AUTOMATED BIKE SHARING SYSTEM FOR UNIVERSITIES

E/14/154 – SHYAMAN MADHAWA

E/14/141 – PUWASURU IHALAGEDARA

E/14/194 - SANDALI LOKUGE

TABLE OF CONTENTS

SECTION 01: USER MANUAL

1. INTRODUCTION
2. SAFETY INSTRUCTIONS
3. GETTING STARTED
4. TROUBLESHOOTING

SECTION 02: TECHNICAL NOTE

1. INTRODUCTION
2. TECHNOLOGY SPECIFICATION
3. METHODOLOGY
4. TESTING

USER MANUAL

TABLE OF CONTENTS

1. INTRODUCTION----------5

1.1 ABOUT PRODUCT----------5

1.2 FEATURES ----------5

2. CAUTIONS & WARNINGS ----------6

3. GETTING STARTED ----------7

3.1 SETUP CONSIDERATIONS ----------7

3.2 USER ACCESS CONSIDERATIONS---------- 8

3.3 ACCESSING THE SYSTEM ----------9

3.4 SYSTEM ORGANIZATION & NAVIGATION ----------9

4. TROUBLESHOOTING---------- 17

INTRODUCTION 1

1.1 ABOUT PRODUCT

This system provides and automated bike sharing facility for university students and staff to evaluate the mobility patterns of academic campuses and to assess the energy consumption and pollutant emissions produced by the universities. This system has a simple process and anyone in the universities can use this just only following few steps. The whole product includes a smart lock for bicycle dock stations, administrative application and a mobile application.

We encourage users to read this user manual before going to launch a bicycle sharing system using this product.

Note :

- The term 'user' is used throughout this document to refer to a person who has administrative privileges on launching this system and accessing the administrative app
- The term 'rider' is used throughout this document to refer to a person who wish to get the service that is provided by this bicycle sharing system.
- 'Dock station' refers to a place where a set of bicycles are locked enabling the bicycle sharing service.

1.2 FEATURES

- Smart lock to hold the bicycle in the dock stations
- Can unlock the bicycle using the QR code reader provides in the mobile app
- Can search the availability of the bikes in the sub stations using the mobile app
- Administrative app provides facilities to register riders, bicycles and dock stations
- Can monitor the bicycle usage with the admin app

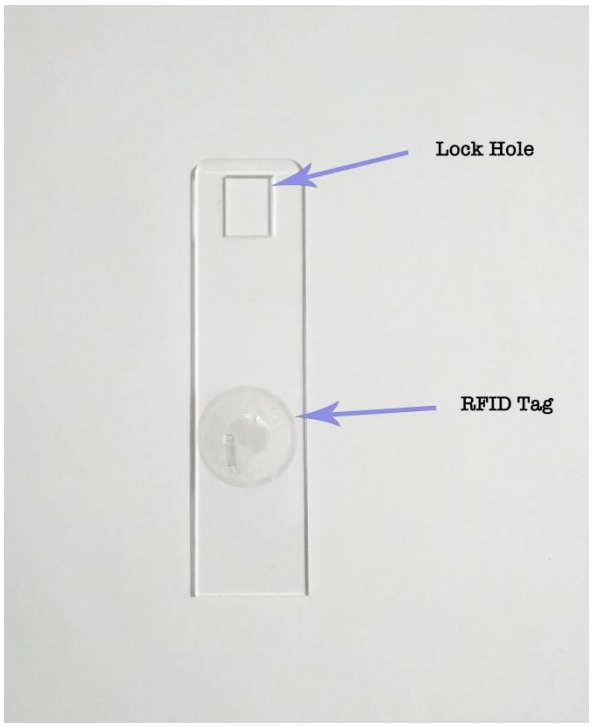
SAFETY INSTRUCTIONS

2

Make sure to read the following instructions carefully and thoroughly as it contains important safety instructions to prevent damages to the product.

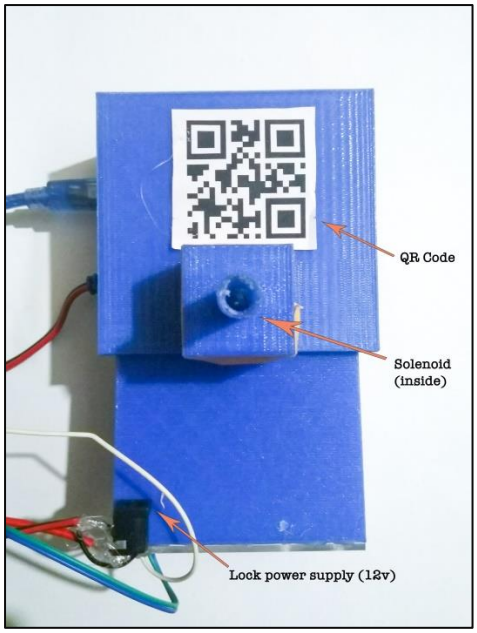
- Do not dissemble the lock and its components
- Make sure to provide 12V and 2A power to the lock and 12V and 1A power to master Arduino.
- Use this product only for its intended purpose as described in this user manual
- Do not use damaged power cords, plug or lose power outlet.
- Do not remove or damage to the RFID tag which is connected to the component that should be fixed to the bicycle.
- The product and the mobile app are continuously being updated. Their functions and language can slightly differ from this manual. Use this document as a reference only.

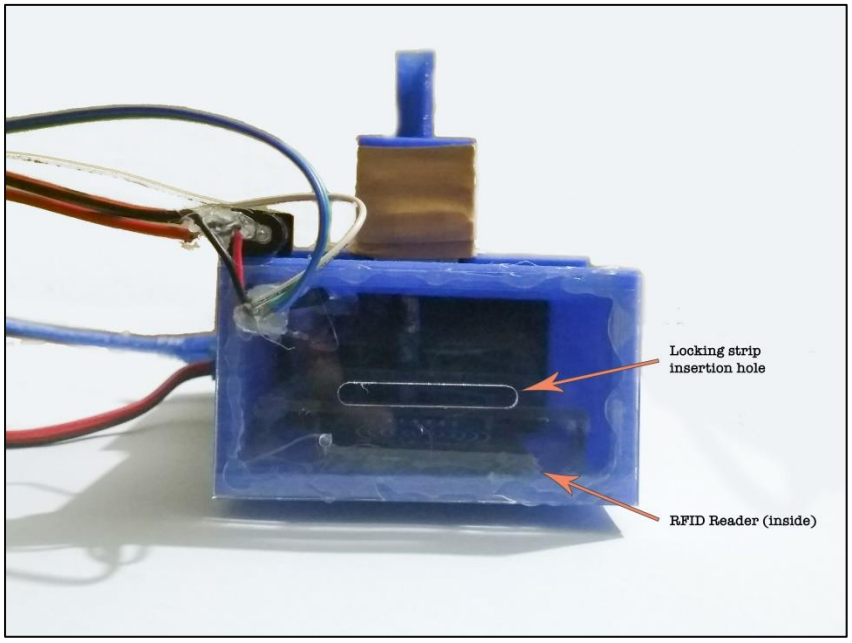
3.1 HARDWARE SETUP CONSIDERATION



The above component should be fixed to the bicycle.

Users have nothing to do with the lock as the lock is already assembled.





3.2 USER ACCESS CONSIDERATIONS

For this system, there are two types of users.

1. Administrators
2. Riders

The admin app is provided for the use of administrators for registering users, bicycles and dock stations. Similarly, the administrators can monitor the overall bicycle usage and usage of each rider.

Riders have access to unlock the bike using the QR code provided in the mobile application. At the very first, riders have to be registered by the admin and then he/she can login to the mobile app using his/her registration number as the username and the NIC number as the password. The riders have privileges to reset the password because when he/she is registered by the admin, the default password is his/her NIC number. The mobile app provides facility to check the about the dock stations; number of free bike locks and the number of available bicycles.

3.3 ACCESSING THE SYSTEM

- Admin username password can be changed by setting ADMIN_USERNAME, ADMIN_PASS environmental variables of the server machine.

- Default admin username and password are as follows

  Username : admin
  Password: perarideadmin

- To access the mobile app, the rider should be registered by the admin

3.4 SYSTEM ORGANIZATION AND NAVIGATION

3.4.1 ADMINISTRATIVE APP

This web application is for the administrative purposes and it can be only accessed by the previously registered admins.

- Login interface
  As mentioned above, at the very first you(admin) have to use above password and username for the login. After login using that username and password, you can create new admins

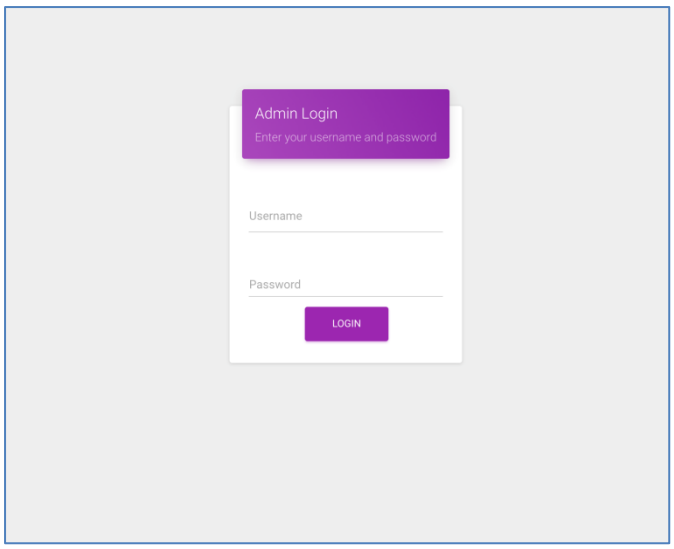


Figure 3.4.1 – Login interface

- Register new riders interface

Following interface provides the facility to register new riders and also the admin can view the list of the registered users. All the fields are required to be filled in order to register a rider.

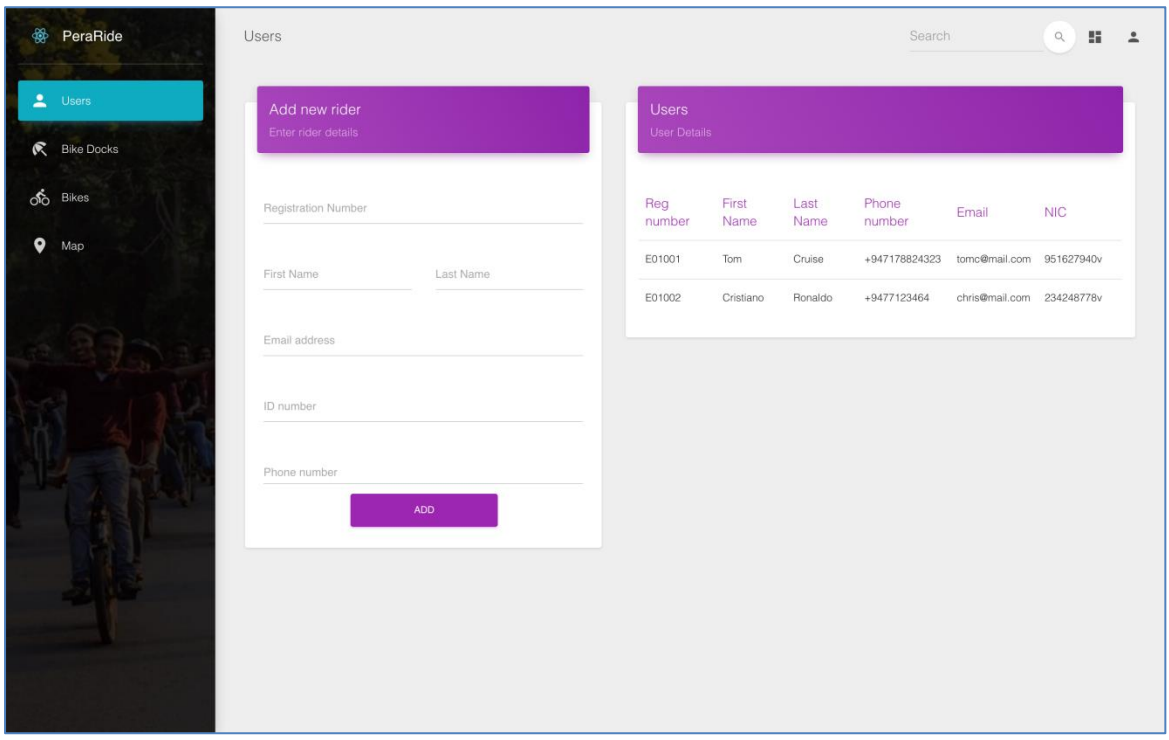


Figure 3.4.2 – Rider Register interface

- Dock station and bicycle register interface

When a new dock station is going to be located within the university, admin have to register that dock station details. When a new bicycle is arrived, admin doesn't have to register that bicycle manually, but it will be automatically registered when it put into the lock.

