

2019

**APLICACIÓN DE APERTURA Y CONTROL DE PUERTA DE ENTRADA**

**MANUAL TÉCNICO**

**PROGRAMACIóN DE SISTEMAS TELEMÁTICOS**

**profesora:**

Msig. Adriana Collaguazo Jaramillo

**PARALELO:** 1 **GRUPO:** 5

**INTEGRANTES:**

Luis Armijos Cevallos

Lorena Campoverde Vilela

Henry Fernández Bedoya

Jair Marcillo Acosta

Contenido

[**Resumen ejecutivo** 3](#_Toc17179598)

[**Introducción** 3](#_Toc17179599)

[**Descripción del problema** 4](#_Toc17179600)

[**Objetivos del Proyecto** 4](#_Toc17179601)

[**Funcionamiento de la Solución** 5](#_Toc17179602)

[**Propuesta Solución del Problema** 5](#_Toc17179603)

[**Recursos de hardware y de software** 6](#_Toc17179604)

[**Implementación del proyecto** 8](#_Toc17179605)

[**Diagramas del Proyecto** 11](#_Toc17179606)

[**Diagrama de diseño** 11](#_Toc17179607)

[**Diagrama de Despliegue** 14](#_Toc17179608)

[**Diagrama de Gantt** 15](#_Toc17179609)

[**Diagrama del Circuito** 16](#_Toc17179610)

[**Diagrama de clases** 18](#_Toc17179611)

[**Diagrama del modelo Entidad Relación** 18](#_Toc17179612)

[**Descripción de los campos** 18](#_Toc17179613)

[**Códigos Fuente del Aplicativo** 23](#_Toc17179614)

[**Aplicativos en Android** 23](#_Toc17179615)

[**Aplicación de Apertura y Control de Puerta de Entrada – Administrador** 23](#_Toc17179616)

[**Aplicación de Apertura y Control de Puerta de Entrada – Residente** 31](#_Toc17179617)

[**Código en IDE Arduino para ESP8266 en NodeMCU** 39](#_Toc17179618)

[**Análisis de presupuesto** 42](#_Toc17179619)

[**Conclusiones** 43](#_Toc17179620)

[**Referencias bibliográficas** 43](#_Toc17179621)

# **Resumen ejecutivo**

Los sistemas de apertura convencionales RIFD presentes en un gran número de urbanizaciones en la ciudad y en el país presentan múltiples desventajas como mal uso o mala implementación por parte de los residentes, los cuales tienen la responsabilidad de manejar esta tarjeta para poder acceder, sin embargo las urbanizaciones no pueden tener un control del mismo, cuando se presentan estos inconvenientes por parte de los residentes.

Ofrecemos un sistema de aplicativos móviles los cuales permiten la apertura del mecanismo de entrada de la puerta de acceso de la urbanización y control a través de una base de datos remota desde internet, en el cual los administradores de la localidad podrán manejar y permitir el ingreso de todos los residentes y sectorizar cada una de las villas a las cuales pertenecen y a su vez los residentes tener un ingreso más seguro y responsable del mismo a la urbanización, cuidando la integridad y la seguridad de todas las personas.

Basados en la economía del país y para poder ser implementado en cualquier sector de la Ciudad de Guayaquil buscamos un mecanismo de seguridad eficaz, seguro y económico para todas las empresas de inmobiliaria que deseen implementar este mecanismo. Buscamos un valor de inversión bajo y de alcance para todos los que deseen emprender este nuevo sistema y referencial debido a que nuestro control de apertura e ingreso a las urbanizaciones es por medio de bases de datos remotas en internet. Y solo se tomaría el valor de los aplicativos móviles y los equipos implementados en el mismo.

Somos un grupo de estudiantes politécnicos de la carrera de Mecatrónica, buscando la innovación y la apertura al mercado para crear sostenibilidad e innovación a problemas sociales actuales. Y estamos convencidos que nuestro sistema es rentable y funcional para someter a este problema actual en la sociedad.

# **Introducción**

En esta guía se detalla todo el proceso de realización de una propuesta de solución a un problema convencional presente en la sociedad, específicamente en las Urbanizaciones o Conjuntos Residenciales. En el luego de plantear la solución al problema se realizan diferentes diagramas para especificar las funcionalidades de esta solución además de expresar todo el funcionamiento y la implementación de todo el sistema que se presenta en el cual se pueden notar diferentes métodos implementados aprendidos en el curso de Programación de Sistemas Telemáticos, conocimientos como:

Acceso a Internet: El cual nos permite tener multiples conexiones con los aplicativos, sensores, e incluso con la base de datos para mantener control especifico de lo que el aplicativo realice.

Sincronizacion con google: Google nos ofrece multiples funcionalidades de diseño, una de esas es la de google maps, la cual permite generar mapas de google y manejar estos mapas según la conveniencia del programador.

Sincronizacion con base de Datos: Las bases de datos ofrece multiples beneficies de trabajo para el desarrollo remoto, y el sincronizarlos con el aplicativo ofrece seguridad y eficacia para el desarrollador. Ejecución de Mecanismos a través de sensores y módulos gestionados remotamente. Diseño y construcción de aplicativos móviles eficaces y seguros.

# **Descripción del problema**

Con el pasar de los años, la tecnología ha ido evolucionando, resolviendo un sinnúmero de problemas alrededor del mundo y logrando que las actividades diarias y servicios sean más productivos y eficientes. Por ejemplo, la seguridad en las inmobiliarias ha ido mejorando gracias al desarrollo de nuevas tecnologías, siempre buscando la manera de mejorar el servicio y que los costos no sean tan elevados. Sin embargo, los sistemas que se usan comúnmente para el control de la entrada de vehículos en las inmobiliarias, aunque estén ligadas con la tecnología, estas no son lo suficientemente eficientes, generando consecuencias a largo plazo, con costos un poco elevados.

En el caso de aperturas convencionales con rfid, por ejemplo, que mediante una llave electrónica (tarjeta), permite que el usuario entre a la urbanización, estas no son totalmente eficientes, ya que no son tan seguras para ciertas situaciones. Una de estas situaciones es, que estas llaves electrónicas pueden fácilmente ser clonadas por otras personas, algo que es muy peligroso ya que puede haber el caso que una persona externa a la urbanización entre a esta, y aunque puede mejorarse la tarjeta para que esto no suceda, esta mejora requiere de recursos económicos, sin considerar que así como las maneras de mejorar la seguridad aumentan, también la forma de robar, por lo que no es una solución a largo plazo. Otra situación, es que estas tarjetas con el tiempo se desgastan por lo que necesitan ser cambiadas constantemente, y sin considerar que puede darse el caso de pérdida o robo de estas, por lo que aunque este servicio nos ofrece beneficios, de igual forma posee sus desventajas, algo que siempre se busca que sean mínimas.

Otro servicio usado para el mismo fin, es aquel que usa TAGS, sin embargo de la misma forma que el anterior posee sus desventajas, ya que puede darse el caso en que el vehículo que tiene el Tag sea robado y como consecuencia el ladrón entre a la urbanización, y así otras situaciones que no proveen seguridad totalmente. Por lo tanto, la necesidad de implementar nuevos servicios que desplacen a los servicios mencionados anteriormente, es una necesidad que en la actualidad se requiere.

# **Objetivos del Proyecto**

**Objetivo General**

* Diseñar una aplicación móvil para implementar un sistema telemático de apertura de puerta para entrada, de automóviles de una urbanización, como alternativa a los sistemas convencionales mediante el uso de Arduino, Android Studio y phpMyAdmin.

**Objetivos Específicos**

* Implementar un sistema telemático eficiente e innovador para el control de apertura de puerta de entrada de una urbanización mediante la aplicación de los conocimientos adquiridos a lo largo de la materia, acerca de la creación y manejo de aplicaciones móviles.
* Usar una base de datos remota para el control de los residentes que habitan y entran en una urbanización con ayuda de un servidor Web remoto llamado Remote MySql
* Identificar problemas de la sociedad actual y plantear una propuesta solución integradora en base a los conocimientos adquiridos en el curso de Programación de Sistemas Telemáticos, como es el uso, diseño y construcción de aplicativos móviles y sus funcionalidades.

# **Funcionamiento de la Solución**

Para el sistema de apertura y control de puerta de entrada se crean dos aplicativos móviles en Android Studio, uno para que maneje los administradores de las urbanizaciones y otro para los residentes, esto se debe a que los residentes, en un aplicativo, deben estar delimitados única y exclusivamente en abrir e ingresar por la puerta de entrada ingresando sus credenciales y que no tenga ningún contacto con los administradores, o con las opciones que tienen los administradores, esto permite la integridad y seguridad del aplicativo para las urbanizaciones.

En cambio en el aplicativo proporcionado a los administradores de la urbanización podrán manejar el ingreso de los residentes registrados en la misma; en el podrán agregar nuevos administradores y a su vez nuevos residentes, agregar villas y sectorizar las villas a cada uno de los residentes para habilitarlos. Además de visualizar y controlar los ingresos de los residentes en el día, todas estas funcionalidades en los dos aplicativos se manejaran en conjunto con la base de datos remota.

En el aplicativo de los residentes, le aparecerá para ingresar sus credenciales de usuario y habilitar para ingresar su clave de acceso a la residencia que de ser correcta podrá ingresar; una vez que el aplicativo verifica a través de la base de datos sus credenciales correctas y esta aceptado para ingresar, se ejecuta la ventana de acceso y una bienvenida junto con funcionalidades de diseño del aplicativo, también se comunica con la base de datos y escribe un nuevo registro de entrada para que el administrador, desde su aplicativo, pueda manejar esto a su conveniencia, además el aplicativo envía al módulo de WIFI ESP8266 el comando de aceptación y manda a ejecutar el motor instalada en la entrada junto con el sensor infrarrojo para evitar que se cierre mientras se encuentra todavía en la puerta.

# **Propuesta Solución del Problema**

Debido a los problemas que poseen los sistemas de apertura de puerta de urbanizaciones convencionales como se mencionó anteriormente, la propuesta del proyecto consiste en un sistema más eficiente con gastos reducidos.

Para esto, se propone un sistema telemático usando una aplicación móvil. Esta propuesta, es muy favorable en comparación con las convencionales, ya que, no depende de tarjetas o de componentes que el usuario fácilmente pueda perder o puedan ser robados o se desgasten, basta con digitar valores a la aplicación, para que si estos valores son los adecuados y la persona que esté usando la aplicación permanezca en la urbanización, es decir que viva en ella, con estos dos requisitos la puerta se abra, y que mediante un sensor infrarrojo, la puerta no se cierre hasta que el usuario entre totalmente, es decir, que se usará base de datos para almacenar toda la información necesaria, como la información de los residentes y en la villa y manzana que vivan.

Cabe destacar, que la aplicación aunque pueda ser usada en cualquier parte, también viene implementado con dígitos que varían constantemente, es decir que el usuario aparte de digitar la clave para que la puerta se abra, también debe digitar otros dígitos que le aparecerán por pantalla, dígitos de forma aleatoria, para asegurar de que el usuario realmente se encentre afuera de la urbanización y como consecuencia la puerta se le abra.

Como se aprecia, es una solución innovadora y muy amigable ya que en la actualidad lo que más se usan son aplicaciones móviles para cualquier servicio, además de que por lo general una persona posee celular, entonces los usuarios se sentirán a gusto con la aplicación, considerando que son más seguros ya que todo depende netamente de la clave, algo que difícilmente puede ser robado por otras personas.

# **Recursos de hardware y de software**

|  |  |  |
| --- | --- | --- |
| RECURSOS | | |
| HARDWARE | ***Módulo Wifi (Node MCU junto con ESP8266)***  Resultado de imagen para nodemcu | Es una placa de desarrollo en el ESP12E, además este módulo incluye el firmware WiFi ESP8266. |
| ***Arduino Uno*** | Se trata de una placa electrónica, el cual incorpora un micro controlador, en donde se almacena el código.  Cuenta con entradas/salidas digitales y analógicas. |
| ***Sensor Infrarrojo IR FC-51***  Resultado de imagen para sensor infrarrojo arduino | Este dispositivo envía señales a un controlador cuando detecta un obstáculo delante del LED infrarrojo y fototransistor que tiene incorporado, la distancia máxima de detección es de 4-5cm. |
| ***Motor de paso y driver***  Resultado de imagen para stepper motor arduino driver | Motor eléctrico DC que puede ser controlado con la ayuda del driver ULN2003, este permite además alimentarlo con una fuente externa. |
| ***Pantalla LCD con I2C***  Resultado de imagen para pantalla lcd con i2c | El circuito integrado I2C permite controlar la pantalla LCD con dos pines analógicos. |
| SOFTWARE | ***Arduino IDE***  Imagen relacionada | Arduino es una plataforma de desarrollo, en la cual se puede realizar la programación para distintas placas como Arduino uno, Arduino nano, entre otros |
| ***Android Studio***  Resultado de imagen para android studio logo | Android Studio es un entorno de desarrollo integrado para aplicaciones específicamente de Android, y en este podemos utilizar el lenguaje de programación JAVA. Además, está basado en Intellij IDEA de JetBrains. |
| ***Marvel App***  Resultado de imagen para marvel app | Marvel App es una herramienta, la cual permite diseñar prototipos digitales de aplicaciones de una forma dinámica. |

# **Implementación del proyecto**

En el implementación del proyecto se debe seguir un correcto orden de conexiones y de manejo del mismo, debido a existen procesos en los cuales no se pueden llevar a cabo si no se finalizan otros, o no pueden trabajarse simultáneamente, sino uno seguido del otro, esto se detalla de igual forma y de manera esquemática en el diagrama de Gantt de la sección de diagramas de esta guía de usuario.

Creación de la base de datos: Para empezar con el control del sistema de apertura y de la urbanización se debe definir la base de datos que se va a manejar. Para esto se escoge la página [www.remotemysql.com](http://www.remotemysql.com) la cual nos permite crear base de datos remotas y de manera gratuita, si el usuario desea una base de datos más protegida puede solicitarla, recordemos que la seguridad y lo protegido de la base de datos la da según los proveedores que escojamos y cuanto estemos dispuestos a pagar por ello.

Se escoge una base de datos remota debido a que se trabajan en dos aplicativos simultáneamente dentro de la urbanización, y las bases de datos locales solo podrán ser accedidas a través de este aplicativo, por tal razón se escoge una base de datos remota para que ambos aplicativos se intercomuniquen a través de ella. La creación de esta base de datos remota se encuentra en la sección de descripción de campos y códigos fuentes.

**Aplicativos del sistema**

Se realizan las ventanas y de más funcionalidades de los aplicativos, se aplican diferentes acciones de diseño para ambos aplicativos, como el splash de las aplicaciones y las funciones de cada uno de los botones de los Activity en Android Studio, se sincroniza además con los permisos correspondientes del aplicativo, como los permisos de internet, acceso a red, google y de más.

**Conexión del aplicativo con la base de datos**

Finalizada las funcionalidades de diseño de los aplicativos se procede a realizar la conexión de los aplicativos con la base de datos remota, debido a que ambos intervienen de una forma u otra con el mismo; en el aplicativo de Administradores, se lee en la base de datos todas las credenciales y los registros de ingresos a la residencia, además de que se escribe cuando se crean nuevos administradores, residentes, villas y sectorización.

Mientras que en el Aplicativo exclusivo para los residentes, lee las credenciales de los residentes y las claves de acceso; si estas son correctas procede a ingresar a la residencia y a su vez escribe en la base de datos en la tabla de registros de entradas, por tal razón se conectan ambos a la base de datos e interactúan de manera dinámica entre ellos. Código de las conexiones a la base de datos se encuentra en la sección de códigos de esta guía.

**Conexión del aplicativo con el sensor ESP8266**

Para poder ejecutar el código y el mecanismo de apertura de la puerta de entrada se está implementando el sensor ESP8266 en el módulo NodeMCU, en conjunto forman un micro controlador que proporciona múltiples funcionalidades a través de WIFI; este módulo es programado con el IDE de Arduino, este código se encuentra en la sección de códigos, el cual lee los comandos que le envíen a la dirección IP que se le proporciona el servidor al que esté conectado y ejecuta lo que deseemos; en resumen, luego de validar las credenciales del residente, el aplicativo le envía a la dirección IP del sensor lo que desea que ejecute, en nuestro caso, que ejecute el motor para accionar la puerta de entrada y el sensor. Es por ello que en el aplicativo, se le envía a través de internet, los comandos al sensor.

