name

SAVITRIBAI PHULE PUNE UNIVERSITY A SDL MINI PROJECT REPORT ON

DAILY VISIT REPORT APPLICATION

SUBMITTED TOWARDS THE

PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

BACHELOR OF ENGINEERING (Computer

Engineering)

BY

Name of the students Exam No:

Neha Kale 71922610E

Harshita Shinde 71922826D

Darshan Deshmukh 71922503F

Mohit Jha 71922677F

Under The Guidance of

Prof. Ketaki Bhoyar

Academic Year 2020-21



DEPARTMENT OF COMPUTER ENGINEERING Dr. D. Y. Patil Institute of Engineering, Management & Research

Akurdi, Pune.



Dr. D. Y. Patil Institute of Engineering, Management & Research

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the Project Entitled

DAILY VISIT REPORT APPLICATION

Submitted by

Name of the students Exam No:

Neha Kale 71922610E

Harshita Shinde 71922826T

Darshan Deshmukh 71922503F

Mohit Jha 71922677F

is a bonafide work carried out by Students under the supervision of **Prof. Ketaki Bhoyar** and it is submitted towards the partial fulfillment of the requirement of Bachelor of Engineering (Computer Engineering).

**Prof. Ketaki Bhoyar** Prof. P.P. Shevatekar

Internal Guide H.O.D

Dept. of Computer Engg. Dept. of Computer Engg.

Dr. A. V. Patil

Principal

Dr. D. Y. Patil Institute of Engineering, Management & Research

Sign of Internal Examiner Sign of External Examiner

PROJECT APPROVAL SHEET

DAILY VISIT REPORT APPLICATION

Is successfully completed by

Name of the students Exam No:

At

Neha Kale 71922610E

Harshita Shinde 71922826T

Darshan Deshmukh 71922503F

Mohit Jha 71922677F

DEPARTMENT OF COMPUTER ENGINEERING

(Dr. D. Y. Patil Institute of Engineering, Management & Research) SAVITRIBAI PHULE PUNE UNIVERSITY,PUNE ACADEMIC YEAR 2020-2021

**Prof. Ketaki Bhoyar** Prof. P.P.Shevtekar

Internal Guide H.O.D

Dept. of Computer Engg. Dept. of Computer Engg.

Abstract

Daily Visit report is an application which is basically made for

the employees who have on-site duties like marketing, selling products,

meetings, business visits, etc.

Through this application the employee can ping their visit locations and the data will directly get uploaded on the real-time database. As a result, the company can view all the check-in information and location of a particular employee. Also, the employees can add bills and expenses and upload them so that company can refund their accounts with the total expenses of the employee. We have also given the functionality of a TO-DO list as it becomes easy for the employee/user to add, review, update and delete(once the task is done) the tasks inside the application itself.

There is one performance tab where daily ranking of the employees will be visible for all the users which will create a healthy competition among them and they will perform more efficiently, in turn is beneficial for their personal as well as business growth.

The company have to create an account and then will add employees into it, and the login credentials will be sent to the employee through email service. Employee can login using those credentials only.

Acknowledgments

It gives us great pleasure in presenting the preliminary project report on

‘Daily Visit Report Application’.

I would like to take this opportunity to thank my internal guide **Prof. Ketaki Bhoyar** for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful.

I am also grateful to Prof. P.P. Shevtekar, Head of Computer Engineer- ing Department, Dr. D. Y. Patil Institute of Engineering, Management

& Research for his indispensable support, suggestions.

In the end our special thanks to **Prof. Ketaki Bhoyar** for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for Our Project.

Name & Roll No of Students

Neha Kale 18101

Harshita Shinde 18104

Darshan Deshmukh 18147

Mohit Jha 18118

(T.E. Computer Engg.)

