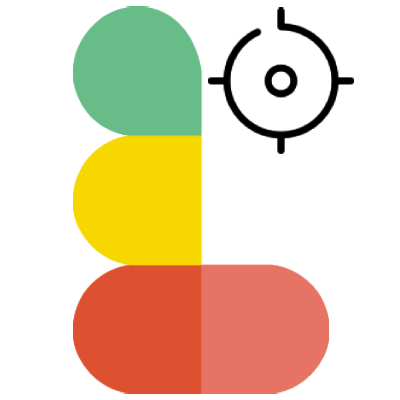
SOFTWARE DOCUMENTATION 



**LQ Net Tracker App**

### PREPARED BY

Piyush Nanwani

Linkquest Technologies Ltd

AUG 05, 2022

CONTENTS

# 1. Project Overview

# 2. User Documentation

# 3. Technical Documentation

# a. Android app

**i. About**

# ii. Getting started

**iii. Deployment**

**iv. Built using**

**b. Backend API**

**i. About**

# ii. Getting started

**iii. Deployment**

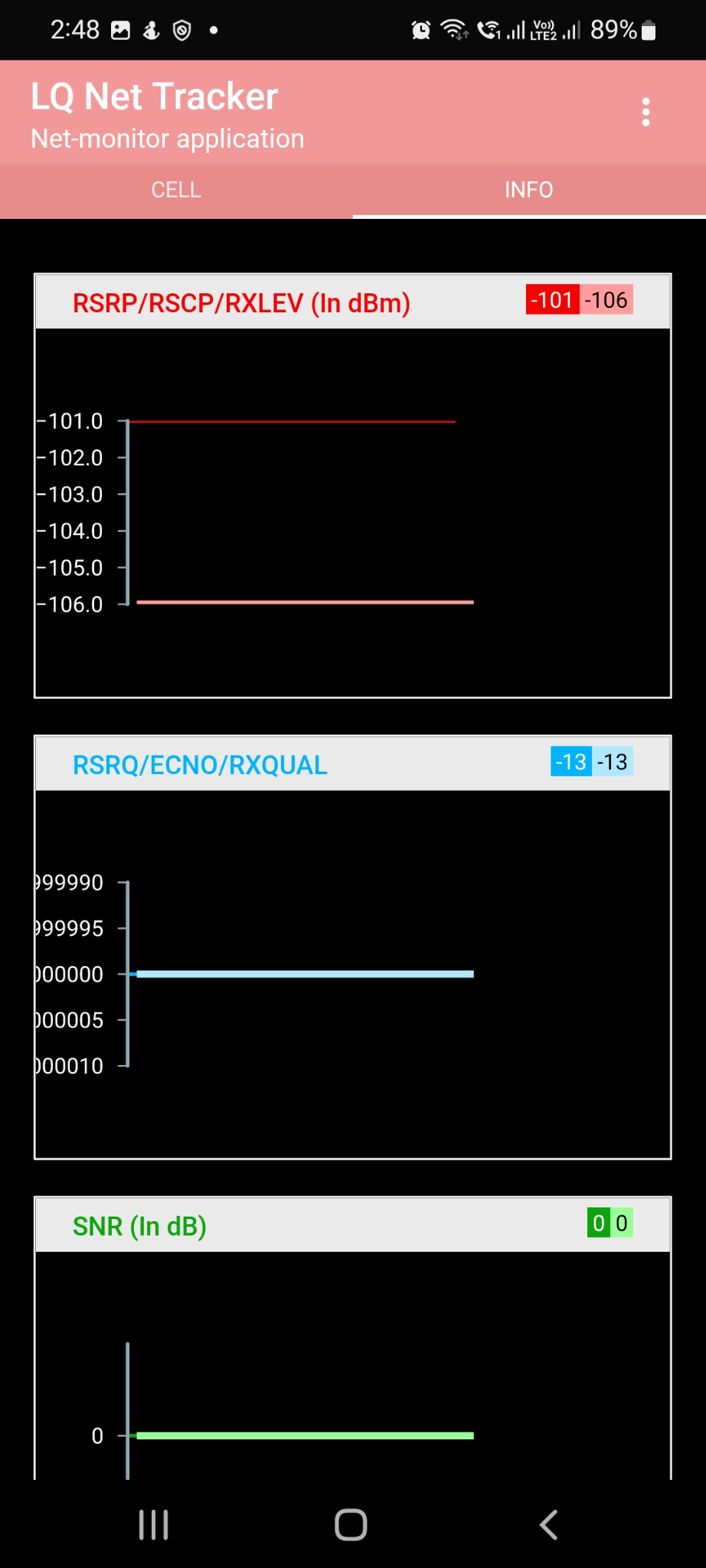
**iv. Built using**

**v. Screenshots**

**c. Database schema**

# 1. Project Overview

**LQ Net Tracker / LQ Pocket** is a net monitor and drive test tool application for 5G NSA/4G/network. It allows monitoring and logging of mobile network serving and neighbour cells information without using specialised equipment.