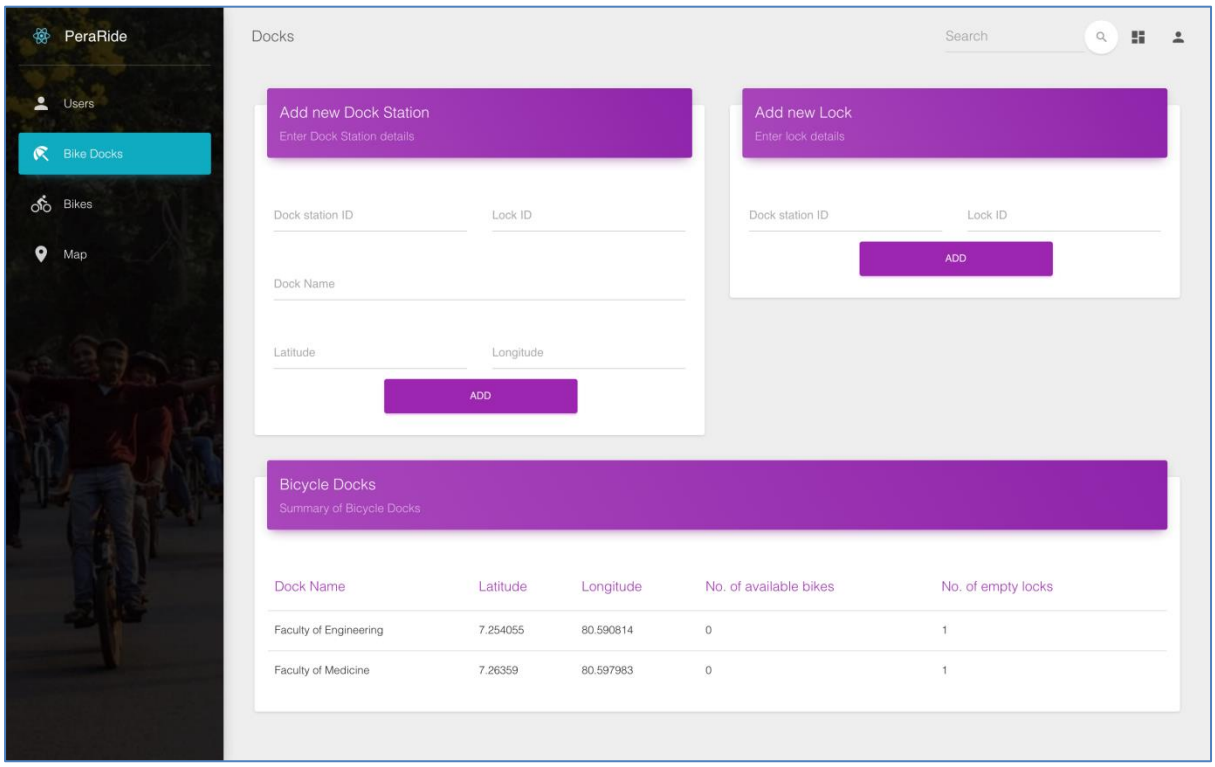


Figure 3.4.3 – Dock station and bicycle register interface

- Monitor the active riders
  The admin have privileges to monitor the active riders and the bicycle details using the following interface

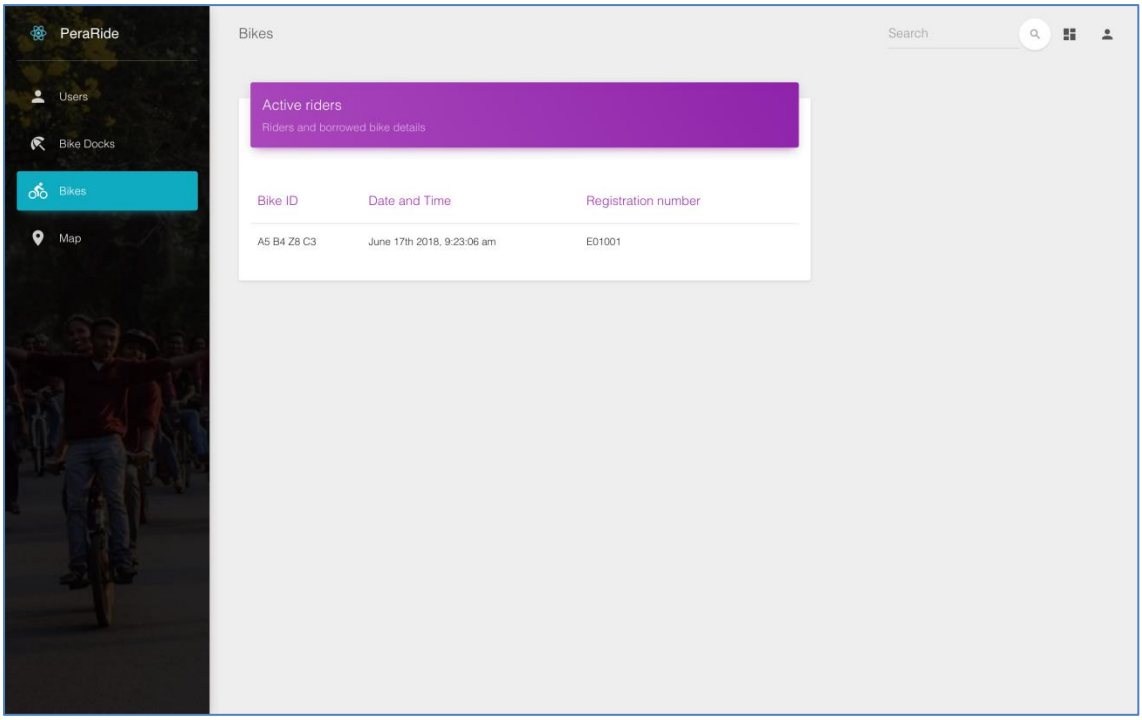


Figure 3.4.4 – Active riders monitoring interface

- Map of dock stations

  The admin can view the each dock station details using this interface.

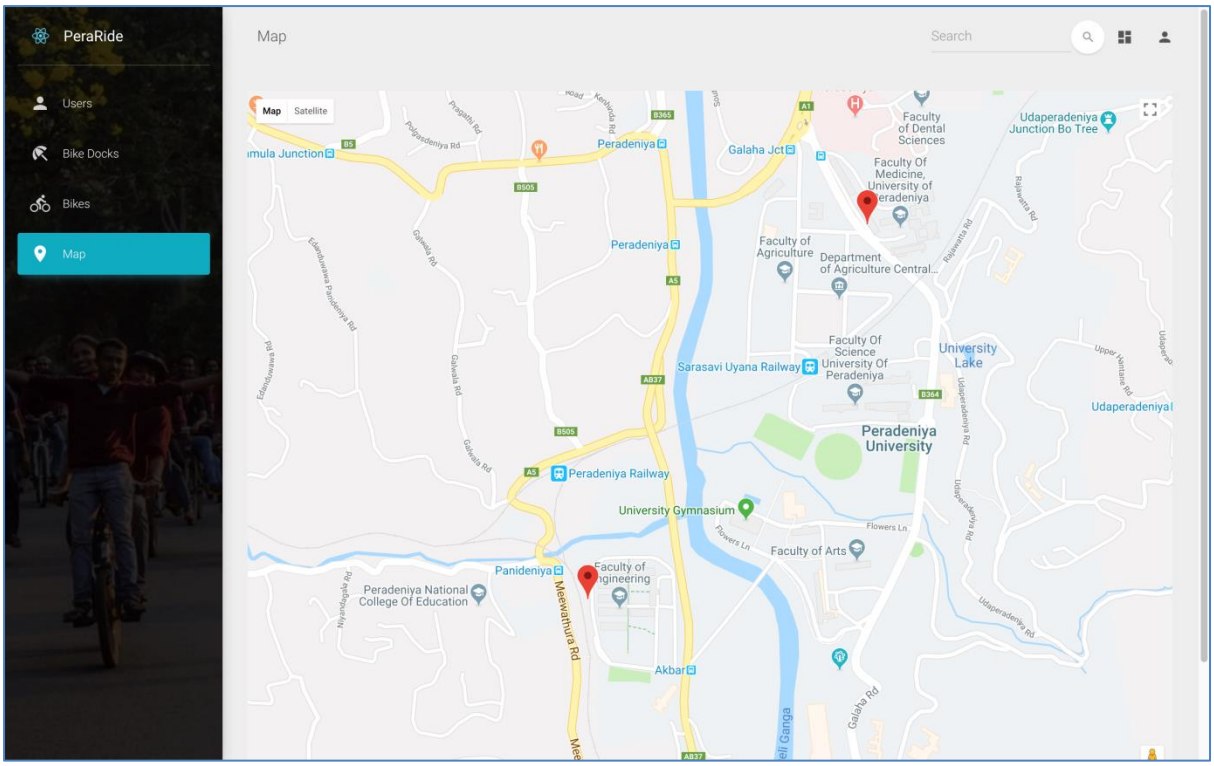
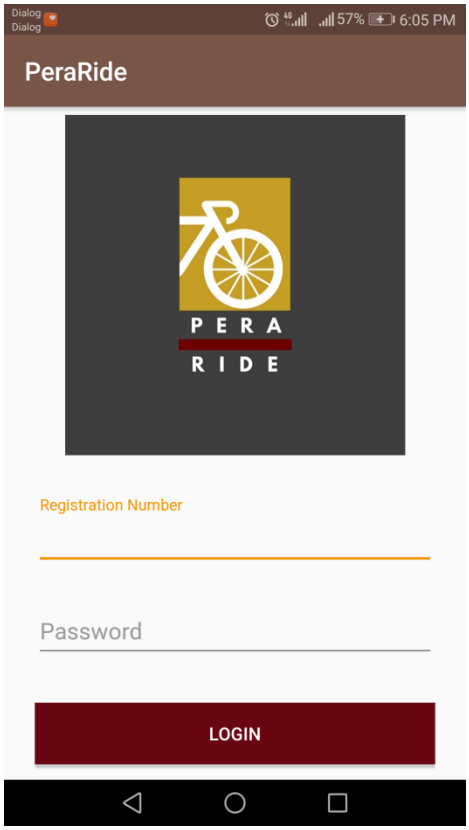


Figure 3.4.5 – Map view of dock stations

3.4.2 MOBILE APPLICATION

This mobile app is for the unlocking purposes and anyone who is already registered can use this mobile app for riding activity.

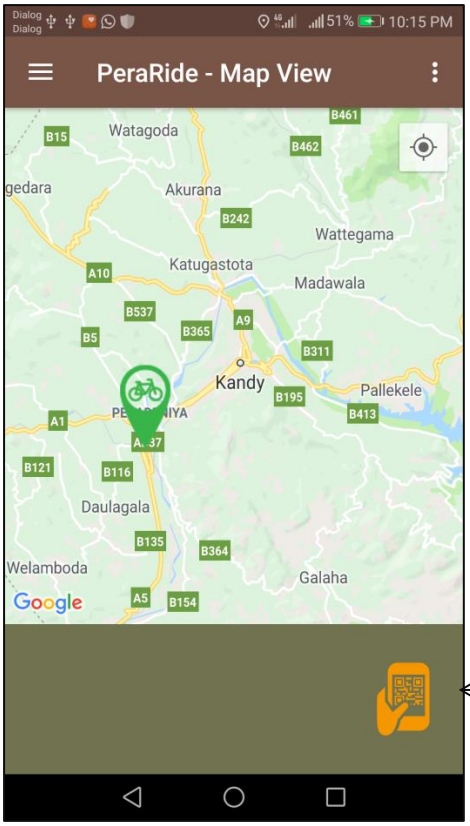
- Login interface
  If you are going to login at the very first time you have to use your university register number as the user name and NIC as the password. If you have already changed it, you can use that password.



