



CMRL Internship

Project:
Android App &
Web Development

Chennai Metro Rail Limited

Address

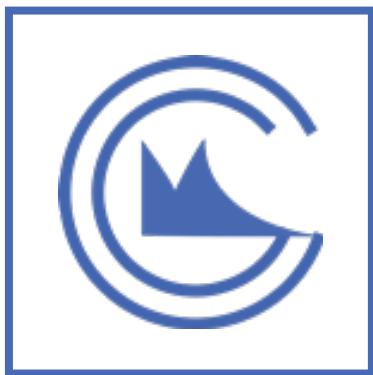
Chennai Metro Rail Limited,
Admin Building, CMRL Depot,
Poonamallee High Road,
Koyambedu,
Chennai – 600107,
TamilNadu, India.

Phone & Fax

Phone : +91 - 44 - 23792000
Fax : +91 - 44 - 23792000

Online

Email : chennaimetrorail@gmail.com
Website : www.chennaimetrorail.gov.in



Dr.B.Umesh Rai,PhD.,IRSEE

Position: Cheif General Manager

| Experience: 31 years

Dr. Umesh sir pursued his engineering in Delhi College of Engineering and did his masters in technology of Electronics Design in Indian Institute of Science. He further did his Masters in Business Administration in MDI, Gurgaon. Furthermore he did his PhD in Indian Institute of Science and did research and cited papers on Bond Graph Models and put them in use.

*The content and the logo used are for representation purpose only. The above logo design and the artwork you are viewing is the intellectual property of the copyright and/or trademark holder and is offered to you as a convenience for lawful use with proper permission from the copyright and/or trademark holder only. It is being used for non commercial purpose without infringing on the rights of the copyright and/or trademark holder and in compliance with the DMCA act of 1998.

ACKNOWLEDGEMENT

I express my deep sense of gratitude to Dr.B.Umesh Rai sir, Cheif General Manager(System Dept.) for encouraging me to take this internship at this reputed organization Chennai Metro Rail Ltd.

I acknowledge with thanks the kind of patronage, loving inspiration and timely guidance which i have received from my course coordinator Mr.Simson Antony (AM-TVS) and Mr.Hari Prasad sir.

I am very much thankful to Mr.Praveen Kumar sir,Mr.Rajasekar sir and IT Department for their valuable guidance, keen interest and encouragement at various stages of my internship period along with answering all my curious questions in the field of Networking.

I express deep and sincere gratitude to the whole team who guided me in knowing about the working and the management of Chennai Metrol Rail Limited.

R.MOHAN RAJAN



CONTENTS

In this Report,I will be explaining on my work done in CMRL and how it can be put into use.

01

02

07

11

14

INTRODUCTION

A brief introduction about the project and how its made use of. Also a short introduction about Chennai Metro Rail Ltd.

INSTALLATION

Step by Step process on installing the webapp in both server and localhost and creating models for use inside by the company.

USING WEBSITE

Clear Detailed picture on how to use the website in a step by step manner with each query or button stated.

USING APP

Starting from Login, all the procedures to execute a query in an app is being explained.

CONCLUSION

Stating the work done in CMRL and the experience i gained from it.



INTRODUCTION

The Government of Tamil Nadu created a Special Purpose Vehicle (SPV) for implementing the Chennai Metro Rail Project. This SPV named as “Chennai Metro Rail Limited” was incorporated on 03.12.2007 under the Companies Act. It has now been converted into a Joint Venture of Government of India and Government of Tamil Nadu with equal equity holding. It is functioning at the following address.

This report provides guidance for smart phone based App Development for asset maintenance of Tunnel Ventilation System (TVS) of Chennai Metro Rail (CMRL) Project (Phase-1).

The Chennai Metro Rail Project has been designed with a high degree of reliability in order to provide dependable service to the public. Best current practices from International Metro standards & Design have been adopted with particular emphasis on life safety systems and necessary inbuilt electrical plant redundancy to support these.

The Tunnel Ventilation System provided for the Station as well as for Tunnel comprises of Tunnel Ventilation Fans, Trackway Exhaust Fans, Trackway Supply Fan, Jet Fans Tunnel Ventilation Dampers, MCC Panels, LCP Panels, IBP Panels and Sound Attenuators.

The main aim of this report is to provide guidance to the App Developers on the requirements of the TVS assent maintenance.

