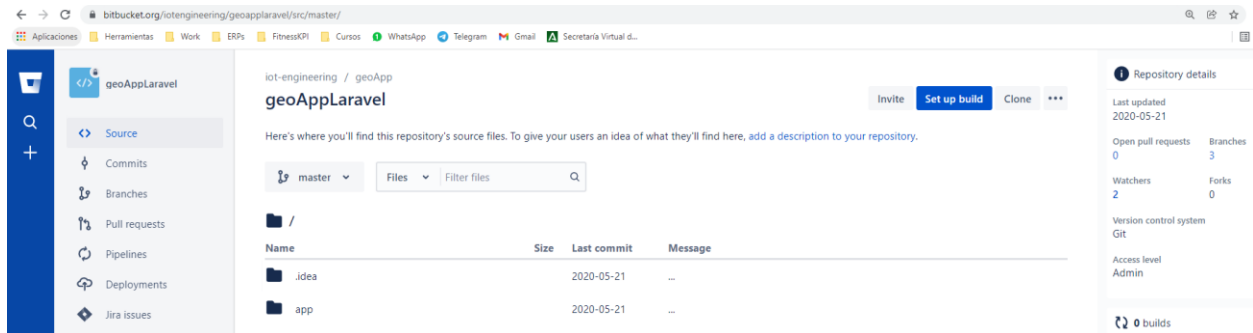


GLOSITA

Para realizar la instalación es conveniente trabajar en entorno Ubuntu (Linux) o Mac. Se deberá de tener instalado la última versión de node.