**Funcionamiento del mecanismo de apertura ESP8266**

Luego de realizar la conexión del sensor con el aplicativo, ya podríamos programar lo que deseemos que el ESP8266 mande a ejecutar, como el motor y el infrarrojo, para esto se debe crear el código de ejecución en el IDE de Arduino, en el cual le asignamos a que red se va a conectar por WIFI y esta su vez le da una IP para poder ejecutar según esta IP, en ella lee el comando y ejecuta el motor Stepper y a través de un sistema mecánico de engranaje y cremallera, accionar la puerta de entrada y abrirla. Luego de su apertura total, el residente podrá ingresar y la puerta seguirá activada hasta que ingrese, y no se cerrara si detecta que el vehículo sigue en esta zona, esto lo realiza a través del sensor infrarrojo incorporado.

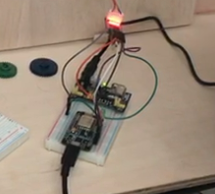


Ilustración 1: Circuito con módulo wifi

**Construcción del Prototipo de presentación**

Para la construcción de toma en cuenta el mecanismo mencionado anteriormente, de engranaje y cremallera para accionar la puerta; en esta puerta se encuentra incorporado el mecanismo de cremallera en la parte superior, y junto a él se encuentra el motor Stepper que tiene un engranaje incorporado en el eje para poder realizar el mecanismo de traslado del movimiento rotacional en lineal.



Ilustración 2: Construcción del prototipo

**Funcionalidades de los aplicativos**

Una vez que se encuentras las conexiones realizadas independiente, se procede a realizar la activación del mecanismo junto con el sensor y el aplicativo que valida a través de la base de datos, para poder accionar las funcionalidades que tienen los residentes, para esto se debe someter a múltiples validaciones de ingreso al sistema, y que permita que el usuario ingrese.

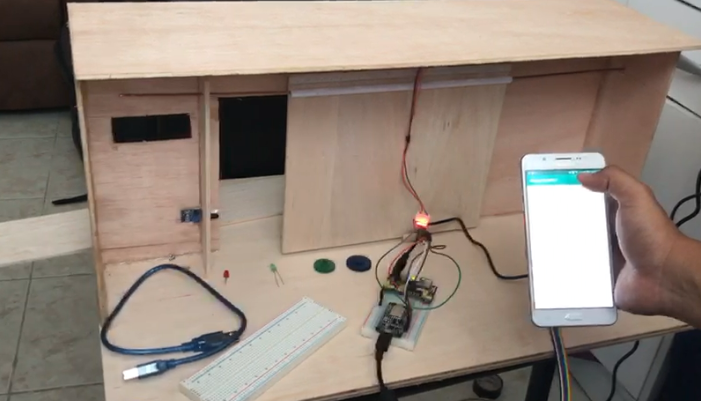


Ilustración 3: Prueba de funcionamiento

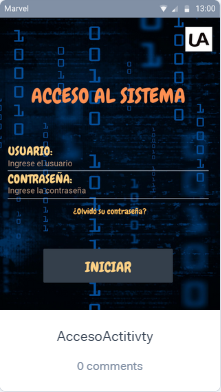
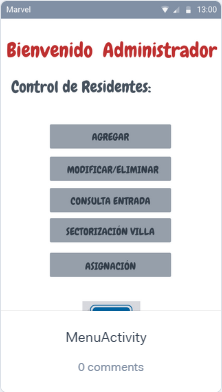
# **Diagramas del Proyecto**

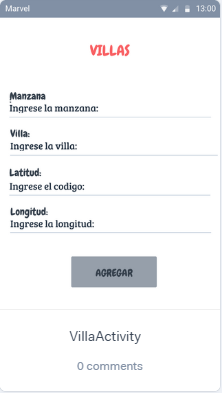
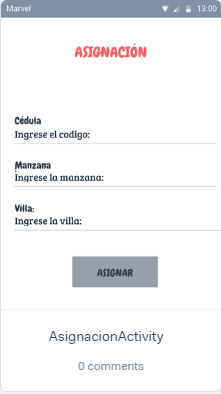
## **Diagrama de diseño**

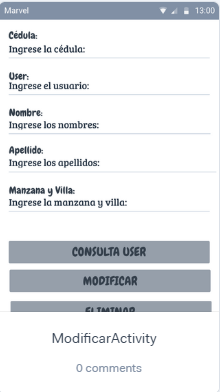
Los diagramas de diseño de ambos aplicativos móviles fueron realizados bajo la herramienta de marvelapp que nos permite realizar los diseños previos todas las ventanas que vamos a ejecutar.