A smart phone based app needs to be developed for record and update all periodic maintenance of the TVS equipment installed at all underground stations.

In general the app should have following functionalities:

1. To scan/read the Quick Response (QR) code of every equipment.
2. To display checklist for periodic maintenance.
3. To aid fault reporting and closing out existing faults.
4. To send SMS/email alerts.
5. To store and update data from the cloud based database.
6. To upload and download data from the database.
7. Store history on the smart phone.
8. Generate short reports.

The Detailed guide on how to use the app and website will be given in the following pages.



INSTALLATION

We'll start with how to install the webapp on the server. If you want to test the webapp ,and i've classified this part into two steps. Either way i would start off with one main thing, that is the github repository.

Username : cmrlweb

Both Web and Android : https://github.com/cmrlweb/CMRL_ALL.git

Only Web : https://github.com/cmrlweb/CMRL_LARAVEL.git

Now to install the Webapp,there are 2 ways.

- 1.Localhost
- 2.Web Server (Cloud)

LocalHost: (tested on Ubuntu 14.04)



1.Open Terminal.

2.Run the following commands in the same order. (From ~ directory (cd ~))

```
$sudo apt-get update  
//Updates the Ubuntu to Latest Version
```

```
$sudo apt-get install apache2  
//Installing Apache
```

```
$sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt php5-curl  
//php , type yes if necessary
```

```
$sudo apt-get install mysql-server php5-mysql  
//Installing Mysql Server specify username-root , password your wish
```

```
$sudo apt-get install git-core  
//installing git
```

```
$sudo apt-get install curl  
//Installing curl
```

```
$git clone https://github.com/cmrlweb/CMRL_ALL.git CMRL_ALL  
//Cloning to your Host.
```



```
$curl -sS https://getcomposer.org/installer | php  
//Installing Composer
```

```
$sudo mv composer.phar /usr/local/bin/composer  
//Making composer Global.
```

- 3.Now you have to import Mysql into your Localhost.
- 4.Install Mysqlworkbench

```
$sudo apt-get install mysql-workbench
```

5.Now connect to localhost by clicking + option.

6.When you install Mysql , you must have specified username and password , now specify that here and connect.

7.Now go to your FileManager and go to Home Directory and find CMRL_ALL directory

8.Now go to CMRL_ALL ->Work File->MYSQLBACKUP->MAINSQL->MAINMYSQL.

9.Open CMRL.sql

10.Copy all and go to workbench.

11.Create Database called CMRL.

12.Now run Copied queries and paste in query window. (Click on Thunderbolt icon to run ALL queries)

13.Now You need to create an admin.This is tricky.I have made that the first register in the name of email will have permission to be admin.Ill explain you that.

14.Go back to terminal and type these.

```
$cd ~
```

```
$cd CMRL_ALL/Work\ File/web/
```

```
$composer install  
//installing the webapp
```

```
$vim .env
```

//To set DB Credentials. Set the DB_HOST to 127.0.0.1 and DB_USERNAME to root and other details needed. This is main file to connect to db.

```
$php artisan serve  
//Now your webapp will be ready in firefox.
```

15.Now go to Firefox or Chrome and type http://localhost:8000/

16.Youll see login , Go to Register and register with this credential to be admin

Name : ***ANY***

email : cmrlweb@gmail.com

password : ***ANY***



17.No Go to terminal ahain and hit Ctrl+c

18.Now run this command

```
$php artisan db:seed.
```

//To populate the ROLE of admin in roles and permissions table.

```
$php artisan serve
```

19.Now Login to your Webapp.

Cloud Server: (tested on Ubuntu 14.04)

1.Same steps until level 2 of installing composer global.

2.Note : Since you cannot see mysql workbench on Cloud Server , you can use mysqldump to put the file.

```
$cd ~
```

```
$sudo apt-get libapache2-mod-auth-mysql
```

```
$sudo mysql_install_db
```

```
$sudo nano /etc/apache2/mods-enabled/dir.conf
```

// Add index.php infront of index.html and after Directory Index.

