



Manual Técnico de la Aplicación RoutesHello

Materia:

Desarrollo y Programación de Software Multiplataform

Ingeniera:

Karens Medrano

Alumnos:

Hernández Gonzales, Carlos Alejandro	#HG231549
Quinteros Guevara, Tatiana Yamileth	# QG232367
Vásquez Pineda, Carlos Eduardo	# VP232418

Indice

Contenido

Indice	2
1. Introducción	3
2. Arquitectura de la Aplicación.....	4
3. Configuración del Entorno de Desarrollo	5
4. Estructura del Proyecto	6
5. Tecnologías Utilizadas	7
6. Funcionalidades Principales	8
7. Configuración de API.....	9
8. Resolución de Problemas Comunes	10
9. Licencia.....	11

1. Introducción

El manual técnico de la aplicación "RutasHola" proporciona una descripción detallada de la arquitectura, funcionamiento y tecnologías utilizadas en el desarrollo de la aplicación. Este manual está dirigido a desarrolladores, colaboradores y cualquier persona interesada en comprender la estructura y el funcionamiento interno de la aplicación.

La aplicación "RutasHola" tiene como objetivo principal facilitar la búsqueda de ubicaciones y proporcionar indicaciones de rutas entre dos puntos dados. Utiliza tecnologías modernas y servicios de mapas para ofrecer una experiencia de usuario intuitiva y práctica.

Este manual proporciona una visión general de la arquitectura de la aplicación, incluida la estructura del proyecto, las tecnologías utilizadas y las funcionalidades principales. También aborda aspectos relacionados con la configuración del entorno de desarrollo, la configuración de API, la resolución de problemas comunes y el mantenimiento futuro de la aplicación.

La documentación está diseñada para ayudar a los desarrolladores a comprender el código, realizar contribuciones y realizar cambios en la aplicación de manera efectiva. Además, ofrece orientación sobre cómo abordar problemas técnicos, realizar pruebas y mantener la calidad del código a lo largo del tiempo.

2. Arquitectura de la Aplicación

La aplicación "RoutesHello" sigue una arquitectura basada en componentes que permite una fácil escalabilidad y mantenimiento. A continuación se detallan los principales componentes de la arquitectura:

2.1. Componentes Principales:

Pantallas (Screens): Representan las diferentes vistas de la aplicación, como la pantalla de inicio, la pantalla de búsqueda de lugares y la pantalla de visualización de rutas. Cada pantalla está compuesta por componentes de React Native y utiliza navegación para permitir la transición entre ellas.

Componentes Reutilizables: Incluyen elementos de interfaz de usuario compartidos en varias pantallas, como botones, listas y tarjetas de ubicación. Estos componentes están diseñados para mejorar la modularidad y la coherencia visual de la aplicación.

Lógica de Negocio: Se encarga de procesar la lógica de la aplicación, como la obtención de ubicaciones, el cálculo de rutas y la gestión de permisos de geolocalización. Esta lógica se organiza en funciones y módulos separados para facilitar la prueba y el mantenimiento.

2.2. Tecnologías Utilizadas:

React Native: Framework de desarrollo de aplicaciones móviles que utiliza componentes de interfaz de usuario basados en React para crear aplicaciones nativas para iOS y Android.

Expo: Plataforma y conjunto de herramientas que facilita el desarrollo y la implementación de aplicaciones React Native.

APIs de Mapas: Se utilizan APIs de mapas gratuitas para mostrar mapas, obtener ubicaciones y calcular rutas entre dos puntos.

2.3. Flujo de Datos:

Obtención de Ubicaciones: La aplicación permite al usuario buscar ubicaciones utilizando un servicio de geocodificación, que convierte nombres de lugares en coordenadas geográficas.

Cálculo de Rutas: Una vez seleccionadas las ubicaciones de origen y destino, la aplicación utiliza un servicio de enrutamiento para calcular la ruta más corta entre los dos puntos y mostrarla en el mapa.

Interfaz de Usuario: La interfaz de usuario de la aplicación permite al usuario interactuar con el mapa, ver las ubicaciones seleccionadas y obtener indicaciones de ruta.

3. Configuración del Entorno de Desarrollo

Para configurar el entorno de desarrollo de RoutesHello, sigue los siguientes pasos:

3.1. Instalación de Node.js y npm

Asegúrate de tener Node.js y npm instalados en tu sistema. Puedes descargar e instalar la última versión desde el sitio web oficial de Node.js: nodejs.org.

3.2. Instalación de Expo CLI

Expo CLI es una herramienta que facilita el desarrollo de aplicaciones Expo. Puedes instalar Expo CLI globalmente ejecutando el siguiente comando en tu terminal:

```
npm install -g expo-cli
```

3.3. Creación de un Nuevo Proyecto

Para crear un nuevo proyecto RoutesHello, ejecuta el siguiente comando en tu terminal:

```
expo init RoutesHello
```

Esto creará un nuevo proyecto Expo con una estructura de directorios básica.

