBaVwHyjGfTnR.',1,1,1),*

Esto nos va a crear un usuario administrador con el password 12345678 que esta encriptado. Ya con este primer usuario luego podemos crear mas cuando nos logueamos a la aplicación y la empecemos a administrar





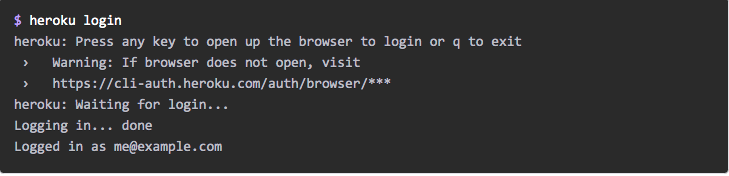
## Entorno remoto

Para montar el servidor en remoto elegimos el hosting HEROKU <https://www.heroku.com/>

Para deployar nuestra aplicación en el servidor heroku seguimos esta guía <https://devcenter.heroku.com/articles/deploying-spring-boot-apps-to-heroku#connecting-to-a-database>

Primero creamos una cuenta , la que es gratuita nos alcanza perfectamente para deployar la aplicación , la creamos aquí <https://signup.heroku.com/>

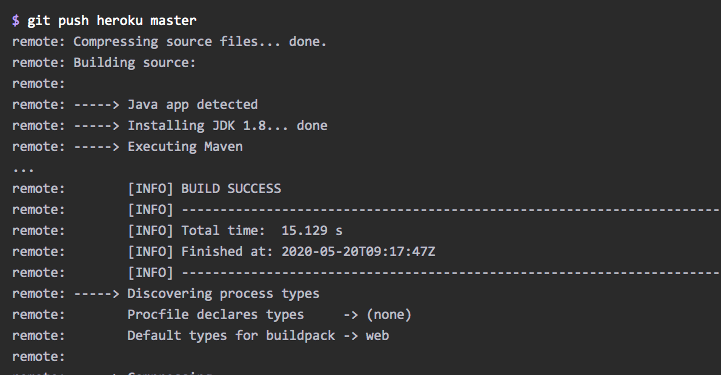
Luego nos bajamos el ciente heroku y desde una terminal abierta **en el directorio donde esta nuestra aplicación** ejecutamos el comando *heroku login* y ponemos las credenciales de la cuenta que nos creamos anteriormente



una vez hecho esto corremos el comando *heroku créate*  ,este comando nos va a crear la aplicación en heroku, también nos va a crear un repositorio remoto en nuestro git local, nos va a crear la aplicación con un nombre random que vamos a cambiar con el comando *heroku apps:rename*

Para deployar la aplicación en heroku vamos a corer el siguiente comando

*git push heroku master*

**

para abrir la aplicación corremos el comando *heroku open* y nos abre el navegador con la pagina de la aplicación. Aunque todavía no va a funcionar ya que debemos configurar la base de datos

## Base de datos Remoto

Para crear la base de datos desde la terminal ,parados en el directorio donde tenemso nuestro proyecto ejecutamos el siguiente comando *heroku addons:créate haroku-postgressql*

heroku addons:create heroku-postgresql

esto nos crea una base de datos de postgressql

con el siguiente comando obtenemos la url de conexión *heroku config*

heroku config === tranquil-mountain-19785 Config Vars DATABASE\_URL: postgres://fhehmrcgzmpjhn:30af34cb97bae5f97d6c6a13ceb6a74d6511d6f16ec77bdaadab09a398857784@ec2-35-169-254-43.compute-1.amazonaws.com:5432/defp5q4qsijggp

podemos obtener mas detalle de la base de datos con el siguiente comando

heroku pg === DATABASE\_URL Plan: Hobby-dev Status: Available Connections: 0/20 PG Version: 12.2 Created: 2020-05-20 09:18 UTC Data Size: 7.9 MB Tables: 0 Rows: 0/10000 (In compliance) Fork/Follow: Unsupported Rollback: Unsupported Continuous Protection: Off Add-on: postgresql-animated-55555

basta hacer un restart de la aplicación para que se conecte a la base de datos, esto se hace automáticamente ya que heroku tiene la cadena de conexión en la variable de entorno **${JDBC\_DATABASE\_URL}** que ya la tenemos configurada en el aplication properties.

Para conectarse a la base de datos remoto seguimos los mismos pasos que con la base de datos local solo que cambiamos los parámetros de conexión por los que nos da heroku.

Podemos obtener mas información desde el dashboard de la cuenta heroku que hemos creado <https://dashboard.heroku.com/apps/adoptameapp>