//itll look like this

```
<IfModule mod_dir.c>
DirectoryIndex index.php index.html index.cgi index.pl
index.php index.xhtml index.htm
</IfModule>
```

```
$mysql -u root -p
//give password
```

```
sql>CREATE DATABASE CMRL;
```

```
sql>exit;
```

```
$cd CMRL_ALL/Work\ File/MySQLBackup/MAINMySQL/
```

```
$mysqldump -u root -p --databases CMRL > CMRL.sql
```



```
$cd ~
```

```
$rm -rf CMRL_ALL
```

//Since we'll clone into apache directory (if you followed localhost setup)

//If no such file exists , no problem.

```
$cd /var/www
```

```
$git clone https://github.com/cmrlweb/CMRL_LARAVEL.git laravel
```

```
$cd laravel
```

```
$composer install
```

```
$nano /etc/apache2/sites-available/000-default.conf
```

// Change the Document Root to the following and add the following lines below it. The File will look like this

```
DocumentRoot /var/www/laravel/public
```

```
<Directory /var/www/laravel/public>
```

```
    AllowOverride All
```

```
    Require all granted
```

```
</Directory>
```

```
$sudo a2enmod rewrite
```

```
$sudo nano /etc/apache2/apache2.conf
```

//Add the lines after <Directory />.....</Directory> Listing

```
<Directory /var/www/laravel/public>
```

```
    Options Indexes FollowSymLinks MultiViews
```

```
    AllowOverride All
```

```
    Order allow,deny
```

```
    allow from all
```

```
    Require all granted
```

```
</Directory>
```

```
$sudo service apache2 restart
```

```
$cd /var/www/laravel/public
```

```
$vim .htaccess
```

//New File



```
Options +FollowSymLinks  
RewriteEngine On
```

```
RewriteCond %{REQUEST_FILENAME} !-d  
RewriteCond %{REQUEST_FILENAME} !-f  
RewriteRule ^ index.php [L]
```

//Save file.

```
$cd /var/www/
```

```
$chmod -R 777 laravel  
//644 if you want
```

```
$cd laravel
```

```
$composer install
```

```
$vim .env  
//Change the DB config. Given in Localhost Setup.
```

3.Now Follow the same for creating an admin as in LocalHOST. using php artisan db:seed

Done. You will now have the webapp running on your IP address.

Installing Android App.

- 1.Install the APK.
- 2.Only if the user is registered in webapp , the user can login.
Everything done in the apk will revolve around the logged in user.
- 3.Follow the tutorial given to the course co-ordinator for use.



Using Webapp (Laravel Framework):

Laravel is an open source PHP Framework, Hence the website is done so that you can resolve any errors by asking the community.

First of all. The website is very simplistic and easy to understand. No complications.

- 1.Login Page.
- 2.Asset Codes
- 3.Maintainence
- 4.Manufacturers
- 5.History
- 6.Report
- 7.Roles
- 8.Register Web User.

The screenshot shows a web application interface with the following elements:

- Header:** CMRL WEB, Welcome, AssetCode ▾, Manufacturers ▾, Maintainence ▾, History, Roles, Register Web User, Mohan Rajan ▾.
- Top Bar:** Home, You are logged in!
- Errors Section:** Errors
 - 1 Header Sample Error C2/STI/LHS/TVF/01/FW/Sept 2015
 - 2 Header Sample Error C2/STI/LHS/TVF/01/FW/Sept 2015
- Buttons:** Remove

