DATA ANALYTICS APPLICATION DEVELOPMENT

LOGASHWINN R 17E622

NAVEEN KUMAR S 17E629

report submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING

BRANCH: ELECTRICAL AND ELECTRONICS ENGINEERING (SANDWICH)

OF THE ANNA UNIVERSITY

NOVEMBER 2020

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

PSG COLLEGE OF TECHNOLOGY

(AUTONOMOUS INSTITUTION)

COIMBATORE – 641 004

CONTENTS

ACKNOWLEDGEMENT

At the outset, we express our gratitude to the almighty who has been with us during each and every step that we have taken towards this work.

We would like to express our sincere thanks to **Dr.Prakasan K**, Principal, PSG College of Technology, for his kind patronage.

Our sincere thanks to **Dr.Kanakaraj J**, Professor & Head In-charge, Department of Electrical and Electronics Engineering, for providing the opportunity and the necessary facilities for carrying out this report work.

Our sincere thanks to our Innovation Practices faculties **Mr.P.Komakhan Sudar Vendan,** Assistant professor, **Ms.B.Sathya,** Assistant Professor (Sr. Gr.), **Mr.A.Mahaboob Subahani,** Assistant Professor (Sr. Gr.), Department of Electrical and Electronics Engineering, for providing us valuable inputs and supporting us throughout the work.

We are grateful to our program coordinator **Dr.Soundarrajan A**, Professor, Department of Electrical and Electronics Engineering and our tutor **Mr.Komakhan sudar vendan P**, Assistant Professor for their constant encouragement.

Finally, we thank all the faculty members of Department of Electrical and Electronics Engineering for their support.

LIST OF FIGURES

1.INTRODUCTION

1.1 OVERVIEW

The focus of this project is fullfill the students knowledge about the companies and requirement of the companies at the time of placement. We have built a mobile application which is used to enrich their knowledge. It is done by building a app using the programming language called python in kivyMD framework.

1.2 PURPOSE

In this app first we are fetching the details such as cgpa, area of interest domain from the students. The another side it is used by the placement representative in real time. The placement representative also upload the details of company requirements such as students cgpa, area of interest and company's details such as salary, designation of job and important materials for the placement..

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

Normally at the time of placement, the placement representative share the companies info such as core of company, requirement of the company, timeline for the company to examine the students through aptitude test, GD and personal interview.

2.2 PROPOSED SOLUTION

By creating the mobile application the placement representative can update the information through this app so the students can view the company details according to their academic data .

3.FLOW OF DATA

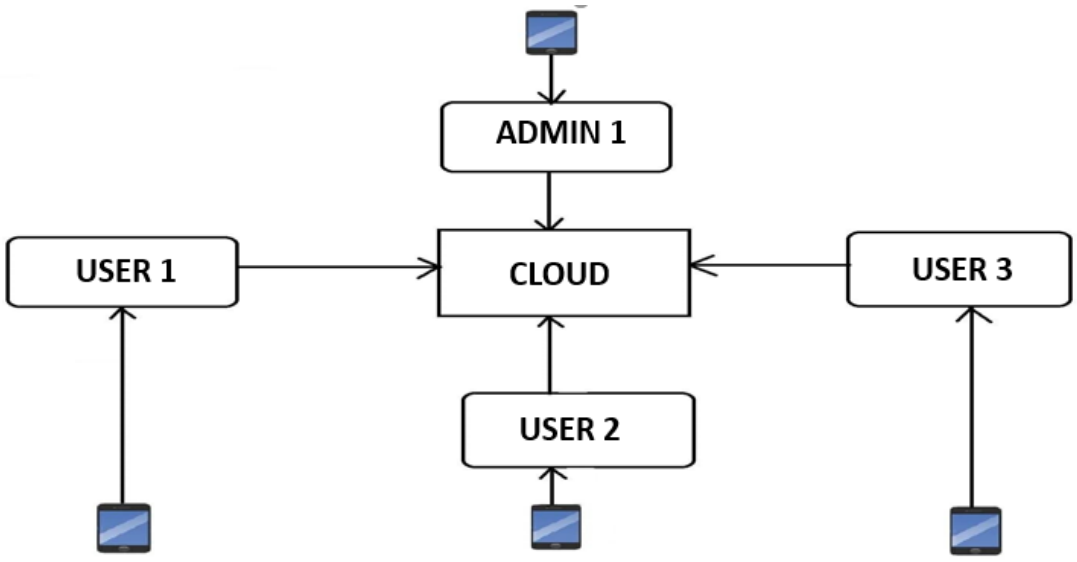


Figure 3.1. Block diagram

3.2 SOFTWARE DESIGNING

- Frame work designing .
- Data collection and Data base creation.
- Algorithm designing.
- Cloud computing.
- Application debugging and deployment.

4.PHASE-1

4.1 MODULES

There are two types of modules

- Admin module
- User module

4.1.1 ADMIN MODULE

First, we have to create an account for the admin and then admin want to login the app. After that admin want to upload details of company requirements such as students cgpa, area of interest and company's details such as salary, designation of job and important materials for the placement. According to the cgpa of students the company list will be listed to the users.

4.1.2 USER MODULE

In this also the user wants to create an account and want to login. after login the user want to enter the details such as cgpa and area of interest. Based on their academic data company details and list will be displayed.