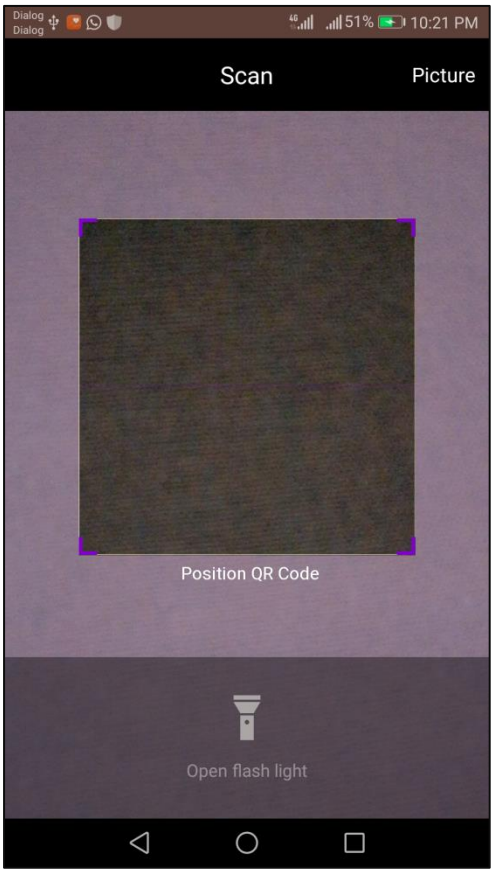
- Map Interface

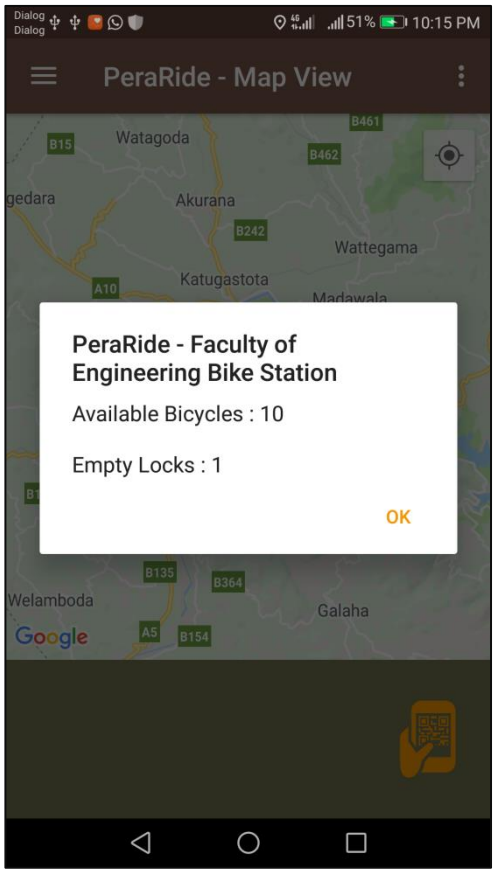
  Riders can view the map of the dock stations and can see where the dock stations are located. Not only that you can check the number of available bicycles and number of free locks in a particular dock station.

  In the same interface there is a QR reader. It can be used to unlock a bicycle.



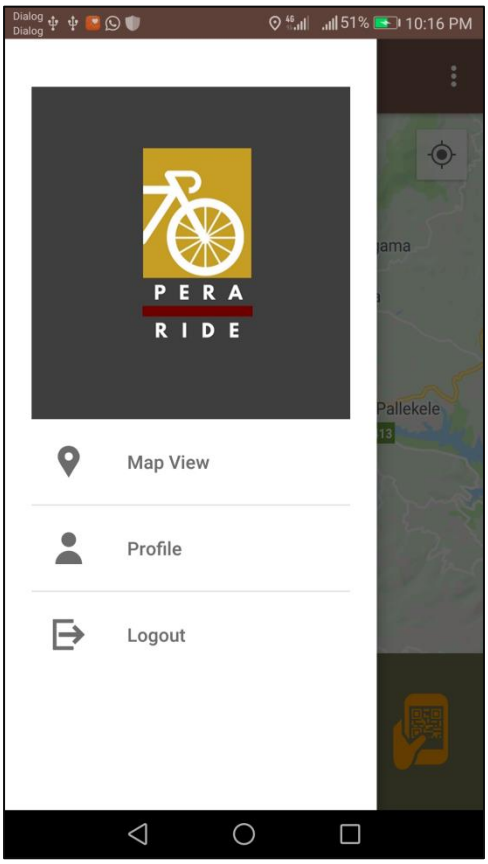
QR code scanner



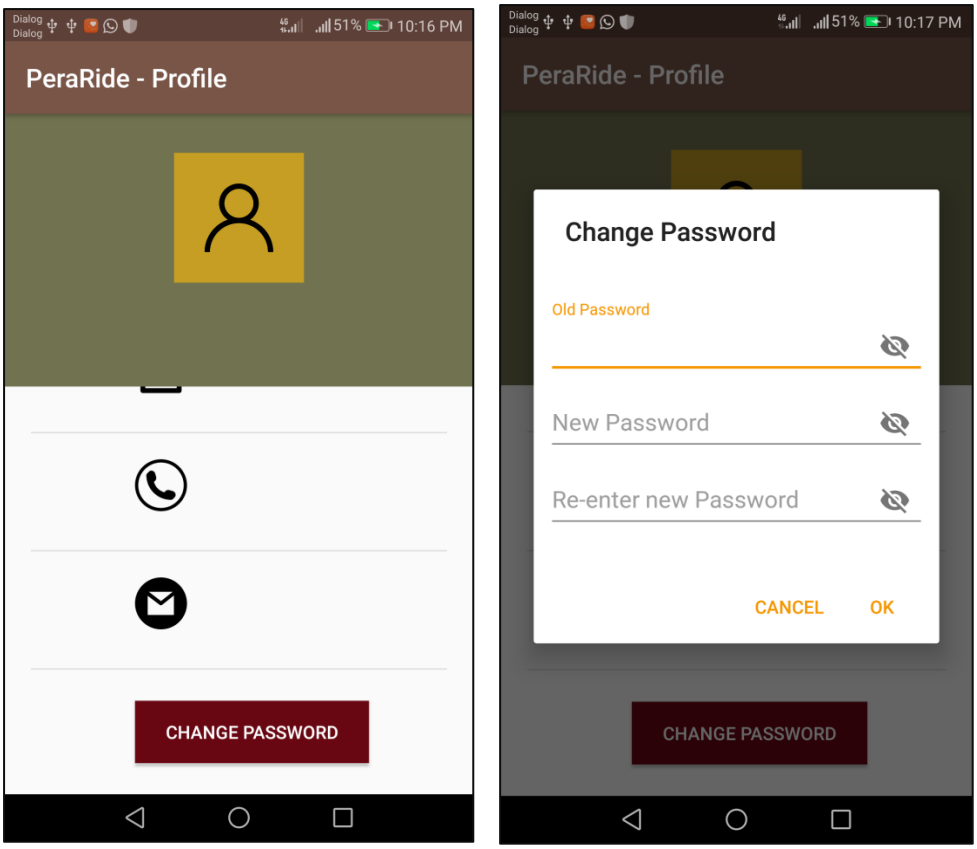


Selected dock station details view

- Navigation Window
  You can navigate into Map view, Profile and Logout using this window



- Profile Interface
  This shows your registered details. But riders don't have privilege to change the profile details but he/she can change the password.



Password changing interface

- If there is an error message of Connection Error when you are using the mobile app, please check your network connection.
- If you forget your username or password, please go to the admin and make sure to get new password.
- If QR Code can be seen in PeraRide mobile app, but no values are picked up then,

  a. Try restarting the app. If this does not work, restart the mobile device

  b. Ensure the full QR Code is contained within the “focus square” of the reader

  c. Ensure front camera does not have obstructed view (e.g., dirty lens)

  d. Ensure reader is able to zoom-in properly on QR Code

  e. Ensure the device is being held steadily during reading

- Internet is available on device, but PeraRide Mobile app cannot reach server (e.g., time-out)
  Resolve various possible issues leading to the inaccessibility
  i. Check if the mobile device has a connection filter / restriction enforced

  ii. Check if the mobile device is connecting to a network firewall / filter

  iii. Check if you are having a particularly slow connection to the server

  iv. Check if your ISP is experiencing problems

TECHNICAL NOTE

TABLE OF CONTENTS

CHAPTER 1 - INTRODUCTION ---------- 20

1.0 PROJECT BACKGROUND ---------- 20

1.1 OBJECTIVES ---------- 31

CHAPTER 2 - TECHNOLOGY SPECIFICATION ---------- 22

2.0 EMBEDDED SYSTEM DESIGNING ---------- 22

2.1 WEB AND NETWORK APPLICATION DESIGNING ----------23

2.2 NETWORK SECURITY ---------- 24

CHAPTER 3 - METHODOLOGY ----------25

3.0 INTRODUCTION ---------- 25

3.1 ANDROID APPLICATION AND WEB APPLICATION DESIGN- 25

3.2 BACK-END DESIGN ----------30

3.3 HARDWARE DESIGN ---------- 34

3.4 OVERALL SYSTEM DESIGN ---------- 41

CHAPTER 4 - TESTING ----------42

1.1 PROJECT BACKGROUND

Bike-share has taken many forms over the course of its development, from free bikes left for a community to use at will to more technologically advanced and secure systems. In every iteration, the essence of bike-share remains simple: anyone can pick up a bike in one place and return it to another, making point-to-point, human powered transportation feasible. Today, more than 600 cities and more than 200 universities around the globe have their own bike-share systems, and more programs are starting every year. The largest systems are in China, in cities such as Hangzhou and Shanghai. In Paris, London, and Washington, D.C. highly successful systems have helped to promote cycling as a viable and valued transport option.

As the most of the universities have wide area of land, transportation within the university causes time waste, accidents, congestion because of using the private vehicles, parking problems and the energy consumption related to the mobility of workers and students of the universities. The bicycle sharing programs have received increasing attention in recent years with initiatives to increase bike usage, better meet the demand of a more mobile public and lessen the environmental impacts of our transportation activities. So the project aims to introduce automated bike sharing system to minimize above impacts while evaluating the mobility patterns of academic campuses and assessing the energy consumption and pollutant emissions produced by the universities. This system provides the users to unlock the chosen bicycle in the substations via a mobile app and start riding, check the availability of bicycles and authorized people to track the path of rides of all users.

1.1 OBJECTIVES

As the most of the universities have wide area of land, transportation within the university causes time waste, accidents, congestion because of using the private vehicles, parking problems and the energy consumption related to the mobility of workers and students of the universities.

- **Mobility for employees** – providing transportation options that assist University employees to conduct their duties and responsibilities in an efficient, environmentally-friendly manner
- **Safety through reduced motor vehicle traffic** – reducing the amount of motor vehicle traffic in areas with complicated infrastructure or high pedestrian volumes
- **Health through increased physical activity** – providing methods for employees and students to add more movement into their daily routine, thus impacting alertness, health, longevity, and more
- **Environmental benefits of reduced motor vehicle traffic** – reducing greenhouse-gas emissions, decreasing impervious surfaces for parking lots, decreasing need for road maintenance
- **Social benefits through enhancement of bicycling culture** – supporting a culture that sees cycling as a preferred transportation mode choice, and a community that respects and works with the different transportation options
- **Convenience factor** – providing efficient transportation choices to aid users in arriving at destinations quickly and safely by reducing the need to always be in search of a bicycle or car parking area as well as the need to have to do own maintenance on bicycles
- **Economically self-sustaining** – implement a system that will pay for itself and reduce transportation costs for the campus community
- **Research opportunities** – provide the potential for research in urban planning, kinesiology and community health, marketing, environment, engineering, and more, with opportunities to publicize findings internationally
- **Education campaign** – participate with the campus-wide bicycle education campaign, by creating a platform for information sharing
- **Enhanced Image of the Campus** – Implementing a bicycle sharing program would improve the campus' standing as a preferred employer and be an attractive feature for prospective students. This could positively impact recruitment and retention.

2.0 EMBEDDED SYSTEM DESIGNING

1. Measuring and controlling

- **RFID reader and tags/stickers** - To identify each bicycle is in the exact position and to identify the bicycle when returning to the dock station.
- **Electric lock** - To lock the bicycle

2. Embedded Platform

- **Arduino**

  Arduino is an open source computer hardware and software platform which is very easy to use. There are enough libraries and compatible modules which can connect to the Arduino board. For serial communication we can have hardware serial ports or software serial ports. To control the locking mechanism there are digital I/O pins and ICSP pins.

3. Connecting the system to the network

- Whole locking system will connect to internet using a GSM module
- Users will connect to the system using a Mobile App

4. Peripheral devices

- **RFID reader**

RFID reader use to read the RFID stickers in the bicycle. These stickers have a unique id which we use as the identification of bicycle. It is a 5V device, so you don't need a external power source. ICSP pins are going to use for the communication between the reader and the Arduino board.

