EPI-USE LABS

value through innovation

Technical Installation Manual

Contents

1. Introduction

Gym King is a product designed to offer a unique motivational tool for gym users. Gym King takes the concept of gamification and combines it with working out.

The application uses Ionic React to create the Front-end and NodeJS and ExpressJS for the Backend. The application is primarily intended for use on mobile devices and has been created to work on both Android and iOS devices.

The backend hosts a PostgreSQL database that uses the ORM (Object Relational Mapping) design pattern. The API calls that are used by the front-end to communicate with the database are written in JavaScript. The backend also uses Firebase to store images and videos.

The backend is currently hosted on Heroku – a cloud-based platform. However, the backend is generic enough that the backend can be replicated/transferred to another hosting/cloud-platform.

For the AR (Augmented Reality) component, ARCore and ARKit are used respectively to create the models for Android and iOS. The map component uses the open-source technology Pigeon Maps, and the map models are taken from MapTiler.

2. Requirements

This section covers all the minimum software and hardware requirements to be able to operate Gym King. These requirements hold for all possible use cases.

2.1. Hardware

The Minimum Hardware Requirements are:

- 2GB RAM
- 1GB of Permanent Storage

2.2. Operating Systems

Gym King has been designed to work primarily on Android and iOS. The current Operating Systems that the Gym King can work on is:

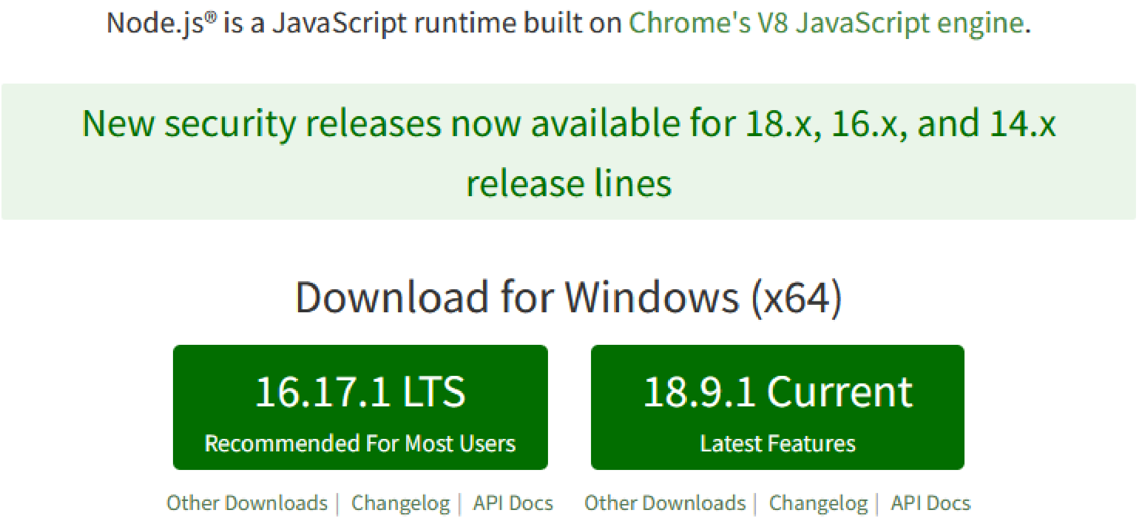
- Android Nougat 7.0 and above.
- iOS version 9.3.5. and above.
- Windows 10 and up
- Linux

2.3. Technology

2.3.1. Node.JS

It is recommended that the latest versions of both NodeJS and NPM are used. Current versions are determined using the commands node -v and npm -v respectively in the terminal.

NPM can be installed using the command npm install in the terminal.



Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

New security releases now available for 18.x, 16.x, and 14.x release lines

Download for Windows (x64)

16.17.1 LTS
Recommended For Most Users
18.9.1 Current
Latest Features
Other Downloads | Changelog | API Docs
Other Downloads | Changelog | API Docs

NodeJS can be installed by downloading the installer from the official NodeJS website https://nodejs.org. You can then run the installer, and it should properly install NodeJS.

2.3.2. Yarn

Yarn is an important tool for running the application on any operating system for development. The current version is 1.22.18, which is recommended, though versions above work as well. To install yarn, simply run yarn install in the terminal, be sure to install yarn in the correct folder repository, particularly in the same folder that contains the clone of GitHub repository.