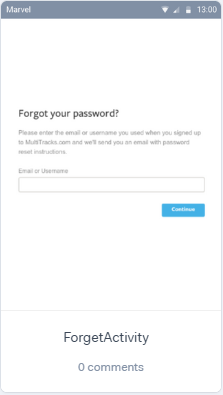
**Aplicativo Administrador**

<https://marvelapp.com/643j40g>

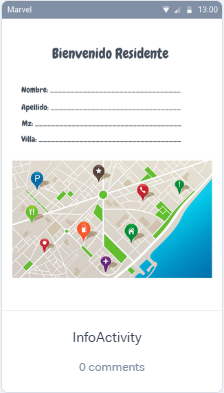
 



**Aplicativo Residentes**

<https://marvelapp.com/1h596116>



## **Diagrama de Despliegue**

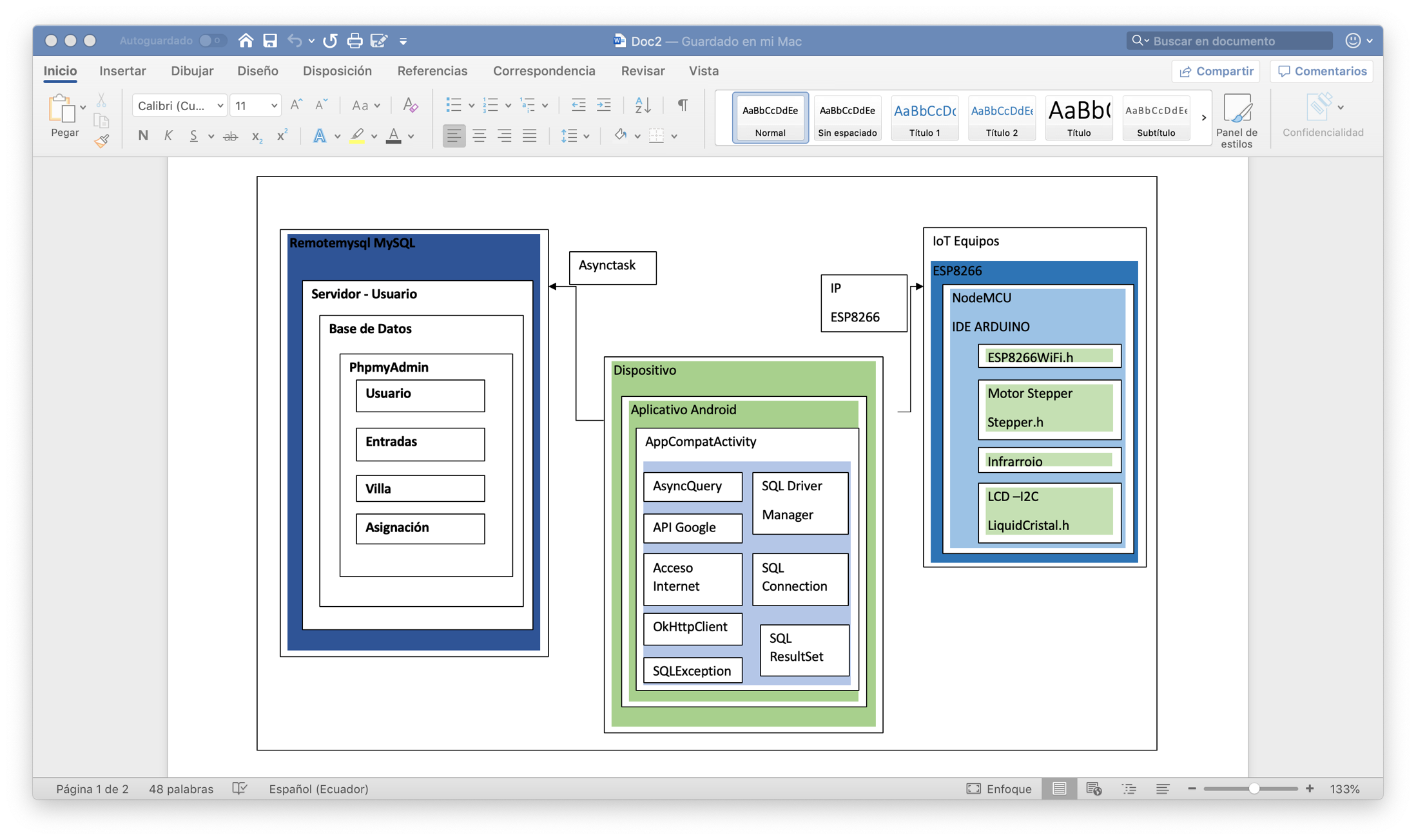


Ilustración 4: Diagrama de despliegue

## **Diagrama de Gantt**

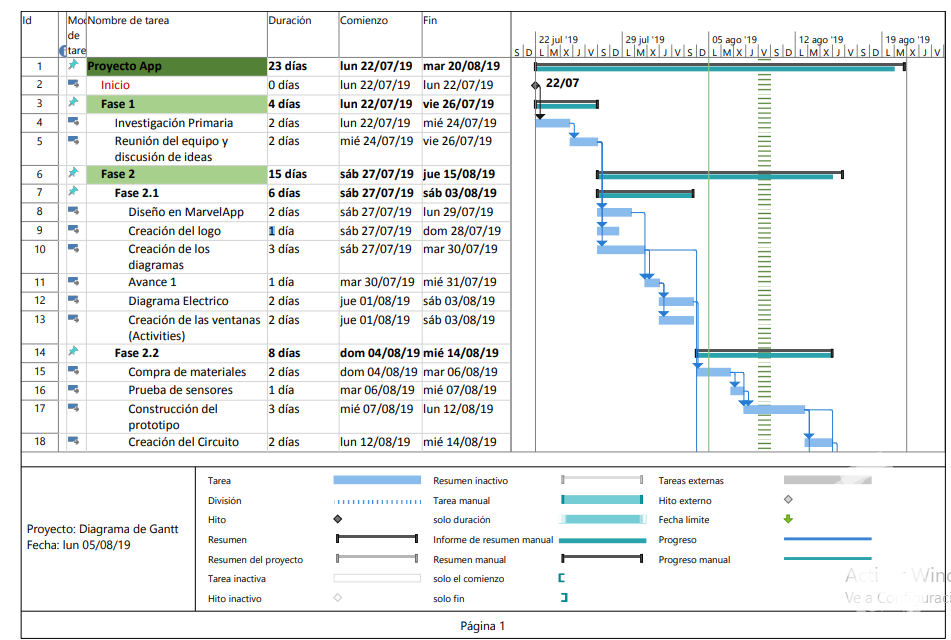


Ilustración 5: Diagrama de Gantt parte 1

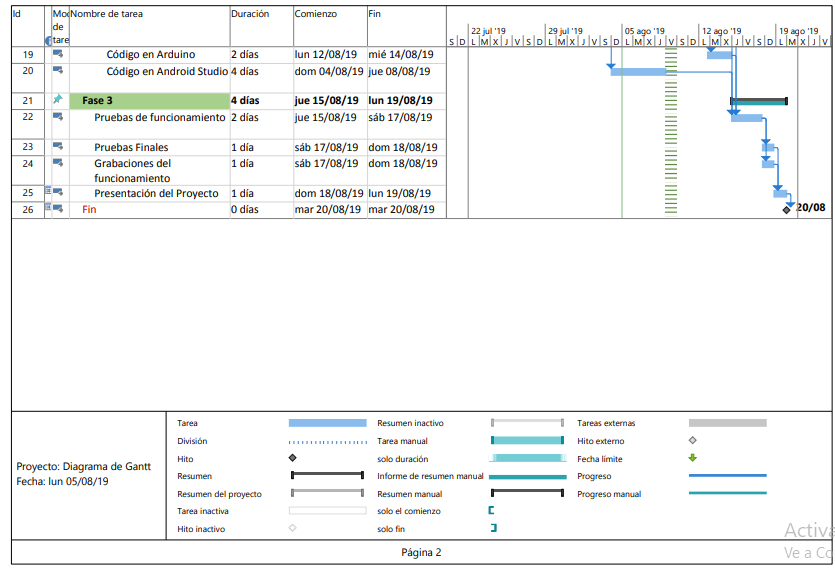


Ilustración 6: Diagrama de Gantt parte 2

## **Diagrama del Circuito**

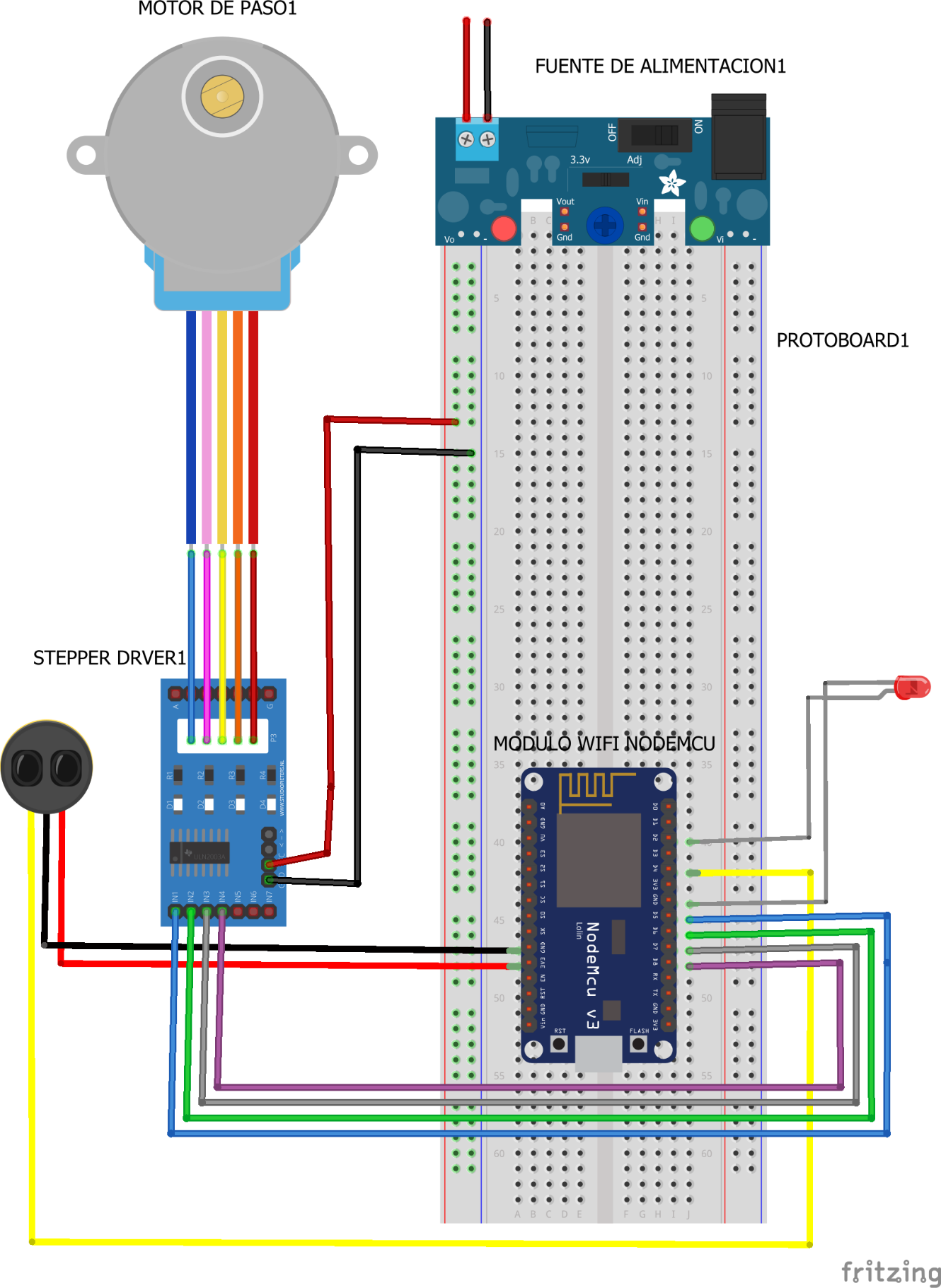


Ilustración 7: Diagrama del Circuito

## **Diagrama de clases**

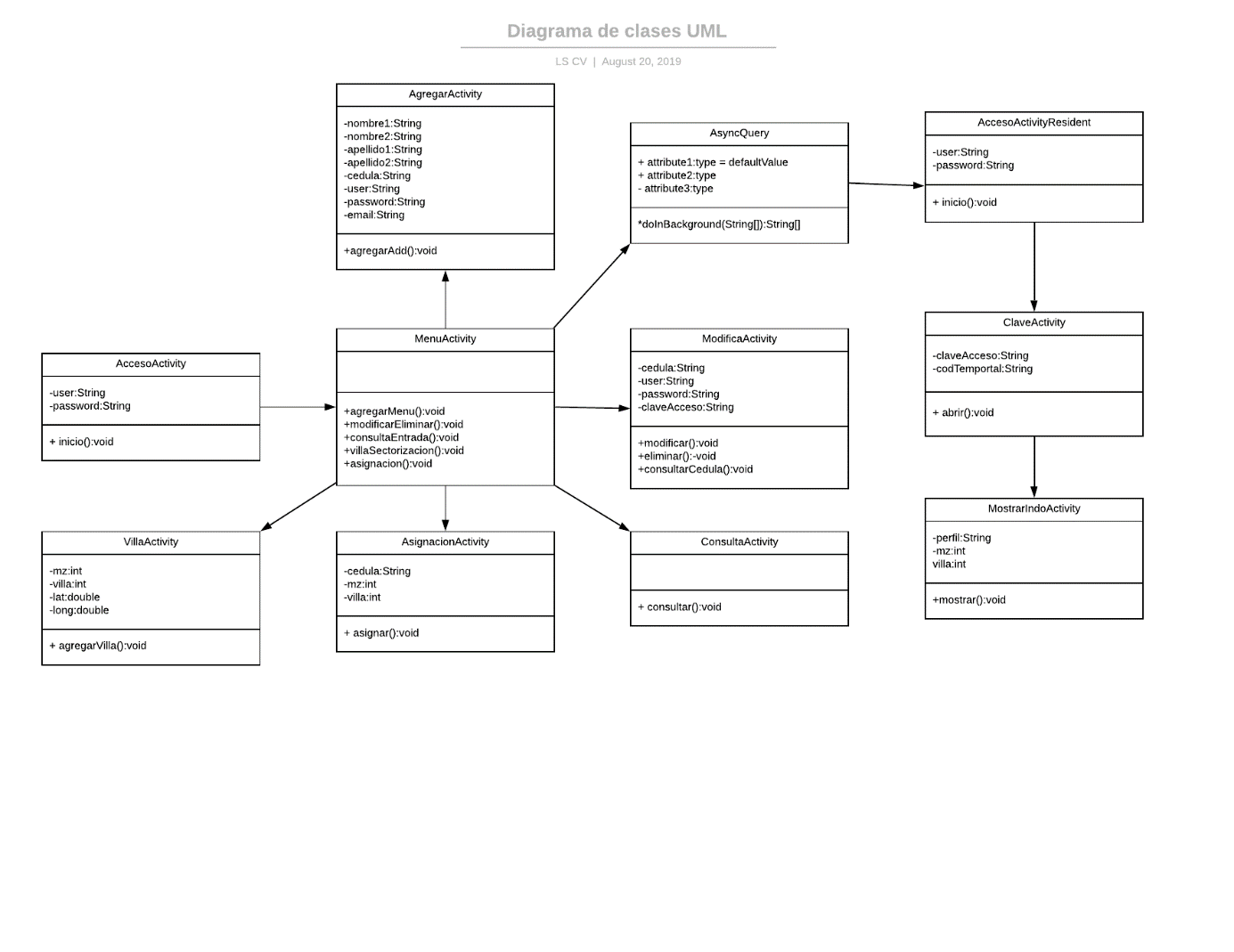


Ilustración 8: Diagrama de clases

## **Diagrama del modelo Entidad Relación**

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Ilustración 9: Modelo Entidad-Relación

# **Descripción de los campos**

Nuestra base de datos remota fue creada en la plataforma remotemysql.com, esta plataforma nos permite acceder a la base de datos y trabajarla en MySQL a través de la página de phpMyAdmin en la cual se puede sectorizar y visualizar los valores ingresados en la base de datos de manera dinámica y eficiente.

Para crearla Ingresamos a la página [www.remotemysql.com](http://www.remotemysql.com) y procedemos a ingresar con nuestras credenciales, si no se tiene una cuenta, se crea una.

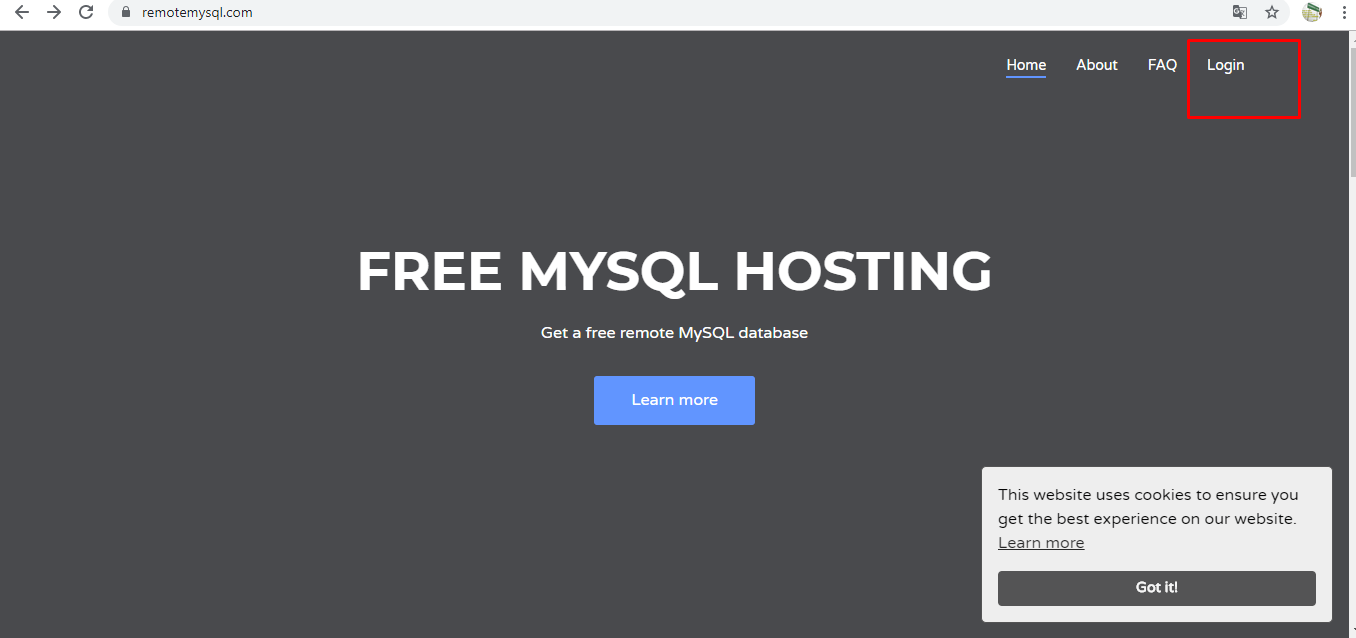


Ilustración 10: Vista inicial de remoteMySQL

Ingresamos nuestras credenciales de usuario de la página para poder acceder a sus funciones.

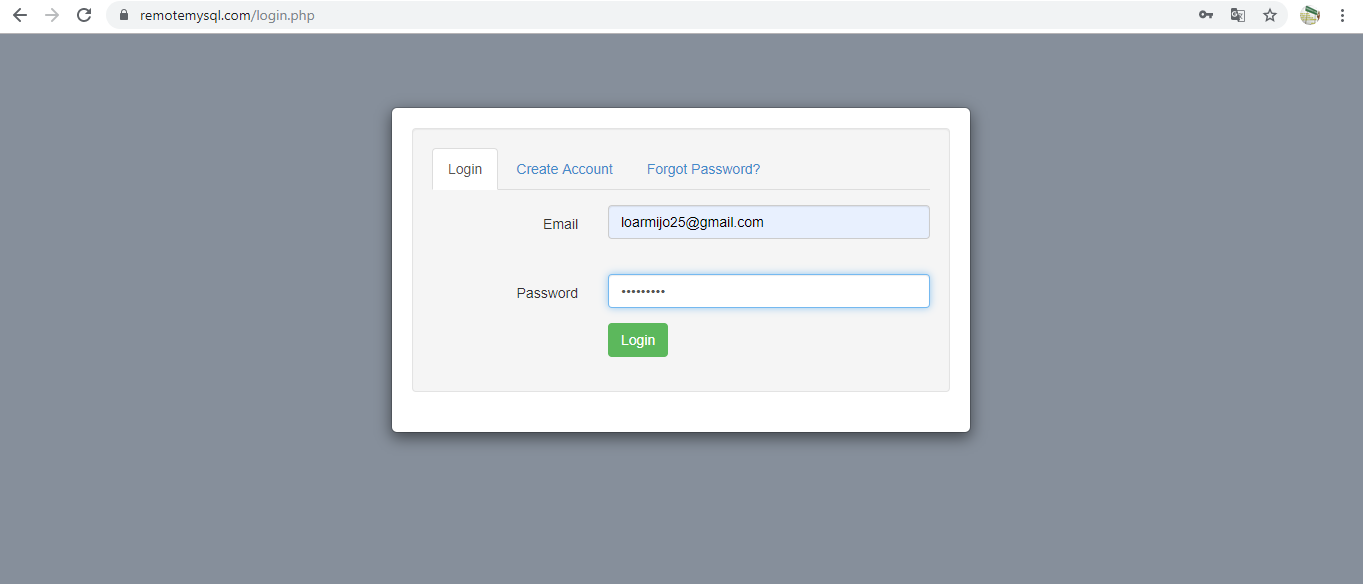


Ilustración 11: Ingreso a plataforma

Una vez ingresados procedemos a crear una nueva base de datos en el botón de la siguiente imagen en la cual se le asigna un nombre y una clave para poder acceder al mismo, guardar estos campos.

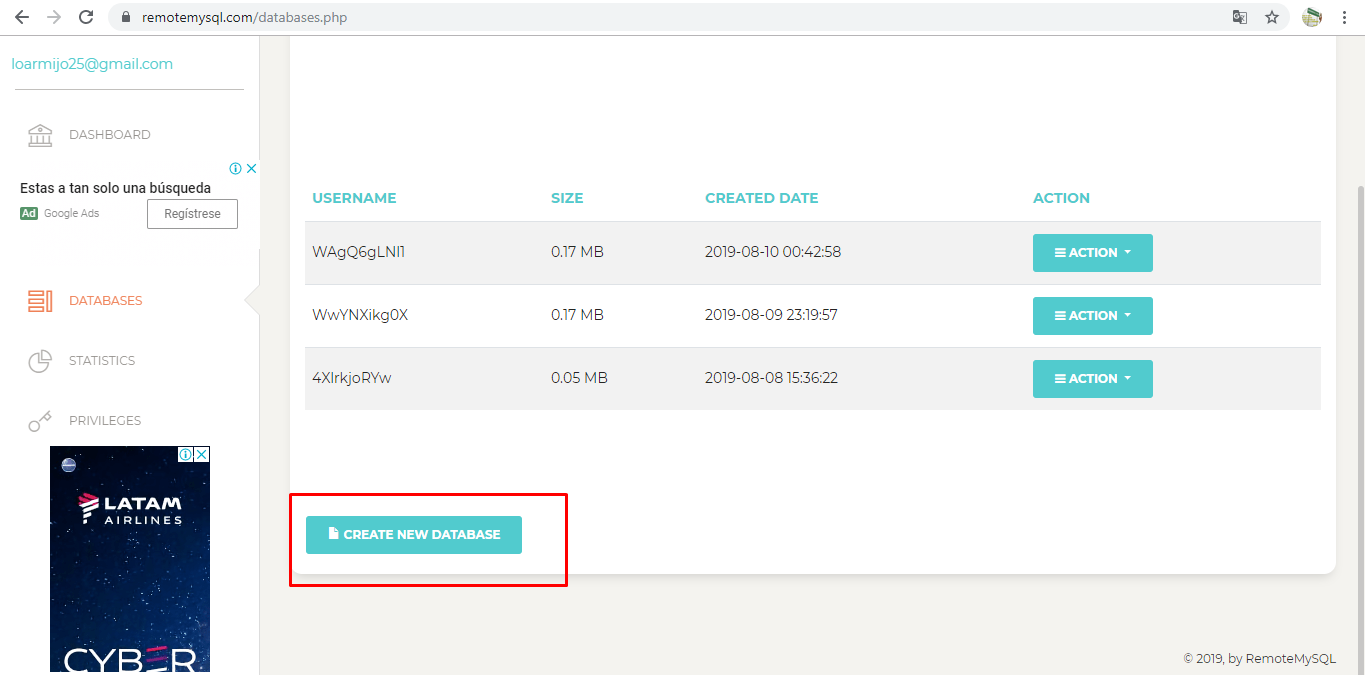


Ilustración 12: Creación de base de datos remota

Una vez creada la base de datos, se procede a entrar al phpMyAdmin con el usuario y clave de la base de datos creada.

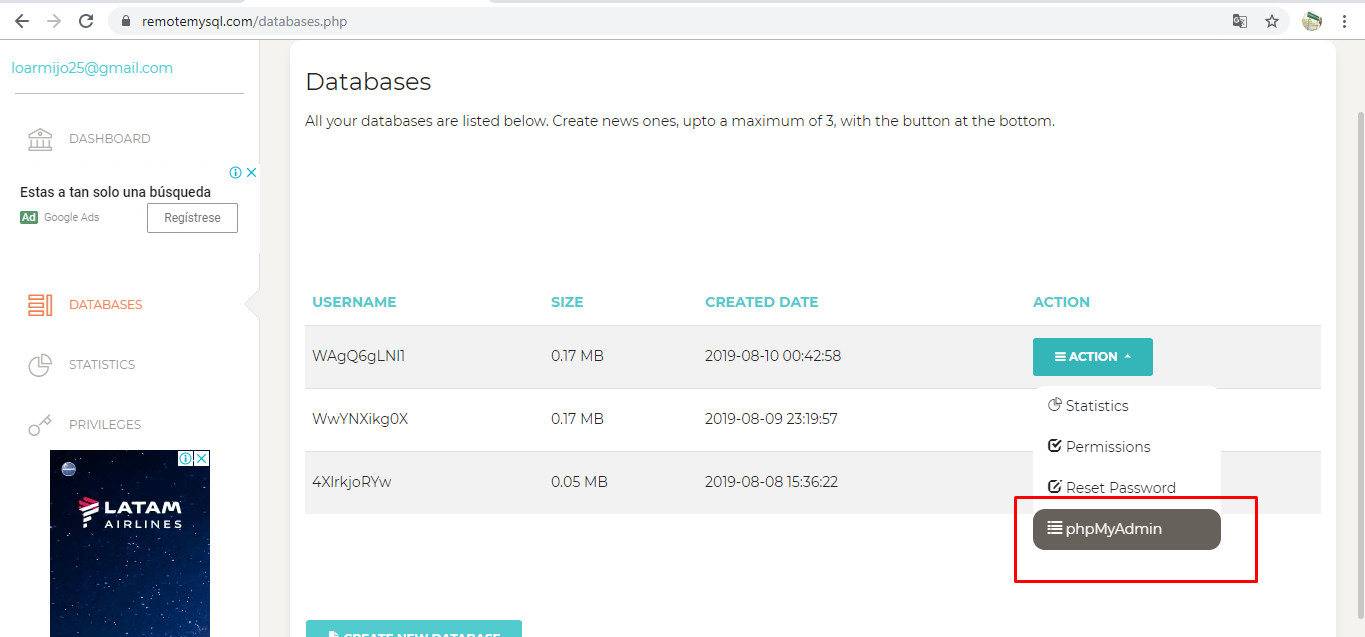


Ilustración 13: Selección de opción phpMyAdmin

Luego se procede a crear todas las tablas de la base de datos.

Todas las tablas de la base de datos ya con las relaciones entidad relación detalladas en los diagramas.