- **GSM module**

A GSM module will use to connect with central server. TTL pins in GSM module will use to connect the module to Arduino board.

- **A linear actuator**

A linear actuator is use in locking mechanism of the bikes. Digital I/O pins will be used to send control signals to the actuator.

5. **Limitations of peripheral devices**

- There are various security problems with locking mechanism. Additional sensors have to use in order to make more secure.

2.1 WEB AND NETWORK APPLICATION DESIGNING

1. **Protocols and middle-wares**

- HTTP

HTTP protocol was used to maintain the communication between dock stations and central server.

- I2C protocol

This protocol was used to communicate between locks and the relay node.

- A central server

Central server was used to control the locks. A user scan the QR code in the lock and send the information with his login details to the server. Then the server will unlock the relevant lock and start to track the bicycle using the GPS system of the mobile using the given mobile app. It maintains a database of users and bicycles.

- MQTT Protocol

MQTT protocol is a Machine to Machine (M2M) protocol widely used in Internet of things. The MQTT protocol is a message based protocol, extremely light-weight and for this reason, it is adopted in IoT ecosystem. Almost all IoT platforms support MQTT protocol to send and receive data from smart objects. There are several implementations for different IoT
boards like Arduino, Raspberry and so on.

2. **Back end and Front end**

- **Back end**
  - Node.js was used as the server side language

- Mongodb as database management system
- Server, MQTT broker and database are hosted in the Docker Cloud
- SMS Gateway- Twilio is used as the SMS API

- **Front end**
  - Web interface for administrational usage
  - HTML, CSS, Javascript(React), Redux
  - Mobile Application
  - Android studio

3. **Connecting components through APIs**

- **REST API**
  Was used to exchange information among components (lock and mobile app)

- **Google Maps API**
  Was used to show to location of the bicycle

- **Bar-code API**
  Was used to parse the QR code with different format

2.2 NETWORK SECURITY

- **Sensitive data**

Details of users are stored in central server. Mobile app is used to login to the system. These login requests need to be secure. Passwords of users need to be stored in hash representation.
Controlling responses from server should be secured.

- **Security features**

Encryption of the requests and responses.

- **Making MQTT secure with SSL**

MQTT brokers may require username and password authentication from clients to connect. To ensure privacy, the TCP connection may be encrypted with SSL. SSL (Secure Socket Layer) is the accepted standard for encrypted communication between the server and a client ensuring that all data passed between the server and client remain private and integral.

3.0 INTRODUCTION

This chapter describes the method which will be used to implement this project.

3.1 ANDROID APPLICATION AND WEB APPLICATION DESIGN

For the riders' side, a mobile application is going to be implemented using Android studio. The client can check the map of the dock stations around the University and he/she can unlock by reading the QR code via this mobile application at the desired dock station. Here android volley was used to make HTTP requests. Google map API is used to get the locations of the dock stations and Google Barcode API is used to scan the QR code. A web application will be developed for the administrational purposes such as to register the riders and monitor the bike usage. HTML, CSS, Javascript, React, Redux were used as the front end technologies.