4.1.3 BLOCK DIAGRAM

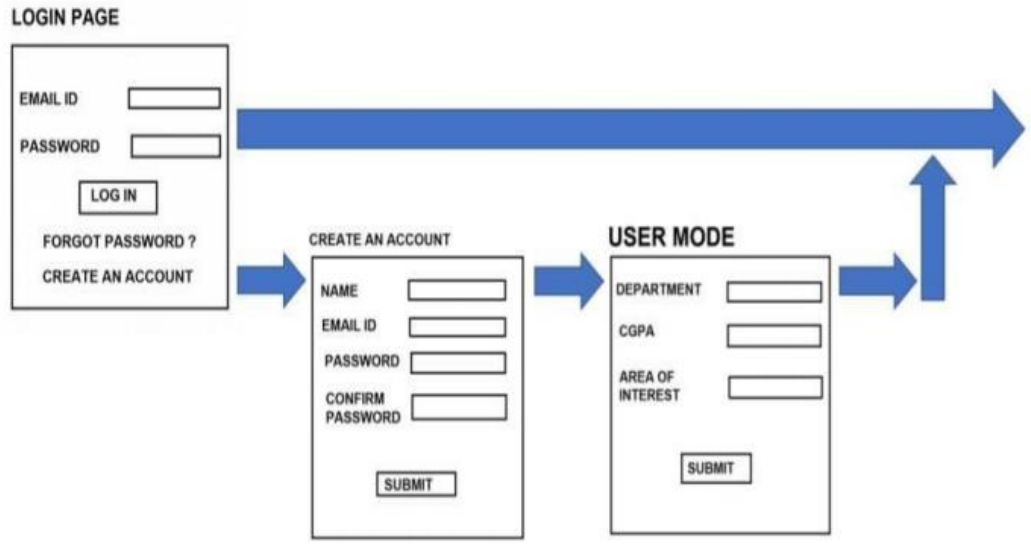


Fig 4.1.1 user side

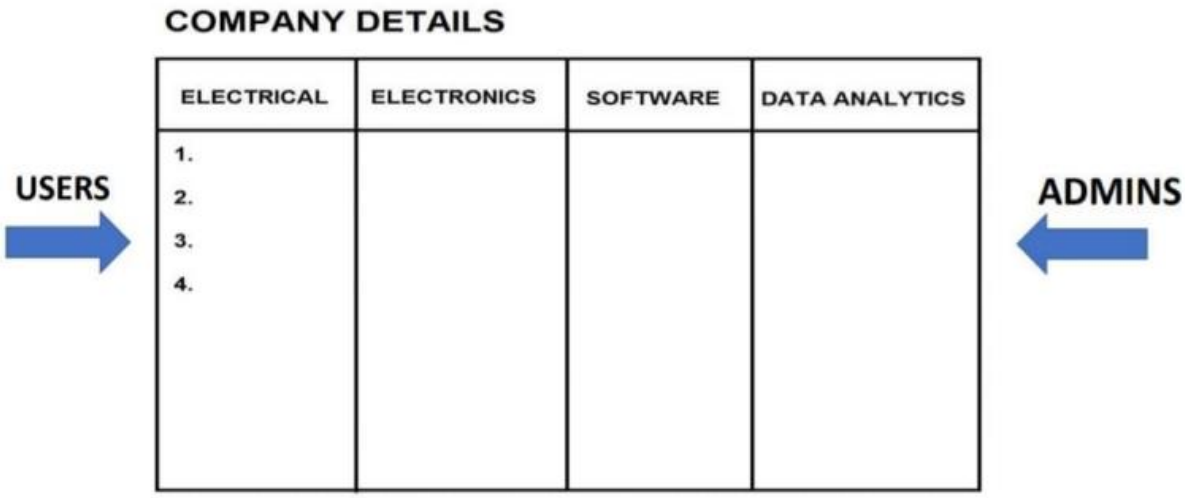


Fig 4.1.2 data table

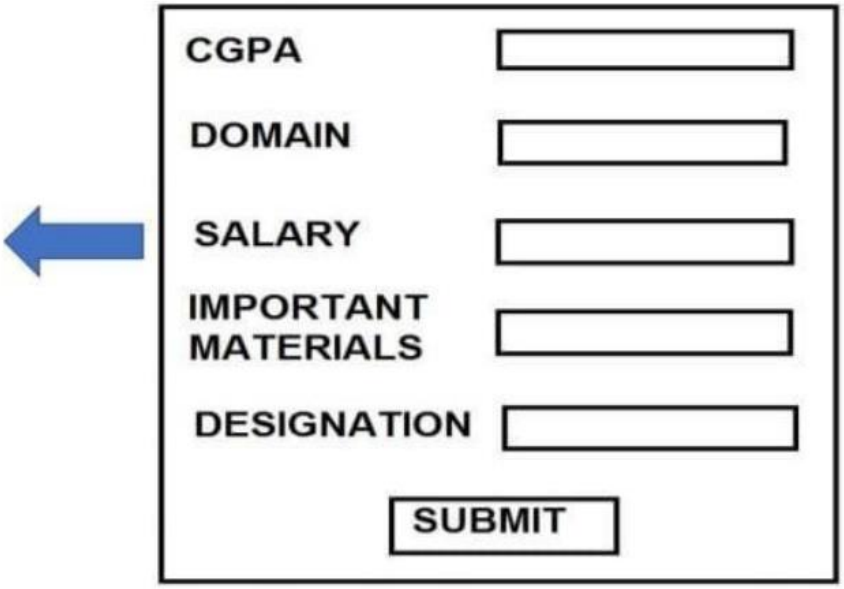


Fig 4.1.3 admin side

4.2 FRAME WORK BY KIVY:

Kivy is a free and open source Python framework for developing mobile apps and other multitouch application software with a natural user interface (NUI). It is distributed under the terms of the MIT License, and can run on Android, iOS, GNU/Linux, macOS, and Windows.

The framework contains all the elements for building an application such as:

- extensive input support for mouse, keyboard, TUIO, and OS-specific multitouch events,

- a graphic library using only OpenGL ES 2, and based on Vertex Buffer Object and shaders,
- a wide range of widgets that support multitouch,
- an intermediate language used to easily design custom widgets.

Example:

```
from kivy.app import App
from kivy.uix.button import Button

class TestApp(App):
    def build(self):
        return Button(text="Hello World")

TestApp().run()
```