Contents

1 Synopsis 10

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1.1 | | Project Title . . | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 10 |
| 1.2 | | Project Option | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 10 |
| 1.3 | | Internal Guide . | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 10 |
| 1.4 | | Problem Statement | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 10 |
| 1.5 | | Abstract . . . . . . | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 10 |
|  | 1.6 Goals and Objectives | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 11 |
| 2 | Technical Keywords | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 12 |
|  | 2.1 Area of Project . . . | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 12 |
|  | 2.2 Technical Keywords . | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 12 |
| 3 | Introduction | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 13 |
|  | 3.1 Project Idea . . . . . | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 13 |
| 3.2 Motivation of the Project . . . . . . . . . . . . . . . . . . . . 13 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 Problem Definition and scope | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 14 |
| 4.1 Problem Statement: . . . . . . . . . . . . . . . . . . . . . . . . 14 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4.1.1 | | | Goals and objectives: | | | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 14 |
| 4.1.2 | | | Statement of scope . | | | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 14 |
|  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4.2 | | Applications: . . . . . . . . . | | | | | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 15 |
| 4.3 | | Hardware Resources Required | | | | | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 16 |
| 4.4 | | Software Resources Required . | | | | | | | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 16 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | Software requirement specification |  |  |  |  |  |  |  |  |  |  |  |  |  | 17 |
|  | 5.1 Data Model and Description . . . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | 18 |
|  | 5.1.1 Data Description . . . . . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | 18 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 5.2 Functional Model and Description . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | 18 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.2.1 | ER Diagram: . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 22 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 6 | Project Implementation | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 23 |
|  | 6.1 Introduction . . . . . | | | | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 23 |
| 6.2 Tools and Technologies Used . . . . . . . . . . . . . . . . . . . 23 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6.2.1 | | Tool: . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 23 |
| 6.2.2 | | Technology: | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 23 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | Software Testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 25 |
|  | 7.1 Type of Testing Used . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 25 |
|  | 7.1.1 Unit testing: . . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 25 |
|  | 7.1.2 Integration testing: . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 26 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | Results |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 28 |
|  | 8.1 Screen shots | . . . . . . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 28 |
|  | 8.2 Outputs . . | . . . . . . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 29 |
| 9 | Conclusion and | future scope |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 32 |
| 10 | References |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 33 |

List of Figures

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5.1 | PMS 0 diagram | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 19 |
| 5.2 | PMS 1 diagram | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 19 |
| 5.3 | PMS 2 diagram | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 20 |
| 5.4 | Activity diagram |  | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 21 |
| 5.5 | ER diagram . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 22 |
| 7.1 | Unit Test . . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 26 |
| 7.2 | Integration test | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 27 |
| 8.1 | Screenshot . . . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | . | 28 |

List of Tables

4.1 Hardware Requirements . . . . . . . . . . . . . . . . . . . . . 16

Chapter 1

Synopsis

1.1 Pro ject Title

Daily Visit Report Application

1.2 Pro ject Option

Internal project

1.3 Internal Guide

Prof. Ketaki Bhoyar

1.4 Problem Statement

Implement a location based Daily Visit Report Application with additional

features for convenience.

1.5 Abstract

Daily Visit Report is an application to record all the visit reports by the employee which will be

reported directly to the unit head. This app is a must-have for anyone working in the field and on

sites. Easily log in hours each employee has spent to complete a task will get recorded through this application.

This app will also have the GPS based location service that will help to managers to get the live

location of the employee working on the different field or sites.

1.6 Goals and Objectives(Darshan)

Improve and grow your sales by optimizing your sales teams travels and activity on the field.

Record each sales visits, later review, follow up and convert them to leads & clients.

Employees can keep records of their expenses during working hours to the organization. They can attach the details of the expense. These expenses later can be approved/rejected by higher management.

Don’t Miss a Certain with a built-in todo in hand.

Get Motivated by watching how other colleagues are doing.

Chapter 2

Technical Keywords

2.1 Area of Project

Business

2.2 Technical Keywords

Daily Visit Report, Checkins, Marketing, Business, Gps Based, Sales Reporting, Real time, Expenses with bill Image, Todo List,

Rank List, Monitoring

Chapter 3

Introduction

3.1 Project Idea

The daily visit report application records all the visit routines of the employees in the respective company/ business. This helps in keeping everything online and at one single place. It will have the functionalities of Checkins(which also ping the visit location), expenses and bills which are paid by the employee and the total expenditure will be refunded to the employee. This is a overall package for any company or business which will manage all the problems faced by a them.