3.1.1 WEB APPLICATION UI DESIGN

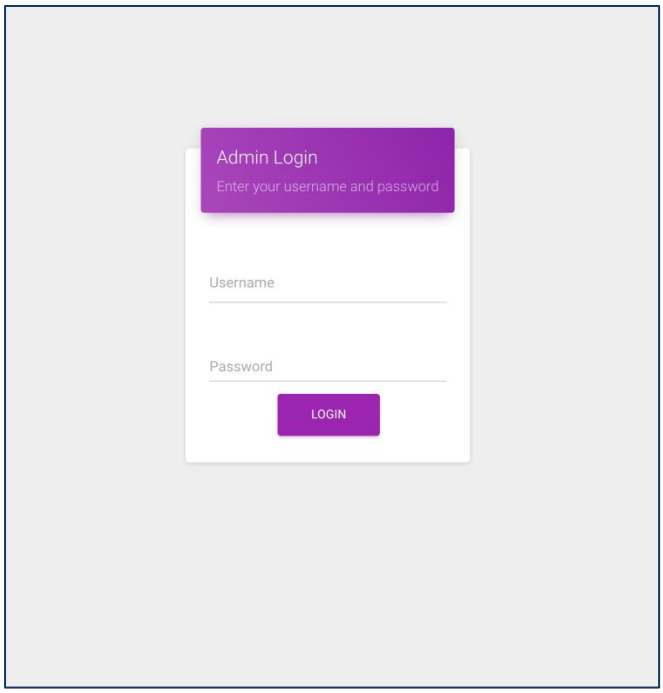


Figure 3.1.1 - Login Interface

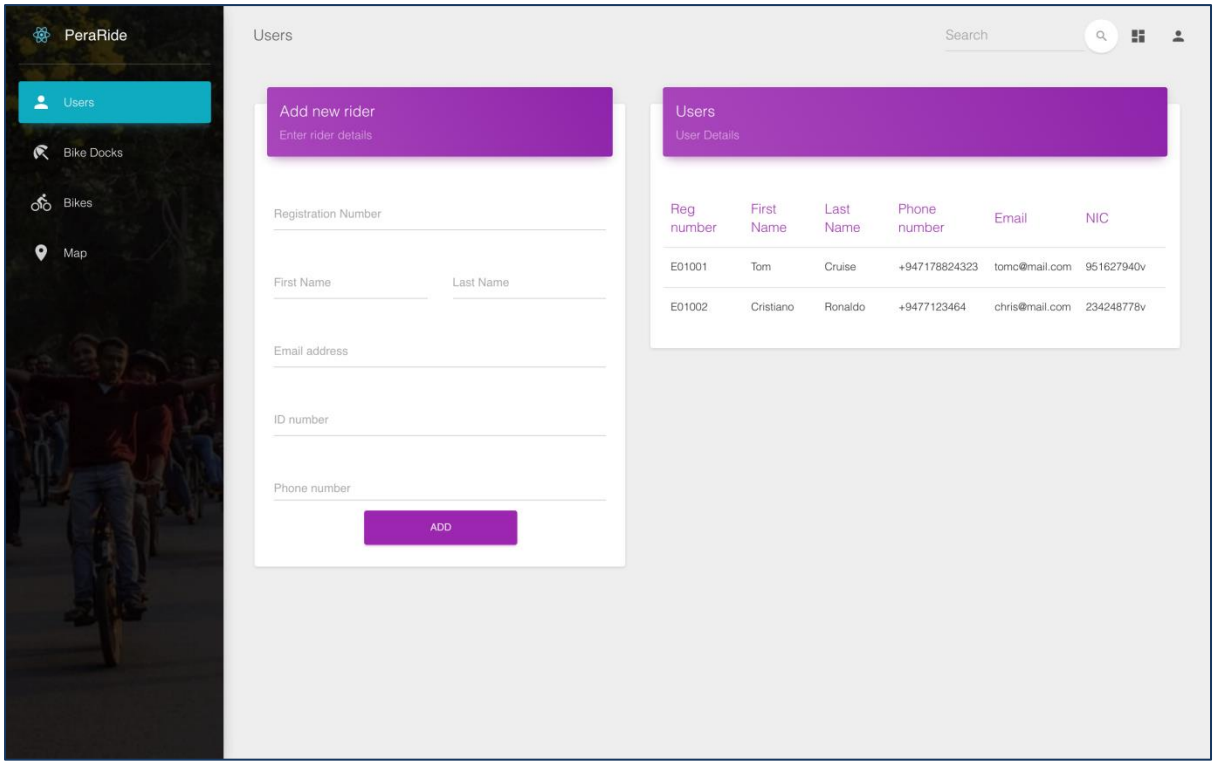


Figure 3.1.2 – Rider Register Interface

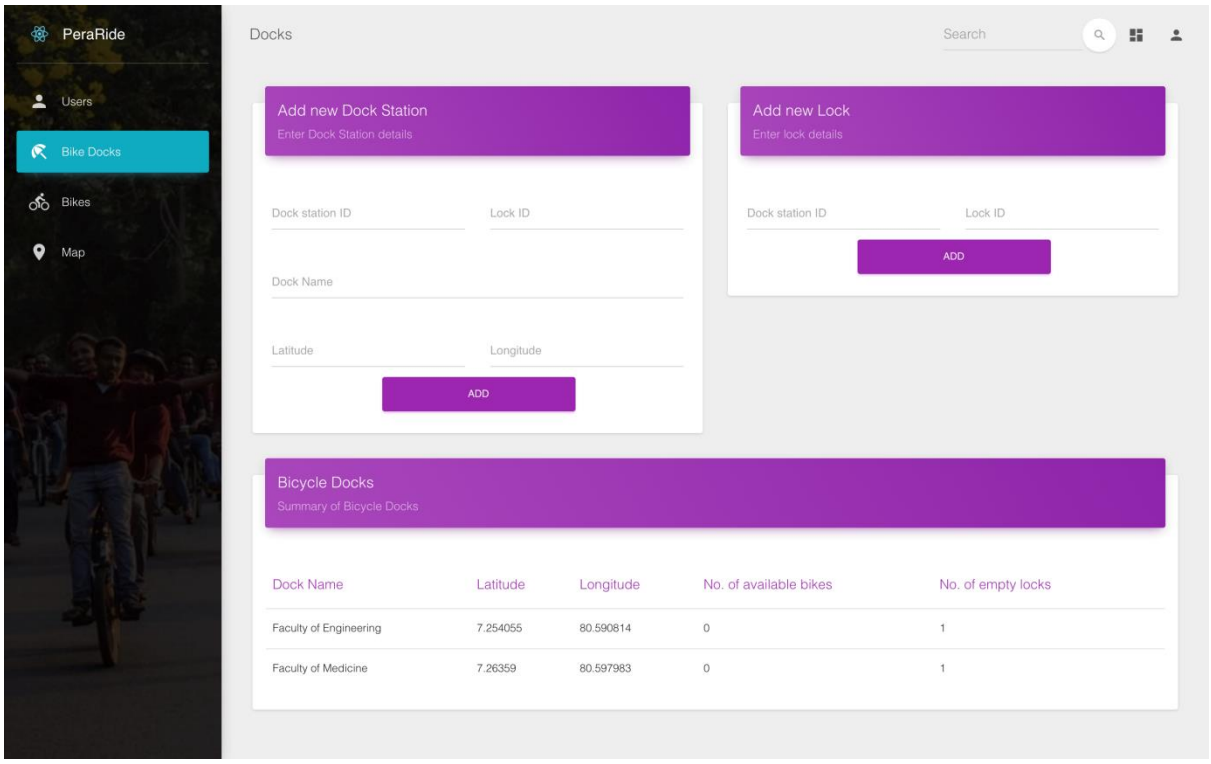


Figure 3.1.3 – Dock Station Register and monitor Interface

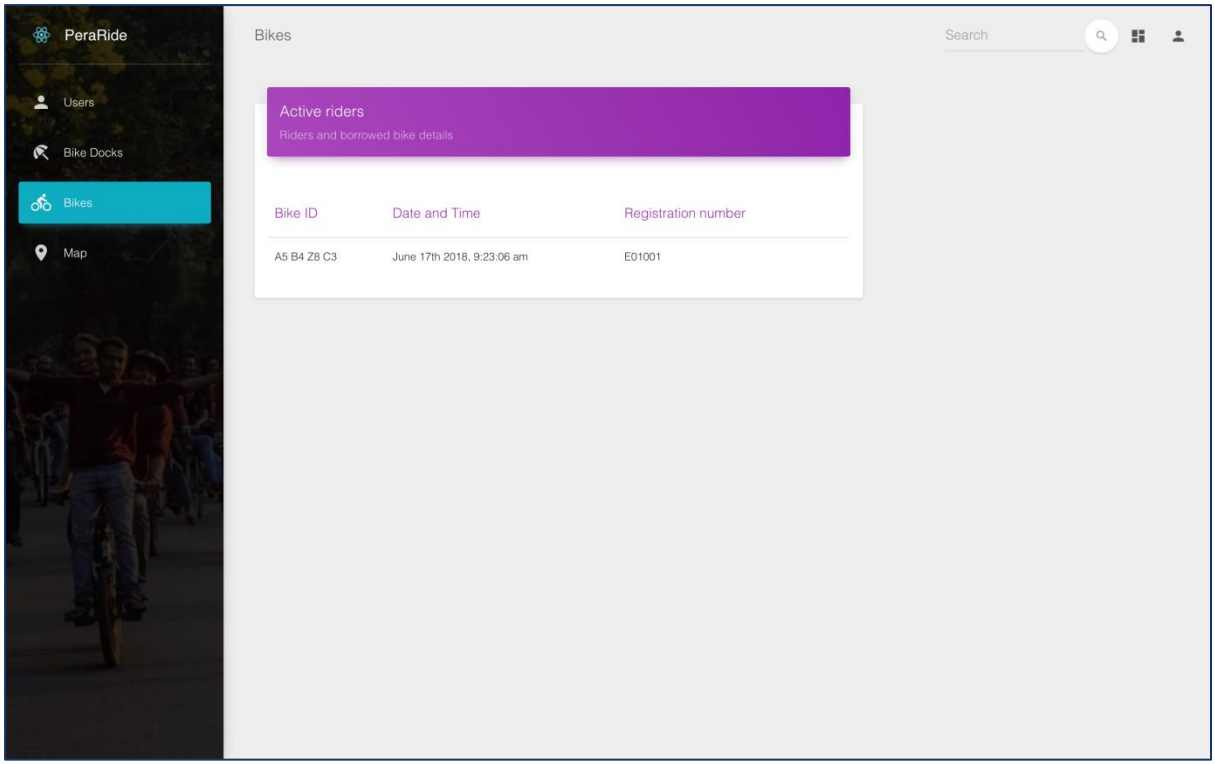


Figure 3.1.4 – Bicycle usage monitoring Interface

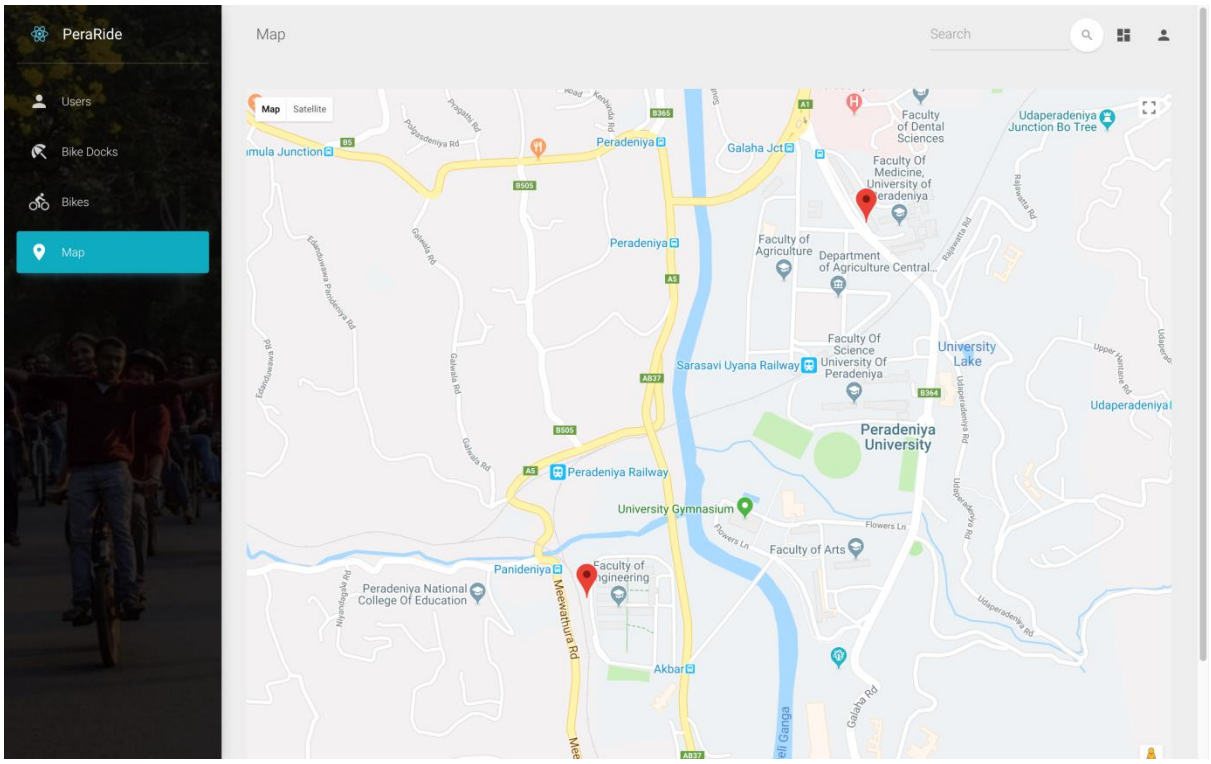


Figure 3.1.5 – Map view of the dock stations

3.1.2 MOBILE APPLICATION UI DESIGN

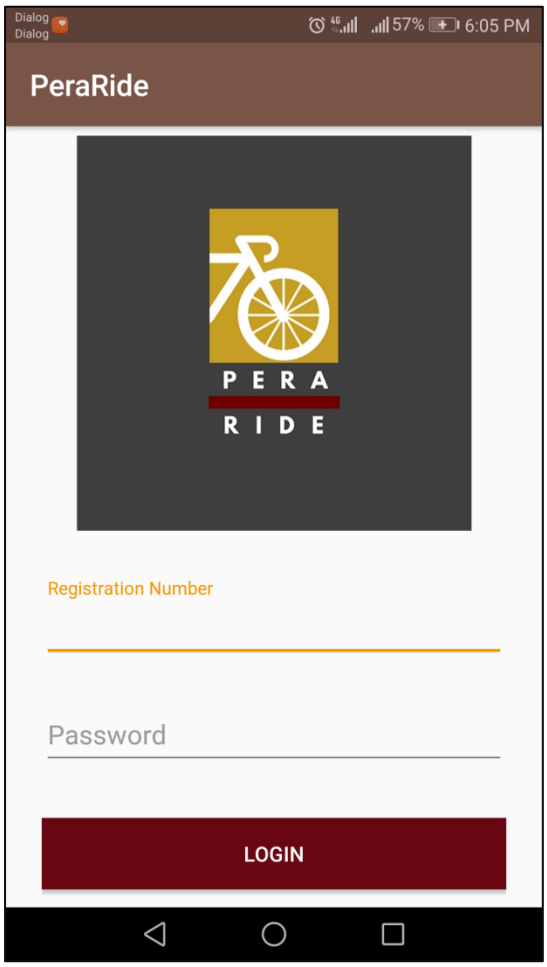


Figure 3.1.6 – Login Interface

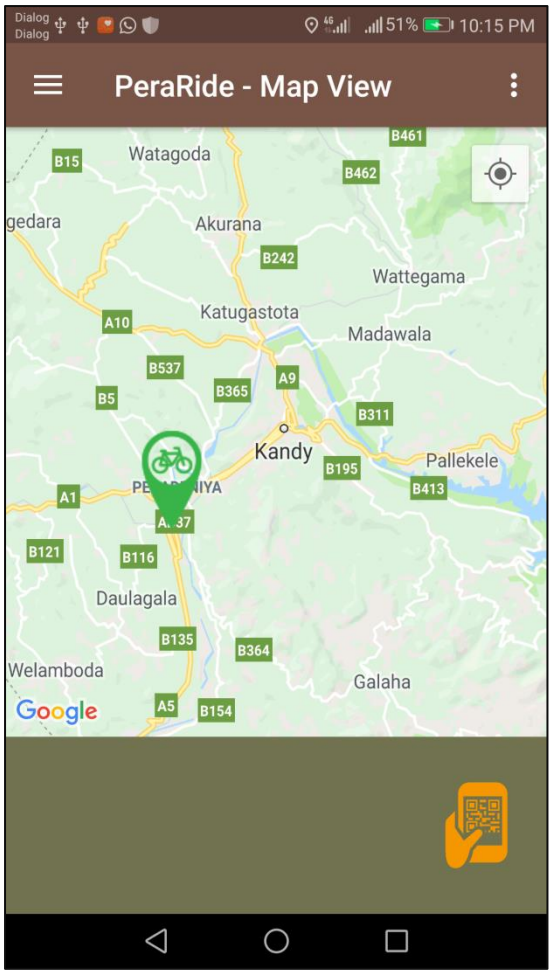


Figure 3.1.7 – Map View

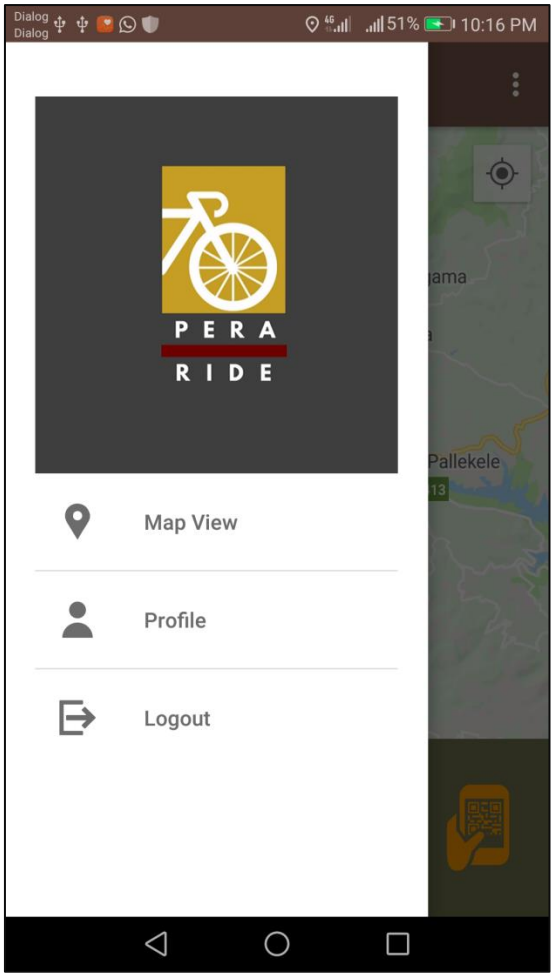


Figure 3.1.8 – Navigation Window

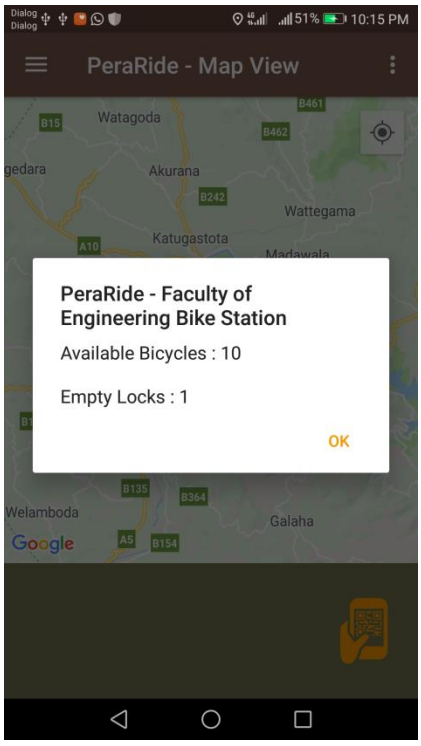


Figure 3.1.9 – Dock Station Details UI

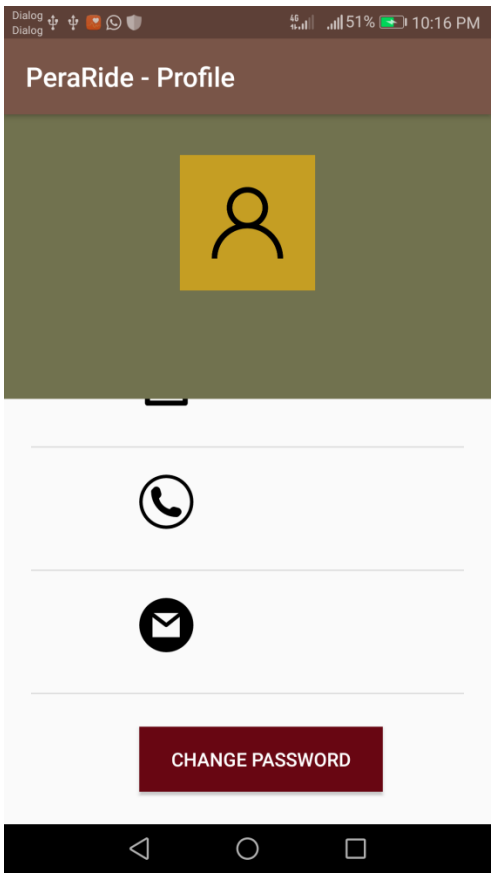


Figure 3.1.10 – Profile Interface

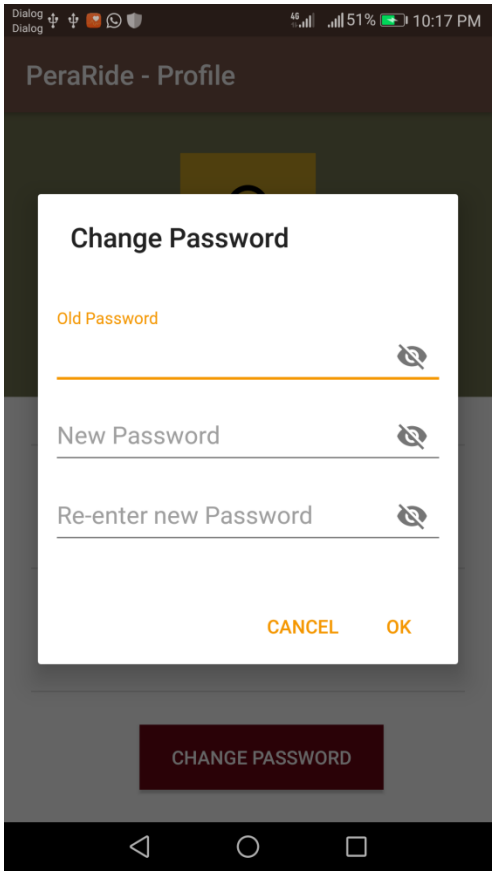
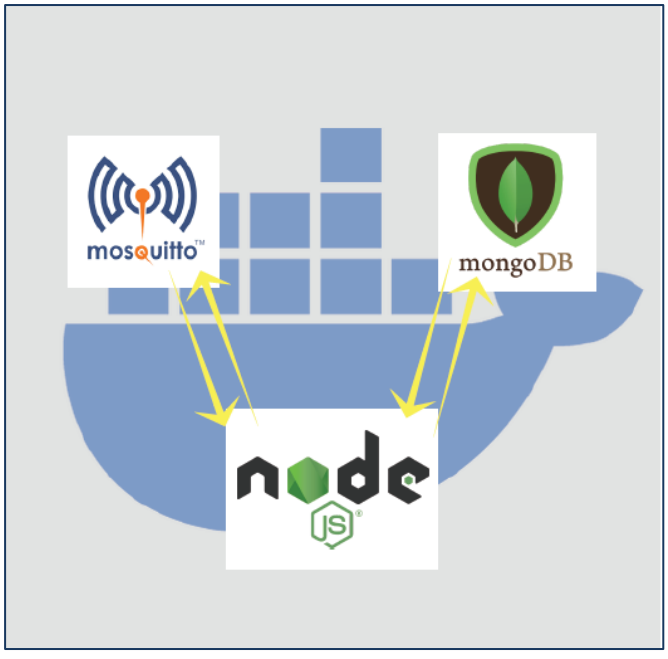


Figure 3.1.11 – Password change ui

3.2 BACK END DESIGN



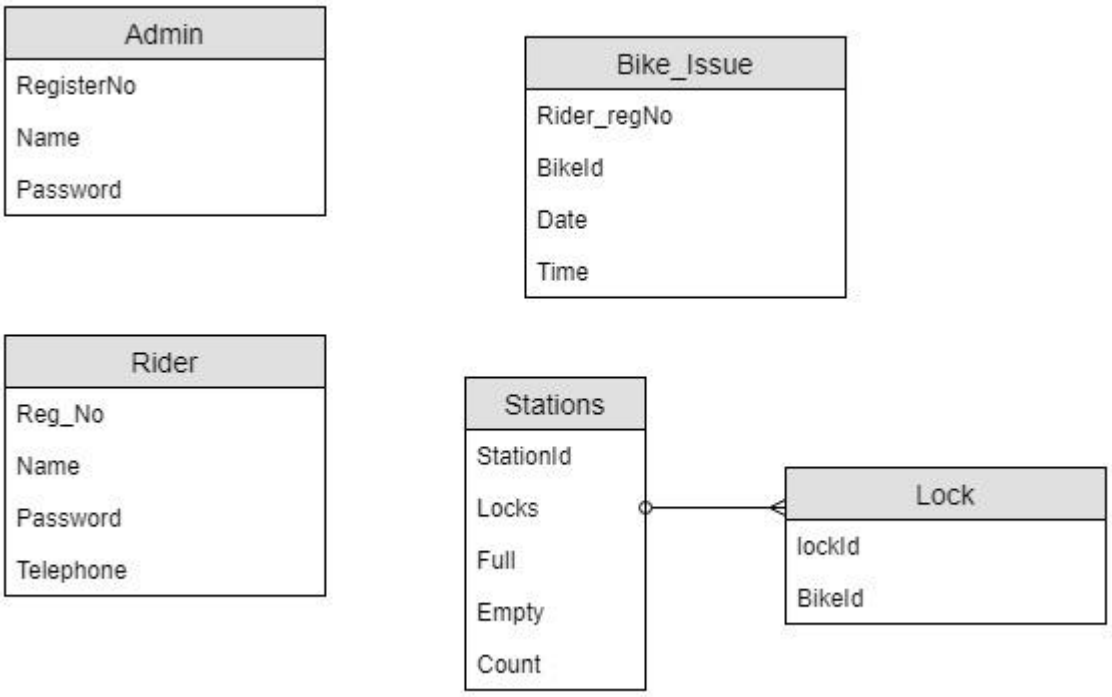
3.2.1 MQTT BROKER

We use Mosquito as our MQTT message broker. It is an open source message broker. Both server and the station subscribe to the relevant topics and publish the message. TLS/SSL will be used to provide a secure communication channel between broker and the client.

3.2.2 MONGODB

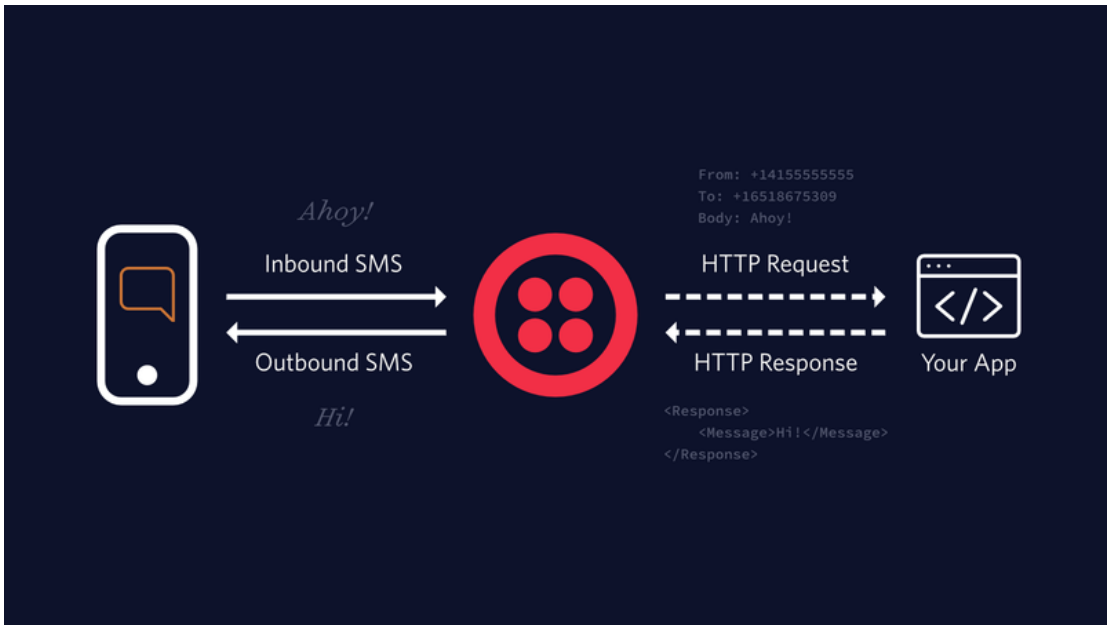
MongoDB is an object oriented, simple, dynamic and scalable no SQL database. The data objects are stored as separated documents inside the collection. The motivation of the MongoDB language is to implement a data store that provides high performance, high availability, and automatic scaling. MongoDB uses JSON or BSON documents to store data.

Database Schema



3.2.3 SMS GATEWAY

