RAJSHAHI UNIVERSITY OF ENGINEERING AND TECHNOLOGY

PROJECT REPORT

Course No. : CSE 2100

Course Title: Software Development Project I

SUBMITTED TO-

Dr. Md. Ali Hossain

Professor

Department of Computer Science and Engineering

Rajshahi University of Engineering & Technology

SUBMITTED BY-

Name: Mahbuba Sultana Prangan Roll: 1803168 Department: Computer Science & Engineering	Name: Mondol Mridul Provakar Roll: 1803062 Department: Computer Science and Engineering

i

Acknowledgement

Firstly, we would like to express our gratitude to Almighty. Without Him, it wouldn't be completed.

Then we would like to thank our beloved and respected course supervisor Md. Ali Hossein for his guidance. His supervision and consistent guidance helped us learn a lot and our work became easier.

Finally, we would like to thank each other for the idea-generation, hard work and for holding on till the last.

ii

Abstract

Heaven's light is our Guide

In the age of digitalization, every field is trying to make everything easier and reduce the time spent in unnecessary things using the most of technological advancement. Education system is nothing different. Everything's becoming more and more updated. So A student should be smart enough to cope up with these, so that he/she doesn't have to waste time and energy.

Our android application 'smart Student Guide' is a small contribution to this. It helps a student get most of the tools he/she needs regarding learning in one. It saves his/her time and energy.

For now, it only contains RUET's information and materials, but it can be upgraded and expanded later.

It's very easy to use, cheap to maintain and efficient to invest.

As requirements to perform, it only needs to have internet connection and memory space in cellphone.

Keywords: Education, Student, smart, efficient, help

Table of Contents

List of Figures

Chapter 1

Introduction

This is the age of modernization. Everything is getting faster and easier.

Education system is not as simple as before either. As the world-population is growing, the number of students is also increasing, so maintaining them is becoming harder and complicated. So we hardly see any institution without technological advanced tools or methods.

So a student should learn to utilize his/her time and energy properly too. Our application can be of great use to students regarding this. It includes most information of RUET, PDF files of necessary books, video recordings and slides of classes in drive, important notices which is controlled by the admins who will have the access to change the information.

It's efficient in many ways like, there might be a scenario where a student can't carry his/her books or just forgot to bring. He/she can just bring out his/her cellphone and go through the application as everyone generally carry phones everywhere.

We used firebase for the database. As it's cloud storage so it's huge and anyone from anywhere can connect their devices and operate using this application.

We hope it to be helpful for students at its most. It can be enhanced through using various facilities like light or dark mode, youtube video links, VR etc.

Chapter 2

Description

Part 1-Short Description

It is mainly divided into two parts while making-

i. Admin app
ii. User app

While operating, it'll be a single application which will act different upon the accounts (by emails and passwords) recognizing as admins or just users.

Simply, it'll give the admin access to write or edit the app information and won't give the user though it'll look the same. An admin will be able to see the student view as well.

There will be firstly the login form, uploading and editing page or admin dashboard, one of the options in dashboard will be student view which will include the main useful things students can use.

Part 2-Detail

The application is mainly divided into two parts-

a. Login Form
b. Admin Dashboard

   Student-view is an option from the admin dashboard, only which can be accessed by the not-admin users. The rest options are only accessed by the admin(s); an admin can also see the student-view.

Figure 1: Login Form

1. **Login Form:** The First step here is the login form, here this page would be shown. This is not actually the part of the application. It's a separate essential part that almost every application needs. Here, if a person is already registered, He/she just have to enter his/her email and password with which he/she created the account, the press login. If one didn't register then after pressing 'Register', one have to give an email and a strong 6 character password to create an account.

If the password is weak, it would show and suggest to give a strong one.

Then repeat the first step mentioned.

If one forgets one's password, just by providing the email address, the option to reset one's password would be available to one's email account as a mail.

It's to prevent anybody's accesss through the application randomly. Here, the user info will be saved in a database.

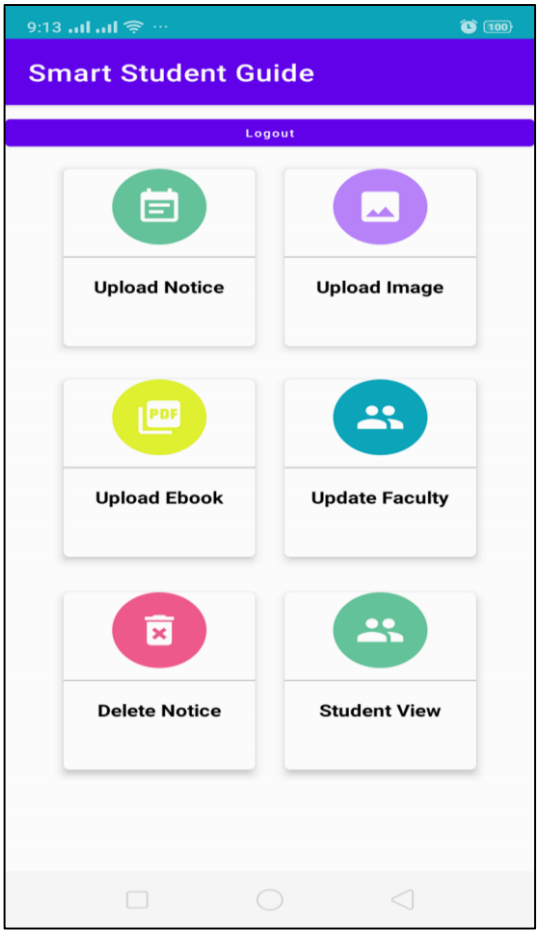

Figure 2: Admin Dashboard

2. **Admin dashboard:** This is the first page from the application. All those options here are accessible for admin account but only 'Student View' can be accessible through the student account.