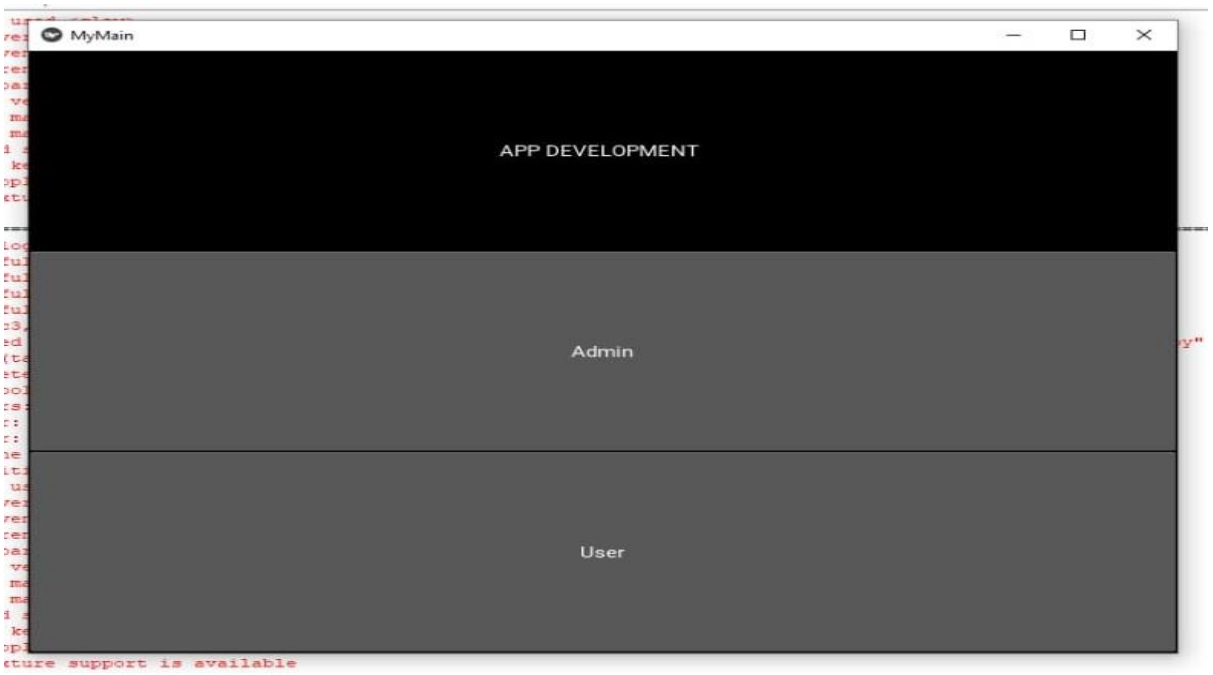


Fig 4.2.1 kivy1

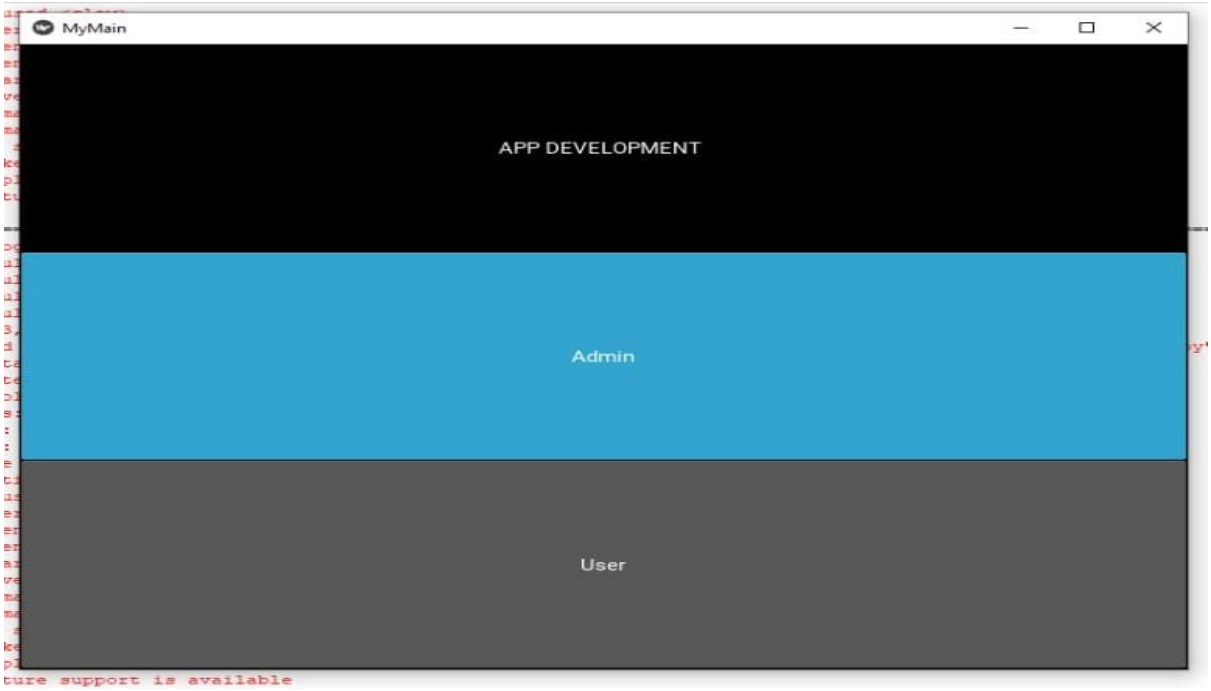


Fig 4.2.2 kivy 2

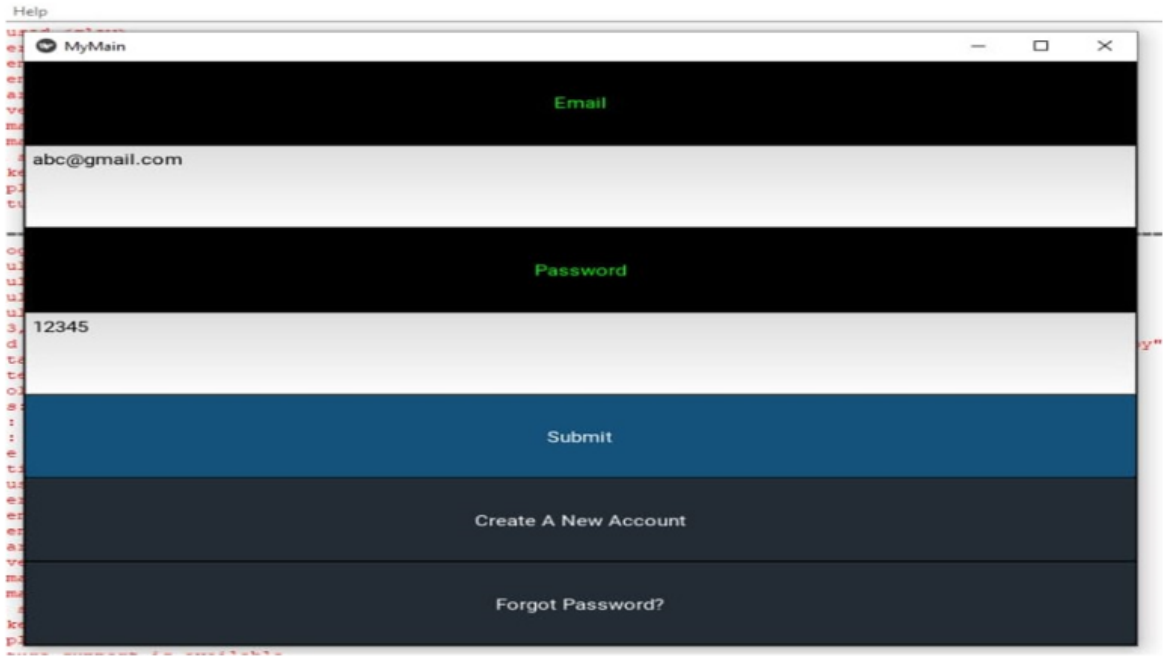


Fig 4.2.3 kivy 3

5.PHASE-2

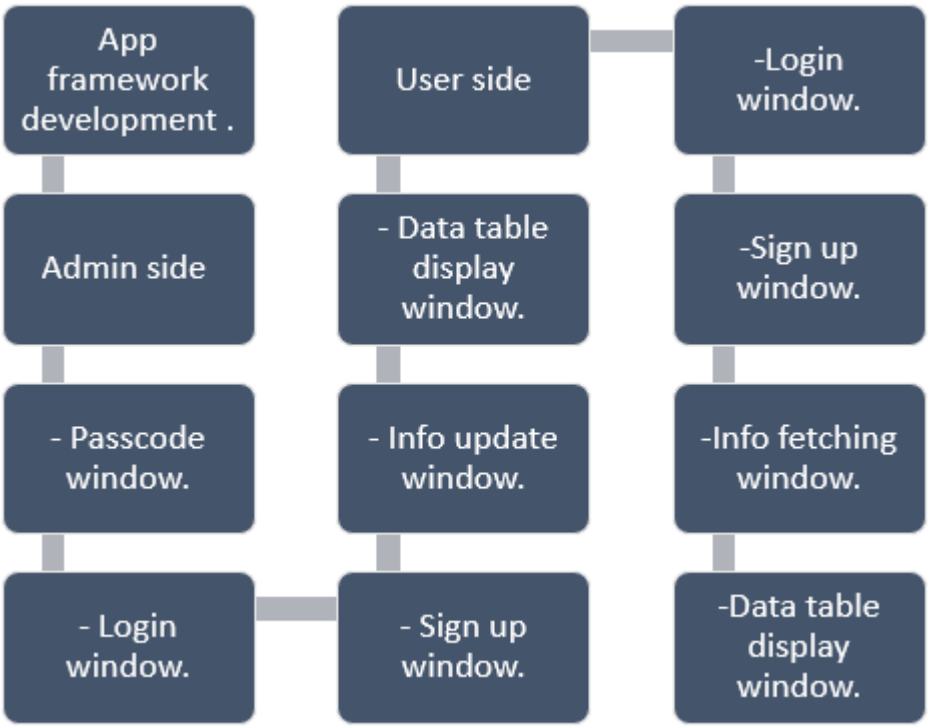
5.1 FRAME WORK BY KIVYMD

- KivyMD is a collection of Material Design compliant widgets for use with Kivy, a framework for cross-platform, touch-enabled graphical applications. The project's

goal is to approximate Google's Material Design spec as close as possible without sacrificing ease of use or application performance.

- Material Design uses more grid-based layouts, responsive animations and transitions, padding, and depth effects such as lighting and shadows.

5.2 PROGRESS OF THE PROJECT



5.2.1 APP FRAMEWORK DEVELOPMENT

We develop the simplified and easily usable frame work which provide the well defined details to the user. Initially by entering the app there will be two buttons **ADMIN** and **USER.** when we select the admin button it moves to the admin module and selecting the user button it moves to the user side.