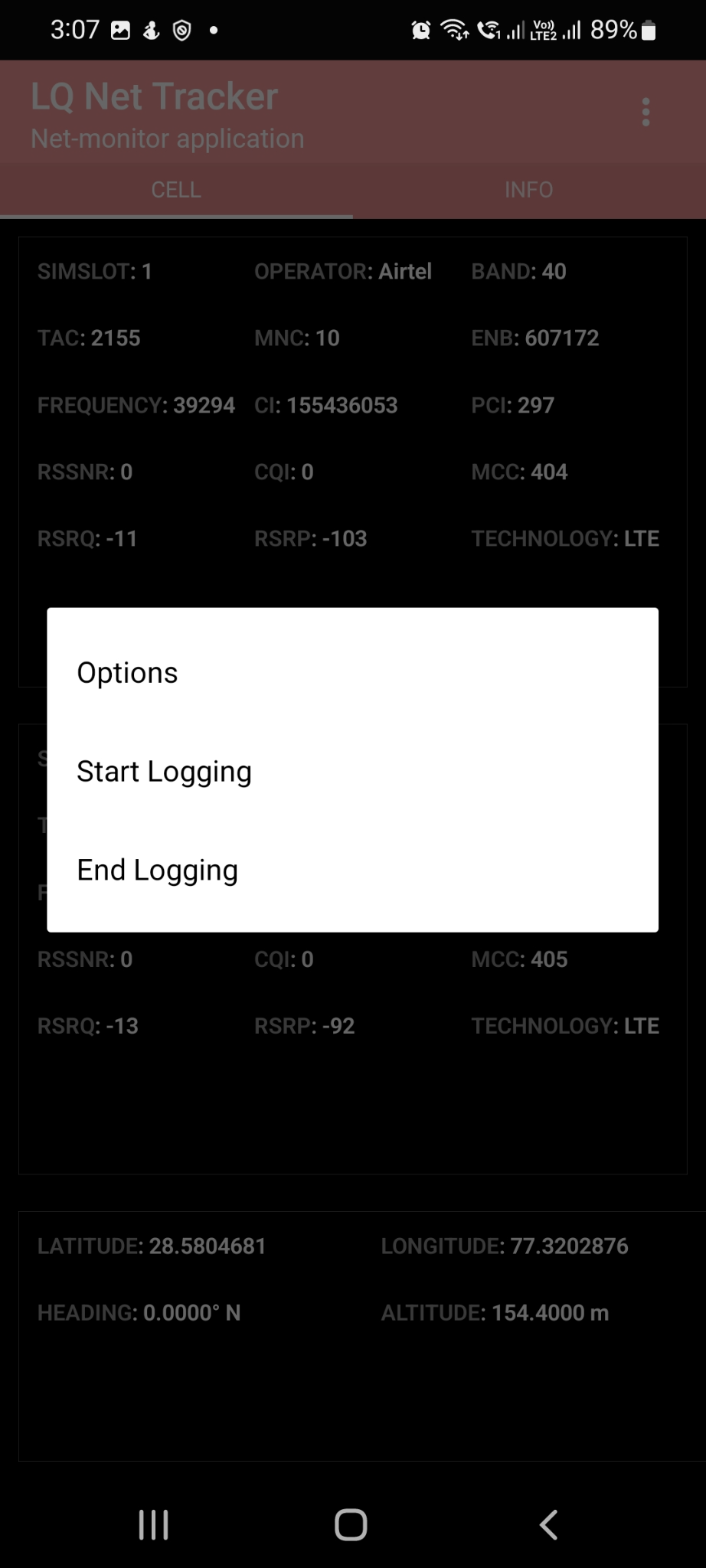
# 2. User Documentation

1. Download the latest APK (0.9.3) by contacting Linkquest Technologies Ltd
2. Install the APK on your Android device
3. Open app to information related to your cellular network ( Currently app supports networks: 5G NSA/LTE.
4. To upload logs to the server, refer to the following steps.. This will send logs to the server after every 5 seconds. ( For reference : See the below image )

**Sending logs to server:**:

1. Click on 3 dots in the top-right corner of the screen.
2. Click on Start Logging option

**Screenshots:**



# 

# 3. Technical Documentation

# Android app

**i. About**

This app shows user cellular data KPIs like MCC, MNC, operator name, device info. Users can also start logging this data to LQ Net Tracker backend APIs by clicking on Start Logging in the app. App will post data every 5 seconds on the API.