3.2 Motivation of the Project

In this tech freak world, everyone wants everything online in one click and want to get rid of all the paperwork. But, there are no applications made as such to handle businesses and marketing effectively

This is why we thought: Why not make an application which will handle all the business and marketing chores at one place which will be beneficial for the growth of the company as well as the employees? As a result, we came up with the idea of the Daily Visit Report Application.

Chapter 4

Problem Definition and scope

4.1 Problem Statement:

Daily Visit Report Application

4.1.1 Goals and objectives:

Improve and grow your sales by optimizing your sales teams travels and activity on the field.

Record each sales visits, later review, follow up and convert them to leads & clients.

Employees can keep records of their expenses during working hours to the organization. They can attach the details of the expense. These expenses later can be approved/rejected by higher management.

Don’t Miss a Certain with a built-in todo in hand.

Get Motivated by watching how other colleagues are doing.

4.1.2 Statement of scope

Business andMarketing

4.2 Applications:

For marketing employees and companies so they can monitor day-to-day activities/visits of their sales team.

The tracker is location based which will use GPS.

Tracking becomes so easy for the employers so that they can focus on right things to help their business grow.

Expenses, billings and total work hours are also monitored by the DVR system.

4.3 Hardware Resources Required

|  |  |  |
| --- | --- | --- |
| Sr. No | Parameter | Minimum Requirement |
| 1 | Processor | Quad-core or above |
| 2 | RAM | 1 Gb or above |
| 3 | ROM | 4 Gb or above |
|  |  |  |

Table 4.1: Hardware Requirements

4.4 Software Resources Required

Tool : Android Studio Platform: Android

Operating System : Android 4.4 (API level 19) or above

Programming Languages: Java, XML

Database: Firebase

Chapter 5

Software requirement specification

5.1 Data Model and Description

5.1.1 Data Description

1. Firebase Data:

Firebase is Google’s mobile application development platform that helps us build, improve and grow our app.

Component of Firebase:

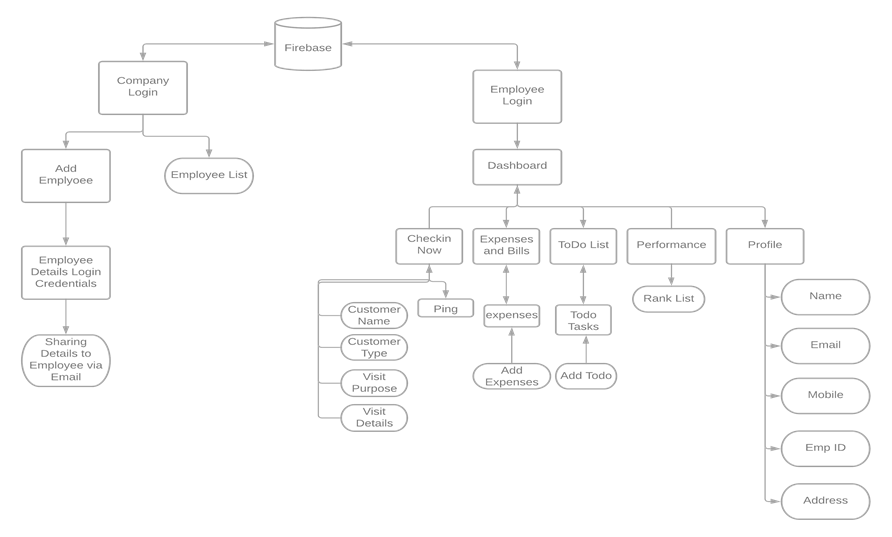
1. Realtime Database : The Firebase Realtime database lets us build collaborative applications by allowing secure access to the database directly from the client-side code.

2. Firebase Storage : Stores Images

3. Authentication: Authenticates the user with ID and password.

4. Record: Records are stored in the form of a tree.

5.2 Functional Model and Description



We Have used Firebase Database as Realtime Database. There is a company Register and Login, after logging in company can add Employee and make their login credentials. Company can see their employee.

In Employee Login Section, the employee can Add Checkin, Expenses and Bills, ToDo List, Performance, Profile.

In Checkin Now, Employee can add Customer Name, Customer Type, Visit Purpose, Visit Details.

Employee can add their bills in expenses and Bills.