1. Clonar repositorio Laravel - [geoAppLaravel](#)

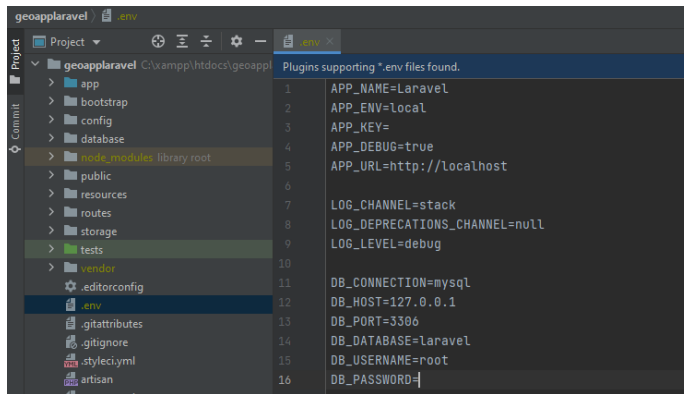


2. Ejecutar los siguientes comandos en la terminal:

- a. ***composer install***
- b. ***npm install***

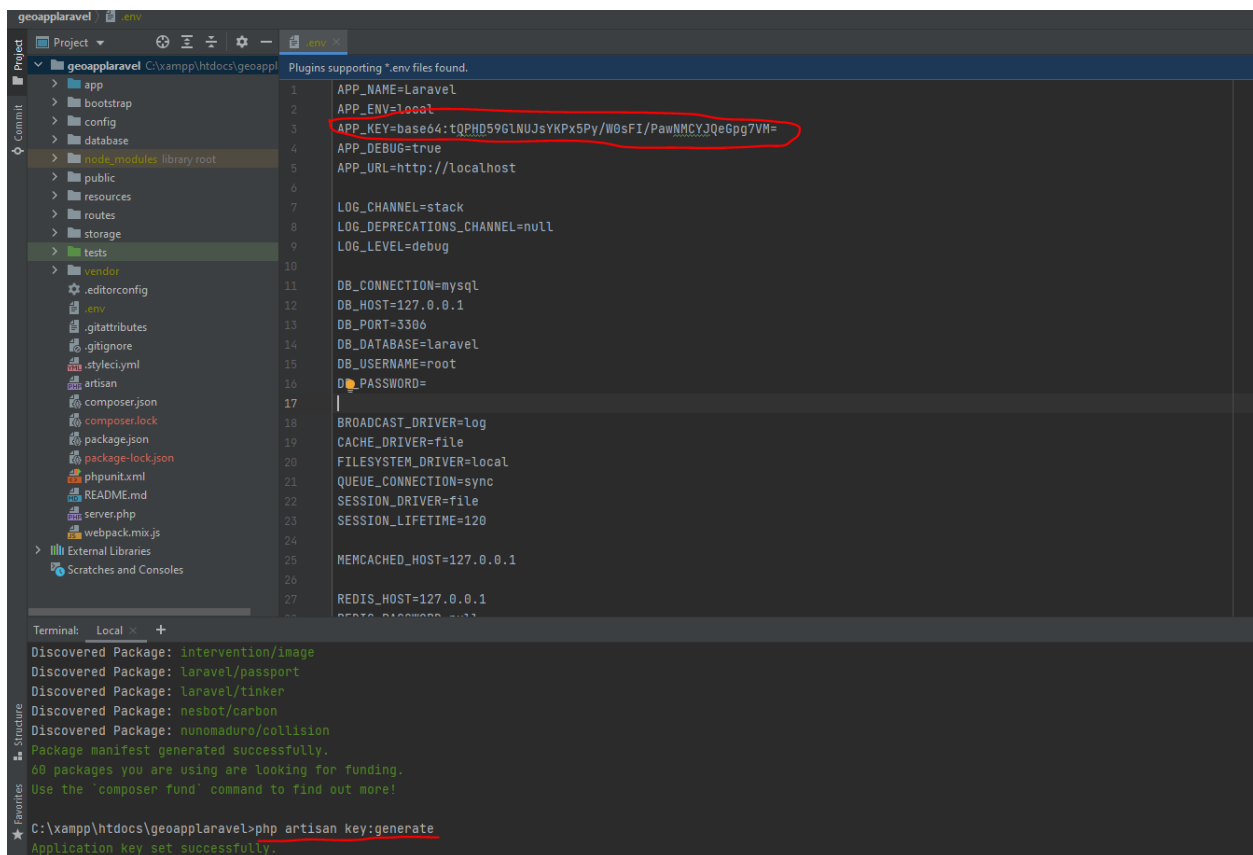
```
C:\xampp\htdocs\geoapplaravel>composer install
Installing dependencies from lock file (including require-dev)
Verifying lock file contents can be installed on current platform.
Warning: The lock file is not up to date with the latest changes in composer.json.
Package operations: 95 installs, 0 updates, 0 removals
- Downloading symfony/polyfill-php72 (v1.12.0)
- Downloading symfony/polyfill-mbstring (v1.12.0)
- Downloading symfony/var-dumper (v4.3.4)
- Downloading symfony/polyfill-ctype (v1.12.0)
- Downloading phpoption/phpooption (1.5.0)
```

3. Crear un archivo llamado **.env** en la raíz del proyecto con la siguiente estructura



```
geoapllaravel .env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=laravel
15 DB_USERNAME=root
16 DB_PASSWORD=
```

4. Ejecutar los comandos **php artisan key:generate**, **php artisan passport:install** y **composer require lcobucci/jwt=3.3.3** (error a).



```
geoapllaravel .env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:tQPH0596LNUJsyKPx5Py/W0sFI/PawNMCYJQeGpg7VM=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=laravel
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DRIVER=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28
```

```
Terminal: Local
Discovered Package: intervention/image
Discovered Package: laravel/passport
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package Manifest generated successfully.
60 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
C:\xampp\htdocs\geoapllaravel>php artisan key:generate
Application key set successfully.
```

