

Setup Instruction

Guide for Setting Up a Local Parking Project Development and Testing Environment

Table of Contents

- [Requirements overview](#)
- [Setup database with docker](#)
- [Setup account verification service with Twilio verify](#)
- [Setup cloud storage with Firebase storage](#)
- [Setup Google map services](#)
- [Setup backend](#)
- [Setup web app](#)
- [Setup mobile apps](#)

Requirements overview:

requirements:

- nodejs 20+
- golang 1.22+
- Expo
- Firebase keys
- Twilio Verify keys
- Google cloud map API keys
- Docker (optional)
- Ngrok (optional)

Setup database with docker (skip this if you use your own database)

Install docker: <https://docs.docker.com/get-docker>

Config database by creating .env file in backend project with these variable, for example:

```
DB_USER=postgres
DB_PASS=1
DB_NAME=postgres
DB_PORT=5432
DB_HOST=localhost
```

Go to the backend project root and run:

docker compose up

Setup account verificaiton service with Twilio verify

Register your account and project: <https://www.twilio.com/en-us/user-authentication-identity/verify> Get your account SID and auth token in Twilio console: <https://console.twilio.com> Get your Twilio service ID: <https://console.twilio.com/us1/develop/verify/services> Update backend environment variables (.env file in backend project)

```
TWILIO_ACCOUNT_SID=your_account_sid  
TWILIO_AUTH_TOKEN=your_twilio_auth_token  
VERIFY_SERVICE_SID=your_verify_service_id
```

Setup cloud storage with Firebase storage

Setup your Firebase account and project: <https://firebase.google.com/> Navigate to Firebase console: <https://console.firebase.google.com/> Create a web app: Project overview -> Add app -> Web Enable firebase storage of your project: (Build -> Storage -> Get started) Get project config in Project settings, it will look like this:

```
const firebaseConfig = {  
  apiKey: "your-api-key",  
  authDomain: "your-auth-domain",  
  projectId: "your-project-id",  
  storageBucket: "your-bucket-storage",  
  messagingSenderId: "your-message-sender-id",  
  appId: "your-app-id"  
};
```

Update enviroment variables of mobile app for user (.env file in mobile project for user)

```
FB_API_KEY=your-api-key  
FB_AUTH_DOMAIN=your-auth-domain  
FB_PROJECT_ID=your-project-id  
FB_STORAGE_BUCKET=your-bucket-storage  
FB_MESSAGING_SENDER_ID=your-message-sender-id  
FB_APP_ID=your-app-id
```

Setup Google map services

Register Google cloud account and project <https://developers.google.com/maps/> Google console and enable the following services: <https://console.cloud.google.com/>

- Geo Encoding
- Maps JavaScript API
- Directions API

Get your google map api key: Google Maps Platform -> Credentials -> API key Get map ID: go to map managent page -> Create New Map ID (choose map javascript) <https://console.cloud.google.com/google/maps-apis/studio/maps> Update environment variables (.env files) for web project, mobile projects (both user and employee)

```
# web project
VITE_MAP_API_KEY=your_map_api_key
VITE_MAP_ID=your_map_id
```

```
# mobile projects
MAP_API_KEY=your_map_api_key
```

Setup backend

Install go 1.22: <https://go.dev/dl> Config backend in .env file

```
APP_ENV=dev
PORT=8088
GIN_MODE=debug
ENABLE_DB=true
```

these variables might be configured in previous steps

```
DB_USER=postgres
DB_PASS=1
DB_NAME=postgres
DB_PORT=5432
DB_HOST=localhost
TWILIO_ACCOUNT_SID=your_account_sid
TWILIO_AUTH_TOKEN=your_twilio_auth_token
VERIFY_SERVICE_SID=your_verify_service_id
```

Run project with command:

```
go run main.go
```

Optionally, you can use Ngrok to expose your backend service without deployment <https://ngrok.com>

```
ngrok http http://localhost:8088
```

Run database migration by sending a POST request to the backend server

```
curl -X POST http://your_backend_domain/internal/migrate
```

Setup web app

Install nodejs version 20 <https://nodejs.org/en/download/package-manager>
Config environment variables

```
VITE_API_BASE_URL=your_backend_domain
```

```
# these variables might be configured in previous steps
```

```
VITE_MAP_API_KEY=your_map_api_key
```

```
VITE_MAP_ID=your_map_id
```

Install dependencies and run

```
npm install
```

```
npm run dev
```

Setup mobile apps (both user app and employee app)

Install nodejs version 20 <https://nodejs.org/en/download/package-manager>

- Config environment variables

```
# for employee app
```

```
VITE_API_BASE_URL=your_backend_domain
```

```
# for user app
```

```
VITE_API_BASE_URL=your_backend_domain
```

```
MAP_API_KEY=your_map_api_key
```

```
# this variables might be configured in previous steps
```

```
FB_API_KEY=your-api-key
```

```
FB_AUTH_DOMAIN=your-auth-domain
```

```
FB_PROJECT_ID=your-project-id
```

```
FB_STORAGE_BUCKET=your-bucket-storage
```

```
FB_MESSAGING_SENDER_ID=your-message-sender-id
```

```
FB_APP_ID=your-app-id
```

Install dependencies and run, after this a QR code will appear

```
npm install  
npx expo start
```

On your mobile devices, install Expo go

- Android: <https://play.google.com/store/apps/details?id=host.exp.exponent>
- IOS: <https://apps.apple.com/us/app/expo-go/id982107779>

Open expo go and scan the QR code