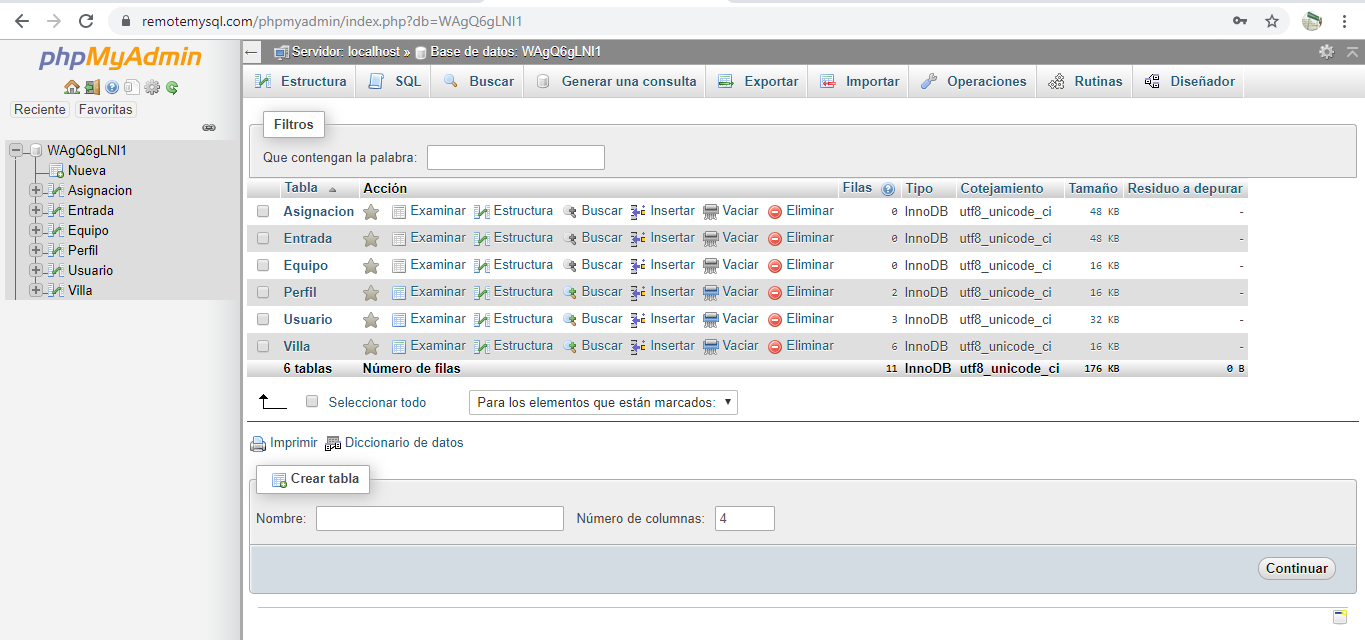


Ilustración 14: Base de datos con todas sus tablas

Tabla de asignación de villas a los residentes.

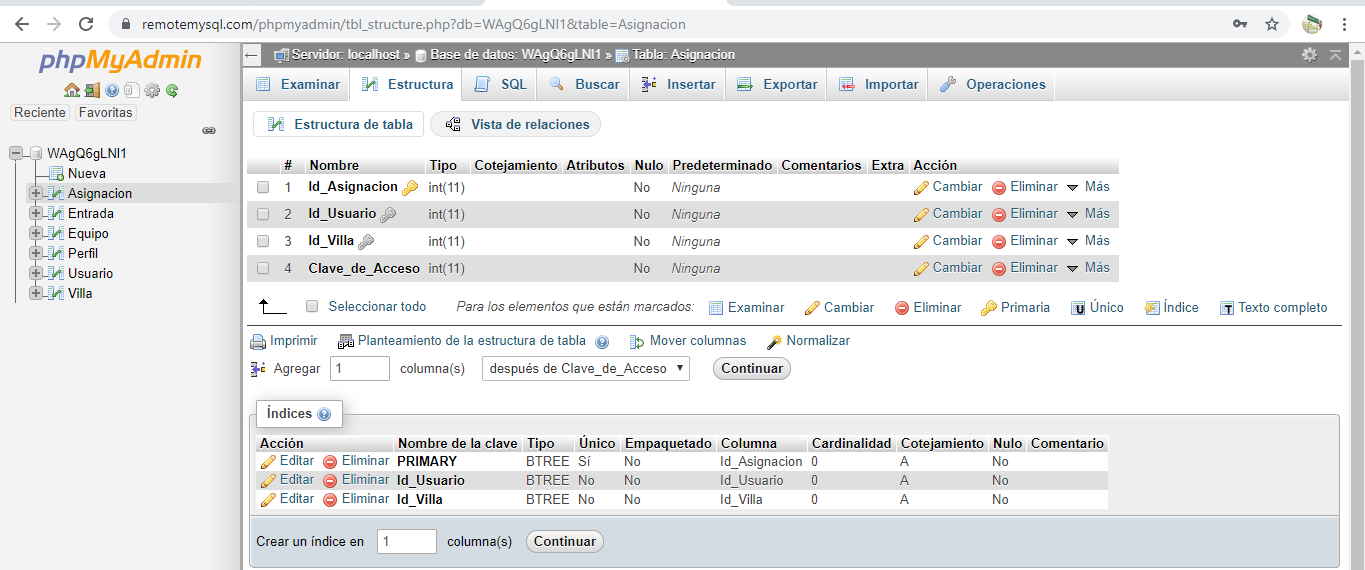


Ilustración 15: Tabla Asignación

Tabla del registro de Entrada de los residentes.

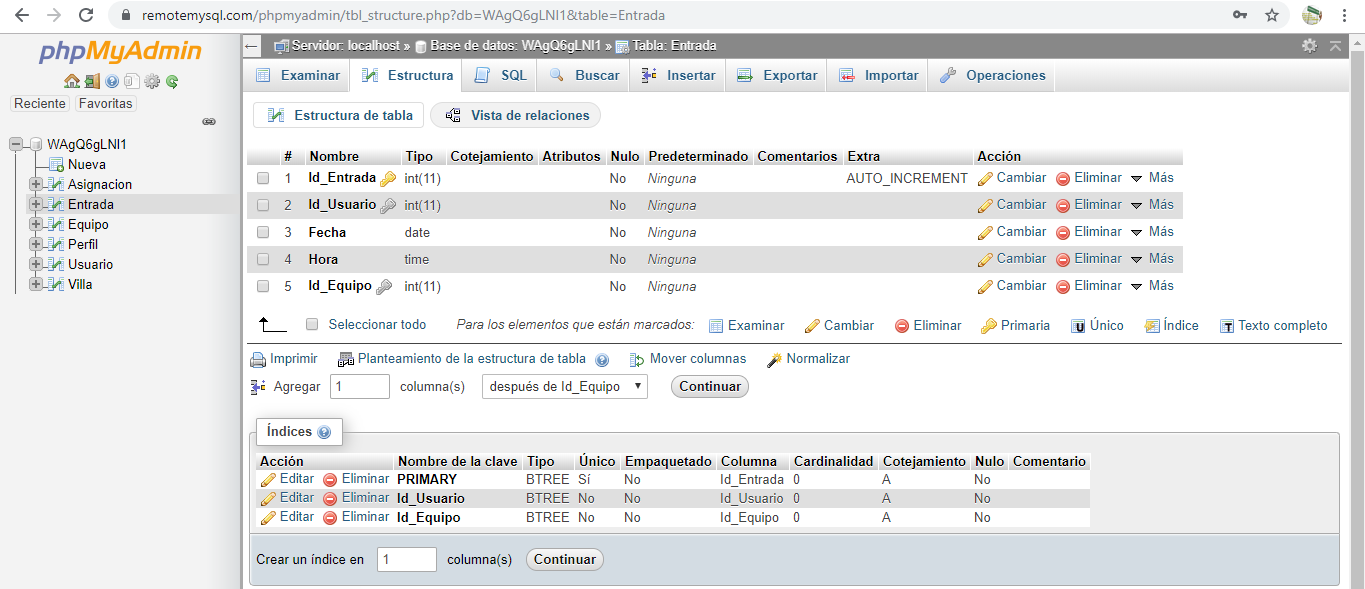


Ilustración 16: Tabla Entrada

Tabla de los Equipos implementados.

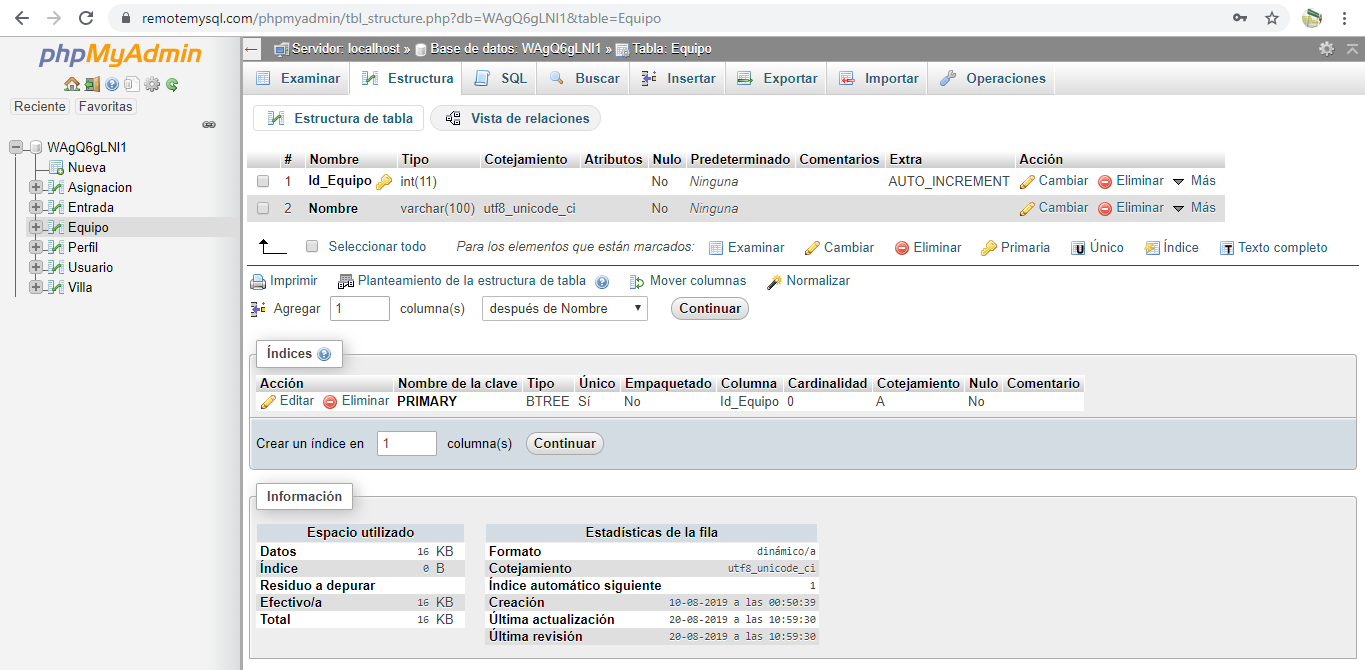


Ilustración 17: Tabla Equipos

Tabla de los perfiles existentes, en este caso, solo serán dos, o Administrador o Residente.



Ilustración 18: Tabla Perfil

Tabla de Usuarios.



Ilustración 19: Tabla Usuario

Tabla de las villas

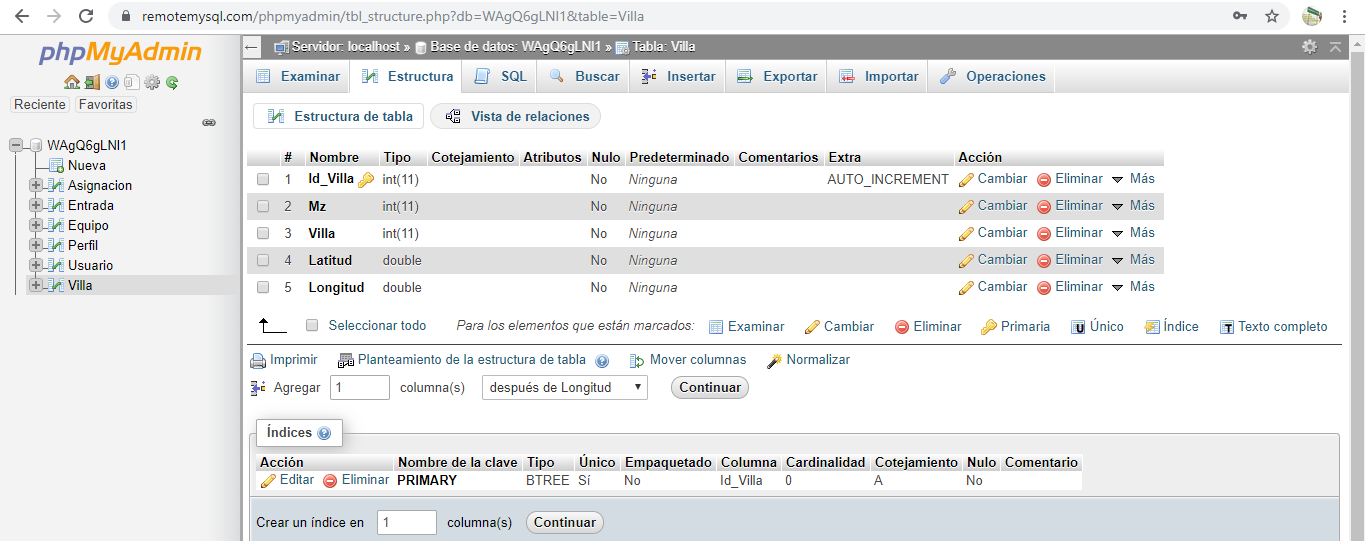


Ilustración 20: Tabla Villa

# **Códigos Fuente del Aplicativo**

## 

## **Aplicativos en Android**

### Aplicación de Apertura y Control de Puerta de Entrada – Administrador:

* AccesoActivity (Java)

En esta clase se crea el método ingresar para posteriormente añadirselo a un botón el cual tiene como objetivo validar que exista el usuario ingresado y a su vez, que la contraseña del usuario coincida con la que se encuentra en la base de datos.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** androidx.appcompat.app.AppCompatActivity; **import** android.content.Intent; **import** android.os.Bundle; **import** android.view.View; **import** android.widget.EditText; **import** android.widget.Spinner; **import** android.widget.Toast;  **public class** AccesoActivity **extends** AppCompatActivity {   **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** EditText **txt1**,**txt2**;  **private** String **codigo**,**user**,**psw**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_acceso***);  **txt1** = (EditText)findViewById(R.id.***editText***);  **txt2** = (EditText)findViewById(R.id.***editText2***);  }   *// Metodo para ingresar al menu principal* **public void** inicio(View view){  Intent i = **new** Intent(**this**, MenuActivity.**class** );  startActivity(i);  }   *// Metodo para realizar la consulta a la base de datos cuando el administrador inicia sesion* **public void** ingreso(View view){  String[] resultadoSQL = **null**;  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"2"**,  **txt1**.getText().toString(),  **txt2**.getText().toString()   };   **if**(**txt1**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe ingresar todos los datos."**, Toast.***LENGTH\_LONG***).show();  }**else**{  resultadoSQL = **new** AsyncQuery().execute(**datosConexion**).get();  Toast.*makeText*(AccesoActivity.**this**,**"Conexión Establecida"**, Toast.***LENGTH\_LONG***).show();  **user**=resultadoSQL[0];  **psw**=resultadoSQL[1];  **if**(**txt2**.getText().toString().equals(**psw**)){  Intent i = **new** Intent(**this**, MenuActivity.**class** );  startActivity(i);  Toast.*makeText*(**this**, **"Ingreso Exitoso."**, Toast.***LENGTH\_LONG***).show();  }**else**{  Toast.*makeText*(**this**, **"Se ha producido un error."**, Toast.***LENGTH\_LONG***).show();  }  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  } } |

* AccesoActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".AccesoActivity"**>   <**TextView  android:id="@+id/textView"  android:layout\_width="310dp"  android:layout\_height="63dp"  android:layout\_marginStart="45dp"  android:layout\_marginTop="35dp"  android:text="Urban Access Admin"  android:textAlignment="center"  android:textSize="30sp"  android:textStyle="bold"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**ImageView  android:id="@+id/imageView"  android:layout\_width="145dp"  android:layout\_height="129dp"  android:layout\_marginStart="133dp"  android:layout\_marginTop="18dp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView"  app:srcCompat="@drawable/logoua"** />   <**EditText  android:id="@+id/editText"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="98dp"  android:layout\_marginTop="34dp"  android:ems="10"  android:hint="Ingrese su Usuario:"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/imageView"** />   <**EditText  android:id="@+id/editText2"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="98dp"  android:layout\_marginTop="18dp"  android:ems="10"  android:hint="Ingrese su Contraseña:"  android:inputType="textPassword"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText"** />   <**Button  android:id="@+id/button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="160dp"  android:layout\_marginTop="84dp"  android:onClick="ingreso"  android:text="INICIO"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText2"** />   </**androidx.constraintlayout.widget.ConstraintLayout**> |

* AsyncQuery (Java)

En esta clase se crearon todos los métodos necesarios para realizar la conexión entre la base de datos y los métodos creados en las diferentes clases del proyecto. En caso de corresponder al codigo, se consultará, agregará, eliminará o modificará algún valor de la base de datos.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** android.os.AsyncTask; **import** android.os.Bundle;  **import** java.io.Console; **import** java.sql.Connection; **import** java.sql.DriverManager; **import** java.sql.ResultSet; **import** java.sql.Statement; **import** java.sql.SQLException; **import** android.util.Log; **import** android.view.View; **import** android.widget.Toast; **import** java.util.Random;  **public class** AsyncQuery **extends** AsyncTask<String[],Void,String[]> {   **private** Connection **conexionMySQL**=**null**;  **private** Statement **st** = **null**;  **private** ResultSet **rs** = **null**;   @Override  **protected** String[] doInBackground(String[]... datos) {  String nombre1 ;  String nombre2 ;  String apellido1;  String apellido2;  String cedula;  String user;  String password;  String mz,villa,longitud,latitud,fecha;  String email ;  String perfil ;  String codigo=datos[0][5];  String resultadoSQL = **""**;  String[] totalResultadoSQL = **null**;  **int** numColumnas = 0;  **int** numFilas = 0;  String SERVIDOR = datos[0][0];  String PUERTO = datos[0][1];  String BD = datos[0][2];  String USUARIO = datos[0][3];  String PASSWORD = datos[0][4];  String clave\_acceso,id\_residente,id\_villa;   **try**{  **conexionMySQL** = DriverManager.*getConnection*(**"jdbc:mysql://"** + SERVIDOR + **":"** + PUERTO + **"/"** + BD,  USUARIO,PASSWORD);   **st** = **conexionMySQL**.createStatement();   *//agg admin y resident* **if**(codigo.equals(**"1"**)){  nombre1 = datos[0][9];  nombre2 = datos[0][10];  apellido1 = datos[0][11];  apellido2 = datos[0][12];  cedula = datos[0][13];  user= datos[0][6];  password= datos[0][7];  email = datos[0][14];  perfil = datos[0][8];  **if**(perfil.equals(**"Administrador"**)){   String q = **"INSERT INTO Usuario (Cedula,Nombre1,Nombre2,Apellido1,Apellido2,User,Password,email,clave\_acceso,Id\_Perfil) VALUES "** +  **"('"** + cedula + **"','"** + nombre1 + **"','"** + nombre2 + **"','"** + apellido1 +**"','"** + apellido2 +**"','"** + user +**"','"** + password +**"','"** + email + **"','0000','1')"**;  Log.*d*(**"Query: "**,q);  **st**.executeUpdate(q);   }**else**{   Random r=**new** Random();  **int** val1=r.nextInt(10);  **int** val2=r.nextInt(10);  **int** val3=r.nextInt(10);  **int** val4=r.nextInt(10);  clave\_acceso=String.*valueOf*(val1)+String.*valueOf*(val2)+String.*valueOf*(val3)+String.*valueOf*(val4);   String q = **"INSERT INTO Usuario (Cedula,Nombre1,Nombre2,Apellido1,Apellido2,User,Password,email,clave\_acceso,Id\_Perfil) VALUES "** +  **"('"** + cedula + **"','"** + nombre1 + **"','"** + nombre2 + **"','"** + apellido1 +**"','"** + apellido2 +**"','"** + user +**"','"** + password +**"','"** + email + **"','"**+clave\_acceso+**"','2')"**;  Log.*d*(**"Query: "**,q);  **st**.executeUpdate(q);  }   *//validar acceso* }**if**(codigo.equals(**"2"**)){  user= datos[0][6];  password= datos[0][7];  **rs** = **st**.executeQuery(**"SELECT User,Password FROM Usuario WHERE User='"**+user+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if**(numFilas == 0)  {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  }**else** {  **rs**.beforeFirst();  **while** (**rs**.next())  {  numColumnas = **rs**.getMetaData().getColumnCount();  totalResultadoSQL=**new** String[2];  **for**(**int** i=1;i<=numColumnas;i++){  totalResultadoSQL[i-1]= **rs**.getString(i);  }  }  }   *//agg villa* }**if**(codigo.equals(**"3"**)){  mz=datos[0][6];  villa=datos[0][7];  latitud=datos[0][8];  longitud=datos[0][9];  String q = **"INSERT INTO Villa (Mz,Villa,Latitud,Longitud) VALUES "** +  **"('"** + mz + **"','"** + villa + **"','"** + latitud + **"','"** +longitud+**"')"**;  Log.*d*(**"Query: "**,q);  **st**.executeUpdate(q);   *//consulta por cedula* }**if**(codigo.equals(**"4"**)){  cedula=datos[0][6];  **rs** = **st**.executeQuery(**"SELECT Id\_Usuario, clave\_acceso FROM Usuario WHERE Cedula='"**+cedula+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if**(numFilas == 0)  {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  }**else** {  **rs**.beforeFirst();  **while** (**rs**.next())  {  numColumnas = **rs**.getMetaData().getColumnCount();  totalResultadoSQL=**new** String[2];  **for**(**int** i=1;i<=numColumnas;i++){  totalResultadoSQL[i-1]= **rs**.getString(i);  }  }  }   *//consulta entrada* }**if**(codigo.equals(**"5"**)) {  fecha = datos[0][6];  **rs** = **st**.executeQuery(**"SELECT \* FROM Entrada WHERE Fecha='"**+fecha+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if** (numFilas == 0) {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  } **else** {  **for** (**int** i = 1; i <= numColumnas; i++) {  **if** (i > 1) resultadoSQL += **","**;  resultadoSQL += **rs**.getMetaData().getColumnName(i);  }  resultadoSQL += **"\n"**;  **rs**.beforeFirst();  **while** (**rs**.next()) {  numColumnas = **rs**.getMetaData().getColumnCount();  **for** (**int** i = 1; i <= numColumnas; i++) {  **if** (i > 1) resultadoSQL += **","**;  resultadoSQL += **rs**.getString(i);  }  resultadoSQL += **"\n"**;  }  }  totalResultadoSQL = **new** String[]{resultadoSQL};   *//modificar eliminar* }**if**(codigo.equals(**"6"**)) {  cedula = datos[0][6];  **rs** = **st**.executeQuery(**"SELECT clave\_acceso,Cedula,Nombre1,Nombre2,Apellido1,Apellido2,User,Password,email FROM Usuario WHERE Cedula = '"**+cedula+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if**(numFilas == 0)  {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  }**else** {  **rs**.beforeFirst();  **while** (**rs**.next())  {  numColumnas = **rs**.getMetaData().getColumnCount();  totalResultadoSQL=**new** String[9];  **for**(**int** i=1;i<=numColumnas;i++){  totalResultadoSQL[i-1]= **rs**.getString(i);  }  }  }   *//consulta por villa y mz* }**if**(codigo.equals(**"7"**)) {  mz = datos[0][6];  villa = datos[0][7];  **rs** = **st**.executeQuery(**"SELECT id\_Villa FROM Villa WHERE Mz='"** + mz + **"' AND Villa='"**+villa+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if** (numFilas == 0) {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  } **else** {  **rs**.beforeFirst();  **while** (**rs**.next()) {  numColumnas = **rs**.getMetaData().getColumnCount();  totalResultadoSQL = **new** String[1];  **for** (**int** i = 1; i <= numColumnas; i++) {  totalResultadoSQL[i - 1] = **rs**.getString(i);  }  }  }   *//asignacion* }**if**(codigo.equals(**"8"**)){  id\_residente=datos[0][6];  id\_villa=datos[0][7];  clave\_acceso=datos[0][8];  String q = **"INSERT INTO Asignacion (Id\_Usuario,Id\_Villa,Clave\_de\_Acceso) VALUES "** +  **"('"** + id\_residente + **"','"** + id\_villa + **"','"** +clave\_acceso+**"')"**;  Log.*d*(**"Query: "**,q);  **st**.executeUpdate(q);   *//consulta para modificar* }**if** (codigo.equals(**"9"**)){  cedula = datos[0][6];  **rs** = **st**.executeQuery(**"SELECT Nombre1,Nombre2,Apellido1,Apellido2,User,Password,email,clave\_acceso FROM Usuario WHERE Cedula='"**+cedula+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if** (numFilas == 0) {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  } **else** {  **rs**.beforeFirst();  **while** (**rs**.next()) {  numColumnas = **rs**.getMetaData().getColumnCount();  totalResultadoSQL = **new** String[8];  **for** (**int** i = 1; i <= numColumnas; i++) {  totalResultadoSQL[i - 1] = **rs**.getString(i);  }  }  }  }   *//modificar* **if** (codigo.equals(**"10"**)){  cedula = datos[0][6];  nombre1= datos[0][7];  nombre2= datos[0][8];  apellido1= datos[0][9];  apellido2= datos[0][10];  email= datos[0][11];  user= datos[0][12];  password= datos[0][13];  clave\_acceso= datos[0][14];  String q = **"UPDATE Usuario SET Nombre1='"**+nombre1+**"',Nombre2='"**+nombre2+**"',Apellido1='"**+apellido1+**"',Apellido2='"**+apellido2+**"',User='"**+user+**"',Password='"**+password+**"',email='"**+email+**"',clave\_acceso='"**+clave\_acceso+**"' WHERE Cedula='"**+cedula+**"'"**;  Log.*d*(**"Query: "**,q);  **st**.executeUpdate(q);  }   *//eliminar* **if** (codigo.equals(**"11"**)){  cedula = datos[0][6];  String q = **"DELETE FROM Usuario WHERE Cedula='"**+cedula+**"'"**;  Log.*d*(**"Query: "**,q);  **st**.executeUpdate(q);  }   }**catch**(SQLException ex)  {  Log.*d*(**"Error de conexión"**, ex.getMessage());  }  **finally** {  **try** {  **if**(**rs** != **null**)  {  **rs**.close();  }  **st**.close();  **conexionMySQL**.close();  } **catch** (SQLException e)  {  e.printStackTrace();  }  }  **return** totalResultadoSQL;  }  } |

* ConsultaActivity (Java)

En esta clase se crea el método consulta de un residente en el cual se realiza una conexión con la base de datos a través de la clase AsyncQuery.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** androidx.appcompat.app.AppCompatActivity;  **import** android.content.Intent; **import** android.os.Bundle; **import** android.view.View; **import** android.widget.EditText; **import** android.widget.Spinner; **import** android.widget.TextView; **import** android.widget.Toast;  **public class** ConsultaActivity **extends** AppCompatActivity {   **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** Spinner **spinnerPerfil**;  **private** EditText **txt1**;  **private** TextView **txt2**;  **private** String **codigo**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_consulta***);  **txt1** = (EditText)findViewById(R.id.***editText13***);  **txt2** = (TextView) findViewById(R.id.***textViewConsulta***);  }   **public void** consulta(View view){  String[] resultadoSQL = **null**;  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"5"**,  **txt1**.getText().toString()   };   **if**(**txt1**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe ingresar todos los datos."**, Toast.***LENGTH\_LONG***).show();  }**else**{  resultadoSQL = **new** AsyncQuery().execute(**datosConexion**).get();  **txt2**.setText(resultadoSQL[0]);  Toast.*makeText*(ConsultaActivity.**this**,**"Conexión Establecida"**, Toast.***LENGTH\_LONG***).show();  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  } } |

* ConsultaActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".ConsultaActivity"**>   <**TextView  android:id="@+id/textView6"  android:layout\_width="307dp"  android:layout\_height="36dp"  android:layout\_marginStart="49dp"  android:layout\_marginTop="61dp"  android:text="CONSULTA"  android:textAlignment="center"  android:textSize="24sp"  android:textStyle="bold"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**TextView  android:id="@+id/textView7"  android:layout\_width="156dp"  android:layout\_height="31dp"  android:layout\_marginStart="48dp"  android:layout\_marginTop="16dp"  android:text="Escoja la Fecha:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView6"** />   <**TextView  android:id="@+id/textViewConsulta"  android:layout\_width="305dp"  android:layout\_height="121dp"  android:layout\_marginStart="52dp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView9"** />   <**TextView  android:id="@+id/textView9"  android:layout\_width="151dp"  android:layout\_height="25dp"  android:layout\_marginStart="48dp"  android:layout\_marginTop="8dp"  android:text="Ingresos:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText13"** />   <**Button  android:id="@+id/button8"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="161dp"  android:layout\_marginTop="160dp"  android:onClick="consulta"  android:text="CONSULTA"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textViewConsulta"** />   <**EditText  android:id="@+id/editText13"  android:layout\_width="305dp"  android:layout\_height="46dp"  android:layout\_marginStart="48dp"  android:layout\_marginTop="8dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView7"** />  </**androidx.constraintlayout.widget.ConstraintLayout**> |

* MenuActivity (Java)

En esta clase se crearon varios métodos para posteriormente ser aplicados a los distintos botones del menú, llevándolos a distintas ventanas dependiendo del botón escogido.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** androidx.appcompat.app.AppCompatActivity;  **import** android.content.Intent; **import** android.os.Bundle; **import** android.view.View;  **public class** MenuActivity **extends** AppCompatActivity {   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_menu***);  }   *// Metodos para que con cada boton se abra la ventana que le corresponda* **public void** agregarBtn(View view) {  Intent i = **new** Intent(**this**, AgregarActivity.**class** );  startActivity(i);  }   **public void** consultaBtn(View view) {  Intent i = **new** Intent(**this**, ConsultaActivity.**class** );  startActivity(i);  }   **public void** asignacionBtn(View view) {  Intent i = **new** Intent(**this**, AsignacionActivity.**class** );  startActivity(i);  }   **public void** villaBtn(View view) {  Intent i = **new** Intent(**this**, VillaActivity.**class** );  startActivity(i);  }   **public void** modificarEliminar(View view) {  Intent i = **new** Intent(**this**, ModificarEliminarActivity.**class** );  startActivity(i);  }  } |

* MenuActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MenuActivity"**>   <**TextView  android:id="@+id/textView3"  android:layout\_width="379dp"  android:layout\_height="80dp"  android:layout\_marginStart="19dp"  android:layout\_marginTop="43dp"  android:text="BIENVENIDO AL MENÚ PRINCIPAL"  android:textAlignment="center"  android:textSize="24sp"  android:textStyle="bold"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**ImageView  android:id="@+id/imageView3"  android:layout\_width="126dp"  android:layout\_height="110dp"  android:layout\_marginStart="140dp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView3"  app:srcCompat="@drawable/logoua"** />   <**Button  android:id="@+id/button2"  android:layout\_width="112dp"  android:layout\_height="38dp"  android:layout\_marginStart="148dp"  android:layout\_marginTop="36dp"  android:onClick="agregarBtn"  android:text="agregar"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/imageView3"** />   <**Button  android:id="@+id/button3"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="116dp"  android:layout\_marginTop="20dp"  android:onClick="modificarEliminar"  android:text="Consulta Residente"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/button2"** />   <**Button  android:id="@+id/button4"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="120dp"  android:layout\_marginTop="24dp"  android:onClick="consultaBtn"  android:text="consulta entrada"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/button3"** />   <**Button  android:id="@+id/button5"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="104dp"  android:layout\_marginTop="24dp"  android:onClick="villaBtn"  android:text="sectorización de villas"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/button4"** />   <**Button  android:id="@+id/button11"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="124dp"  android:layout\_marginTop="20dp"  android:onClick="asignacionBtn"  android:text="ASIGNACIÓN VILLAS"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/button5"** />  </**androidx.constraintlayout.widget.ConstraintLayout**> |