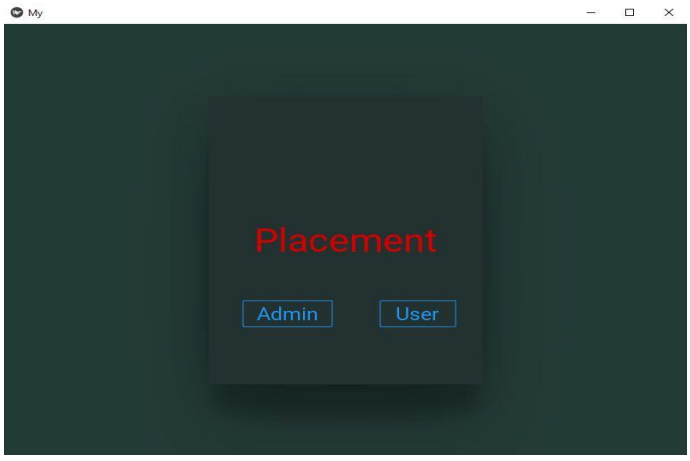


Fig 5.2.1 kivy MD 1

5.3 ADMIN SIDE

5.3.1 PASSCODE WINDOW

To prevent enter of user in this window and allow only the admin to enter in this window we have set the passcode to this window. So the admin knows the passcode and access this window. if the admin enters the correct passcode it allows to the next window or else it shows INVALID PASSCODE.

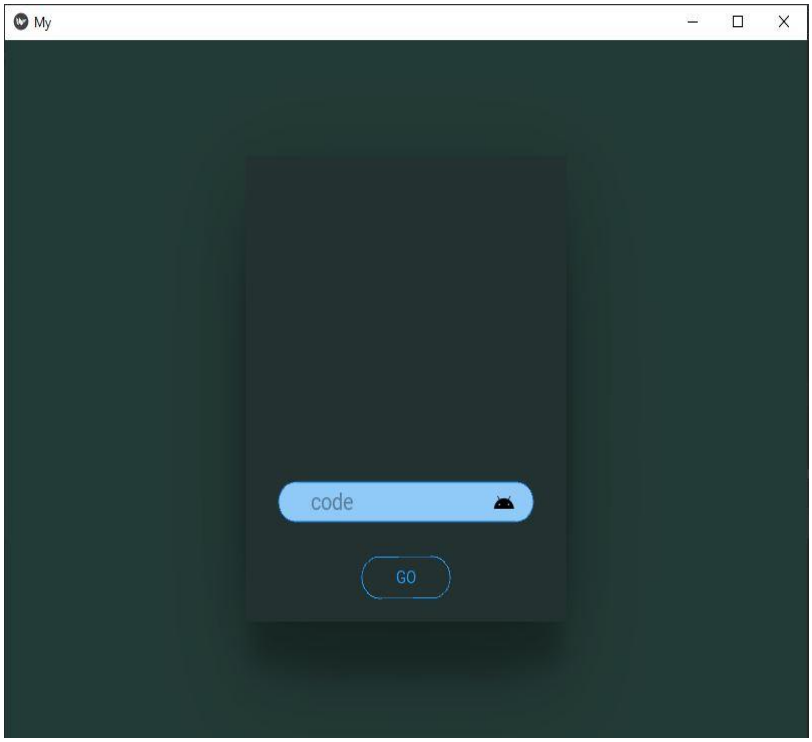


Fig 5.3.1 admin passcode

5.3.2 SIGN IN WINDOW

When the admin enter the email ID and password , if the email ID and password exist in the data base by clicking the login button then it moves to the data entry window but if the email ID and password not exist in the data base then it show NOT EXIST ,so we should create the new account by clicking create an account.