3.4. Configuración del Proyecto

Una vez creado el proyecto, asegúrate de instalar todas las dependencias necesarias ejecutando el siguiente comando en la carpeta del proyecto:

```
cd RoutesHello
```

```
npm install
```

3.5. Ejecución del Proyecto

Para ejecutar el proyecto en un emulador o dispositivo físico, utiliza el siguiente comando:

```
npm start
```

Esto iniciará el servidor de desarrollo de Expo y mostrará un código QR en tu terminal. Escanea este código QR con la aplicación Expo Go en tu dispositivo móvil para cargar la aplicación.

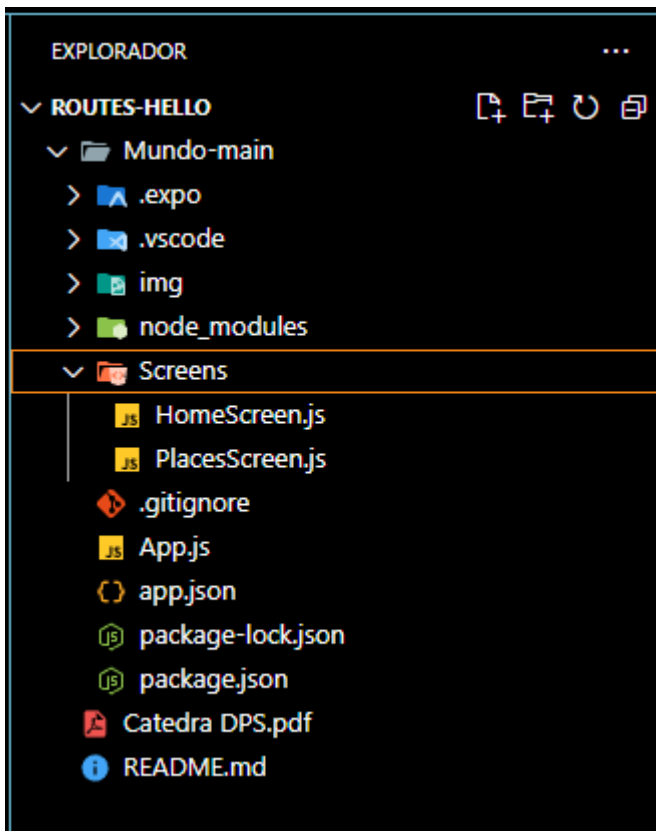
3.6. Desarrollo y Depuración

Una vez cargada la aplicación en tu dispositivo, podrás realizar cambios en el código fuente y ver los resultados en tiempo real. Utiliza las herramientas de depuración de Expo para resolver problemas y optimizar el rendimiento de la aplicación.

Con estos pasos, tendrás configurado correctamente el entorno de desarrollo de RoutesHello y podrás comenzar a desarrollar tu aplicación móvil.

4. Estructura del Proyecto

La estructura del proyecto RoutesHello está organizada de la siguiente manera:



La estructura del proyecto sigue las mejores prácticas de organización de archivos para una aplicación móvil basada en React Native. Cada directorio tiene un propósito específico:

assets/: Almacena recursos estáticos como imágenes y fuentes utilizadas en la aplicación.

Img/: Almacena Imágenes

screens/: Contiene las diferentes pantallas de la aplicación, cada una representando una interfaz de usuario específica.

App.js: Es el punto de entrada de la aplicación donde se configuran y registran los componentes principales.

app.json: Archivo de configuración de Expo que define la configuración específica de la aplicación, como el nombre, el icono y las versiones de Expo.

package.json: Archivo de configuración de npm que contiene todas las dependencias del proyecto y los scripts de inicio y construcción.

5. Tecnologías Utilizadas

RoutesHello utiliza las siguientes tecnologías y herramientas para su desarrollo:

React Native: Framework de desarrollo de aplicaciones móviles basado en JavaScript y React. Permite crear aplicaciones nativas para iOS y Android con un código base común.

Expo: Plataforma y conjunto de herramientas para el desarrollo de aplicaciones móviles con React Native. Proporciona acceso a APIs y componentes nativos, así como herramientas de desarrollo como Expo CLI y Expo Client.

React Navigation: Biblioteca de navegación para React Native que proporciona una forma flexible y fácil de implementar la navegación entre pantallas en la aplicación.

OpenStreetMap: Servicio de mapas de código abierto utilizado para obtener datos de ubicación y calcular rutas y direcciones entre puntos.

API de Geolocalización de OpenStreetMap: API gratuita utilizada para buscar lugares y obtener coordenadas geográficas a partir de direcciones.

API de Direcciones de OpenStreetMap: API gratuita utilizada para obtener indicaciones y rutas entre dos puntos geográficos.

JavaScript (ES6+): Lenguaje de programación utilizado para escribir la lógica de la aplicación.

HTML y CSS: Utilizados para la representación y el diseño de la interfaz de usuario de la aplicación móvil.