Todo List Tasks are stored in Firebase database.

In Performance, checkins of all other employees of company are fetched from database

5.2.1 ER Diagram:

Company Login

Company email

Empolyee ID

Has

user

Employee list

Employee Name

userID

Serial No.

Add and View

V

employee

name

Address

email

Login

moble

password

Employee User

Expenses

profile

Checkin now

Performance

Chapter 6

Project Implementation

6.1 Introduction

Daily Visit Report is an android application used to record all the visit reports by the employee which will be reported directly to the unit head. This application is fully designed in Java and layout is designed using the XML. Firebase is used in the backend to store the information and data related to the application.

6.2 Tools and Technologies Used

6.2.1 Tool:

Tool : Android Studio Platform: Android

Operating System : Android 4.4 (API level 19) or above

Programming Languages: Java, XML

Database: Firebase

6.2.2 Technology:

**Java**:

Java is a **programming language** and a **platform**. Java is a high level, robust, object-oriented and secure programming language.

**XML:**

XML stands for Extensible Markup Language. XML is a markup language much like HTML used to describe data. In Android we use xml for designing our layouts because xml is lightweight language so it doesn't make our layout heavy.

Chapter 7

Software Testing

\*Below data is given for your reference\*

7.1 Type of Testing Used

7.1.1 **Local unit tests**:

Located at module-name/src/test/java/.

These are tests that run on your machine's local Java Virtual Machine (JVM). Use these tests to minimize execution time when your tests have no Android framework dependencies or when you can mock the Android framework dependencies.

At runtime, these tests are executed against a modified version of android.jar where all final modifiers have been stripped off. This lets you use popular mocking libraries, like Mockito.

