Snap Vision

Technical Installation Manual

Demo 4 – Updated Version

Table of Contents

1. Introduction

Snap Vision is a comprehensive AR-powered indoor and outdoor navigation application built with React Native. Firebase is used for backend services such as user authentication and database storage.
This installation manual outlines everything required to set up Snap Vision the project locally, including:

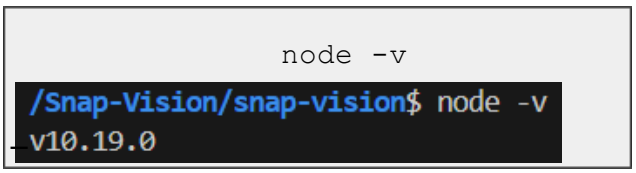
- Cloning the source code
- Installing dependencies and system requirements
- Setting up Firebase services
- Running the application on Android

2. Prerequisites

Ensure the following tools and packages are installed before starting installation:

- Node.js
  Version: >=10.0.0
  Download link: https://nodejs.org/en and install Node.js.
  Verify the installation by running this command in your terminal, as seen in the screenshot below:

```
node -v
```

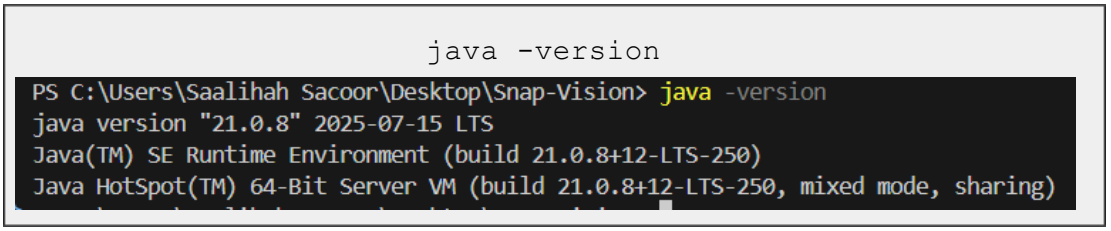


- React Native CLI
  Install globally:

```
npm install -g react-native-cli
```

- Java Development Kit (JDK)
  Version: >=11
  Download link:
  https://www.oracle.com/java/technologies/javase-jdk11-downloads.html and install JDK.
  Verify the installation by running this command in your terminal, as seen in the screenshot below:

```
java -version
```



- Android Studio (for Android development):
  Install Android Studio and configure the Android SDK.
  Ensure the following SDK tools are installed:
  - Android SDK Platform-Tools
  - Android SDK Build-Tools
  - Android Emulator

- Firebase Configuration
  Install Firebase CLI globally:

```
npm install -g firebase-tools
```

  Verify installation:

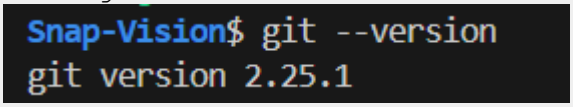
```
firebase --version
```

- Windows Long Path Support
  Windows has a default path length limitation that can cause build failures with React Native projects due to deep node_modules structures.
  1. Enable Long Path Support:
     -Open Group Policy Editor:
     -Press Win + R
     -Type *gpedit.msc* and hit Enter

  2. Navigate to the Policy:
     -Go to:
     Local Computer Policy → Computer Configuration → Administrative Templates → System → Filesystem

  3. Enable Long Paths:
     -Find "Enable Win32 long paths" policy
     -Double-click to open it
     -Select "Enabled"
     -Click "Apply" then "OK"
     -Restart your computer (required for changes to take effect)
- Git
  Version: >= 2.0.0
  Download and install Git.
  Verify the installation by running this command in your terminal, as can be seen in the screenshot below:

```
git --version
```



3. Installation

Step 1: Clone the Repository

1. Open a terminal or command prompt.
2. Run the following command to clone the repository:

```
git clone https://github.com/COS301-SE-2025/Snap-Vision
```

3. Navigate to the project directory in your terminal:

```
cd snap-vision
```

Step 2: Install Dependencies

1. Ensure you have Node.js and npm installed.
2. Install the required packages in your terminal:

```
npm install
```

Step 3: Setting up Android Studio

Based on your path, you should set your environment variables as follows:

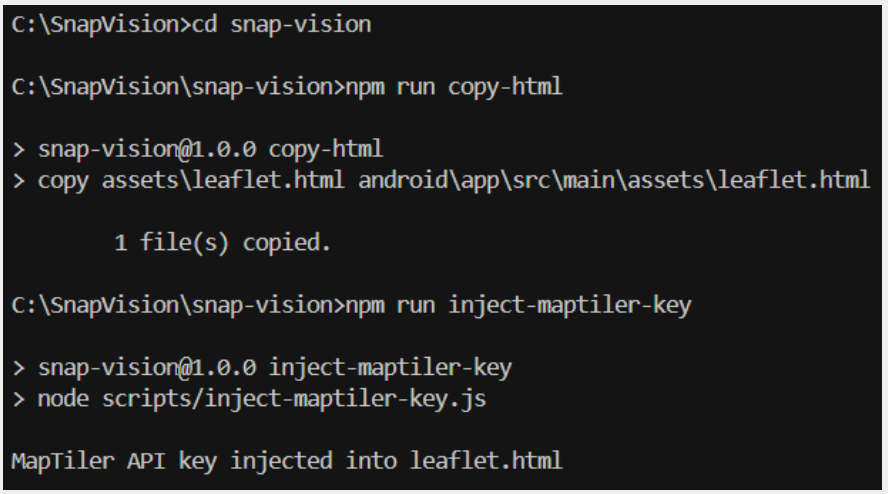
1. Right-click on "This PC" or "My Computer" and select "Properties"
2. Click on "Advanced system settings"
3. Click the "Environment Variables" button Under "System variables" and click "New"
4. Add the following variables:
   Variable name: ANDROID_HOME
   Variable value: Sdk Find the "Path" variable in the list, select it, and click "Edit"
5. Click "New" and add these entries one by one:
   %ANDROID_HOME%\emulator %ANDROID_HOME%\platform-tools
6. Click "OK" on all dialogs to save the changes
7. You'll need to restart any open command prompts or IDEs for these changes to take effect.
8. Enable a virtual device (Android Emulator):
   Open Android Studio Go to More Actions → Virtual Device Manager
   Create a Pixel device (or any other)
   Choose a system image (e.g. API 33) and finish the setup

4. Deployment/Running

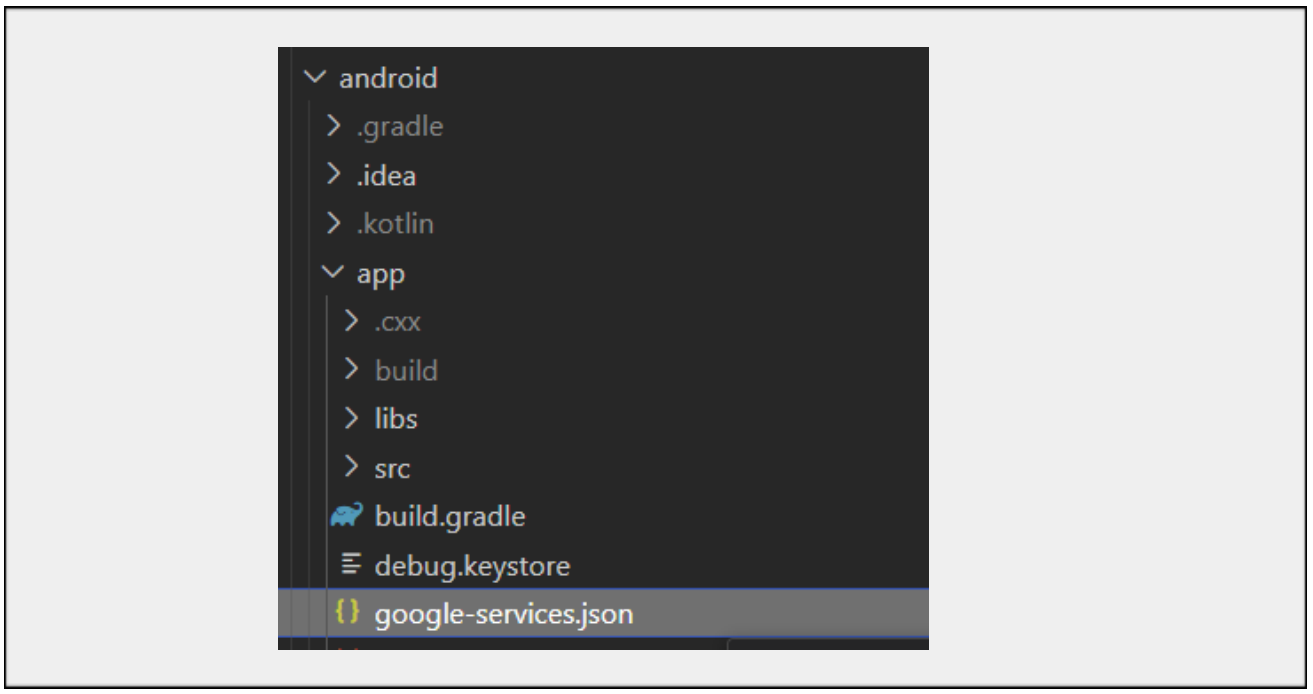
NB: .env files and Firebase setup files can be requested from the development team (bltscapstone@gmail.com)

Step 1: Inject Maptiler API Key
In terminal:

```
cd snap-vision
npm run copy-html
```