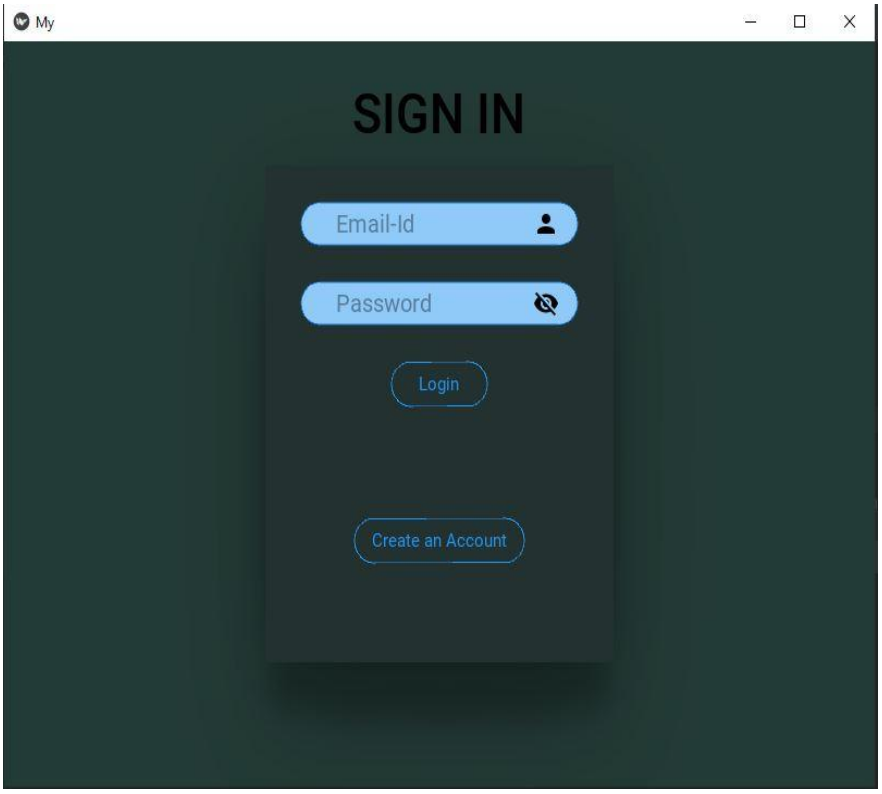


Fig 5.3.2 admin sign in

5.3.3 SIGN UP WINDOW

For create the new account the admin should enter the user name , email ID, password and the details will stored in the database, so admin can login.

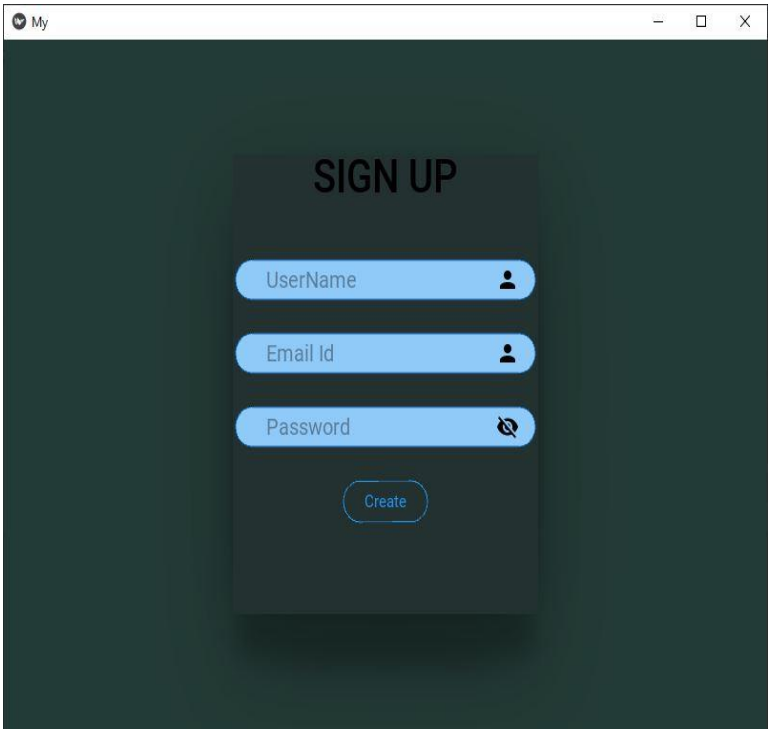


Fig 5.3.3 admin sign up

5.3.4 DATA ENTRY WINDOW

In this window the admin can enter the company details such as name , cgpa , salary, important study material that required by the students to reach the company which can be entered in the particular domain such as electrical, electronics, software and data analytics.