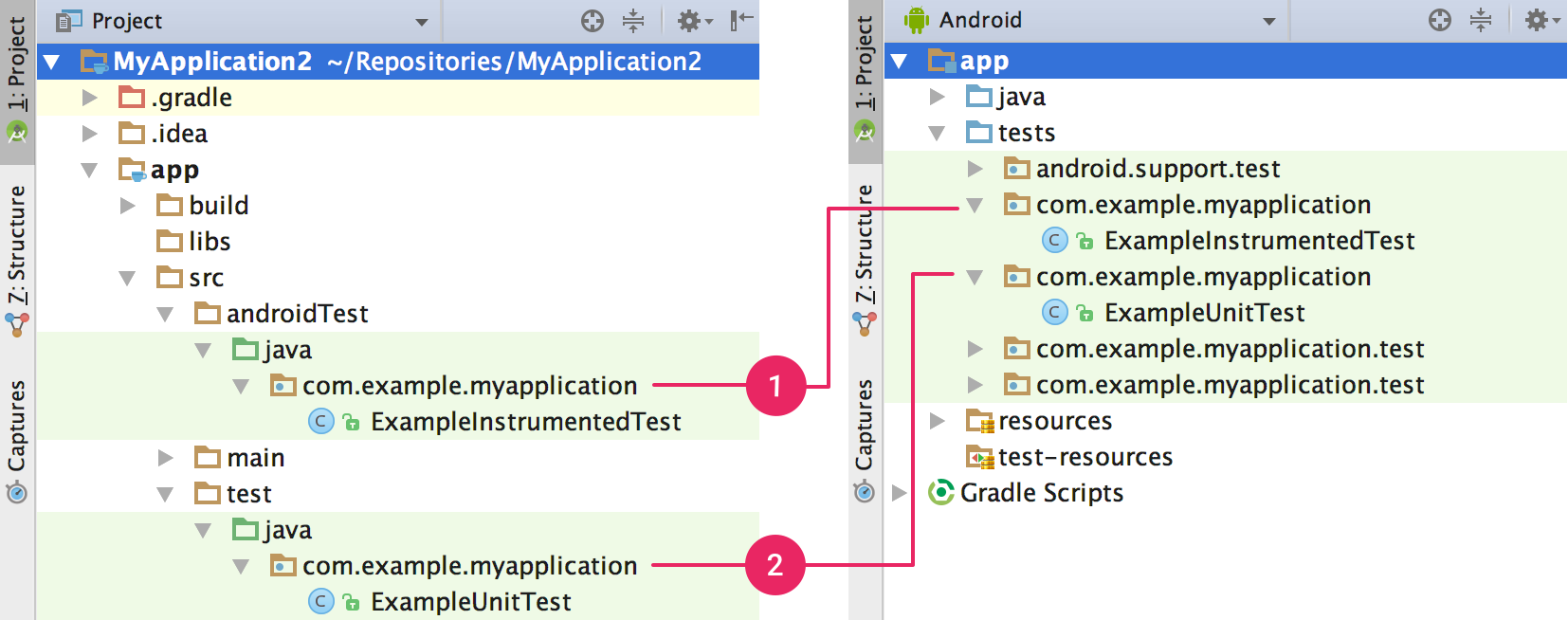


Figure 7.1: Local Unit Test

7.1.2 **Instrumented tests**:

Located at module-name/src/androidTest/java/.

These are tests that run on a hardware device or emulator. These tests have access to [Instrumentation](https://developer.android.com/reference/android/app/Instrumentation) APIs, give you access to information such as the [Context](https://developer.android.com/reference/android/content/Context) of the app you are testing, and let you control the app under test from your test code. Use these tests when writing integration and functional UI tests to automate user interaction, or when your tests have Android dependencies that mock objects cannot satisfy.

Because instrumented tests are built into an APK (separate from your app APK), they must have their own [AndroidManifest.xml](https://developer.android.com/guide/topics/manifest/manifest-intro) file. However, Gradle automatically generates this file during the build so it is not visible in your project source set. You can add your own manifest file if necessary, such as to specify a different value for `minSdkVersion` or register run listeners just for your tests. When building your app, Gradle merges multiple manifest files into one manifest.

