**Manual Técnico - SLIM**

1. **Introducción:**

**La aplicación móvil Miamiga, se emerge para dar apoyo y denuncia ante problemas psicológicas, agresiones y abusos. Con roles definidos, los usuarios normales presentan denuncias de feminicidio, mientras que los supervisores proporcionan ayuda y seguimiento, detrás de todo, hay un director, a través de un sistema web, coordina estratégicamente la asignación de casos a los supervisores.**

1. **Descripción del proyecto:**

**WEB**

**Usuario Director :**

* Editar su perfil
* Crear supervisor
* Crear Director
* Ver todos los casos en proceso
* Asignar un supervisor a un caso
* Ver Reporte de los casos finalizados
* Ver un dashboard donde le muestra cantidad de usuario, supervisores y casos

**MOVIL:**

**Usuario Normal:**

* Editar su perfil
* Hacer la denuncia(caso) Agregando evidencias al caso y datos del Denunciado
* Mandar alertas después de hacer la denuncia

**Usuario Supervisor:**

* Editar su perfil
* Ver los casos y detalles del mismo que se les fueron asignados
* Agregar evidencia a un caso
* Conclusión del caso
* Finalizar un caso

1. **Roles / integrantes**

* Jerson Climaco Solis Yucra: DB Architect, Developer
* Luis Alejandro Ugarte Lujan: Git Master, Developer
* Johan Pablo Merida Hinojosa: Team Leader, Developer

1. **Arquitectura del software:**

**4.1. Estructura General del Proyecto:**

* + **Módulos**
  + **WEB: Contiene la logica y componentes necesarios para la interfaz web.**
  + **MOVIL: Contiene la logica y componentes necesarios para la aplicación movil.**

**4.2. Interacción entre Componentes:**

* **WEB-MOVIL:**
* **Se comparten elementos de diseno y logica de presentacion entre la version web y movil para una experiencia coherente.**
* **MOVIL-FIREBASE:**
* **La aplicación movil se comunica con Firebase para realizar un creacion de un caso esto debe llegar primero al director luego debe ser asignado al supervisor lo cual ambos (usuario normal y supervisor) usan la aplicación movil.**
* **WEB-FIREBASE:**
* **La interfaz web se conecta a Firebase para hacer acciones como CRUDs la parte web maneja un director y administra toda la sistema.**

**4.3. Patrones de Diseno:**

* + **Singleton (para Firebase)**
  + **Observer (observaciones en tiempo real)**

1. **Requisitos del sistema:**

Hardware:

* Microsoft® Windows® 7/8/10 (64-bit)
* 4 GB RAM minimo, 8 GB RAM recomendado
* 2 GB de disponibilidad,
* 4 GB recomendado (500 MB para el IDE + 1.5 GB para Android SDK y emulador de sistema de imagen)
* 1280 x 800 minimo resolucion

1. **Instalación y configuración:**

* Necesitamos tener instalado visual code como entorno de desarrollo en nuestra maquina: <https://code.visualstudio.com/>
* Para la instalacion necesitamos primero tener instalado Flutter SDK en nuestro entorno lo cual es el siguiente link: <https://docs.flutter.dev/get-started/install/windows>
* La version de flutter que actualmente tenemos en nuestro proyecto es el 3.13.9 de la canal estable
* Flutter SDK lo ponemos en nuestro disco ejemplo ‘C:’ o ‘D:’ creamos una carpeta lo cual nos guste ejemplo es ‘src’ y adentro de esa carpeta estará nuestro flutter lo extraemos adentro del ‘src’ y estará todo.
* Verificamos si todo esta bien corriendo siguientes comandos como: ‘flutter doctor’ y estará las cosas que tenemos que habilitar para usarlo sin problemas.
* Necesitamos configurar Firebase como entorno de base de datos lo cual es el siguiente enlace: <https://firebase.google.com/docs/android/setup>
* Para instalar el proyecto nos tenemos que dirigir a GitHub lo cual el proyecto se encuentra en el siguiente link: <https://github.com/luisUgarte/SLIM.git>
* Podemos instalar Firebase con CLI que es (Command Line Interface) tenemos la opción de instalación como ‘standalone binary’ o ‘npm’
* Si instalamos el CLI de Firebase entonces necesitamos logearnos para poder tener en nuestro entorno de máquina.
* Tenemos que activar FlutterFire con el siguiente comando ‘dart pub actívate flutterfire\_cli’
* Usar FlutterFire CLI para configurar las apps de flutter para conectar a firebase necesitamos correr el siguiente comando: ‘flutterfire configure’
* Para inicializar firebase en nuestras apps tenemos que poner el siguiente comando: ‘flutter pub add firebase\_core’
* Averiguar con el comando para ver si están actualizados: ‘flutterfire configure’
* Una vez hecho bien en nuestro entorno de Flutter por archivos ‘lib/main.dart’ tenemos que importar el siguiente: import ‘package:firebase\_core/firebase\_core.dart’; --- import ‘firebase\_options.dart’;
* En el ‘lib/main.dart’ igual tenemos que inicializar con un código lo cual es: await Firebase.initializeApp(options: DefaultFirebaseOptions.currentPlatform,);
* Correr la aplicación de flutter con el comando: flutter run

1. **PROCEDIMIENTO DE HOSTEADO / HOSTING (configuración)**

* Necesitamos tener instalado el Firebase (CLI) luego hay que iniciar sesion de Google
* Acceso del SLIM:
* Correo: [slim022023@gmail.com](mailto:slim022023@gmail.com)
* Contraseña: Miamiga2023

1. **GIT :**

El proyecto se encuenta en nuestra repositorio con el rama master y el carpeta se llama miamiga\_app y se encuentra todo nuestra desarrollo del aplicación movil y sistema web. La apk tambien se encuentra en el repositorio lo cual esta el ejecutable del aplicación movil.

1. **Personalización y configuración:**

* Puede personalizar su manera de autenticar
* Crearse cuenta de un usuario normal o supervisor
* Puede configurar su API key del mapa de Google
* Link para configurar Google Maps: https://mapsplatform.google.com/

1. **Seguridad**:

* Manejo de roles para acceso solo a funciones necesarios de los usuarios
* Encriptación de contraseña del usuario
* Usa tokens de ID de Google para autenticar usuarios
* Validación de Datos

1. **Depuración y solución de problemas:**

Para buscar los errores podra debugear y ver que mensajes tiene si por ejemplo lleno todos los datos y en el ‘debug console’ puede observar que errores se encuentra si talvez se guarda bien o no.

1. **Glosario de términos:**

* Proveedores de Autenticación: usar un correo y contraseña para registrarse, o usar terceros para autenticar como Google.
* Manejo de Roles: poder iniciarse con un rol de usuario normal o supervisor.
* CRUD: Create / Read / Update / Delete. Procedimientos para hacer la manipulacion de registros en la base de datos

1. **Referencias y recursos adicionales:**

* *Documentacion de widgets en Flutter:* <https://flutter.dev/docs/development/ui/widgets-intro>.
* Documentacion de autenticacion en Firebase: <https://firebase.google.com/docs/auth>

1. **Herramientas de Implementación:**

* Framework: Flutter
* Lenguaje de Programacion: Dart
* Backend: Firebase

1. **Bibliografía**
   * *Building user interfaces with Flutter*. (n.d.). Retrieved from Flutter: <https://docs.flutter.dev/ui>
   * *Firebase Authentication*. (2023). Retrieved from Firebase: <https://firebase.google.com/docs/auth>