This system is used when the mobile app cannot be used to unlock the bicycle. It is simply sending a sms with the bike number to unlock the bike. Twilio SMS API is used to send the http request to the server to unlock the bicycle.



LIST OF THIRD PARTY SOFTWARE TOOLS

- Reactjs - javascript UI
  https://reactjs.org/
- Electronjs - cross platform desktop app
  https://electronjs.org/
- Google maps api (javascript)
  https://developers.google.com/maps/documentation/javascript/tutorial
- Mosquitto MQTT broker
  https://mosquitto.org/
- Docker platform
  https://www.docker.com/
- Twilio
  https://www.twilio.com/

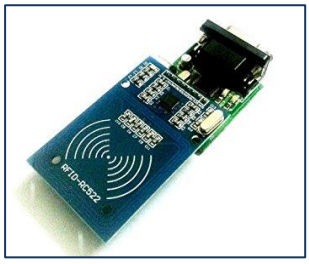
3.3 HARDWARE DESIGN

Implementation of smart locks for bike dock stations is the main task under hardware design. Smart lock is going to be designed using a linear actuator (solenoid) which triggered when the current flows. The smart lock unlocks when the rider reads the QR code and locks when the rider returns the bike, RFID reader reads the RFID tag in the bike and then the smart lock works.

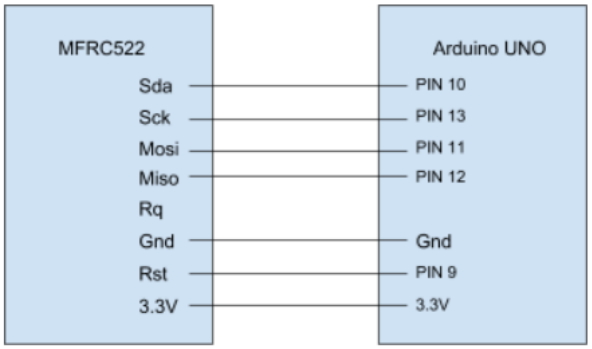
Each lock in dock station contains RFID reader and lock components which are connected to an Arduino nano board. Those Arduino nano boards are connected to an Ardiuno mega board using I2C communication bus. GSM/GPRS module is used for the wireless communication between the hardware components and the server. In between the server and the hardware components, there is a MQTT broker which is primarily responsible for receiving all messages, filtering them, decide who is interested in it and then sending the message to all subscribed clients.

3.2.1 SPECIFICATION OF COMPONENTS

MFRC522 RFID Reader/Writer



- Communication frequency - 13.56 MHz
- Power supply - 2.5V to 3.3V
- Supported host interfaces - *SPI*, I2C-bus interface, RS232 serial UART