27

Chapter 8

Results

\*Below data is given for your reference\*

8.1 Screen shots

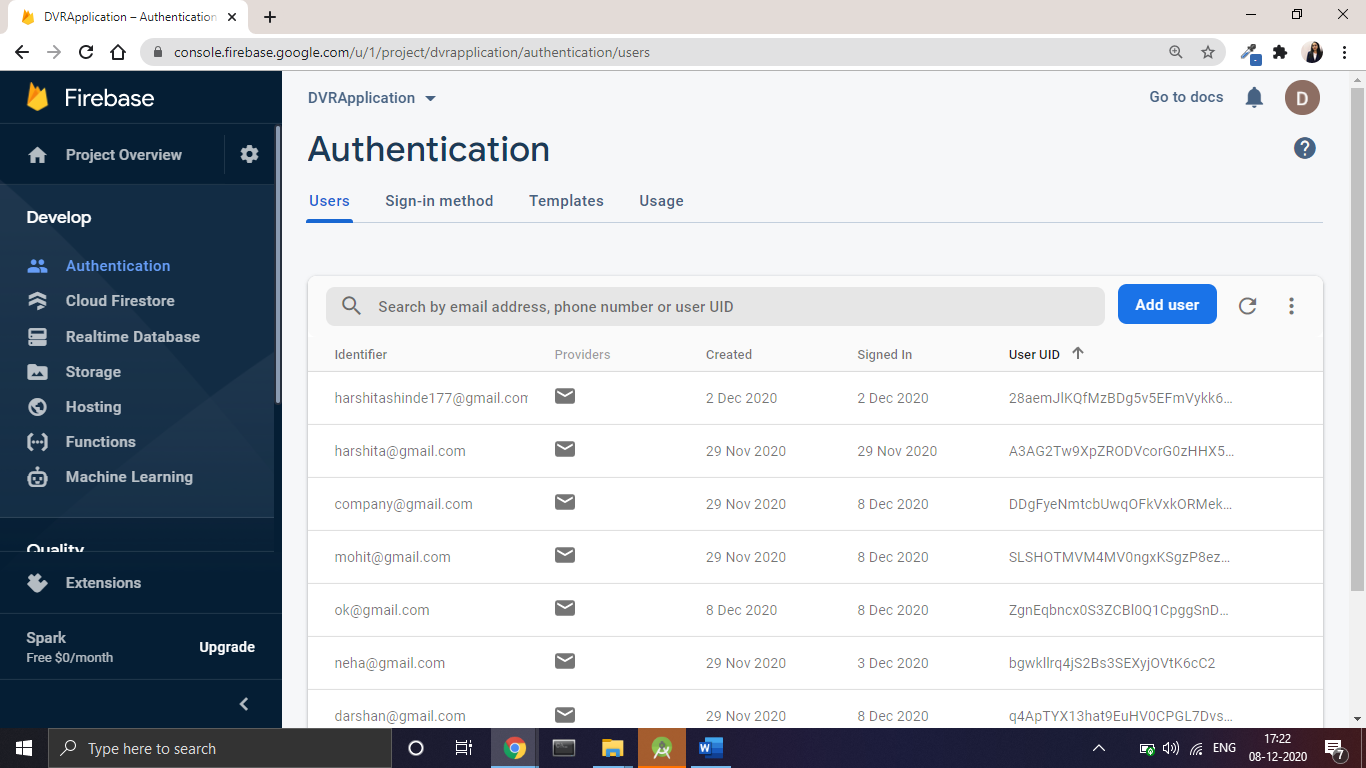