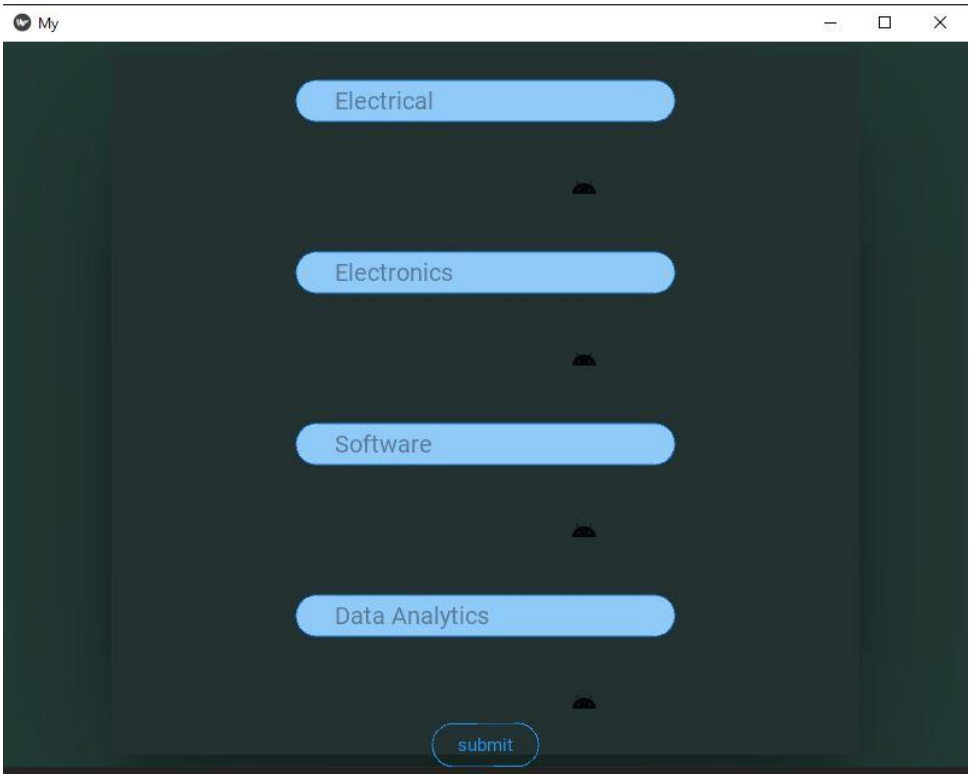


Fig 5.3.4 data entry window

5.4 USER SIDE

5.4.1 SIGN IN WINDOW

When the USER enter the email ID and password , if the email ID and password exist in the data base by clicking the login button then it moves to the data entry window but if the email ID and password not exist in the data base then it show NOT EXIST ,so we should create the new account by clicking create an account.

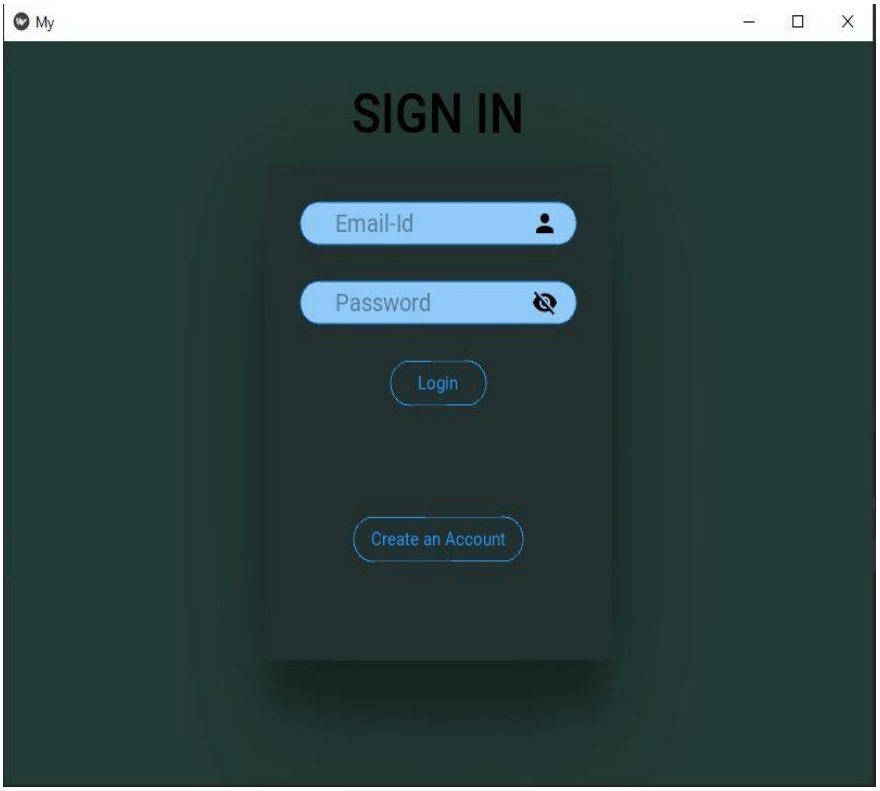


Fig 5.4.1 user sign in

5.4.2 SIGN UP WINDOW

For create the new account the user should enter the user name , email ID, password and the details will stored in the database, so user can login.

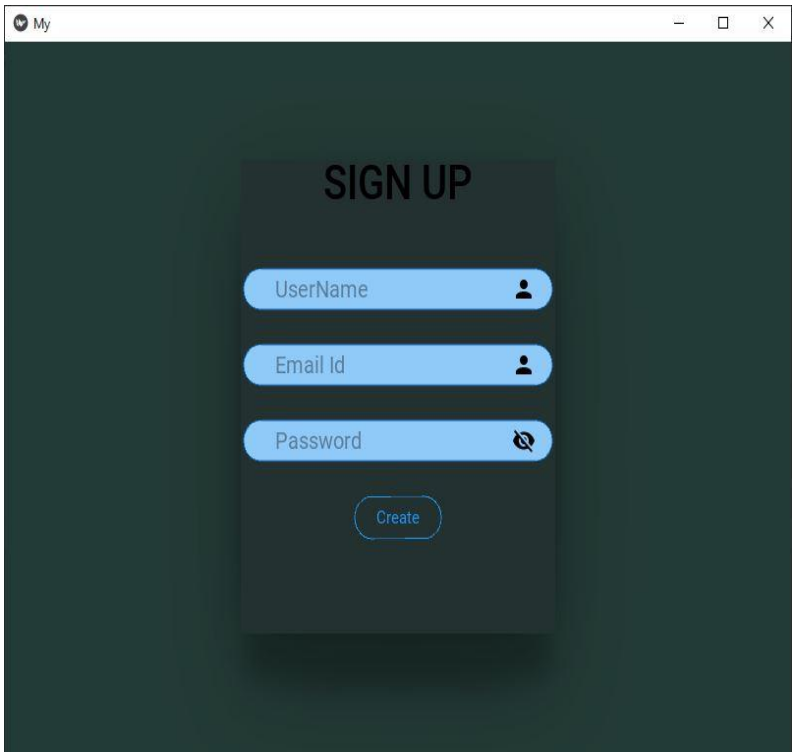


Fig 5.4.2 user sign up

5.4.3 CGPA UPLOAD WINDOW

User should enter their CGPA and click the upload button ,then it will move to the data display window.

