# Reporte de Servicio Social

Prestador: Rodrigo Urtecho Quintal

Responsable: Dr. Jorge Alberto Ríos Martínez

Proyecto: DESARROLLO DE TECNOLOGÍA PARA LA SALUD DIRIGIDA A LA POBLACIÓN DE YUCATÁN CON DIABETES Y OTRAS ENFERMEDADES CRÓNICAS.

## Introducción

En el presente documento se hace la recolección de los resultados de las actividades solicitadas durante la prestación para la actividad del Servicio Social en el proyecto: “DESARROLLO DE TECNOLOGÍA PARA LA SALUD DIRIGIDA A LA POBLACIÓN DE YUCATÁN CON DIABETES Y OTRAS ENFERMEDADES CRÓNICAS”.

## Observaciones del proyecto anterior

**Proyecto:** Diabetics Awareness

**Descripción:** El proyecto a analizar se trata de una aplicación móvil desarrollada con anterioridad para el proyecto en cuestión para el Servicio Social, desconozco la intención que se planteaba con esta aplicación, es decir, desconozco de si se trata de un prototipo o el inicio de una aplicación funcional que se esperaba sea mantenida en un futuro. El proyecto no cuenta con mayor documentación más que el código en sí de la aplicación.

La aplicación fue desarrollada utilizando el IDE de Google, Android Studio.

**Módulos identificados:** Login, Registro, Menú, Home, Buscador, Examen, Perfil, Selec. Edo, Control, Educación, Metas, Recomendación, Podómetro.

**Notas acerca del funcionamiento:**

- El módulo del Login permite el acceso de un usuario registrado previamente de forma satisfactoria. El método de autenticación por Email y Contraseña es el único que parece funcionar. Los métodos OAuth para acceso con Google o Facebook parecen no funcionar correctamente, pues se realiza la conexión con las API, pero no se realiza la autenticación. Desconozco si el problema anterior se debe principalmente a que la prueba de la aplicación la realicé en un emulador o a otros factores.

- El módulo de Registro: El registro de una nueva cuenta para acceso a la aplicación utilizando un correo electrónico y contraseña funciona correctamente. Se ha implementado un proceso de verificación de la dirección de correo electrónico mediante el envío de un correo a la cuenta proporcionada para el acceso.

- Menú: El menú no permite la funcionalidad adecuada para cerrar sesión en la aplicación.

- Home: El módulo de Home muestra de forma correcta las posibles opciones o funcionalidades a las cuales puede uno acceder. Los botones funcionan correctamente y envían al usuario a la vista correspondiente.

- Buscador: El módulo o funcionalidad está implementado correctamente pero no está completo. Permite la identificación de los elementos buscados, pero no la interacción con el elemento encontrado.

- Selec. Edo: Menú desplegable en el que se muestran las 32 Entidades Federativas de México.

- Examen: Módulo interactivo de preguntas y respuestas. Consta de pruebas cortas de pregunta y respuesta (opción múltiple), se valida la elección del usuario.

- Perfil: El módulo de perfil permite la visualización y modificación de la información proporcionada por el usuario.

\*Los demás módulos sólo fueron considerados y se les designó una vista, sin embargo, no son funcionales\*

**Notas desde perspectiva UI/UIX**

La aplicación fue desarrollada requiriendo una versión mínima de Android 9 (Pie/API28), abarcando el 81.2% de los dispositivos disponibles actualmente en el mercado, un poco por debajo de lo que se recomendaría (85%) para el alcance pero el mínimo recomendable para poder permitir la conexión a las API actualizadas y recientemente desarrolladas.

El Login es adecuado en cuando a la estructuración de los elementos, pero el aspecto visual puede mejorarse un poco más.

El Módulo de Registro es el que presenta más elementos que se pudieran mejorar. La adición de los campos para proporción de información referentes al estado del paciente aún siendo opcionales pudiera no ser el mejor momento para ser solicitados. En cuanto a estos campos, se pudiera mejorar el llenado de algunos como por ejemplo la fecha de nacimiento la cual tiene que ser escrita por el paciente cuando existen mejores formas de solicitarla, lo mismo para elementos como el tipo de diabetes del paciente o los medicamentos que toma al igual que su tipo. Si son descritos incorrectamente o el nombre proporcionado no es el correcto, se pudiera afectar el desempeño de futuras funcionalidades de la aplicación y provocar la afectación de la salud del usuario.

Lo mejor sería limitar o establecer opciones de elección para dicha información.

La interfaz de Home presenta demasiados elementos que se pierden de la vista del usuario. Se requiere del desplazamiento para poder visualizarlos. Por lo que, se recomendaría poner dichos elementos en un menú desplegable, permitiendo una interfaz más limpia.

La funcionalidad de búsqueda puede ser completamente reemplazada por el menú mencionado con anterioridad.

La selección de estados de la república mexicana no parece tener utilidad alguna. Si lo que se desea es conocer la entidad federativa en la cual se encuentra dicho paciente se pudiera agregar dicha opción dentro de la información a proporcionar para el llenado del perfil.

Se pueden y se recomendaría hacer mejoras visuales y de funcionalidad general de la aplicación. Habría información que pudiera ser presentada directamente al usuario sin necesidad de que este se encuentre registrado y haya iniciado sesión. De esta forma se lograría que si el usuario desea acceder a las funcionalidades de valor de la aplicación (recomendaciones y metas) este deba proporcionar sus datos para la creación de un perfil, se procese dicha información y se realicen la funcionalidad de los módulos que utilicen la información.

Por otro último, la funcionalidad del examen pudiera ser hasta cierto punto molesta para el usuario. Se debe definir de mejor forma el objetivo de la aplicación, ¿se desea entretener y educar o educar y apoyar al usuario?