Figure 8.1: Firebase Authentication

28

8.2 Outputs

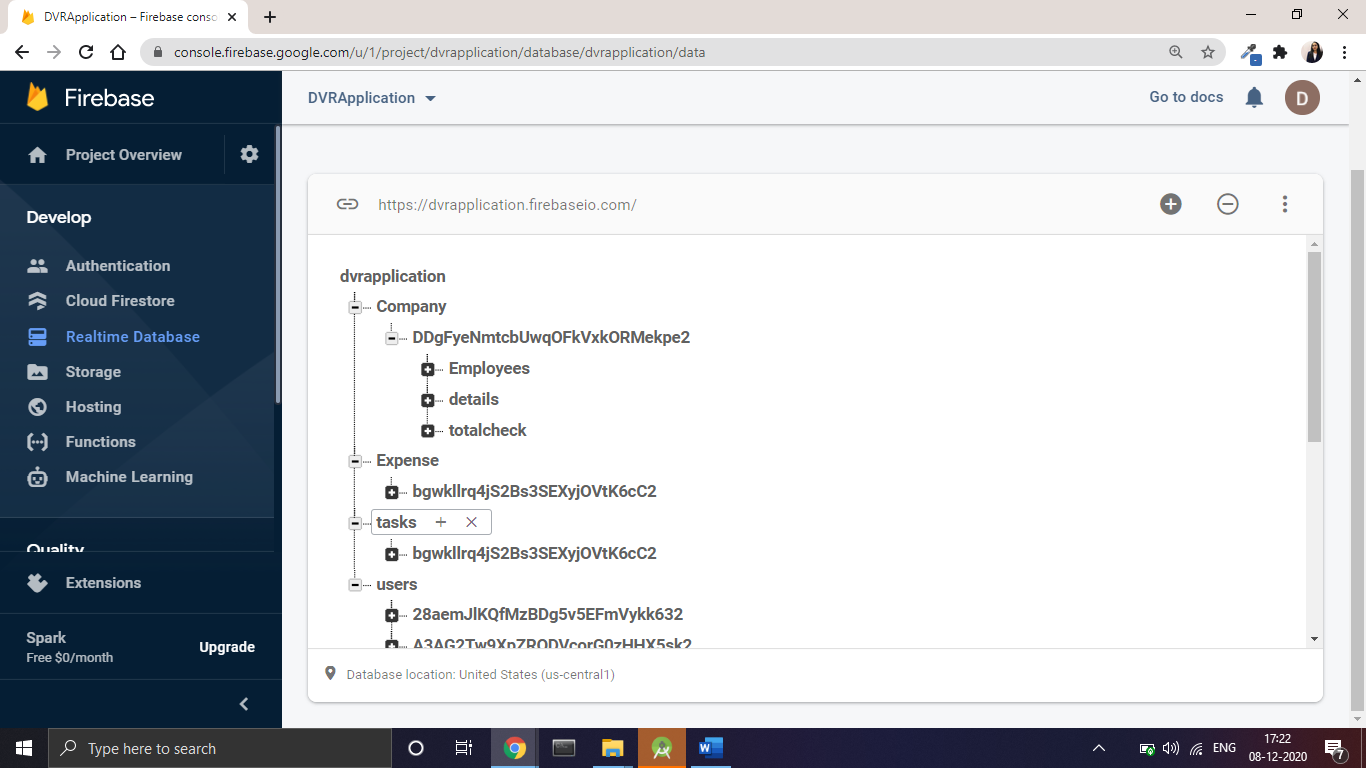
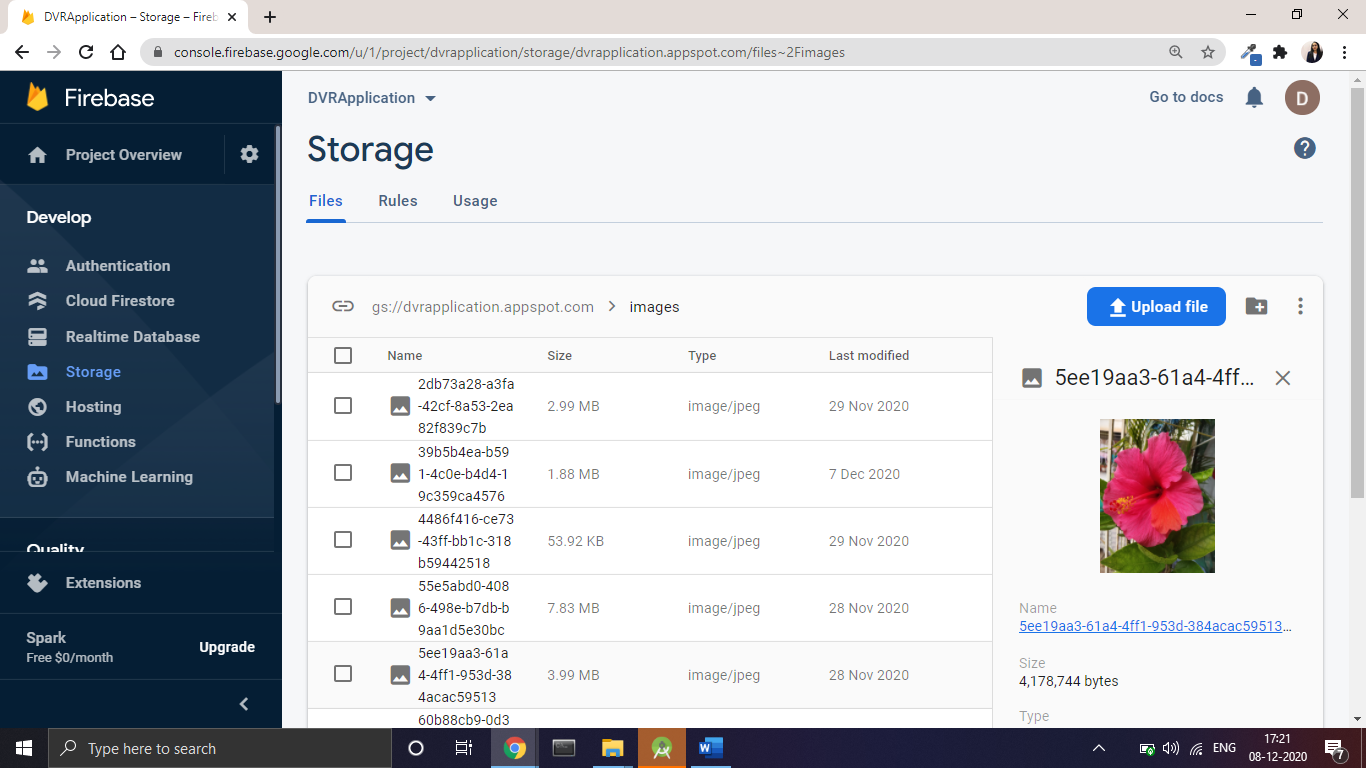