5. Creamos una nueva base de datos MySQL en el servidor y escribimos los parámetros de conexión en el archivo .env

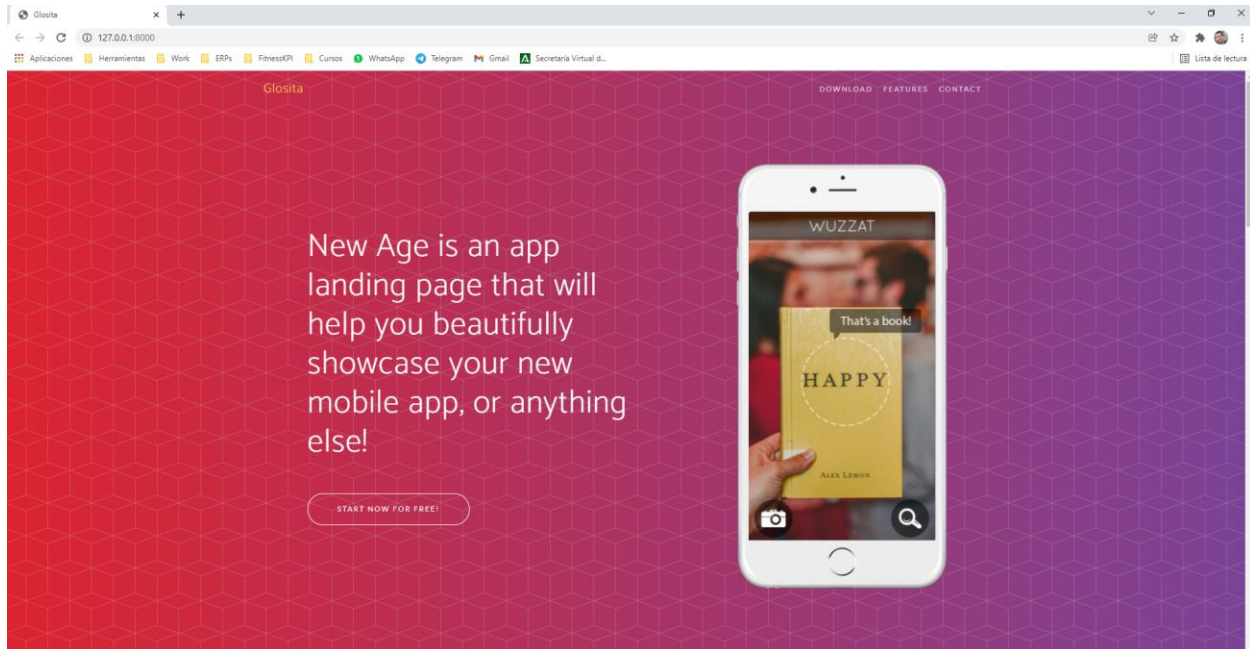


```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=geoapplaravel
DB_USERNAME=root
DB_PASSWORD=
```

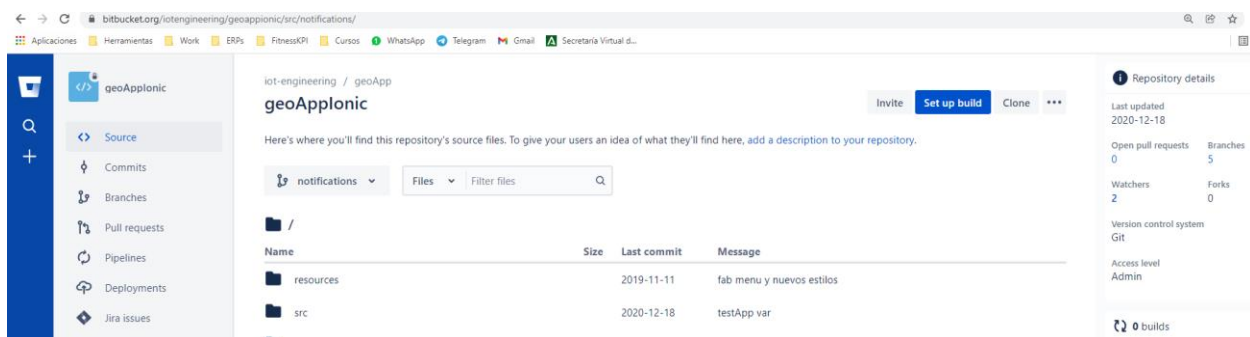
6. Ejecutar el comando **php artisan migrate** (Error b).

```
C:\xampp\htdocs\geoapplaravel>php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (0.02 seconds)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (0.02 seconds)
Migrating: 2016_06_01_000001_create_oauth_auth_codes_table
Migrated: 2016_06_01_000001_create_oauth_auth_codes_table (0.03 seconds)
Migrating: 2016_06_01_000002_create_oauth_access_tokens_table
Migrated: 2016_06_01_000002_create_oauth_access_tokens_table (0.03 seconds)
Migrating: 2016_06_01_000003_create_oauth_refresh_tokens_table
Migrated: 2016_06_01_000003_create_oauth_refresh_tokens_table (0.03 seconds)
```