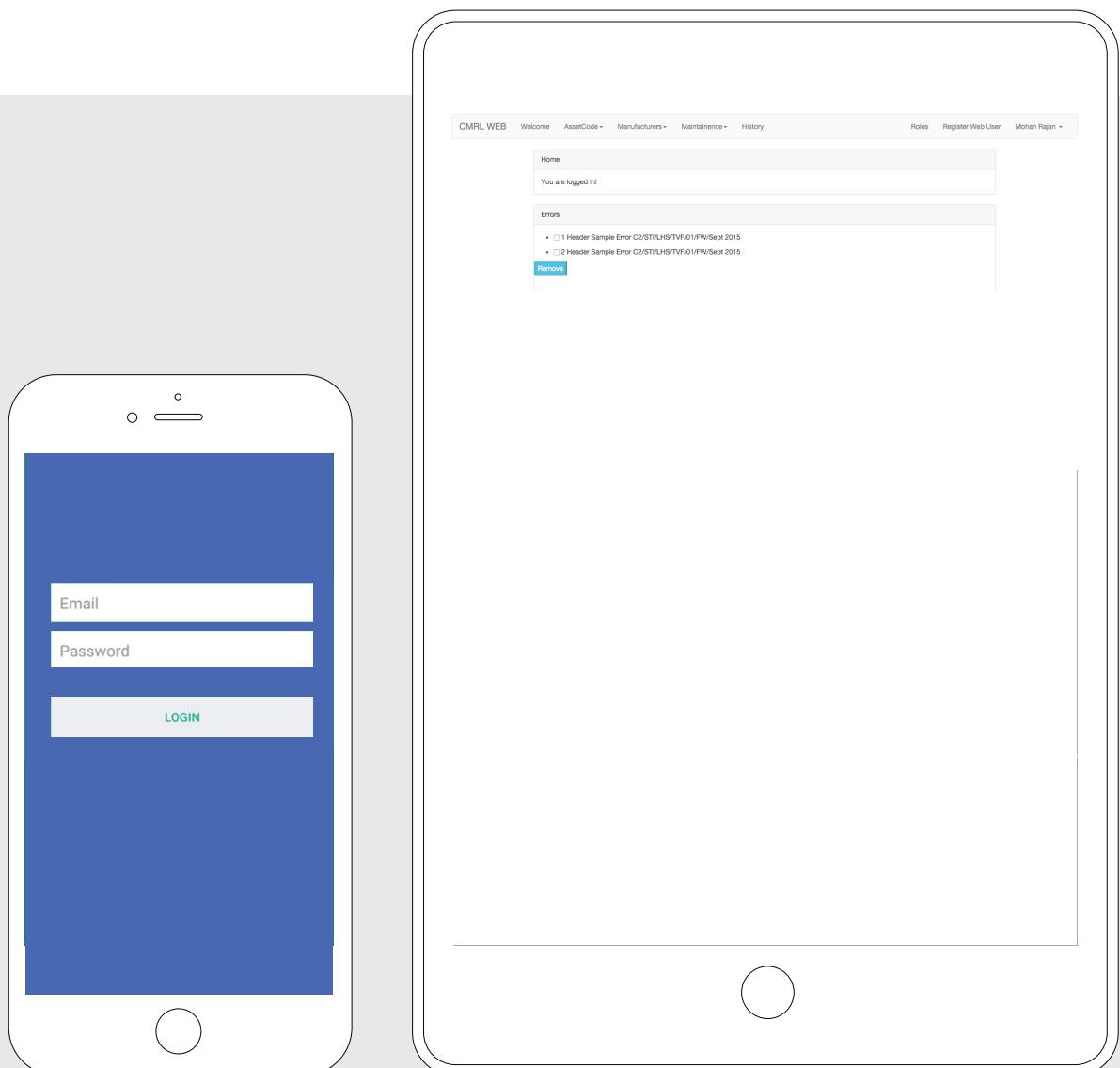


Similiar idea goes for App , Which will be explained later.

Login Page:

The users can be registered in the register page and login in the main page. It is self embedded with laravel so the security is inbuilt. If you want to see the no of customers, you can see it in the database. This page doesnt have soo much history into it.

You can register users and add roles to them by just registering a user in the login page.





Asset Codes Page:

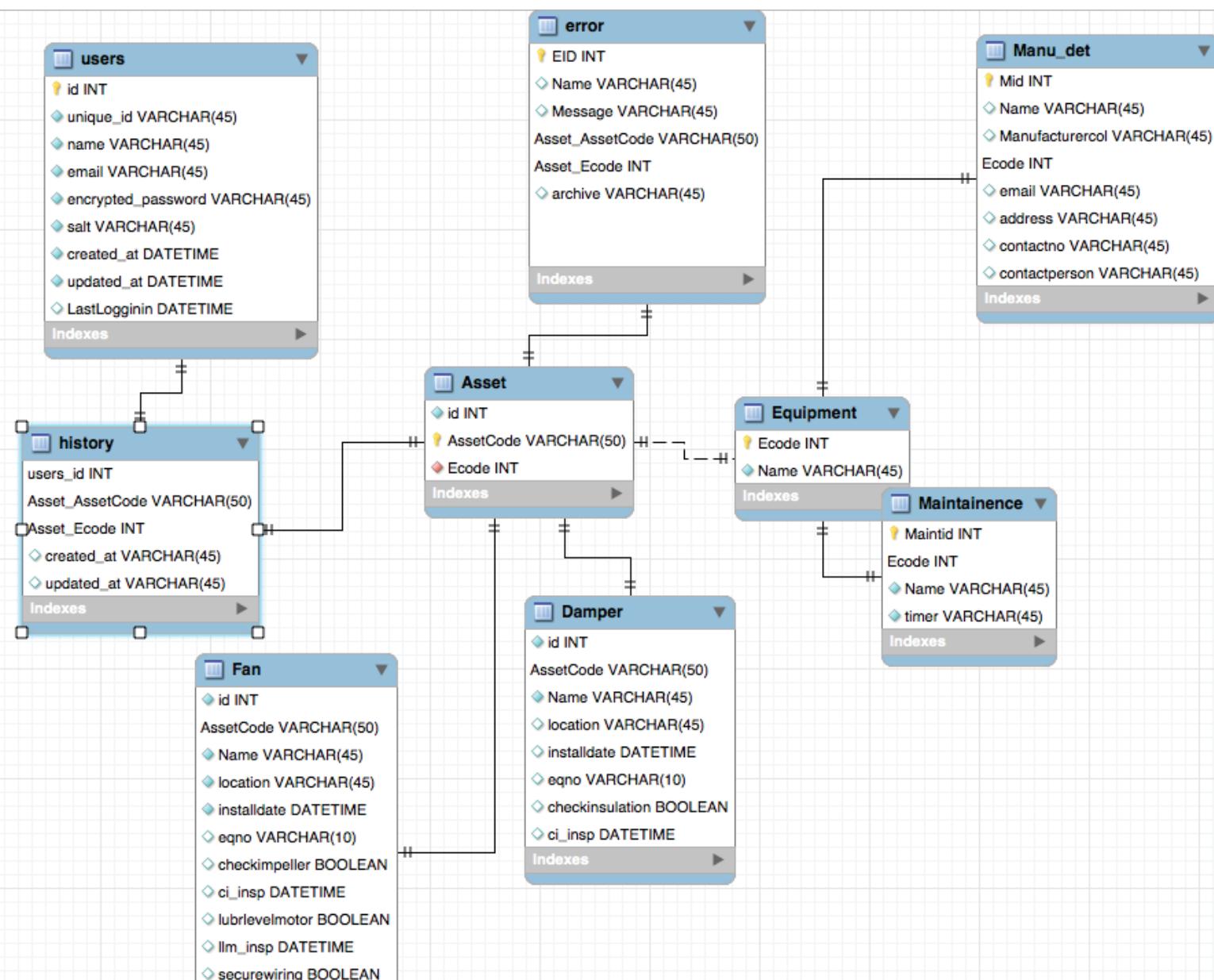
The AssetCodes sets the mark of being an Unique code which is exclusively around which this whole webapp and Android App revolves around. Hence it is Important to Add edit and Maintain them.

Each Assetcode has a unique request to be added. They have to be one of the Equipment. The SQL Workbench Map is listed Below.

So if you need to add an Assetcode. First add Equipment. You can see if there is any pre existing equipments.

Also You can see and Delete Equipments in Mysql Connection from Your Local Machine using Workbench.

You can then add Assetcode. And Note that , it should not have any spelling mistake.





Adding Equipments First:

For Each Equipment, You have to create table and create certain Checkbox Functions. This part of the report is very important because the models and tables are all connected. I have attached a SQL File in CMRL_ALL>Work File>cmrlmai.sql for reference with data for seeing how a model exists.

Basic Things needed in a Equipment table is :

- 1.id
- 2.AssetCode

```
CREATE TABLE `Model_Name` (
`id` int(100) NOT NULL AUTO_INCREMENT,
`AssetCode` varchar(100) NOT NULL,
//Your Checkboxes.
PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;
```

ex:

```
CREATE TABLE `Tunnel_Ventilation_Damper` (
`id` int(100) NOT NULL AUTO_INCREMENT,
`AssetCode` varchar(100) NOT NULL,
`Clean_Blades` tinyint(1) DEFAULT '0',
`Clean_Blades_d` varchar(5) NOT NULL,
`Check_Linkages` tinyint(1) DEFAULT '0',
`Check_Linkages_d` varchar(5) NOT NULL,
`Manual_closeopen` tinyint(1) DEFAULT '0',
`Manual_closeopen_d` varchar(5) NOT NULL,
`Frame_Tightness` tinyint(1) DEFAULT '0',
`Frame_Tightness_d` varchar(5) NOT NULL,
`Actuator_Wiring` tinyint(1) NOT NULL DEFAULT '0',
`Actuator_Wirig_d` varchar(5) NOT NULL,
PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;
```

Note that for each Checkbox you have to do this.