But a student account holder won't be able to upload or delete anything. All those information will be stored in the cloud database.

Creating the dashboard is the first job in the main project after the login form creation. After creating the project determining the language-JAVA, SDK-API 21: Android 5.0(Lollipop), the main work started. Some dependancies(such as-material dependancies) were added in app-levelgradle, colors to use for the options-circles were chosen and added in

colors.xml. In drawables, six drawable resourse files were created with shape-oval, solid and each one with different colors. For each option's icons, in vector asset from drawables, the built-in clip arts were used as icons and six more drawable resourse files were created. The color of the icons were given white. In activity_main.xml, the layout of the dashboard is designed using those previously chosen colors and shapes' drawables. Material cardview was used with properties to show the items. In imageview, the views of the options with background colors, height, weight, gravity, icons-previously-added and padding,

were created for each of the icons. After using margintop, the textview was created to write the names of the options-with layout width, layout height, textstyle. Same things were copied six times. The whole page designing was finished after this.

Five of them are accessible by only the admin and the rest one can be used by the users too. Now to go through the detail of all the options here one by one The first one is 'Upload Notice' which is for only admin's use. When we click here-

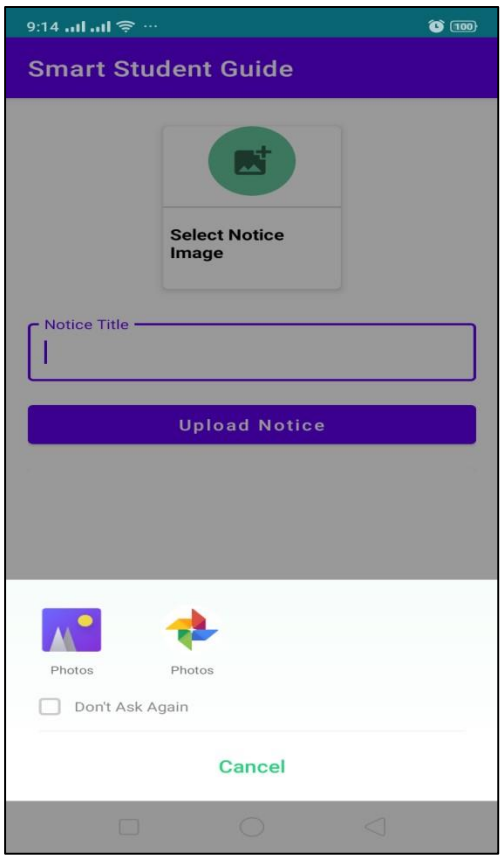

Figure 3: Notice Upload

This would be shown. From google photos or gallery, the admin will be able to upload notice images. From the 'Notice Title' option, he/she will be able to create some title for the notice.

To open a new activity from 'upload notice' option, another empty activity was chosen named 'UploadNotice', scrollview was used to take linearLayout. The layout was designed this way- Layout gravity is center, icon from vector asset and color is customed again for imageview just like before. For the input-taking part, textInputLayout is chosen from 'design' and dropped under the linear layout made before. The input layout is then taken below the cardview. Style was 'outlined', id was 'notice title'. A materialbutton is used with text, layout width, layout height, margintop, textAllCaps, id with 'upload notice' in the button. Then it was connected to firebase database. Default account was selected. Then, For getting the upload_notice activity on with clicking in 'upload notice' option in dashboard, firstly, in mainactivity, onclick method is implemented. Then a cardview variable is created and properties passed. In upload_notice, after

necessary steps, 'openGallery' method is created and called for passing the image from gallery, EXTERNAL_CONTENT_URI is used. In startActivityForResult, 'pickImage' intent is passed. After clicking this, gallery will be opened. The image is picked from gallery with the condition and passed.

Image-upload, Ebook upload option figure is not shown here.