# ii. Getting started

These instructions will get you a copy of the project up and running on your local machine for development and testing purposes. See **deployment** for notes on how to deploy the project on a live system.

Prerequisites

First make sure, you have done your react-native setup ( Refer: https://reactnative.dev/docs/environment-setup)

Installing

1. Clone this repository

2. Create a .env file by refering to .env.sample in the root directory

3. `npm i`

4. Development scripts: `npm start`, `npm run android`

5. Running release on android device : `react-native run-android --variant=release`

6. Generating signed apk (We already have a release.keystore in android folder for this)

- First `cd android`

- `./gradlew app:assembleRelease` (To generate APK)

Debugging

- Shake the device and using react-native debugger that opens in the browser window.

- Or you can also use the Logcat in VS Code. ( Extension : Android logcat)

- Or you can use the Logcat in Android Studio ( Select your device and app)

Notes

- Backend of this project is [here](https://github.com/linkquest-hq/net-tracker-backend)

**iii. Built using**

Tech stack used to build this Android App:

* React Native
* JavaScript
* Native modules in Java

**iv. Deployment**

1. Generating signed apk (We already have a release.keystore in android folder for this)

- First `cd android`

- `./gradlew app:assembleRelease` (To generate APK)

2. Semantic versioning using `npm version 1.0.5` (to update npm package) and then `react-native-version`

**Backend API**

**i. About**

This API is part of the backend for LQ Net Tracker App.

* Users can POST /logs /kpi.
* Database : MS SQL Server ( for database connection details refer .env )

# ii. Getting started

These instructions will get you a copy of the project up and running on your local machine for development and testing purposes. See **deployment** for notes on how to deploy the project on a live system.

Prerequisites

First make sure, you have [node-js](https://nodejs.org/en/) and [git-cli](https://git-scm.com/book/en/v2/Getting-Started-The-Command-Line) installed on your system

Installing & Setup

1. Clone this repository with `git clone https://github.com/linkquest-hq/net-tracker-backend.git`

2. To setup MS SQL database refer to `db-setup/README.md`

3. Create a .env file by refering to .env.sample in the root directory

4. `npm i`

5. Command to start server: `npm start`

6. To view api docs now go to `localhost:3000/api-docs`

7. [Installing MS SQL server, Azure Data Studio on Mac OS](https://phoenixnap.com/kb/install-sql-server-macos), [Ref 2](<https://setapp.com/how-to/install-sql-server>)

Setting up MS SQL DB

1. Create a new Database LQ\_Net\_Tracker using Azure Data Studio / MS SQL Server (after connecting - LQ database credentials )
2. Create tables : KPI, Logs\_NumKPI, Logs\_StrKPI (refer queries in sql/)
3. Create stored procedure: SP\_KPI\_Insert, SP\_LogInsertJson, SP\_Logs\_NumKPI\_Insert, SP\_Logs\_StrKPI\_Insert ( refer sql/ folder )

Summary

1. First we created an `app` object from express()

2. Then we simply specified the config ( port, CORS, parsing JSON, routes)

3. Route `/api/logs` is defined by logRouter in `src/router/log.router.js`

4. Now, LogController is defined in `src/controller/log.controller.js`

5. LogController has a `addLog()` function which connects to MS SQL Server DB and executes a stored procedure to INSERT logs into log table .

**iii. Built using**

Tech stack used to build this Android App:

- [Node JS](https://nodejs.org/en/) - JS framework to build network apps

- [Express JS] - Minimalist node js framework

- [Swagger UI] - To generate API docs

- [Nodemon] - Reloads node js app on file updates

**iv. Deployment**

1. Semantic versioning using `npm version 1.0.5` (to update npm package)

2. API has been deployed on Heroku at https://net-tracker-backend.herokuapp.com/

|  |  |
| --- | --- |
|  |  |