2.3.3. Ionic React

The system uses Ionic CLI of 6.19.0, and Ionic React 6.1.14. Versions above work as well. To install Ionic React and the CLI, download the installer from the Ionic website. https://ionicframework.com/docs/intro/cli.

2.3.4. Mobile App IDE environment

You'll need to have Android Studio(android) or Xcode(ios) installed on your computer in order to use the source code that gets generated by Ionic react.

- https://developer.android.com/studio
- xcode comes natively installed on MAC devices and you won't be able to build to IOS without it

3. Installation

Once all of the required tools and systems have been set up you can install the application. The application can be cloned from the GitHub repository using the command 'gh repo clone COS301-SE-2022/Gym-King'. You can also download the zip file of the GitHub repository from the GitHub page - https://github.com/COS301-SE-2022/Gym-King. Please ensure that the yarn tool and the application are located in the same folder, else yarn will not properly work.

4. Development Usage

Development Usage is for programmers and developers who which to expand or alter the application.

4.1. Programming Languages

4.1.1. Front-end

- Typescript
- CSS
- Python
- Javascript

4.1.2. Backend

- NodeJS
- ExpressJS
- JavaScript

4.2. Technology Used

4.2.1. Front-end

- Ionic React
- Capacitor

4.2.2. Backend

- Cloud – Platform: Heroku (https://dashboard.heroku.com/)
- Database: PostgreSQL
- Image Storage: Firebase

4.2.3. AR Components

- Android: ARCore (https://developers.google.com/ar)
- iOS: ARKit (https://developer.apple.com/augmented-reality/)

4.2.4. Map Component

- Functionality: Pigeon Maps (https://pigeon-maps.js.org/)
- Map: MapTiler (https://www.maptiler.com/)

5. Running the System

Running on a browser

- run development version “yarn dev:api && yarn start:dev”
  1. start the api in develop mode (launches the backend locally which connects the development DB on the cloud)
  2. start the front end in develop mode(connects to a local host API)
  3. optional - you can also run ”yarn dev:api” to start the api in production mode(launches the backend locally which connects the production DB on the cloud)
- run production version “yarn start:prod”
  1. start the front end in production mode(connects to the cloud production API)

Build and Run the app on mobile

- Building from React to Mobile IDE source code:
  - *“yarn nx build:ios front-end”* - build for IOS
  - *“yarn nx build:android front-end”* - build for Android
  - These build commands will build the source code and launch their respective mobile IDEs
- Building the.apk/.ipa files:
  - hit the play button in the ide while your phone is plugged in
  - builds always run in production mode!

The application will either run on localhost in port 3000 or it will open Android Studio, or equivalent IDE.

6. Deployment Model

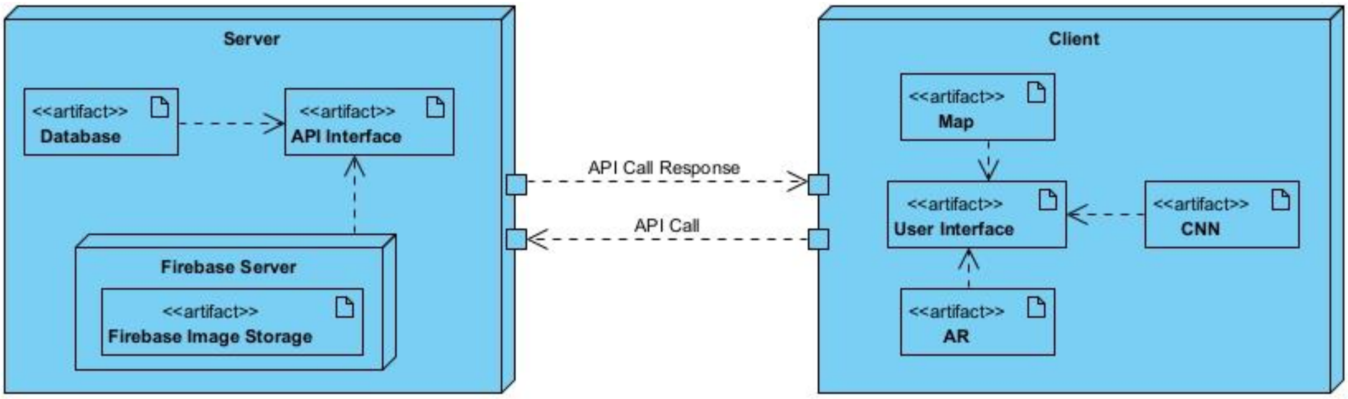


Figure 1 – Client Server Deployment model