* AgregarActivity (Java)

En esta clase se busca añadir información de residentes o administradores a la base de datos a través de la clase AsyncQuery.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** androidx.appcompat.app.AppCompatActivity;  **import** android.os.Bundle; **import** android.view.View; **import** android.widget.ArrayAdapter; **import** android.widget.EditText; **import** android.widget.Spinner; **import** android.widget.Toast;  **public class** AgregarActivity **extends** AppCompatActivity {   **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** Spinner **spinnerPerfil**;  **private** EditText **txt1**,**txt2**,**txt3**,**txt4**,**txt5**,**txt6**,**txt7**,**txt8**;  **private** String **codigo**,**perfil**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_agregar***);  **spinnerPerfil**= (Spinner)findViewById(R.id.***spinnerPerfil***);  **txt1** = (EditText)findViewById(R.id.***editText12***);  **txt2** = (EditText)findViewById(R.id.***editText6***);  **txt3** = (EditText)findViewById(R.id.***editText15***);  **txt4** = (EditText)findViewById(R.id.***editText3***);  **txt5** = (EditText)findViewById(R.id.***editText5***);  **txt6** = (EditText)findViewById(R.id.***editText7***);  **txt7** = (EditText)findViewById(R.id.***editText8***);  **txt8** = (EditText)findViewById(R.id.***editTextCedulaME***);   String [] perfiles= {**"Administrador"**,**"Residente"**};  ArrayAdapter<String> adapter1= **new** ArrayAdapter<String>(**this**, android.R.layout.***simple\_spinner\_item***, perfiles);  **spinnerPerfil**.setAdapter(adapter1);  }   *// Agregar el administrador o residente a la base de datos* **public void** agregar(View view){  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"1"**,  **txt6**.getText().toString(),  **txt7**.getText().toString(),  **perfil**=**spinnerPerfil**.getSelectedItem().toString(),  **txt1**.getText().toString(),  **txt2**.getText().toString(),  **txt3**.getText().toString(),  **txt4**.getText().toString(),  **txt5**.getText().toString(),  **txt8**.getText().toString()   };   **if**(**txt1**.getText().toString().equals(**""**)|| **txt2**.getText().toString().equals(**""**)|| **txt3**.getText().toString().equals(**""**)|| **txt4**.getText().toString().equals(**""**)|| **txt5**.getText().toString().equals(**""**)|| **txt6**.getText().toString().equals(**""**)|| **txt7**.getText().toString().equals(**""**)|| **txt8**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe ingresar todos los datos."**, Toast.***LENGTH\_LONG***).show();  }**else**{  **new** AsyncQuery().execute(**datosConexion**);  **txt1**.setText(**""**);  **txt2**.setText(**""**);  **txt3**.setText(**""**);  **txt4**.setText(**""**);  **txt5**.setText(**""**);  **txt6**.setText(**""**);  **txt7**.setText(**""**);  **txt8**.setText(**""**);  Toast.*makeText*(AgregarActivity.**this**,**"Dato Agregado."**, Toast.***LENGTH\_LONG***).show();  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  } }  } |

* AgregarActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".AgregarActivity"**>   <**EditText  android:id="@+id/editText12"  android:layout\_width="292dp"  android:layout\_height="45dp"  android:layout\_marginStart="50dp"  android:layout\_marginTop="8dp"  android:ems="10"  android:hint="Primer Nombre"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/spinnerPerfil"** />   <**EditText  android:id="@+id/editTextCedulaME"  android:layout\_width="294dp"  android:layout\_height="45dp"  android:layout\_marginStart="50dp"  android:ems="10"  android:hint="Email"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText8"** />   <**EditText  android:id="@+id/editText7"  android:layout\_width="295dp"  android:layout\_height="40dp"  android:layout\_marginStart="50dp"  android:ems="10"  android:hint="Usuario"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText5"** />   <**EditText  android:id="@+id/editText6"  android:layout\_width="292dp"  android:layout\_height="45dp"  android:layout\_marginStart="50dp"  android:ems="10"  android:hint="Segundo Nombre"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText12"** />   <**TextView  android:id="@+id/textView4"  android:layout\_width="191dp"  android:layout\_height="41dp"  android:layout\_marginStart="109dp"  android:layout\_marginTop="24dp"  android:text="CREACIÓN"  android:textAlignment="center"  android:textSize="24sp"  android:textStyle="bold"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**TextView  android:id="@+id/textView5"  android:layout\_width="71dp"  android:layout\_height="23dp"  android:layout\_marginStart="50dp"  android:layout\_marginTop="8dp"  android:text="Perfil:"  android:textSize="18sp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView4"** />   <**Spinner  android:id="@+id/spinnerPerfil"  android:layout\_width="294dp"  android:layout\_height="35dp"  android:layout\_marginStart="50dp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView5"** />   <**EditText  android:id="@+id/editText3"  android:layout\_width="295dp"  android:layout\_height="41dp"  android:layout\_marginStart="50dp"  android:ems="10"  android:hint="Segundo Apellido"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText15"** />   <**EditText  android:id="@+id/editText5"  android:layout\_width="294dp"  android:layout\_height="41dp"  android:layout\_marginStart="50dp"  android:ems="10"  android:hint="Cédula"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText3"** />   <**EditText  android:id="@+id/editText8"  android:layout\_width="294dp"  android:layout\_height="47dp"  android:layout\_marginStart="50dp"  android:ems="10"  android:hint="Contraseña"  android:inputType="textPassword"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText7"** />   <**Button  android:id="@+id/button7"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="160dp"  android:layout\_marginTop="44dp"  android:onClick="agregar"  android:text="AGREGAR"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editTextCedulaME"** />   <**EditText  android:id="@+id/editText15"  android:layout\_width="292dp"  android:layout\_height="41dp"  android:layout\_marginStart="50dp"  android:ems="10"  android:hint="Primer Apellido"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText6"** />  </**androidx.constraintlayout.widget.ConstraintLayout**> |

* ModificarEliminarActivity (Java)

En esta clase se busca modificar, eliminar o enviar un correo electrónico a un residente con sus datos, generandose una conexión con la base de datos a través de la clase AsyncQuery.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** androidx.appcompat.app.AppCompatActivity;  **import** android.content.DialogInterface; **import** android.content.Intent; **import** android.net.Uri; **import** android.os.Bundle; **import** android.util.Log; **import** android.view.View; **import** android.widget.EditText; **import** android.widget.Toast;  **public class** ModificarEliminarActivity **extends** AppCompatActivity {    **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** EditText **cedula**,**n1**,**n2**,**a1**,**a2**,**mail**,**user**,**psw**,**clave**;  **private** String **codigo**;  Intent **emailIntent** = **new** Intent(Intent.***ACTION\_SEND***);    @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_modificar\_eliminar***);   **cedula** = (EditText) findViewById(R.id.***editTextCedulaME***);  **n1**= (EditText) findViewById(R.id.***editTextNombre1***);  **n2**= (EditText) findViewById(R.id.***editTextNombre2***);  **a1**= (EditText) findViewById(R.id.***editTextApellido***);  **a2**= (EditText) findViewById(R.id.***editTextApellido2***);  **mail**= (EditText) findViewById(R.id.***editTextMail***);  **user**= (EditText) findViewById(R.id.***editTextUser***);  **psw**= (EditText) findViewById(R.id.***editTextPsw***);  **clave**= (EditText) findViewById(R.id.***editTextClave***);   }   **public void** buscar(View v){  String[] resultadoSQL = **null**;   **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"9"**,  **cedula**.getText().toString()   };   **if**(**cedula**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe el dato cédula."**, Toast.***LENGTH\_LONG***).show();  }**else**{  resultadoSQL = **new** AsyncQuery().execute(**datosConexion**).get();  **n1**.setText(resultadoSQL[0]);  **n2**.setText(resultadoSQL[1]);  **a1**.setText(resultadoSQL[2]);  **a2**.setText(resultadoSQL[3]);  **mail**.setText(resultadoSQL[6]);  **user**.setText(resultadoSQL[4]);  **psw**.setText(resultadoSQL[5]);  **clave**.setText(resultadoSQL[7]);  Toast.*makeText*(**this**,**"Conexión Establecida"**, Toast.***LENGTH\_LONG***).show();  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }   **public void** Modificar(View view){  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();  **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"10"**,  **cedula**.getText().toString(),  **n1**.getText().toString(),  **n2**.getText().toString(),  **a1**.getText().toString(),  **a2**.getText().toString(),  **mail**.getText().toString(),  **user**.getText().toString(),  **psw**.getText().toString(),  **clave**.getText().toString()  };   **if**(**cedula**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe consultar primero."**, Toast.***LENGTH\_LONG***).show();  }**else**{  **new** AsyncQuery().execute(**datosConexion**);  Toast.*makeText*(ModificarEliminarActivity.**this**,**"Datos Actualizados."**, Toast.***LENGTH\_LONG***).show();  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }   **public void** Eliminar(View view){  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();  **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"11"**,  **cedula**.getText().toString()  };   **if**(**cedula**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Ingrese cédula."**, Toast.***LENGTH\_LONG***).show();  }**else**{  **new** AsyncQuery().execute(**datosConexion**);  **cedula**.setText(**""**);  **n1**.setText(**""**);  **n2**.setText(**""**);  **a1**.setText(**""**);  **a2**.setText(**""**);  **mail**.setText(**""**);  **user**.setText(**""**);  **psw**.setText(**""**);  **clave**.setText(**""**);  Toast.*makeText*(ModificarEliminarActivity.**this**,**"Dato Eliminado."**, Toast.***LENGTH\_LONG***).show();  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }   **public void** enviar\_correo(View view){   **if**(**cedula**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Inicialmente debe consultar la cédula."**, Toast.***LENGTH\_LONG***).show();  }**else**{  String[] to = {**mail**.getText().toString()};  String[] cc = {**mail**.getText().toString()};  **emailIntent**.setData(Uri.*parse*(**"mailto:"**));  **emailIntent**.setType(**"text/plain"**);  **emailIntent**.putExtra(Intent.***EXTRA\_EMAIL***, to);  **emailIntent**.putExtra(Intent.***EXTRA\_CC***, cc);  **emailIntent**.putExtra(Intent.***EXTRA\_SUBJECT***, **"Envío de Datos UrbanAccess Residente"**);    **emailIntent**.putExtra(Intent.***EXTRA\_TEXT***, **"Información\n"** +  **"Usuario:"** + **user**.getText().toString() + **"\n"** +  **"Contraseña:"** + **psw**.getText().toString() + **"\n"** +  **"Clave De Acceso:"** + **clave**.getText().toString() + **"\n"**);   **try** {  startActivity(Intent.*createChooser*(**emailIntent**, **"Enviando Email"**));  Log.*i*(**"termina envio de email"**, **""**);  } **catch** (android.content.ActivityNotFoundException ex) {  Toast.*makeText*(ModificarEliminarActivity.**this**, **"No existe cliente Email instalado."**, Toast.***LENGTH\_SHORT***).show();    } }    } } |

* ModificarEliminarActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".ModificarEliminarActivity"**>   <**EditText  android:id="@+id/editTextNombre1"  android:layout\_width="227dp"  android:layout\_height="45dp"  android:layout\_marginStart="16dp"  android:layout\_marginTop="23dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView9"  app:layout\_constraintTop\_toBottomOf="@+id/button7"** />   <**EditText  android:id="@+id/editTextClave"  android:layout\_width="228dp"  android:layout\_height="51dp"  android:layout\_marginStart="13dp"  android:layout\_marginTop="2dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView8"  app:layout\_constraintTop\_toBottomOf="@+id/editTextPsw"** />   <**EditText  android:id="@+id/editTextPsw"  android:layout\_width="227dp"  android:layout\_height="52dp"  android:layout\_marginStart="26dp"  android:layout\_marginTop="3dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView7"  app:layout\_constraintTop\_toBottomOf="@+id/editTextUser"** />   <**EditText  android:id="@+id/editTextUser"  android:layout\_width="225dp"  android:layout\_height="54dp"  android:layout\_marginStart="28dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView6"  app:layout\_constraintTop\_toBottomOf="@+id/editTextMail"** />   <**EditText  android:id="@+id/editTextMail"  android:layout\_width="226dp"  android:layout\_height="51dp"  android:layout\_marginStart="1dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView5"  app:layout\_constraintTop\_toBottomOf="@+id/editTextApellido2"** />   <**TextView  android:id="@+id/textView4"  android:layout\_width="117dp"  android:layout\_height="41dp"  android:layout\_marginStart="28dp"  android:layout\_marginTop="45dp"  android:text="Cédula:"  android:textAlignment="center"  android:textSize="18sp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**EditText  android:id="@+id/editTextCedulaME"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginTop="41dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView4"  app:layout\_constraintTop\_toTopOf="parent"** />   <**Button  android:id="@+id/button7"  android:layout\_width="149dp"  android:layout\_height="50dp"  android:layout\_marginStart="136dp"  android:layout\_marginTop="28dp"  android:onClick="buscar"  android:text="buscar"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView4"** />   <**TextView  android:id="@+id/textView5"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="28dp"  android:layout\_marginTop="32dp"  android:text="Correo Electrónico:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView15"** />   <**TextView  android:id="@+id/textView6"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="72dp"  android:layout\_marginTop="40dp"  android:text="Usuario:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView5"** />   <**TextView  android:id="@+id/textView7"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="56dp"  android:layout\_marginTop="40dp"  android:text="Password:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView6"** />   <**TextView  android:id="@+id/textView8"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="44dp"  android:layout\_marginTop="28dp"  android:text="Clave Acceso:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView7"** />   <**TextView  android:id="@+id/textView9"  android:layout\_width="wrap\_content"  android:layout\_height="23dp"  android:layout\_marginStart="36dp"  android:layout\_marginTop="30dp"  android:text="Primer Nombre:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/button7"** />   <**TextView  android:id="@+id/textView11"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="36dp"  android:layout\_marginTop="32dp"  android:text="Segundo Nombre:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView9"** />   <**TextView  android:id="@+id/textView13"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="36dp"  android:layout\_marginTop="40dp"  android:text="Primer Apellido:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView11"** />   <**TextView  android:id="@+id/textView15"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="38dp"  android:layout\_marginTop="20dp"  android:text="Segundo Apellido:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView13"** />   <**EditText  android:id="@+id/editTextNombre2"  android:layout\_width="227dp"  android:layout\_height="47dp"  android:layout\_marginStart="2dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView11"  app:layout\_constraintTop\_toBottomOf="@+id/editTextNombre1"** />   <**EditText  android:id="@+id/editTextApellido"  android:layout\_width="229dp"  android:layout\_height="51dp"  android:layout\_marginStart="13dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView13"  app:layout\_constraintTop\_toBottomOf="@+id/editTextNombre2"** />   <**EditText  android:id="@+id/editTextApellido2"  android:layout\_width="228dp"  android:layout\_height="51dp"  android:layout\_marginTop="3dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView15"  app:layout\_constraintTop\_toBottomOf="@+id/editTextApellido"** />   <**Button  android:id="@+id/button12"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="32dp"  android:layout\_marginTop="28dp"  android:onClick="Modificar"  android:text="MODIFICAR"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editTextClave"** />   <**Button  android:id="@+id/button13"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="20dp"  android:layout\_marginTop="28dp"  android:onClick="Eliminar"  android:text="ELIMINAR"  app:layout\_constraintStart\_toEndOf="@+id/button12"  app:layout\_constraintTop\_toBottomOf="@+id/editTextClave"** />   <**Button  android:id="@+id/button14"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="12dp"  android:layout\_marginTop="28dp"  android:onClick="enviar\_correo"  android:text="enviar datos"  app:layout\_constraintStart\_toEndOf="@+id/button13"  app:layout\_constraintTop\_toBottomOf="@+id/editTextClave"** />  </**androidx.constraintlayout.widget.ConstraintLayout**> |

* SplashActivity (Java)

En esta clase se presenta la imagen o logo del grupo apenas se ingresa a la app.

|  |
| --- |
| **import** androidx.appcompat.app.AppCompatActivity;  **import** android.content.Intent; **import** android.os.Bundle; **import** android.os.Handler;  **public class** SplashActivity **extends** AppCompatActivity {   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_splash***);  **new** Handler().postDelayed(**new** Runnable() {  @Override  **public void** run() {   Intent intent= **new** Intent(SplashActivity.**this**, AccesoActivity.**class**);  startActivity(intent);   }  },3000);  } } |