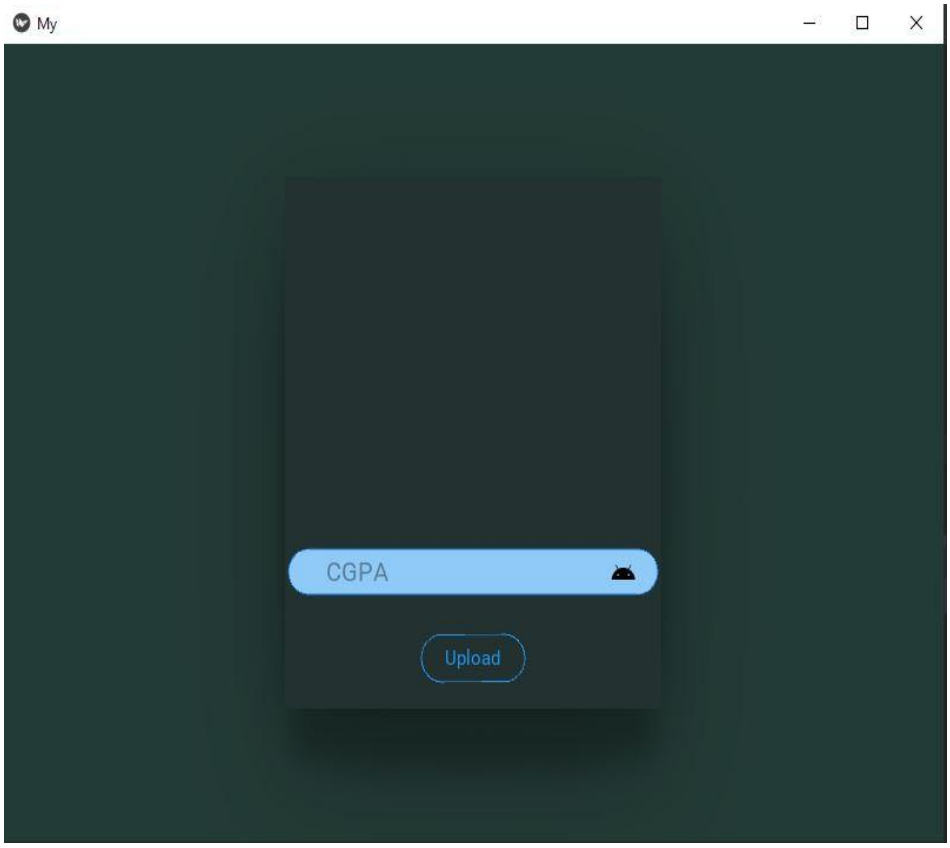


Fig 5.4.3 cgpa upload window

5.4.4 DATA DISPLAY WINDOW

In this data display window the details entered by the admin will be displayed on the data table according to their CGPA in classified manner based on electrical, electronics, software and data analytics.

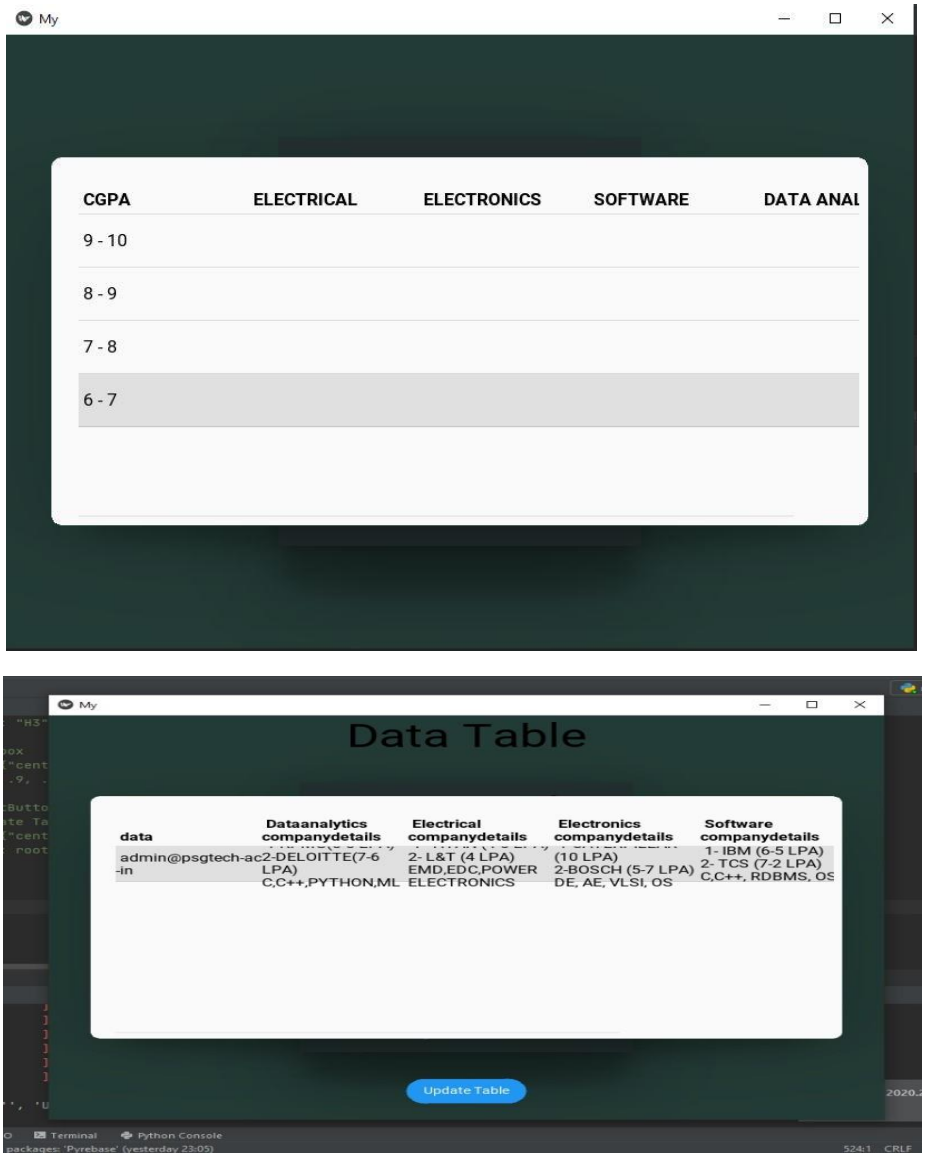


Fig 5.4.4 data table

6.PHASE -3

6.1 INTERFACING WITH CLOUD

The **Open Cloud Computing Interface** is a set of specifications delivered through the Open Grid Forum, for cloud computing service providers. OCCI has a set of implementations that act as proofs of concept. It builds upon World Wide Web fundamentals by using the Representational State Transfer approach for interacting with services.

6.1.1 FIREBASE REALTIME DATABASE

Firebase is a platform developed by Google for creating mobile and web applications. It provides two types of database to you.

- Firebase Realtime Database
- Firebase Cloud Firestore

Both of these databases are real time-savers.

6.1.2 USER MODULE DATA BASE

User side data base stores user's emailID, password, username and CGPA.