7. Ejecutar el comando **php artisan db:seed** para crear las categorías y los tags por defecto de la aplicación en la base de datos.
8. Ejecutar el comando **php artisan serve** para comprobar que todo funciona correctamente.



9. Clonar repositorio Ionic - [geoApplonic](#). La rama **notifications** es la más actualizada.



10. Ejecutar los comandos **npm install --g cordova**, **npm install -g @ionic/cli** y **npm install** en la terminal. (Error c)

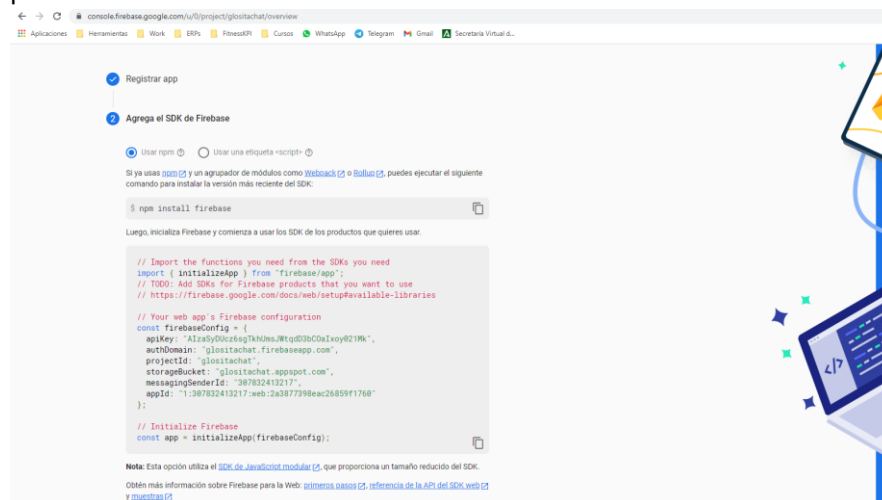
11. Ejecutar el comando **ionic serve**. (Error d)

12. Al lanzar la aplicación en el navegador, nos dará un error de firebase. Esto es porque tenemos que crear una base de datos en firebase para el chat. Nos dirigimos a <https://console.firebase.google.com/>, creamos una nueva cuenta y un proyecto.

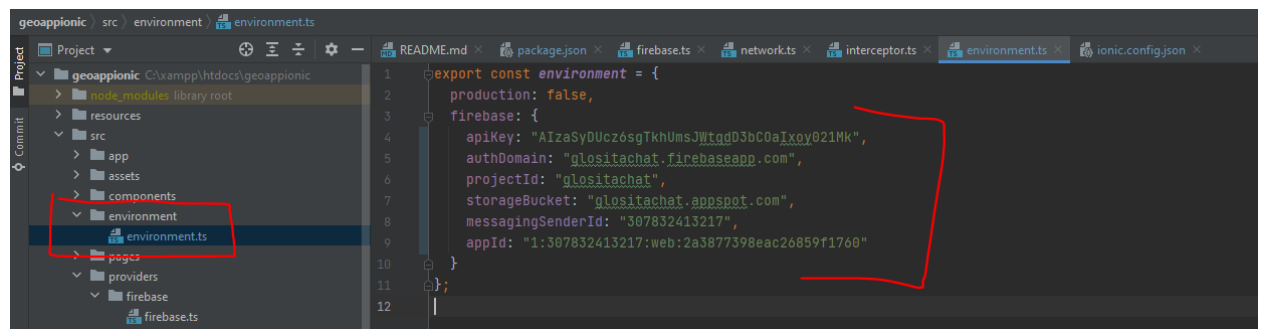
Ejecutamos los comandos **npm install firebase angularfire2 --save** y **npm i angularfire2@5.0.0-rc.7 firebase@4.12.1 --save**

Añado un [tutorial](#) para trabajar con Firebase e Ionic.