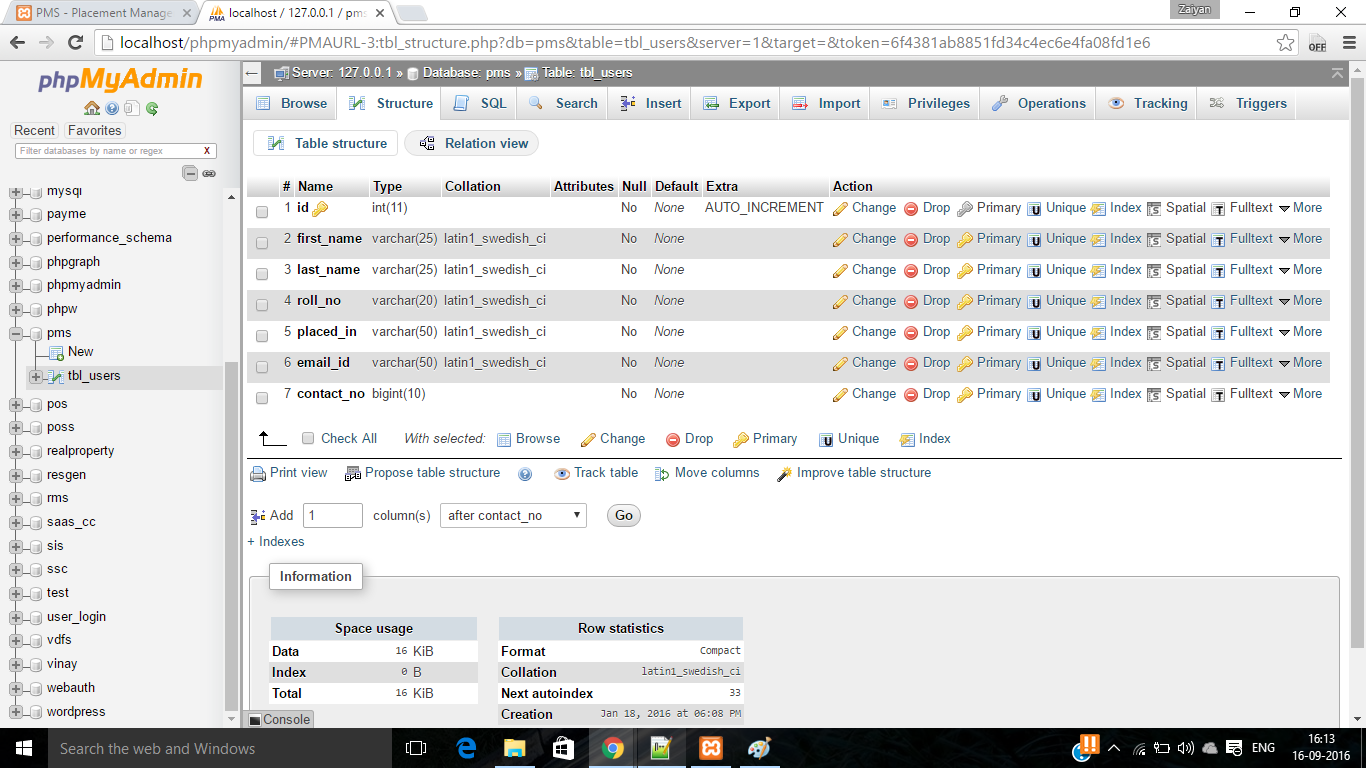
Figure 8.2. Realtime Database

29



8.3. Firebase Storage(for storing images)

30



31

Chapter 9

Conclusion and future scope

\*Below data is given for your reference\*

We can finally conclude that using this project we can have greater efficiency in providing the daily

visit reports of the employee and a great interface between the unit head and the

employees working on sites, thus becoming time saving and error free.

Future Scope:

The future scope of this project is including more funcitonalities such as :

1. Field Report
2. Customer requirement,
3. Extended Functionality.
4. Monthly Attendance.

Chapter 10

References

### • 1. Android: [Android Programming: The Big Nerd Ranch Guide](https://geni.us/d42E" \t "https://hackr.io/blog/_blank)

<https://www.javatpoint.com/android-tutorial>

https://developer.android.com/guide

• **2. Java:** Java: The Complete Reference by Herbert Schildt

<https://www.javatpoint.com/java-tutorial>

• **3. Firebase:** [https://firebase.google.com/docs/android/](https://firebase.google.com/docs/android/setup)