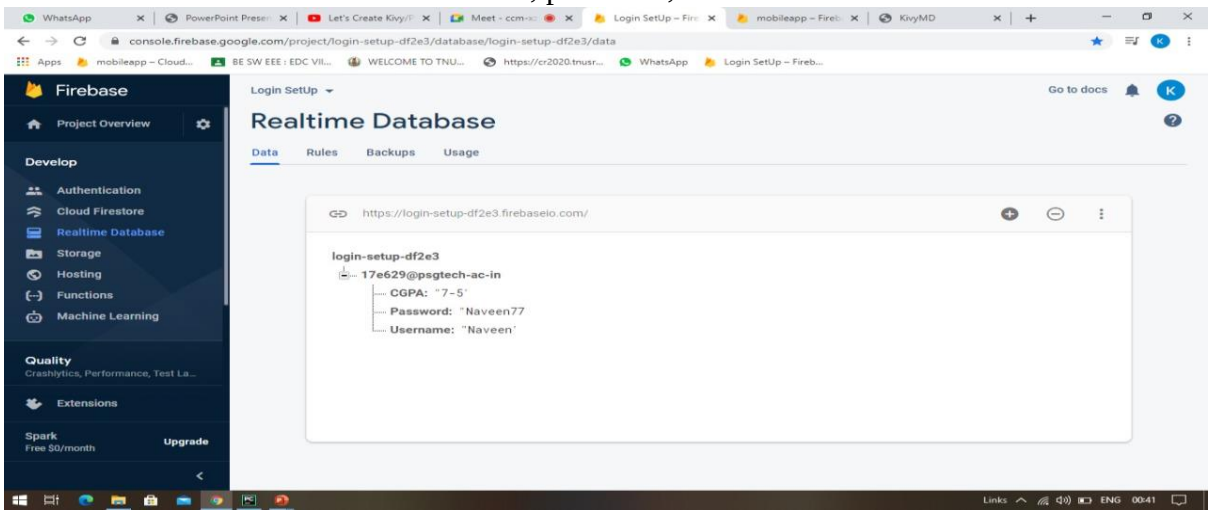


Fig 6.1.2 user database

6.1.3 ADMIN MODULE DATA BASE

Admin side data base stores admin's emailID, password, username and company details.

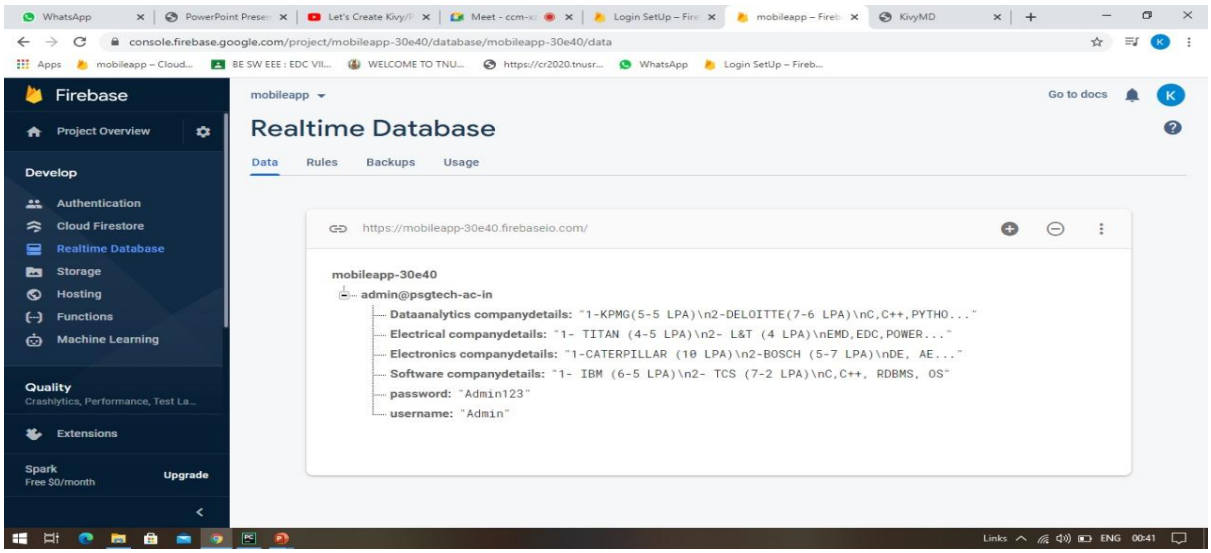


Fig 6.1.3 admin database

7.DEPLOYING AND DEBUGGING THE APK

Buildozer is a tool for creating application packages easily.

- Inside the py file of kivymd directory a buildozer packages are defined by using ubuntu.

- The goal is to have one "buildozer.spec" file in your app directory, describing your application requirements and settings such as title, icon, included modules etc.
- Buildozer will use that spec to create a package for Android, iOS, Windows, OSX and/or Linux.
- Buildozer currently supports packaging for Android via the python-for-android project, and for iOS via the kivy-ios project.

7.1: Packaging App for Android

One great way to do that is to turn your code into an application that can run on your Android phone. To accomplish this, first you'll need to install a package called buildozer with pip:

```
$ pip install buildozer
```

Then, create a new folder and navigate to it in your terminal. Once you're there, you'll need to run the following command:

```
$ buildozer init
```

This will create a buildozer.spec file that you'll use to configure your build. For this example, you can edit the first few lines of the spec file as follows:

```
[app]

# (str) Title of your application
title = Kvmyapp

# (str) Package name
package.name = kvmyapp

# (str) Package domain (needed for android/ios packaging)
package.domain = org.kvmyapp
```

At this point, you're almost ready to build your application, but first, you'll want to install the dependencies for buildozer. Once those are installed, copy your calculator application into your new folder and rename it to main.py. This is required by buildozer. If you don't have the file named correctly, then the build will fail.

```
$ buildozer -v android debug
```

The build step takes a long time! On my machine, it took 15 to 20 minutes. Depending on your hardware, it may take even longer, so feel free to grab a cup of coffee or go for a run while you wait. Buildozer will download whatever Android SDK pieces it needs during

the build process. If everything goes according to plan, then you'll have a file named something like kvcalc-0.1-debug.apk in your bin folder.

8. CONCLUSION

By creating this app both the admin and user gets the benefit. The admin upload the company details to the database and the user gets the details from the database stored by the admin in the form of datatable.

FILES LINK

- Github repository:
  https://github.com/naveensnk15/placementapp

9.REFERENCE

- https://kivymd.readthedocs.io/en/latest/components/datatables/index.html
- https://github.com/kivymd/KivyMD
- https://github.com/kivy/buildozer
- https://pypi.org/project/buildozer/
- https://ubuntu.com/community/VirtualMachines