13. Conectar proyecto de Firebase con app ionic. Para ello nos dirigimos a la consola de firebase y clicamos en “Agregar Firebase a tu app web”. Tras ponerle nombre a la app, nos dará los parámetros de conexión:

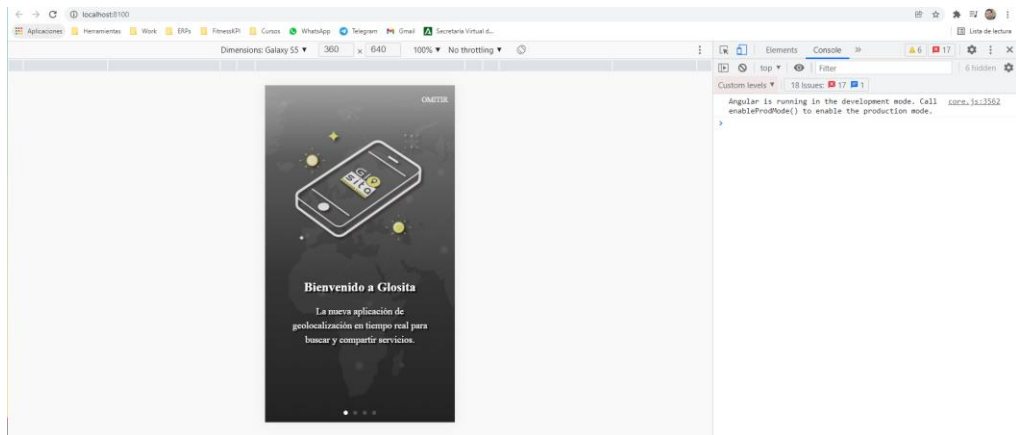


Éstos mismos parámetros serán los que tenemos que conectar en la app de ionic, en el archivo `src/environment/environment.ts`

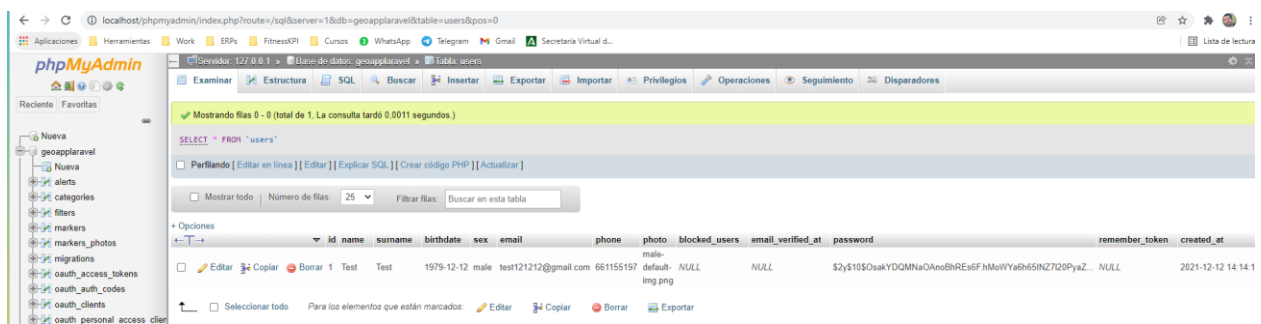


Una vez hecho esto, en la consola de Firebase le damos a guardar y nos dirigimos a Realtime Database para crear la base de datos.