* VillaActivity (Java)

En esta clase se tiene como objetivo crear un método que permita realizar un ingreso de villa y otros valores a la base de datos con ayuda de la clase AsyncQuery.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** androidx.appcompat.app.AppCompatActivity;  **import** android.os.Bundle; **import** android.view.View; **import** android.widget.EditText; **import** android.widget.Spinner; **import** android.widget.Toast;  **public class** VillaActivity **extends** AppCompatActivity {   **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** EditText **txt1**,**txt2**,**txt3**,**txt4**;  **private** String **codigo**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_villa***);  **txt1** = (EditText)findViewById(R.id.***editText14***);  **txt2** = (EditText)findViewById(R.id.***editText15***);  **txt3** = (EditText)findViewById(R.id.***editText16***);  **txt4** = (EditText)findViewById(R.id.***editText17***);  }  **public void** agregar(View view){  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**" 3"**,  **txt1**.getText().toString(),  **txt2**.getText().toString(),  **txt3**.getText().toString(),  **txt4**.getText().toString()   };   **if**(**txt1**.getText().toString().equals(**""**)|| **txt2**.getText().toString().equals(**""**)|| **txt3**.getText().toString().equals(**""**)|| **txt4**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe ingresar todos los datos."**, Toast.***LENGTH\_LONG***).show();  }**else**{  **new** AsyncQuery().execute(**datosConexion**);  **txt1**.setText(**""**);  **txt2**.setText(**""**);  **txt3**.setText(**""**);  **txt4**.setText(**""**);  Toast.*makeText*(VillaActivity.**this**,**"Dato Agregado."**, Toast.***LENGTH\_LONG***).show();  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }  } |

* VillaActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".VillaActivity"**>   <**TextView  android:id="@+id/textView17"  android:layout\_width="217dp"  android:layout\_height="48dp"  android:layout\_marginStart="97dp"  android:layout\_marginTop="33dp"  android:text="Villas"  android:textAlignment="center"  android:textSize="30sp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**TextView  android:id="@+id/textView18"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="63dp"  android:layout\_marginTop="121dp"  android:text="Mz:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**TextView  android:id="@+id/textView19"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="63dp"  android:layout\_marginTop="35dp"  android:text="Villa:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView18"** />   <**TextView  android:id="@+id/textView20"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="50dp"  android:layout\_marginTop="37dp"  android:text="Latitud:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView19"** />   <**TextView  android:id="@+id/textView21"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="38dp"  android:layout\_marginTop="33dp"  android:text="Longitud:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView20"** />   <**EditText  android:id="@+id/editText14"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="42dp"  android:layout\_marginTop="27dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView18"  app:layout\_constraintTop\_toBottomOf="@+id/textView17"** />   <**EditText  android:id="@+id/editText15"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="34dp"  android:layout\_marginTop="9dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView19"  app:layout\_constraintTop\_toBottomOf="@+id/editText14"** />   <**EditText  android:id="@+id/editText16"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="30dp"  android:layout\_marginTop="10dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView20"  app:layout\_constraintTop\_toBottomOf="@+id/editText15"** />   <**EditText  android:id="@+id/editText17"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="30dp"  android:layout\_marginTop="10dp"  android:ems="10"  android:inputType="textPersonName"  app:layout\_constraintStart\_toEndOf="@+id/textView21"  app:layout\_constraintTop\_toBottomOf="@+id/editText16"** />   <**Button  android:id="@+id/button10"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="161dp"  android:layout\_marginTop="126dp"  android:onClick="agregar"  android:text="Crear"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText17"** />  </**androidx.constraintlayout.widget.ConstraintLayout**> |

* AsignacionActivity (Java)

En esta clase se busca asignar una villa a un residente, por lo cual se crean los métodos consulta residente, el cual obtendrá los datos del residente, consulta villa, que obtendrá datos de las villas y finalmente agregar asignación, el cual mediante los valores obtenidos anteriormente ingresa una villa a un residente a la base de datos.

|  |
| --- |
| **package** com.grupo5.urbanaccessadmin;  **import** androidx.appcompat.app.AppCompatActivity;  **import** android.os.Bundle; **import** android.view.View; **import** android.widget.EditText; **import** android.widget.TextView; **import** android.widget.Toast;  **public class** AsignacionActivity **extends** AppCompatActivity {  **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** EditText **cedula**, **mz**,**villa**;  **private** TextView **idR**,**clave**,**idV**;  **private** String **codigo**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_asignacion***);  **cedula**= (EditText)findViewById(R.id.***editTextcedula***);  **mz**= (EditText)findViewById(R.id.***editTextmz***);  **villa**= (EditText)findViewById(R.id.***editTextvilla***);  **idR**= (TextView)findViewById(R.id.textView378);  **idV**= (TextView)findViewById(R.id.***textViewVilla***);  **clave**= (TextView)findViewById(R.id.***textViewClave***);  }   *// Consultar la informacion de un residente,modificarla o eliminarla* **public void** consultaResidente(View view){  String[] resultadoSQL = **null**;  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"4"**,  **cedula**.getText().toString()   };   **if**(**cedula**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe el dato cédula."**, Toast.***LENGTH\_LONG***).show();  }**else**{  resultadoSQL = **new** AsyncQuery().execute(**datosConexion**).get();  **idR**.setText(resultadoSQL[0]);  **clave**.setText(resultadoSQL[1]);  Toast.*makeText*(AsignacionActivity.**this**,**"Conexión Establecida"**, Toast.***LENGTH\_LONG***).show();   }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }   *// Consultar informacion acerca de una villa* **public void** consultaVilla(View view){  String[] resultadoSQL = **null**;  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"7"**,  **mz**.getText().toString(),  **villa**.getText().toString()   };   **if**(**mz**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe el dato cédula."**, Toast.***LENGTH\_LONG***).show();  }**else**{  resultadoSQL = **new** AsyncQuery().execute(**datosConexion**).get();  **idV**.setText(resultadoSQL[0]);  Toast.*makeText*(AsignacionActivity.**this**,**"Conexión Establecida"**, Toast.***LENGTH\_LONG***).show();  }   }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }   *// Relaciona los residentes con su villa* **public void** agregarAsignacion(View view){  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"8"**,  **idR**.getText().toString(),  **idV**.getText().toString(),  **clave**.getText().toString()   };   **if**(**idV**.getText().toString().equals(**""**)|| **clave**.getText().toString().equals(**""**)|| **idR**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Primero debe consultar."**, Toast.***LENGTH\_LONG***).show();  }**else**{  **new** AsyncQuery().execute(**datosConexion**);  Toast.*makeText*(AsignacionActivity.**this**,**"Dato Agregado."**, Toast.***LENGTH\_LONG***).show();  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }  } |

* AsignacionActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/edit1"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".AsignacionActivity"**>   <**TextView  android:id="@+id/textView11"  android:layout\_width="315dp"  android:layout\_height="37dp"  android:layout\_marginStart="45dp"  android:layout\_marginTop="54dp"  android:text="ASIGNACIÓN"  android:textAlignment="center"  android:textSize="24sp"  android:textStyle="bold"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**EditText  android:id="@+id/editTextcedula"  android:layout\_width="291dp"  android:layout\_height="44dp"  android:layout\_marginStart="45dp"  android:layout\_marginTop="42dp"  android:ems="10"  android:hint="Ingrese el número de Cédula"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView11"** />   <**Button  android:id="@+id/button9"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="88dp"  android:layout\_marginTop="16dp"  android:layout\_marginBottom="8dp"  android:onClick="consultaResidente"  android:text="Consultar datos residente"  app:layout\_constraintBottom\_toTopOf="@+id/editTextmz"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textViewClave"** />   <**EditText  android:id="@+id/editTextmz"  android:layout\_width="286dp"  android:layout\_height="39dp"  android:layout\_marginStart="48dp"  android:layout\_marginTop="15dp"  android:ems="10"  android:hint="Ingrese Manzana"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/button9"** />   <**EditText  android:id="@+id/editTextvilla"  android:layout\_width="285dp"  android:layout\_height="42dp"  android:layout\_marginStart="44dp"  android:layout\_marginTop="12dp"  android:ems="10"  android:hint="Ingrese Villa"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editTextmz"** />   <**Button  android:id="@+id/button10"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="112dp"  android:layout\_marginTop="20dp"  android:onClick="consultaVilla"  android:text="consultar dato villa"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textViewVilla"** />   <**TextView  android:id="@+id/textView2"  android:layout\_width="97dp"  android:layout\_height="31dp"  android:layout\_marginStart="45dp"  android:layout\_marginTop="22dp"  android:text="ID Residente:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editTextcedula"** />   <**TextView  android:id="@+id/textView378"  android:layout\_width="194dp"  android:layout\_height="28dp"  android:layout\_marginStart="16dp"  android:layout\_marginTop="24dp"  app:layout\_constraintStart\_toEndOf="@+id/textView2"  app:layout\_constraintTop\_toBottomOf="@+id/editTextcedula"** />   <**TextView  android:id="@+id/textView23"  android:layout\_width="119dp"  android:layout\_height="27dp"  android:layout\_marginStart="45dp"  android:text="Clave de Acceso:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView2"** />   <**TextView  android:id="@+id/textViewClave"  android:layout\_width="188dp"  android:layout\_height="29dp"  app:layout\_constraintStart\_toEndOf="@+id/textView23"  app:layout\_constraintTop\_toBottomOf="@+id/textView378"** />   <**TextView  android:id="@+id/textView25"  android:layout\_width="80dp"  android:layout\_height="24dp"  android:layout\_marginStart="45dp"  android:layout\_marginTop="15dp"  android:text="ID Villa:"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editTextvilla"** />   <**TextView  android:id="@+id/textViewVilla"  android:layout\_width="209dp"  android:layout\_height="26dp"  android:layout\_marginTop="12dp"  app:layout\_constraintStart\_toEndOf="@+id/textView25"  app:layout\_constraintTop\_toBottomOf="@+id/editTextvilla"** />   <**Button  android:id="@+id/button6"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="104dp"  android:layout\_marginTop="36dp"  android:onClick="agregarAsignacion"  android:text="asignar villa a residente"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/button10"** />  </**androidx.constraintlayout.widget.ConstraintLayout**> |

### Aplicación de Apertura y Control de Puerta de Entrada – Residente:

* Conexión

En esta clase de Java se realiza la conexión con el cliente para poder realizar la conexión con internet y enviar los comandos al sensor ESP8266 y que pueda ejecutar las acciones que le enviemos.

|  |
| --- |
| package com.grupo5.urbanaccessresident;  import java.io.IOException;  import okhttp3.OkHttpClient;  import okhttp3.Request;  import okhttp3.Response;  public class Conexion {  public static String getDatos(String urlUsuario){  OkHttpClient client = new OkHttpClient();  Request request = new Request.Builder()  .url(urlUsuario)  .build();  try{  Response response = client.newCall(request).execute();  return response.body().string();  }catch (IOException error){  return null;  }  }  } |

* AsyncQuery (Java)

En esta clase se crearon todos los métodos necesarios para realizar la conexión entre la base de datos y los métodos creados en las diferentes clases del proyecto. En caso de corresponder al codigo, se consultará, agregará, eliminará o modificará algún valor de la base de datos.

|  |
| --- |
| **package** com.grupo5.urbanaccessresident;  **import** android.os.AsyncTask; **import** android.os.Bundle;  **import** java.io.Console; **import** java.sql.Connection; **import** java.sql.DriverManager; **import** java.sql.ResultSet; **import** java.sql.Statement; **import** java.sql.SQLException; **import** android.util.Log; **import** android.view.View; **import** android.widget.Toast; **import** java.util.Random;  **public class** AsyncQuery **extends** AsyncTask<String[],Void,String[]> {   **private** Connection **conexionMySQL**=**null**;  **private** Statement **st** = **null**;  **private** ResultSet **rs** = **null**;   @Override  **protected** String[] doInBackground(String[]... datos) {  String nombre1;  String nombre2;  String apellido1;  String apellido2;  String cedula;  String user;  String pass;  String mz,villa,longitud,latitud;  String email;  String perfil;  String codigo=datos[0][5];  String resultadoSQL = **""**;  String[] totalResultadoSQL = **null**;  **int** numColumnas = 0;  **int** numFilas = 0;  String SERVIDOR = datos[0][0];  String PUERTO = datos[0][1];  String BD = datos[0][2];  String USUARIO = datos[0][3];  String PASSWORD = datos[0][4];  String clave\_acceso;   **try**{  **conexionMySQL** = DriverManager.*getConnection*(**"jdbc:mysql://"** + SERVIDOR + **":"** + PUERTO + **"/"** + BD,  USUARIO,PASSWORD);   **st** = **conexionMySQL**.createStatement();   **if**(codigo.equals(**"1"**)) {  user= datos[0][6];  pass= datos[0][7];  **rs** = **st**.executeQuery(**"SELECT User,Password,Id\_Usuario FROM Usuario WHERE User='"**+user+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if** (numFilas == 0) {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  } **else** {  **rs**.beforeFirst();  **while** (**rs**.next()) {  numColumnas = **rs**.getMetaData().getColumnCount();  totalResultadoSQL=**new** String[3];  **for** (**int** i = 1; i <= numColumnas; i++) {  totalResultadoSQL[i - 1] = **rs**.getString(i);  }  }  }   }**if**(codigo.equals(**"2"**)){  clave\_acceso=datos[0][6];  **rs** = **st**.executeQuery(**"SELECT Id\_Usuario,clave\_acceso FROM Usuario WHERE clave\_acceso='"**+clave\_acceso+**"';"**);  **rs**.last();  numFilas = **rs**.getRow();  **if**(numFilas == 0)  {  resultadoSQL = **"No se ha producido ningún resultado. Revise la consulta realizada.\n"**;  }**else** {  **rs**.beforeFirst();  **while** (**rs**.next())  {  numColumnas = **rs**.getMetaData().getColumnCount();  totalResultadoSQL = **new** String[2];  **for**(**int** i=1;i<=numColumnas;i++){  totalResultadoSQL[i-1]= **rs**.getString(i);  }  }  String q = **"INSERT INTO Entrada (Id\_Usuario,Fecha,Id\_Equipo) VALUES "** +  **"('"** + totalResultadoSQL[0] + **"','2019-08-21','1')"**;  Log.*d*(**"Query: "**,q);  **st**.executeUpdate(q);  String vv;  }   }    }**catch**(SQLException ex)  {  Log.*d*(**"Error de conexión"**, ex.getMessage());  }  **finally** {  **try** {  **if**(**rs** != **null**)  {  **rs**.close();  }  **st**.close();  **conexionMySQL**.close();  } **catch** (SQLException e)  {  e.printStackTrace();  }  }  **return** totalResultadoSQL;  }  } |

* CodigoActivity (Java)

En esta clase se define la apertura de la puerta, si los codigos son correctos, manda a ejecutar al NodeMCU y se genera un ingreso en la base de datos.

|  |
| --- |
| **package** com.grupo5.urbanaccessresident;  **import** android.content.Context; **import** android.content.Intent; **import** android.net.ConnectivityManager; **import** android.net.NetworkInfo; **import** android.os.AsyncTask; **import** android.os.Bundle; **import** android.view.View; **import** android.widget.Button; **import** android.widget.EditText; **import** android.widget.Toast;  **import** androidx.appcompat.app.AppCompatActivity;  *//En esta clase se define la apertura de la puerta, si los codigos son correctos, manda a ejecutar al NodeMCU* **public class** CodigoActivity **extends** AppCompatActivity {  **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** EditText **txt1**;  **private** String **codigo**,**user**,**psw**;  Button **btnEntrar**;   EditText **txtResultado**;*//para reflejar resultados de luego presionar el boton* @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_codigo***);  **txt1** = (EditText) findViewById(R.id.***editText***);  **btnEntrar** = (Button) findViewById(R.id.***button***);    }    **public void** ingreso(View view){  String[] resultadoSQL = **null**;  **try** {  String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();   **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"2"**,  **txt1**.getText().toString()   };   **if**(**txt1**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe ingresar todos los datos."**, Toast.***LENGTH\_LONG***).show();  }**else**{  resultadoSQL = **new** AsyncQuery().execute(**datosConexion**).get();  Toast.*makeText*(CodigoActivity.**this**,**"Conexión Establecida"**, Toast.***LENGTH\_LONG***).show();  **psw**=resultadoSQL[1];  **if**(**txt1**.getText().toString().equals(**psw**)){  **new** AsyncQuery().execute(**datosConexion**);  Intent i = **new** Intent(**this**, MapsActivity.**class** );  i.putExtra(**"Lat"**,-2.0641032);  i.putExtra(**"Long"**,-79.8987751);  startActivity(i);  Toast.*makeText*(**this**, **"Ingreso Exitoso."**, Toast.***LENGTH\_LONG***).show();  }**else**{  Toast.*makeText*(**this**, **"Se ha producido un error."**, Toast.***LENGTH\_LONG***).show();  }  }     }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  }    } |