- 1.Add it in Maintenance table CheckBox_Name.
- 2.Add 2 fields in the Equipment table i) CheckBox_Name ii) CheckBox_Name_d

Maintenance table and equipment table are interlinked. CASE SENSITIVE.

- 1.Checkbox_Name : (BOOLEAN)Name of the Checkbox.
- 2.Checkbox_Name : (varchar{5})Days before its expiry. (1 Month - 30 Days.).

It'll Keep on decreasing everyday and we schedule and cron job in your server that will run a POST url to decrease the day Checkbox and Make the boolean go to NULL.



Adding Maintenance List:

For Each Checkbox you have to create a Maintenance List. You can do this through the webapp . You just have to create the equipment model , Then Create an equipment through web , You dont worry about linking db and your webapp creation. Now you have to create Checkboxes exactly same to what you did in db in your webapp. And make sure they are linked to the same Equipment. Theyll Create the Ecode Depending on what Information you Provide.

Adding Manufacturers List:

Its pretty easy considered with Equipments and AssetCodes. You just go to the webapp , select the equipment and add the manufacturer.

The main thing in the webapp is the AssetCode -> Equipment -> Maintenance.

History:

You can check the history of how each assetcode is being accessed and what the user has done.

- 1.QR_FETCH - He has done a qr fetch and didnt change the Values. Ask him to update the values.
- 2.CHANGED - He has updated the values on the given Time.

Using Android:

Its easy to go Activity by Activity.

Main Activity - It will Show what username it has logged into.

- 1.Qrcode - To go to Qr Activity and Scane QR Code
- 2.Pending - Files that are not yet uploaded in server.
- 3.Sync - Sync the tables of the Server with the app . This step is important.
- 4.Logout - Logs out the user.

QrCode Activity - Nothing Unusual . You scan QR CODE and it will send you automatically to next activity.

EquipmentActivity- Depending on QR CODE you can do two thing either online or offline.

Online:

- 1.Fetch Data : The Checkboxes get accumulated automatically and dynamically from server.
- 2.Save Data : Save the Checkbox value in the server.



Offline

1.Fetch Data : From Sync - It'll show which table it should show and you'll be shown the respective checkboxes.

2.Save Data - It will create a pending activity and you will be guided to there. PRESS Back button and come out.

Once you come to internet location you can go to pending activity and then Push all the data into the server.

Pending Activity - Local Sqlite database which stores the data which has to be uploaded into the server.

Final Note : To change the IP address to connect to in your Android App , you go to android studio and in your app folder , You will see a folder called as helper folder and other folder, There will be a file called as AppConfig.java

Set HOST = “YOUR_IP_ADDRESS” in AppConfig.java and you will be set in your Android application



Extra Details:

GMAIL ID :

Username : cmrlweb@gmail.com

Password : (To be handwritten) _____

Conclusion

Completing 2 and half years of studying at Anna University,I came across different theories in computer science.I was able to apply only few of those in practical assignments and projects given in university.

In the initial days of training i was able to know that industrial experience will be more different from the academic background i had at university.Seeing how to address business world problems, how to communicate with employees and identify the problems were some new experiences.

Being able to know that fact that even small changes should be monitored with such precise timings of regular inspection to be a error prone process.The intensity of analysis done for every part is quite high, so what we see is that one can execute without getting let out.

Chennai Metro Rail LTD. is a great place for interns and implant trainings,as the company uses cutting edge technologies and everyone are being exposed to new tech before its being made thoeretical.

LOGIN



Ready to discuss
your next project?
Let's start.

R.Mohan Rajan
CEG, Anna University.

Address
No-284, 26th Street,N-Block, Anna Nagar East,Chennai -600 102
Phone: 81440-87395 Phone2: 044-26281734

Online
Email 1: mohanrajan1996@gmail.com
Email 2: info@sasipolypack.com