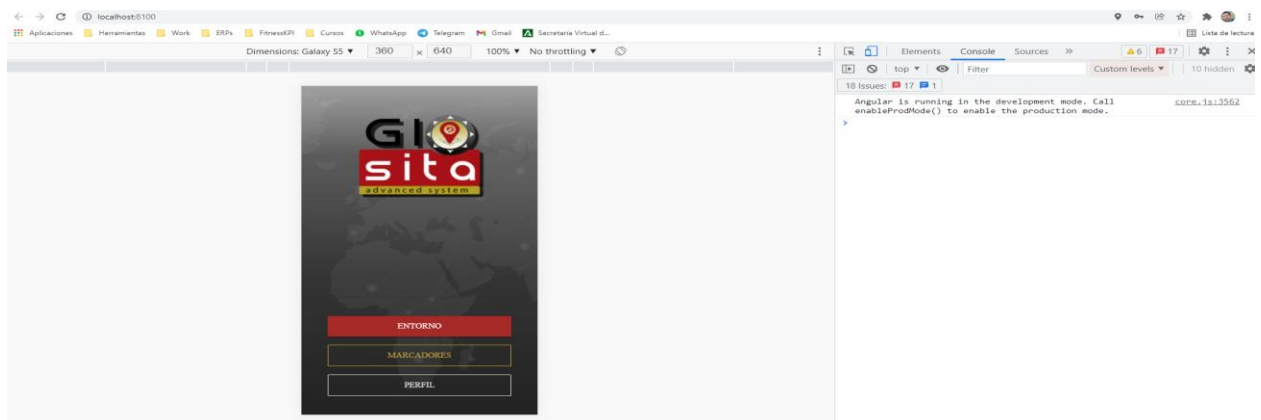
14. Volvemos a lanzar la aplicación con el comando **ionic serve** y ya la tendremos montada en el navegador.

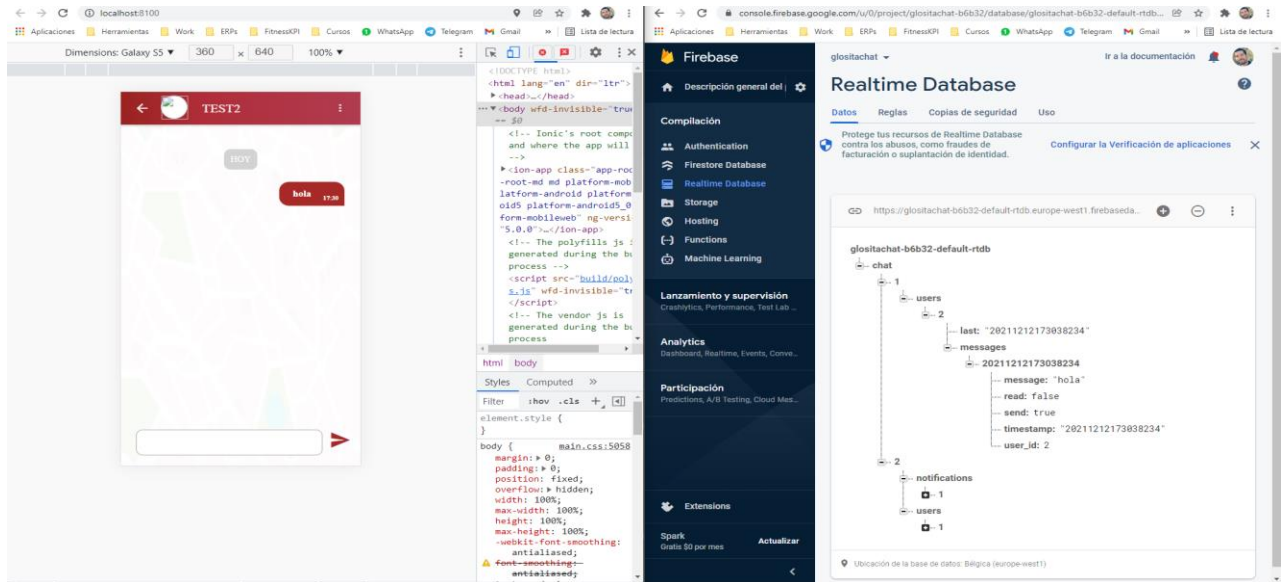


15. Creamos un usuario desde el apartado Registrarse. (Error e).



16. Si no hay ningún servicio de email creado para que el usuario termine el registro desde la web, tendremos que acudir a la base de datos y establecer el parámetro **confirmed** de la tabla **users** a 1. De esa manera nos saltamos el envío del email y el usuario podrá loguear sin problema.