Estas tecnologías se combinan para crear una aplicación móvil funcional y fácil de usar que permite a los usuarios buscar lugares, ver información detallada sobre ellos, y obtener indicaciones para llegar a esos lugares desde su ubicación actual.

6. Funcionalidades Principales

RoutesHello ofrece las siguientes funcionalidades principales:

Búsqueda de Lugares: Los usuarios pueden buscar lugares específicos ingresando su nombre en el campo de búsqueda. La aplicación utiliza la API de Geolocalización de OpenStreetMap para buscar lugares basados en la consulta del usuario.

Visualización de Resultados: Una vez que el usuario ingresa una consulta de búsqueda, la aplicación muestra una lista de resultados coincidentes. Cada resultado incluye información detallada sobre el lugar, como su nombre y dirección.

Selección de Lugar: Los usuarios pueden seleccionar un lugar de la lista de resultados para ver más detalles y obtener indicaciones para llegar a ese lugar desde su ubicación actual.

Visualización de Mapa: La aplicación muestra un mapa interactivo que presenta la ubicación actual del usuario y el lugar seleccionado. Utiliza la API de mapas de OpenStreetMap para mostrar el mapa y marcar los puntos relevantes.

Obtención de Indicaciones: Una vez que se selecciona un lugar, la aplicación utiliza la API de Direcciones de OpenStreetMap para calcular y mostrar las indicaciones paso a paso para llegar a ese lugar desde la ubicación actual del usuario.

Estas funcionalidades permiten a los usuarios buscar lugares de interés, visualizarlos en un mapa y obtener indicaciones detalladas para llegar a ellos, lo que hace que la aplicación sea útil para la planificación de viajes y la exploración de áreas desconocidas.

7. Configuración de API

Para el correcto funcionamiento de RoutesHello, se utilizan las siguientes API:

API de Geolocalización de OpenStreetMap: Esta API se utiliza para buscar lugares basados en consultas de usuario. No se requiere una clave de API para acceder a esta funcionalidad, ya que es de uso gratuito.

API de Mapas de OpenStreetMap: Se utiliza para mostrar mapas interactivos en la aplicación, marcando la ubicación actual del usuario y los lugares seleccionados. Esta API tampoco requiere una clave de API y es de uso gratuito.

API de Direcciones de OpenStreetMap: Utilizada para calcular y mostrar indicaciones paso a paso para llegar a un lugar seleccionado desde la ubicación actual del usuario. Esta API tampoco requiere una clave de API y es gratuita.

La configuración de estas API se realiza dentro del código de la aplicación y no requiere ninguna configuración adicional por parte del usuario. Las solicitudes a estas API se realizan de forma transparente a través de las funciones implementadas en la aplicación.

8. Resolución de Problemas Comunes

Problema 1: La aplicación muestra un error al buscar lugares.

Posible solución: Verifique la conexión a Internet en el dispositivo y asegúrese de que la API de Geolocalización de OpenStreetMap esté disponible. Si el problema persiste, intente ingresar consultas de búsqueda más específicas o verifique si hay errores en la implementación del código de búsqueda.

Problema 2: No se muestran indicaciones para llegar al lugar seleccionado.

Posible solución: Asegúrese de que la API de Direcciones de OpenStreetMap esté disponible y que la ubicación actual del usuario esté correctamente configurada. Verifique también si la API devuelve datos válidos para generar indicaciones. Puede haber restricciones de la API o errores en la implementación del código que causen este problema.

Problema 3: El mapa no muestra la ubicación actual del usuario.

Posible solución: Asegúrese de haber otorgado permisos de ubicación a la aplicación en el dispositivo. Verifique también si la API de Mapas de OpenStreetMap está funcionando correctamente. Si el problema persiste, revise la implementación del código relacionado con la obtención de la ubicación del usuario y la configuración del mapa.

Problema 4: La aplicación se bloquea o muestra un comportamiento inesperado.

Posible solución: Verifique si hay errores en el código de la aplicación y solucione cualquier problema de lógica o sintaxis. También asegúrese de que todas las dependencias estén actualizadas y de que no haya conflictos entre los diferentes componentes de la aplicación. Puede ser útil utilizar herramientas de depuración para identificar y resolver problemas específicos.

Problema 5: Las indicaciones o la ubicación en el mapa son inexactas.

Posible solución: Revise la precisión de los datos proporcionados por las API utilizadas. Es posible que los datos de ubicación o las rutas proporcionadas no sean completamente precisos debido a la naturaleza de las API o a errores en la implementación. Considere ajustar los parámetros de búsqueda o mejorar el manejo de datos en la aplicación para obtener resultados más precisos.

Si los problemas persisten después de intentar estas soluciones, comuníquese con el equipo de desarrollo para obtener asistencia adicional.

9. Licencia

RoutesHello © 2024 por Carlos. Este trabajo está licenciado bajo la Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional. Para ver una copia de esta licencia, visita <http://creativecommons.org/licenses/by-nc-sa/4.0/> o envía una carta a Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.