TIP41C transistor

- Maximum Ratings
- Collector-emitter voltage - 100V
- Collector current - 6.0A

- Characteristics
- Saturation voltage - 1.5V IC=6A, IB=0.6A

D400 transistor

- Maximum Ratings
- Collector-emitter voltage - 25V
- Collector current - 1.0A

- Characteristics
- Saturation voltage - 0.1V; IC=500mA, IB=50mA

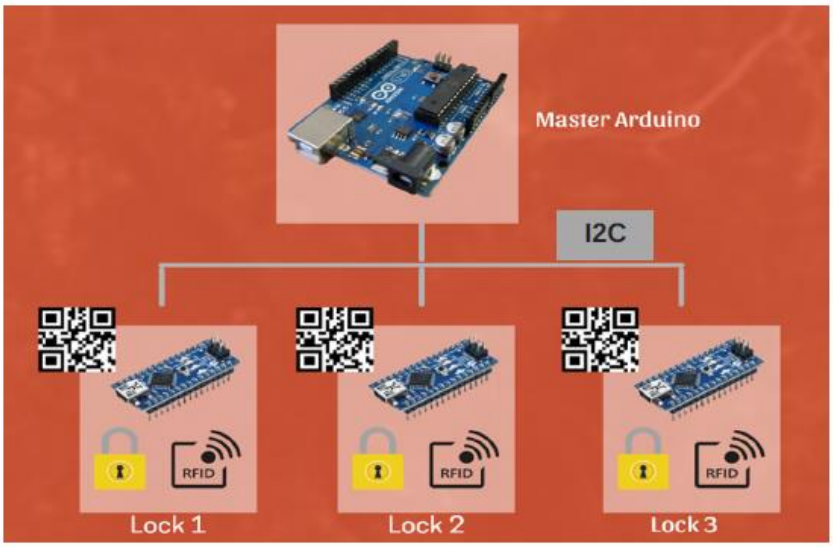
Pull type Solenoid

- Maximum Ratings
- Voltage - 12V
- Current - 2A
- Force - 20N

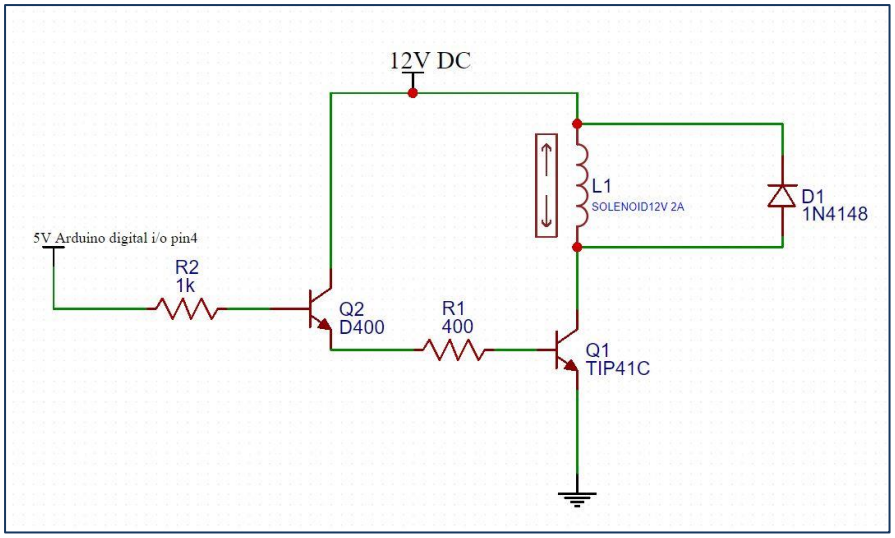
SIM808 Shield - GSM&GPRS + GPS

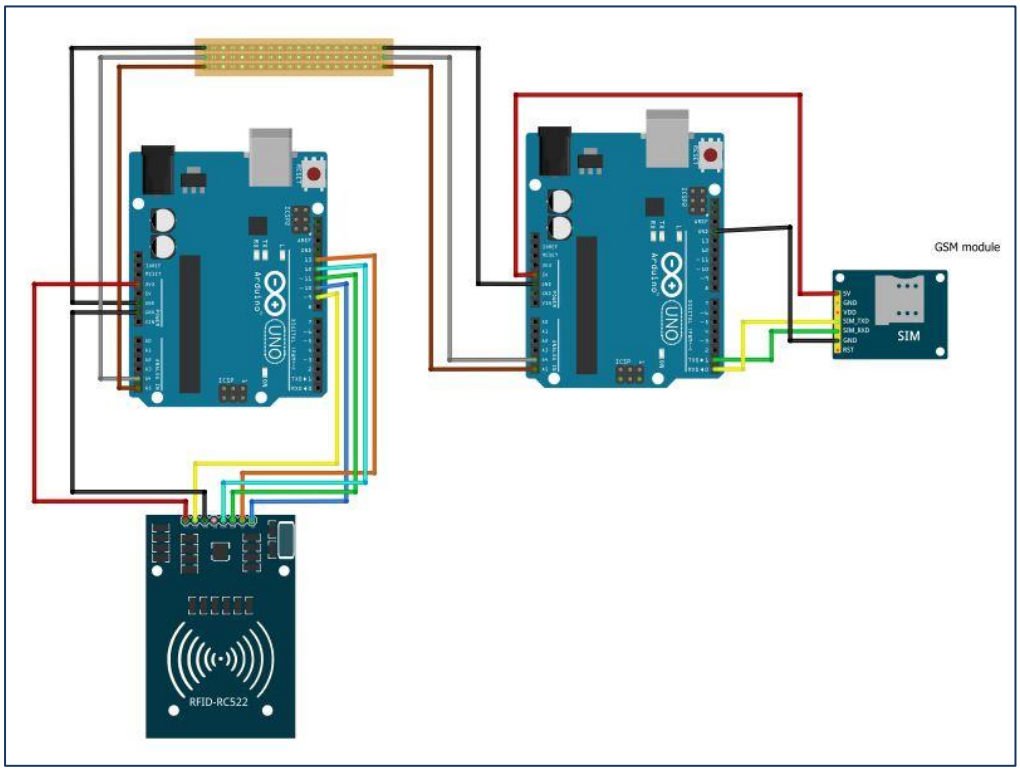
- Compatible with Arduino Uno, Arduino Leonardo, Arduino Mega
- Powered via Arduino USB connection or Arduino external 12V power supply
- Quad-band 850/900/1800/1900MHz – 2G GSM network
- GPRS multi-slot class12 connectivity: max. 85.6kbps(down-load/up-load)
- GPRS mobile station class B

3.3.2 DESIGN OF THE DOCK STATION

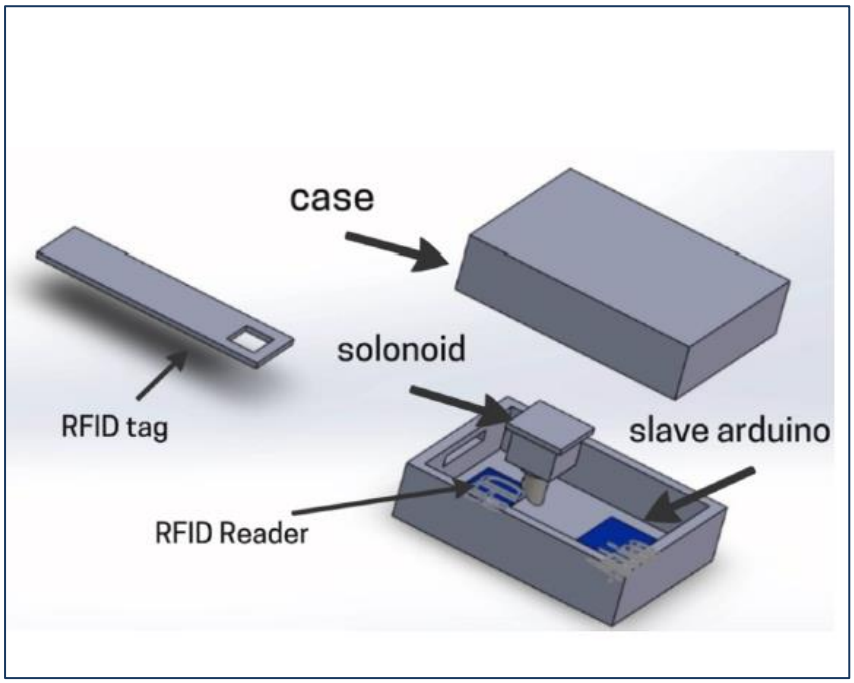


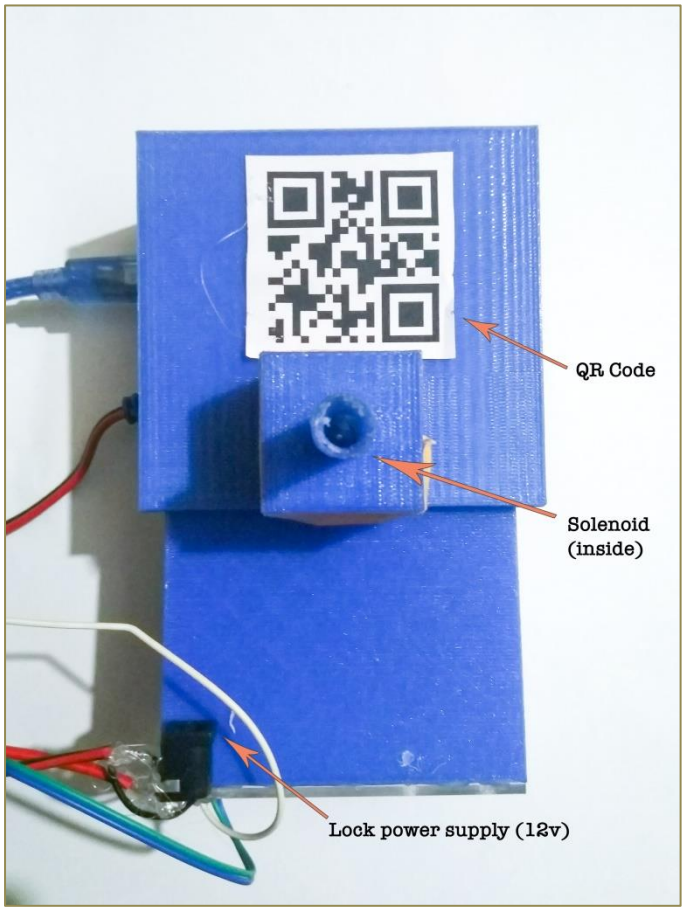
3.3.3 CIRCUIT DESIGN

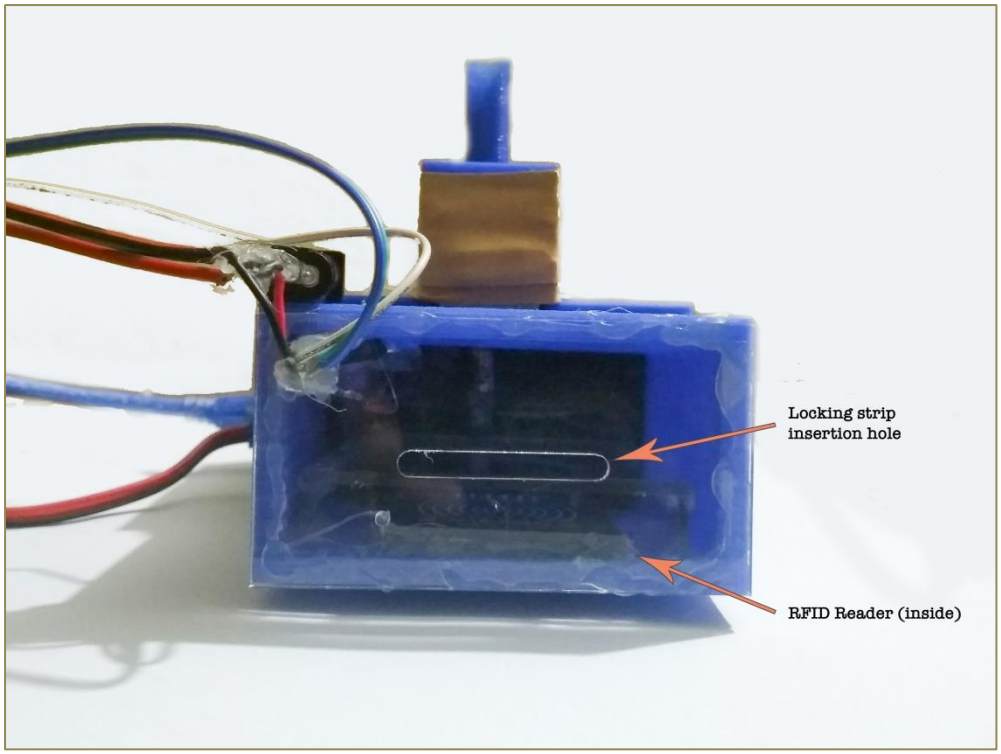

12V DC
L1
SOLENOID12V 2A
D1
1N4148
5V Arduino digital i/o pin4
R2
1k
Q2
D400
R1
400
Q1
TIP41C