17. Para la implementación en Android será necesario instalar Android Studio y seguir los pasos indicados en [esta guía](#).

Posibles errores durante la instalación:

- a) Undefined index: name in PackageManifest.php. Eliminados el archivo composer.lock y la carpeta vendor, luego ejecutamos un composer install
- b) Si la migración nos da error por acceso denegado, habrá que ejecutar los comandos php artisan config:cache y php artisan config:clear para que el archivo .env se guarde correctamente.
- c) npm ERR! code ELIFECYCLE
npm ERR! errno 1
npm ERR! grpc@1.23.3 install: `node-pre-gyp install --fallback-to-build --library=static_library`
npm ERR! Exit status 1

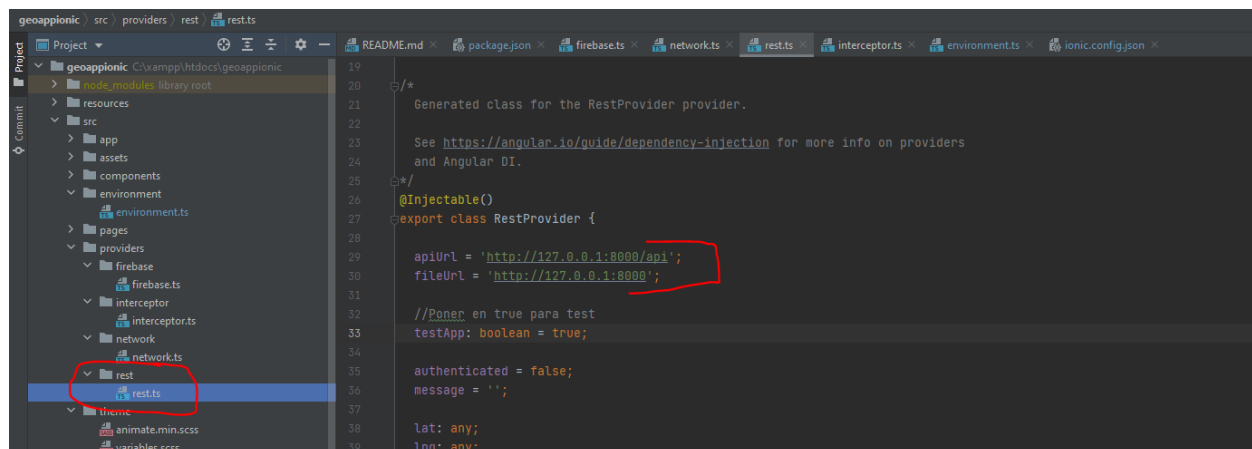
Eliminar la carpeta node_modules, el archivo package-lock.json, ejecutar un npm cache clean --force, npm install y npm i firebase@latest.

Aunque lo ideal es trabajar en un entorno Linux.

- d) [ERROR] ionic-app-scripts has unexpectedly closed (exit code 1)

Ejecutar los comandos "npm uninstall node-sass" "npm install node-sass" "npm rebuild node-sass"

- e) Si no podemos registrar un usuario, es posible que la app de ionic no esté buscando la api de laravel en la url correcta. Para ello, nos dirigimos al archivo src/rest/rest.ts y modificamos los parámetros apiUrl y fileUrl para apuntar al servidor donde esté instalada la aplicación de laravel. También habrá que modificar el parámetro apiUrl del archivo src/app/app.component.ts



Otro error es que no hay ningún servidor de email configurado, por lo que en la aplicación de Laravel habrá que comentar la línea 126 del archivo AuthController.php hasta que se cree un nuevo servicio de email y se vincule en el archivo .env