## Investigaciones

¿Qué plataforma de desarrollo es conveniente para el proyecto?

Parte de los requisitos especificados para el proyecto es que la aplicación móvil sea desarrollada para teléfonos celulares de gama baja y que sea compatible con dispositivos Android y dispositivos con iOS.

Inicialmente se desarrolló un primer proyecto haciendo uso de la plataforma de desarrollo Android Studio.

**Android Studio**

Esta plataforma es ideal para el desarrollo de aplicaciones nativas para dispositivos Android, permite un buen manejo de las librerías y funcionalidades de los dispositivos con dicho sistema operativo, sin embargo, no permite la escalabilidad de compatibilidad con dispositivos con el sistema operativo iOS. Este es una plataforma de desarrollo soportada por Google que hace uso de Java y Kotlin como los lenguajes de programación principales para la construcción de las aplicaciones. La plataforma es ideal para la construcción de aplicaciones móviles que puedan ser ejecutados en dispositivos de gama baja; de hecho, las pruebas de ejecución pueden realizarse con un simulador dentro de la misma plataforma. Los dispositivos simulados poseen las características de la versión del sistema operativo Android que seleccionado y podrá observarse el desempeño de la aplicación en dispositivos con características similares.

**React Native**

Es una plataforma de desarrollo soportada por Meta la cual tiene como característica principal la capacidad de desarrollo de aplicaciones compatibles con Android y iOS, esto al permitir el desarrollo de aplicaciones híbridas, las cuales no requieren el manejo totalitario de las características de los dispositivos en donde se desea implantar la aplicación. Se considera una plataforma de desarrollo muy sencilla de aprender a utilizar utilizando JavaScript como principal lenguaje de programación para la construcción de las aplicaciones.

**Flutter**

Flutter es la reciente iniciativa de Google que busca ser competencia directa de la plataforma de desarrollo de Meta. Es una plataforma sencilla de utilizar y que permite el desarrollo de aplicaciones con características similares a las que posee React Native. Una de las principales diferencias con React Native es la implementación de un lenguaje de programación reciente que tiene por nombre Dart. El lenguaje de programación Dart es de paradigma Orientado a Objetos y posee características similares a los lenguajes Java, C# y C++, por lo anterior, es que se considera que es un lenguaje sencillo de utilizar y aprender.

Una de las desventajas de la plataforma de desarrollo Flutter es la incapacidad de poder utilizar de forma eficiente las características de cada uno de los sistemas operativos a los cuales se enfoca su desarrollo, es por ello, que no es una buena opción para abarcar el aspecto de la ejecución de la aplicación en teléfonos de baja gama. Sin embargo, debido a la creciente popularidad en el uso de la plataforma, así como, su facilidad de aprendizaje sería bueno considerar esta plataforma para el mantenimiento del sistema.

OAuth con Facebook

Bajo la suposición de que se ha definido Flutter como la plataforma a utilizar para el desarrollo de la aplicación, se define la siguiente opción para la autenticación externa a la aplicación usando una cuenta de Facebook.

Para habilitar la autenticación externa es necesario utilizar unas herramientas proporcionadas por Meta para la configuración de nuestro proyecto y habilitar este método de autenticación. Para lo anterior es necesario utilizar Meta for Developers. No es necesaria la creación ni la posesión de una cuenta de Facebook institucional o de negocio.

El uso de la API de autenticación es gratuito tanto para implementación como para uso, e el caso que se deseen otro tipo de beneficios o características, entonces si se requerirá de un costo adicional. Con funcionalidades o beneficios extra se hace referencia a que, por ejemplo, el uso gratuito de la API permite la conexión y uso del método de autenticación un total de , en una hora. Siendo n el número de usuarios de la aplicación.

Para el acceso a los paquetes de herramientas es necesaria la creación de un perfil de desarrollador en Meta y de igual forma, se requiere un registro de la aplicación en la página de las herramientas para la generación de un identificador que será utilizado en el método de autenticación dentro de la app.

Mobile Push Notification

Para el envío y difusión de notificaciones en aplicaciones multiplataforma, existen diferentes alternativas que pudieran ser de utilidad para el caso del proyecto y sus propósitos. Una de las alternativas es usando Native Notify, sin embargo, esta opción permite únicamente el envío de notificación de forma gratuita a 10 subscritores, por lo que, a partir del subscriptor número once se requiere tener un plan para el envío de notificaciones depende de la dimensión de la aplicación, el uso de esta opción pudiera ser o no considerada.

Otra de las opciones es la implementación de Firebase la cual es una plataforma de APIs sencillas e integradas desarrollada por Google, en teoría y según lo encontrado, en comparación la opción de Native Notify, la configuración de Firebase es un poco más tardada y tediosa, sin embargo, para los efectos del proyecto pareciera ser que es posible implementarlo sin requerir la compra de alguno de los planes que ofrece: <https://firebase.google.com/pricing?hl=es-419>

Ambas opciones requieren de la configuración e implementación en código la conexión con las plataformas para enviar las notificaciones desde la plataforma en sí.

Las notificaciones enviadas a través de esta plataforma funcionan a través de la implementación de un patrón de diseño observer. En cuanto se determina que existe un nuevo mensaje o interacción con el sistema desde la plataforma en el navegador web, ésta se comunica con la aplicación dentro de un dispositivo móvil y le hace llegar el mensaje o notificación que fue enviado desde la plataforma por alguna persona designada para dicha tarea. Sin embargo, el patrón observer funciona con una lista de suscriptores para recepción de la nueva información, por lo que, esta información le llagará a todos los dispositivos con la aplicación instalada.