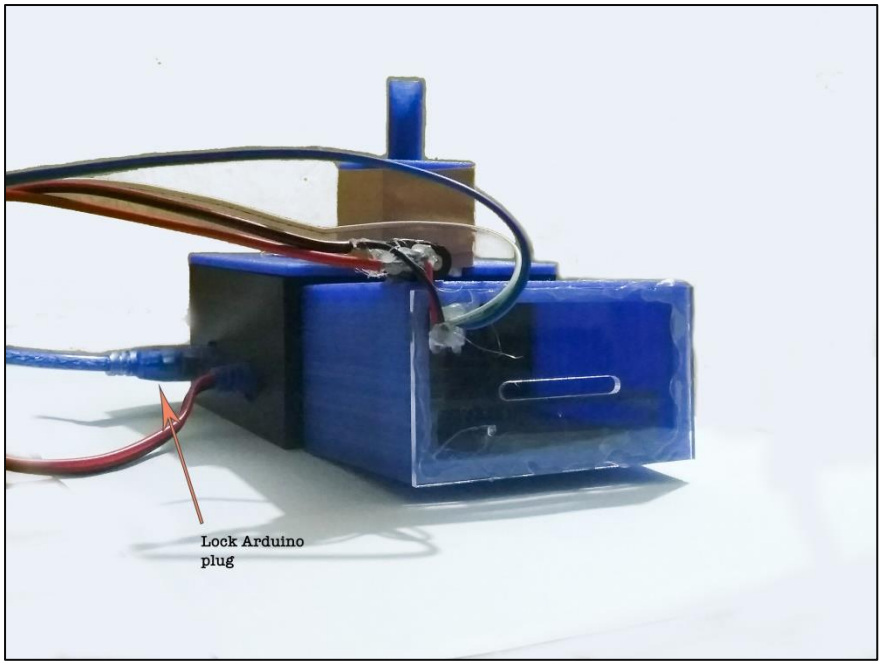

GSM module
SIM
RFID-RC522

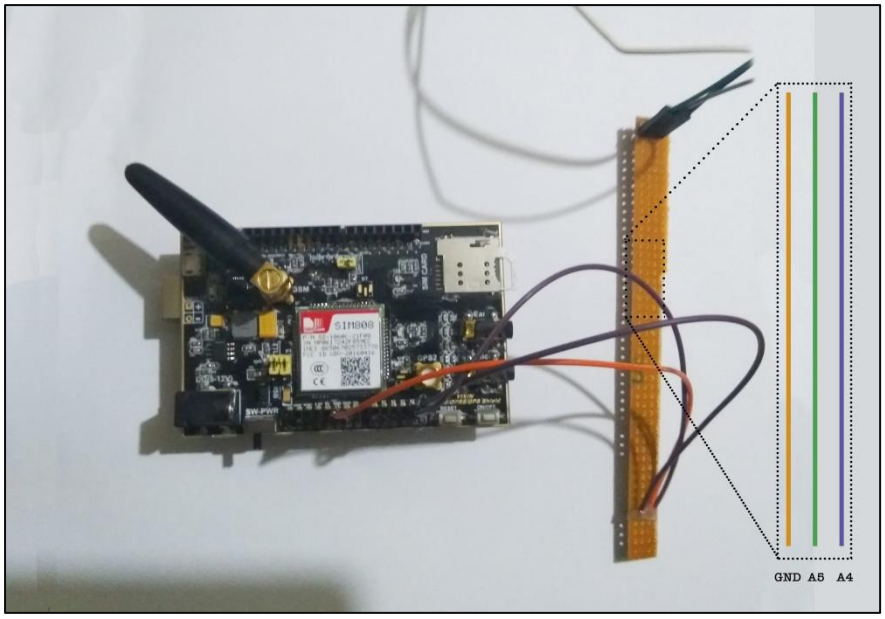
3.3.4 DESIGN OF THE BICYCLE LOCK



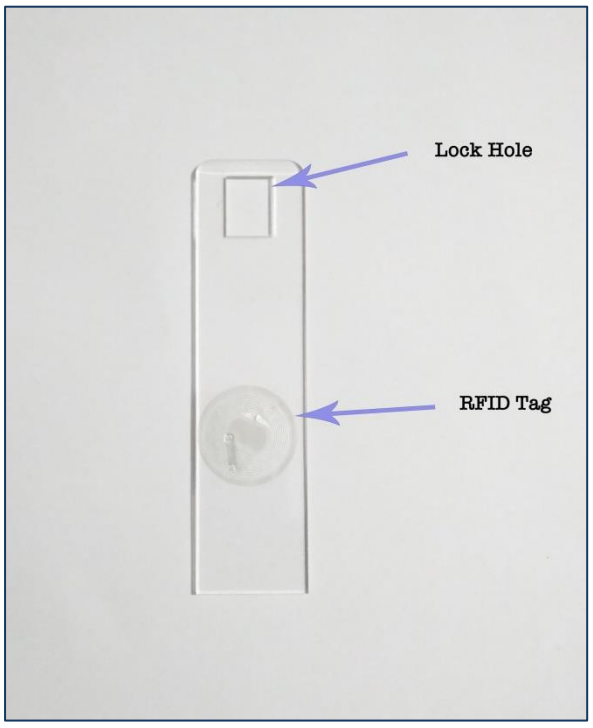



Locking strip
insertion hole
RFID Reader (inside)


Lock Arduino
plug

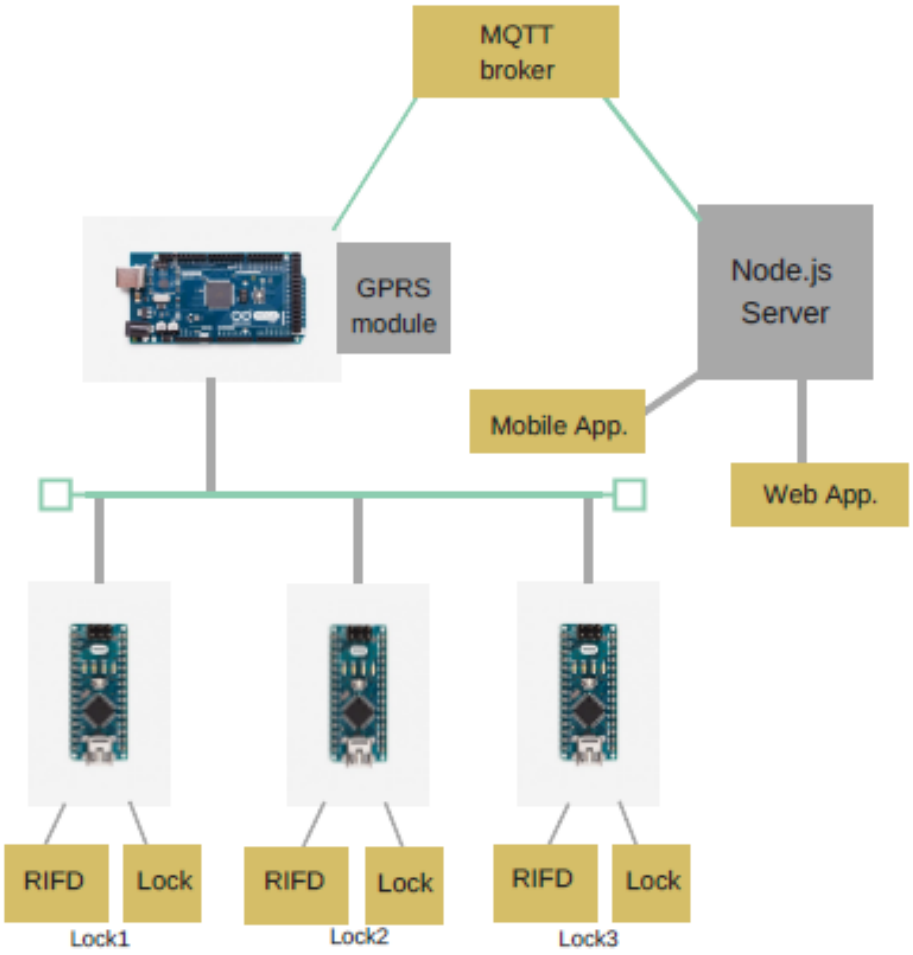


Connecting part to master Arduino



- Component that is fixed to the bicycle -

3.4 OVERALL SYSTEM DESIGN



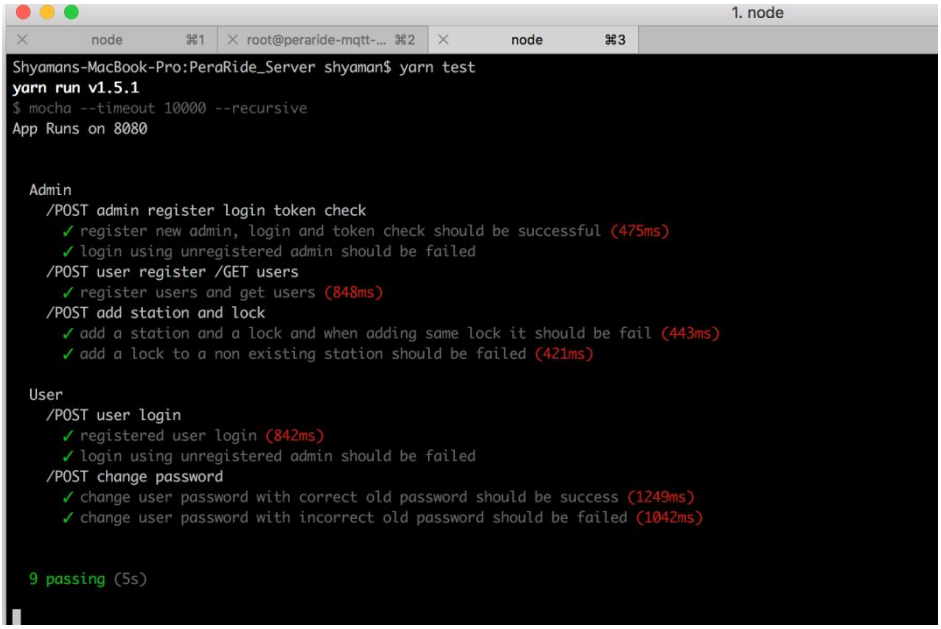
4.1 MOBILE APP TESTING

- Local Unit Tests - used to test isolated the functionality of a certain component
- Instrumented Tests - tests to control the life cycle and user interaction events.

- Android Testing Support library (ATSL) - JUnit 4-compatible test runner (AndroidJUnitRunner), the Espresso test framework and the UI Automator test framework

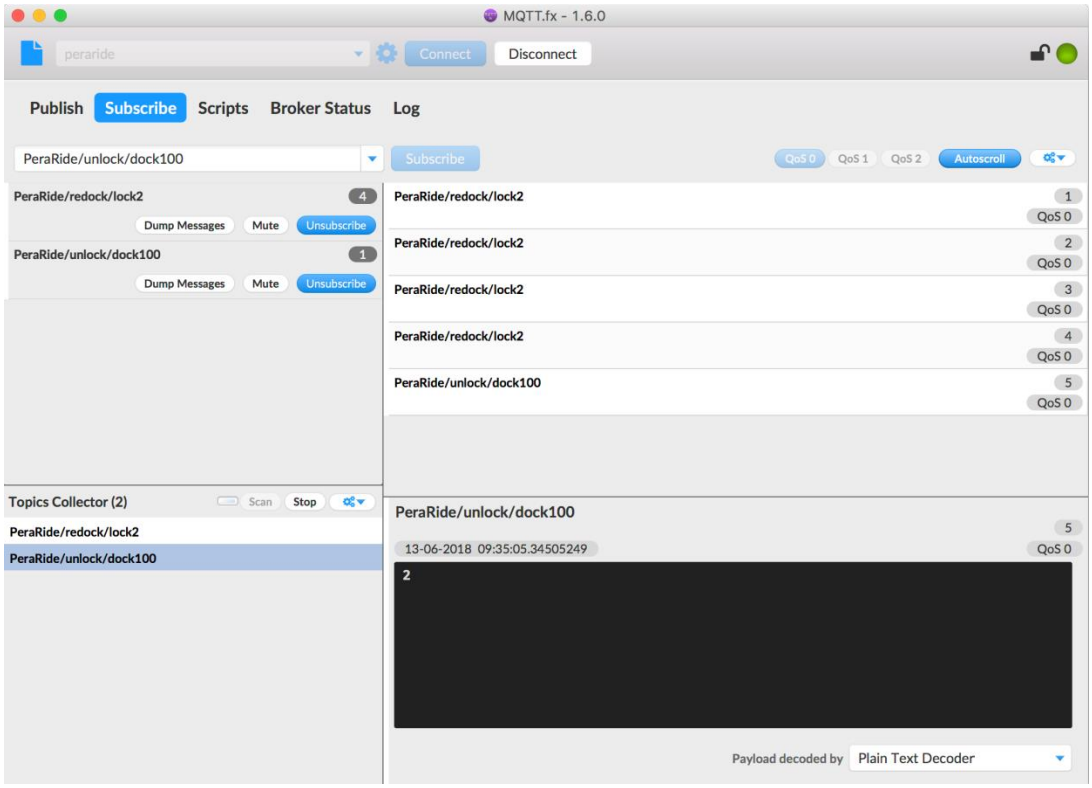
4.2 API TESTING

- Api testing with Mocha and Chai testing framework



4.3 MQTT BROKER TESTING

- MQTT Broker was tested by publishing messages to relevant topics and observed in MQTT.fx client



4.4 HARDWARE

RFID reader

- Tested for various distances between reader and tag. Found that to get correct readings distance should be less than 1cm.

Solenoid

- Tested for different voltages and minimum voltage should be 9.2V.
- Current should be at least 1.2A to get minimum functionality.