* CodigoActivity (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  android:orientation="vertical" android:layout\_width="match\_parent"  android:layout\_height="match\_parent"**>   <**TextView  android:id="@+id/textView"  android:layout\_width="310dp"  android:layout\_height="82dp"  android:layout\_marginStart="45dp"  android:layout\_marginTop="35dp"  android:text="Urban Access Residente"  android:textAlignment="center"  android:textSize="30sp"  android:textStyle="bold"** />   <**ImageView  android:id="@+id/imageView3"  android:layout\_width="match\_parent"  android:layout\_height="157dp"  app:srcCompat="@drawable/logoua"** />   <**EditText  android:id="@+id/editText"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="98dp"  android:layout\_marginTop="34dp"  android:ems="10"  android:hint="Clave de Acceso:"  android:inputType="textPersonName"** />    <**Button  android:id="@+id/button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="160dp"  android:layout\_marginTop="100dp"  android:onClick="ingreso"  android:text="ENTRAR"** />   </**LinearLayout**> |

* MapsActivity (Java)

Clase para una vez ingresado el residente se le direcciona a un mapa de google y se le muestra la ubicación de su villa registrada.

|  |
| --- |
| **package** com.grupo5.urbanaccessresident;  **import** androidx.fragment.app.FragmentActivity;  **import** android.os.Bundle;  **import** com.google.android.gms.maps.CameraUpdateFactory; **import** com.google.android.gms.maps.GoogleMap; **import** com.google.android.gms.maps.OnMapReadyCallback; **import** com.google.android.gms.maps.SupportMapFragment; **import** com.google.android.gms.maps.model.LatLng; **import** com.google.android.gms.maps.model.MarkerOptions;  **public class** MapsActivity **extends** FragmentActivity **implements** OnMapReadyCallback {   **private** GoogleMap **mMap**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_maps***);  *// Obtain the SupportMapFragment and get notified when the map is ready to be used.* SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()  .findFragmentById(R.id.***map***);  mapFragment.getMapAsync(**this**);    }   @Override  **public void** onMapReady(GoogleMap googleMap) {  **mMap** = googleMap;   Bundle extras=getIntent().getExtras();  Double lat=extras.getDouble(**"Lat"**);  Double longitud=extras.getDouble(**"Long"**);   *// Agrego un marcador en la villa.* LatLng villa = **new** LatLng(lat, longitud);  **mMap**.addMarker(**new** MarkerOptions().position(villa).title(**"\*\*\* Urban Access \*\*\*"**));  **mMap**.moveCamera(CameraUpdateFactory.*newLatLngZoom*(villa,15));  }  } |

* Google Api para servicio de Mapa - google\_maps\_api.xml

|  |
| --- |
| <resources>  <string name="google\_maps\_key" templateMergeStrategy="preserve" translatable="false">AIzaSyAZhtuk8teTMzY7dKh\_YSIzYBQPmPFDTZs</string>  </resources> |

* activity\_maps (XML)

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <fragment xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:map="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:id="@+id/map"  android:name="com.google.android.gms.maps.SupportMapFragment"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".MapsActivity" /> |

* AccesoActivityResidente (Java)

En esta clase se crea el método ingresar para posteriormente añadirselo a un botón el cual tiene como objetivo validar que exista el usuario ingresado y a su vez, que la contraseña del usuario coincida con la que se encuentra en la base de datos.

|  |
| --- |
| **package** com.grupo5.urbanaccessresident;  **import** android.content.Intent; **import** android.os.Bundle; **import** android.view.View; **import** android.widget.Button; **import** android.widget.EditText; **import** android.widget.Toast;  **import** androidx.appcompat.app.AppCompatActivity;  **public class** AccesoActivityResident **extends** AppCompatActivity {  **private** String **serverIP** = **"remotemysql.com"**;  **private** String **port** = **"3306"**;  **private** String **userMySQL** = **"WAgQ6gLNl1"**;  **private** String **pwdMySQL** = **"oPAE3gP5fa"**;  **private** String **database** = **"WAgQ6gLNl1"**;  **private** String[] **datosConexion** = **null**;  **private** EditText **txt1**,**txt2**;  **private** String **codigo**,**user**,**psw**,**id\_Usuario**;   Button **btnInicio**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_acceso\_residente***);  **txt1** = (EditText) findViewById(R.id.***editText***);  **txt2** = (EditText) findViewById(R.id.***editText2***);  **btnInicio** = (Button) findViewById(R.id.***button***);  }    **public void** ingreso(View view){  String[] resultadoSQL = **null**;  **try** {  *//Toast.makeText(this, "Debe ingresar todos los datos.", Toast.LENGTH\_LONG).show();* String driver = **"com.mysql.jdbc.Driver"**;  Class.*forName*(driver).newInstance();  *//Toast.makeText(this, "lalaos los datos.", Toast.LENGTH\_LONG).show();* **datosConexion** = **new** String[]{  **serverIP**,  **port**,  **database**,  **userMySQL**,  **pwdMySQL**,  **codigo**=**"1"**,  **txt1**.getText().toString(),  **txt2**.getText().toString()   };   **if**(**txt1**.getText().toString().equals(**""**)){  Toast.*makeText*(**this**, **"Debe ingresar todos los datos."**, Toast.***LENGTH\_LONG***).show();  }**else**{  resultadoSQL = **new** AsyncQuery().execute(**datosConexion**).get();  Toast.*makeText*(AccesoActivityResident.**this**,**"Conexión Establecida"**, Toast.***LENGTH\_LONG***).show();  **user**=resultadoSQL[0];  **psw**=resultadoSQL[1];  **id\_Usuario**=resultadoSQL[2];  **if**(**txt2**.getText().toString().equals(**psw**)){  Intent i = **new** Intent(**this**, CodigoActivity.**class** );  i.putExtra(**"nombre"**,**user**);  i.putExtra(**"contraseña"**,**psw**);  i.putExtra(**"id\_Usuario"**,**id\_Usuario**);  startActivity(i);  Toast.*makeText*(**this**, **"Ingreso Exitoso."**, Toast.***LENGTH\_LONG***).show();  }**else**{  Toast.*makeText*(**this**, **"Se ha producido un error."**, Toast.***LENGTH\_LONG***).show();  }  }    }**catch**(Exception ex)  {  Toast.*makeText*(**this**, **"Error: "** + ex.getMessage(), Toast.***LENGTH\_LONG***).show();  }  } } |

* AccesoActivityResidente (XML)

|  |
| --- |
| *<?***xml version="1.0" encoding="utf-8"***?>* <**androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  tools:context=".AccesoActivityResident"**>   <**TextView  android:id="@+id/textView"  android:layout\_width="305dp"  android:layout\_height="82dp"  android:layout\_marginStart="45dp"  android:layout\_marginTop="35dp"  android:text="Urban Access Residente"  android:textAlignment="center"  android:textSize="30sp"  android:textStyle="bold"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toTopOf="parent"** />   <**ImageView  android:id="@+id/imageView"  android:layout\_width="145dp"  android:layout\_height="129dp"  android:layout\_marginStart="133dp"  android:layout\_marginTop="18dp"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/textView"  app:srcCompat="@drawable/logoua"** />   <**EditText  android:id="@+id/editText"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="98dp"  android:layout\_marginTop="34dp"  android:ems="10"  android:hint="Ingrese su Usuario:"  android:inputType="textPersonName"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/imageView"** />   <**EditText  android:id="@+id/editText2"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="98dp"  android:layout\_marginTop="18dp"  android:ems="10"  android:hint="Ingrese su Contraseña:"  android:inputType="textPassword"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText"** />   <**Button  android:id="@+id/button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginStart="144dp"  android:layout\_marginTop="56dp"  android:onClick="ingreso"  android:text="INICIO"  app:layout\_constraintStart\_toStartOf="parent"  app:layout\_constraintTop\_toBottomOf="@+id/editText2"** /> </**androidx.constraintlayout.widget.ConstraintLayout**> |

* SplashActivity

En esta clase se presenta la imagen o logo del grupo apenas se ingresa a la app.

|  |
| --- |
| **import** android.content.Intent; **import** android.os.Bundle; **import** android.os.Handler;  **public class** SplashActivity **extends** AppCompatActivity {   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_splash***);   **new** Handler().postDelayed(**new** Runnable() {  @Override  **public void** run() {   Intent intent= **new** Intent(SplashActivity.**this**, AccesoActivityResident.**class**);  startActivity(intent);   }  },4000);  } } |

## **Código en IDE Arduino para ESP8266 en NodeMCU**

En este código se detalla la codificación del ESP8266 que el IDE de Arduino le carga al módulo.

|  |
| --- |
| #include <ESP8266WiFi.h>  #include <Stepper.h>  //Le definimos a la red en la cual se va a conectar  #ifndef STASSID  #define STASSID "nombre\_red"  #define STAPSK "contraseña\_red"  //variable de led  #define led1 4  #define sensor 2  int i=0;  int j=0;  //VARIABLES DE STEPPER  const int stepsPerRevolution = 200;  Stepper myStepper(stepsPerRevolution,14,13,12,15);  //Stepper miStepper(pasosporRev,8,10,9,11);  #endif  const char\* ssid = STASSID;  const char\* password = STAPSK;  WiFiServer server(80);  void setup() {  // put your setup code here, to run once:  pinMode(sensor,INPUT);  Serial.begin(115200);  delay(10);  //prepara led  pinMode(led1, OUTPUT);  //PREPARA AL MOTOR  myStepper.setSpeed(60);    // Conectando con red WIFI  Serial.println();  Serial.println();  Serial.print(F("Conectando con "));  Serial.println(ssid);  WiFi.mode(WIFI\_STA);  //WiFi.config(ip,dns,gateway,subnet);  WiFi.begin(ssid, password);  while (WiFi.status() != WL\_CONNECTED) {  delay(500);  Serial.print(F("."));  }    Serial.println();  Serial.println(F("WiFi conectado"));  // Start the server  server.begin();  Serial.print(F("Servidor Iniciado en: "));  // Print the IP address  Serial.println(WiFi.localIP());  }  void loop() {  // Check if a client has connected  WiFiClient client = server.available();  if (!client) {  return;  }  Serial.println(F("Cliente encontrado"));  while (!client.available()){  delay(1);  }  // Read the first line of the request  String request = client.readStringUntil('\r');  Serial.println(F("request: "));  Serial.println(request);  client.flush();  //el sensor lee el comando que le se envia junto con su IP y ejecuta según este comando, en este caso se le envia la palabra abrir para ejecutar el motor  //abrir puerta - girar motor  if(request.indexOf("abrir")!= -1 ){  //digitalWrite(led1, !digitalRead(led1));  digitalWrite(led1, !digitalRead(led1));  delay(100);  while (i<129){  myStepper.step(-stepsPerRevolution);  i++;  if(i==128){  j=0;  }  }  delay(10000);//espera 10 segundos y comienza a cerrar  //para cerrar y ya va detectando el infrarrojo a la vez  while(i==129 && digitalRead(sensor)==HIGH && j<129){  myStepper.step(stepsPerRevolution);  j++;  if(j==128){  i=0;  }    }  delay(1000);  digitalWrite(led1, !digitalRead(led1));    }  //conexiones con el servidor  client.println("HTTP/1.1 200 OK");  client.println("Content-Type: text/html");  client.println("");  //commandos del sensor  if (digitalRead(led1)){  client.print("Pabierta");  } else {  client.print("Pcerrada");  }  client.print(",");  delay(1);  // The client will actually be \*flushed\* then disconnected  // when the function returns and 'client' object is destroyed (out-of-scope)  // flush = ensure written data are received by the other side  Serial.println(F("Disconnecting from client"));    } |

# 

# **Análisis de presupuesto**

Para el presupuesto del sistema de aplicativos sincronizados de manera remota a través de una base de datos en un control de apertura se miran diferentes factores, desde los equipos utilizados, hasta el gasto de programación y codificación de funcionalidad de los aplicativos, los cuales detallamos a continuación.

|  |  |  |  |
| --- | --- | --- | --- |
| Tabla de Equipos Utilizados | | | |
|  | **Descripcion** | **Cantidad** | **Precio en Dólares** |
| Equipos y Materiales para Prototipo | Motor Stepper con Driver | 1 | 6.25 |
| NodeMCU con ESP8266 | 1 | 11.00 |
| Sensor Infrarojo | 1 | 2.00 |
| Fuente de Alimentación Externa | 1 | 5.00 |
| Madera Plywood – Plancha | 1 | 5.00 |
| Programadores y recursos | Base de Datos | 1 | 0.00 |
| Aplicativos Moviles y codigos fuentes de los equipos | 2 | 15.00 |
| TOTAL |  |  | **44.25** |

Tomar en cuenta que este presupuesto está dado para el prototipo de la presentación del proyecto, estos valores varían dependiendo de la sectorización debido a que para implementar este mecanismo se necesitan de equipos más grandes para su uso, como el motor con mayor potencia para mover una cremallera y una puerta de mayor tamaño, además de un sensor infrarrojo de mayor alcance, pero para la prueba de funcionalidad este equipo está presupuestado bajo este valor referencial.

Si el propietario de la urbanización desea una base de datos protegida totalmente, este servicio se los ofrecen diferentes plataformas y servidores, según lo que estén dispuestos a pagar; sin embargo, como en esta actividad se guardan única y exclusivamente registros de todos los agentes que intervienen a nivel de textos y números esto no amerita una protección totalmente restringida y esta base de datos puede funcionar de igual manera para este caso, a diferencia de si guardar archivos protegidos o de carácter privado y de gran tamaño, que no es el caso para este sistema de ingreso.

Además, al ameritar un mecanismo de mayor tamaño con equipos nuevos y de mayor alcance y funcionalidad el trabajo del programador e ingeniero también es de mayor remuneración, debido a que amerita un mecanismo más eficiente.

# **Conclusiones**

Finalizado el proyecto de ingeniería se concluye:

* El manejo de bases de datos remotas nos permite realizar múltiples conexiones de manera eficaz y segura para implementar según sea nuestra conveniencia, en el caso del sistema, permite conectar y comunicarse entre dos aplicativos móviles, además de que puede conectarse junto con los diferentes sensores para trabajar de forma simultánea.
* Android Studio, como desarrollador de aplicativos, tiene múltiples ventajas sobre otro software, debido a que permite realizar conexiones importantes como el acceso a internet, comunicación con plataformas como google, comunicación con diferentes sensores como el módulo NodeMCU para trabajar de manera remota a través de internet.
* Este sistema puede significar un ejemplo de lo que se conoce como IoT, internet de las cosas, debido a que todo se esquematiza y trabaja de manera remota y con diferentes comunicaciones en la red.

# **Referencias bibliográficas**

* Crespo, J. (6 de Enero de 2014). *Aprendiendo Arduino*. Obtenido de Aprendiendo Arduino: https://aprendiendoarduino.wordpress.com/2017/06/19/ejemplos-practicos-arduino/
* EquipoNodeMCU. (2014). *NodeMCU*. Obtenido de NodeMCU: https://www.nodemcu.com/index\_en.html
* Llamas, L. (20 de Agosto de 2019). *DETECTOR DE OBSTÁCULOS CON SENSOR INFRARROJO Y ARDUINO*. Obtenido de DETECTOR DE OBSTÁCULOS CON SENSOR INFRARROJO Y ARDUINO: https://www.luisllamas.es/detectar-obstaculos-con-sensor-infrarrojo-y-arduino/
* Macho, J. (Noviembre de 2008). *Prometec*. Obtenido de Prometec: https://www.prometec.net/esp8266-display-i2c/
* Mendosa, C. (19 de Julio de 2015). *StepperMotor*. Obtenido de StepperMotor: https://steppermotor.top/stepper-h-arduino/
* Peralta, A. (Febrero de 2019). *Rackspace*. Obtenido de Rackspace: https://support.rackspace.com/how-to/mysql-connect-to-your-database-remotely/