Step 2: Add google-services.json File to Project

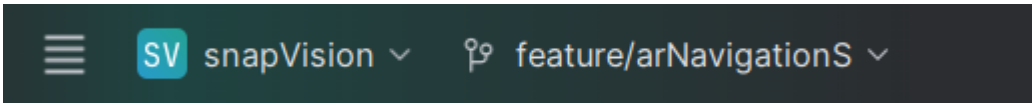


Step 3: Start the Metro Bundler
In terminal:

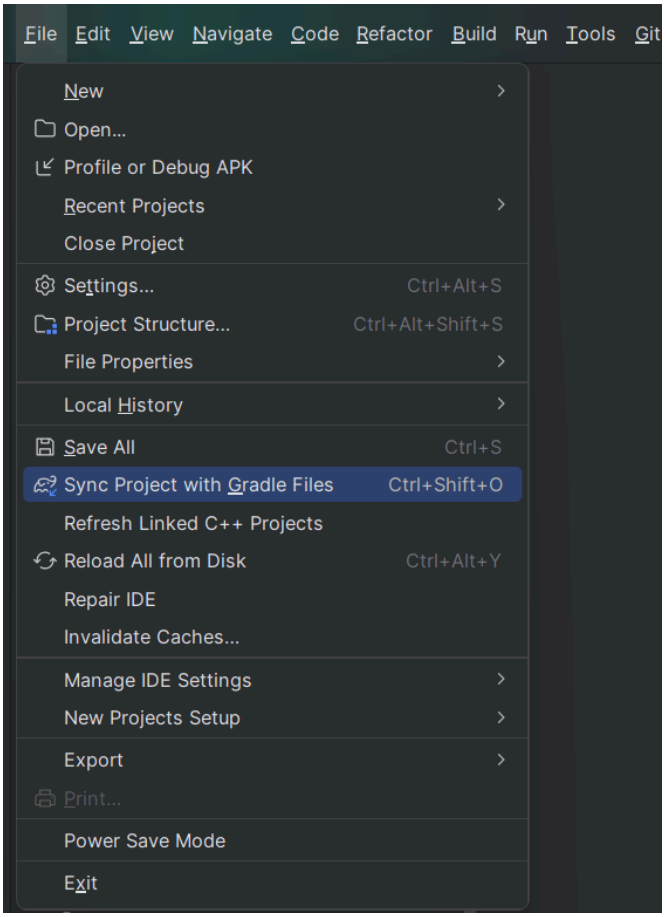
```
npx react-native start
```

Step 4: Run app on Android Studio

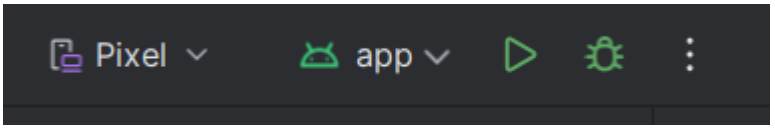
1. Open Android Studio
2. Click the 4 horizontal lines in the menu bar, as seen below:



3. Go to ‘File’ > ‘Sync Project with Gradle Files’ and let it run until it says ‘Build was successful’



4. Ensure an Android emulator is running or connected to a physical device and click the green play button on Android Studio, as seen below:



5. Run the following commands in terminal 1:

```
cd snap-vision
npx react-native start
```

Step 5 : Testing Instructions
Navigate to the project directory in your terminal:

```
cd snap-vision
```

1. To run jest unit and integration tests:

```
npm test
```

2. To run a specific unit test file:

```
npm test __tests__/filename.test.tsx
```

3. To run a specific integration test file:

```
npm test
__tests__/integration/filename.integration.test.tsx
```

Step 6: Uninstallation / Clean Setup Guide
If the build fails or you need to reset your environment, follow these steps:

1. Remove node modules and reinstall dependencies:

```
rm -rf node_modules
rm package-lock.json
npm install
```

2. Clear metro bundler cache:

```
npx react-native start - -reset- cache
```

3. Clear Android build artefacts:
   From the project:

```
cd android
./gradlew clean
```

5. Troubleshooting/ Common errors

1. Emulator Build Errors (Gradle / SDK Issues)

   SDK location not found

   Open Android Studio → SDK Manager → Install latest SDK Platform & Build Tools.
   Run :

   ```
   cd android
   ./gradelw clean
   ```

2. Node/NPM Version Mismatch

   Cannot find module react-native]

   Run:

   ```
   npm npx react-native start --reset-cache
   ```

   If still failing:

   ```
   rm -rf node_modules
   npm install
   ```

3. Firebase configuration errors

   App crashes at launch with missing API key error.
   Fix:

- Ensure google-services.json is placed inside android/app/.
- Check .env file contains correct API keys.

Rebuild the app after adding config:

```
cd android && ./gradlew clean
npx react-native run-android
```

6. User Manual

After installation and deployment, refer to the User Manual Document for detailed instructions on using the Snap Vision system.