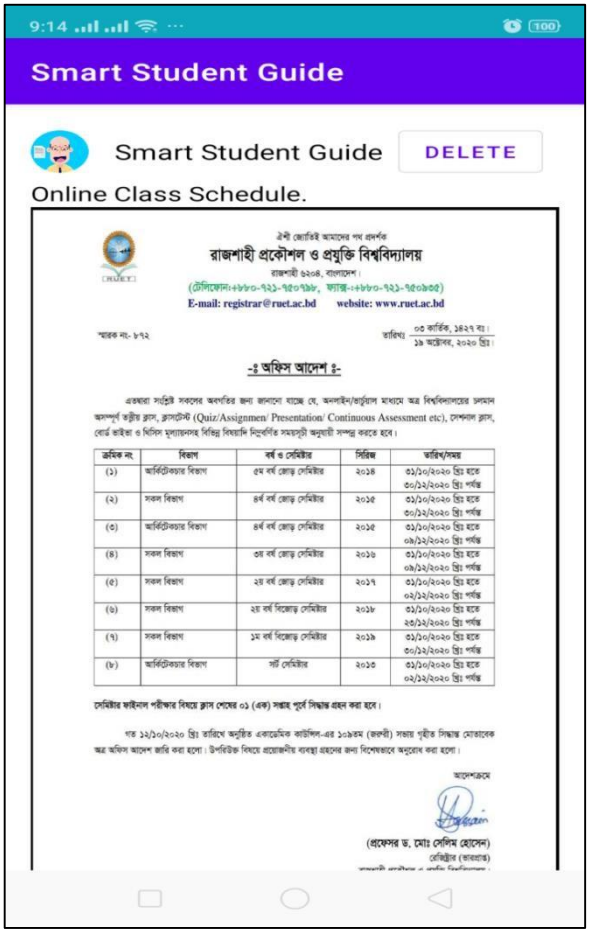

Figure 4: Delete Notice

Delete Notice option is for deleting the notices uploaded before. Only admin can do this. Updating the other options are separated. Delete option is separately created.

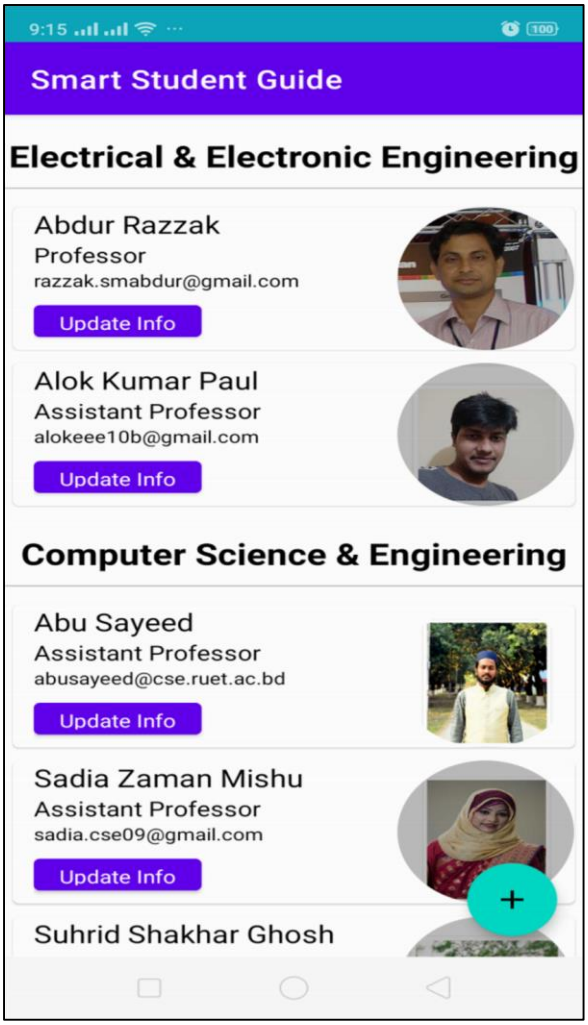

Figure 5: Update Info

By clicking 'Update Info' option, the admin account holder can update or change the information provided before. He/ she can also upload new teacher's information by clicking the plus sign.

Here, the information is separated by means of departments. For now, here is few teachers' information which could be expanded later. The information part contains the teachers' name, designation, email address and a photo.

Inside java, a new package was created for teachers. Inside this, an empty activity called 'update faculty', another called 'add teachers'. In main activity,

intent passed for faculty opening the layout. The activity 'update faculty' design started. Coordinatorlayout was used to add the floatingactionbutton. Only add teacher operation is designed here, how it is showed after adding teachers or updating teachers will be designed in the user application part.

Floatingactionbutton was created with custom width, height, gravity, margin and style. The icon is set from vector asset. An elevation is given. Fabsize is normal. In updatefaculty.java file, a variable for floatingactionbutton is created and found with necessary properties. Add teacher's xml file layout is designed with linearlayout –custom height, weight, gravity. Some textInputLayouts and button are created respectively for taking input(name, information) and confirmation following the same process as before. For selecting the category, spinner is used. In java file, all those were found. Spinner was also used for uploadImage part. The same process was repeated. Then the teachers' data were uploaded in firebase.

These are the parts admin has/have control on. There are some static information as well.

Student-view: The user ID will view the page and get advantage from it.

There are those information described before, which can be updated and expanded by the admin account holder. Accept the works done by admin account holder, there are few options we made for the user, which aren't directly controlled by the admin.

Almost all of them will be directly connected to firebase database, which would be changed with the change in the main source. This part is a separate part but connected to the admin part by the dashboard uploading options.

> There's no option to change information of student-view unless the admin does is from the admin application.

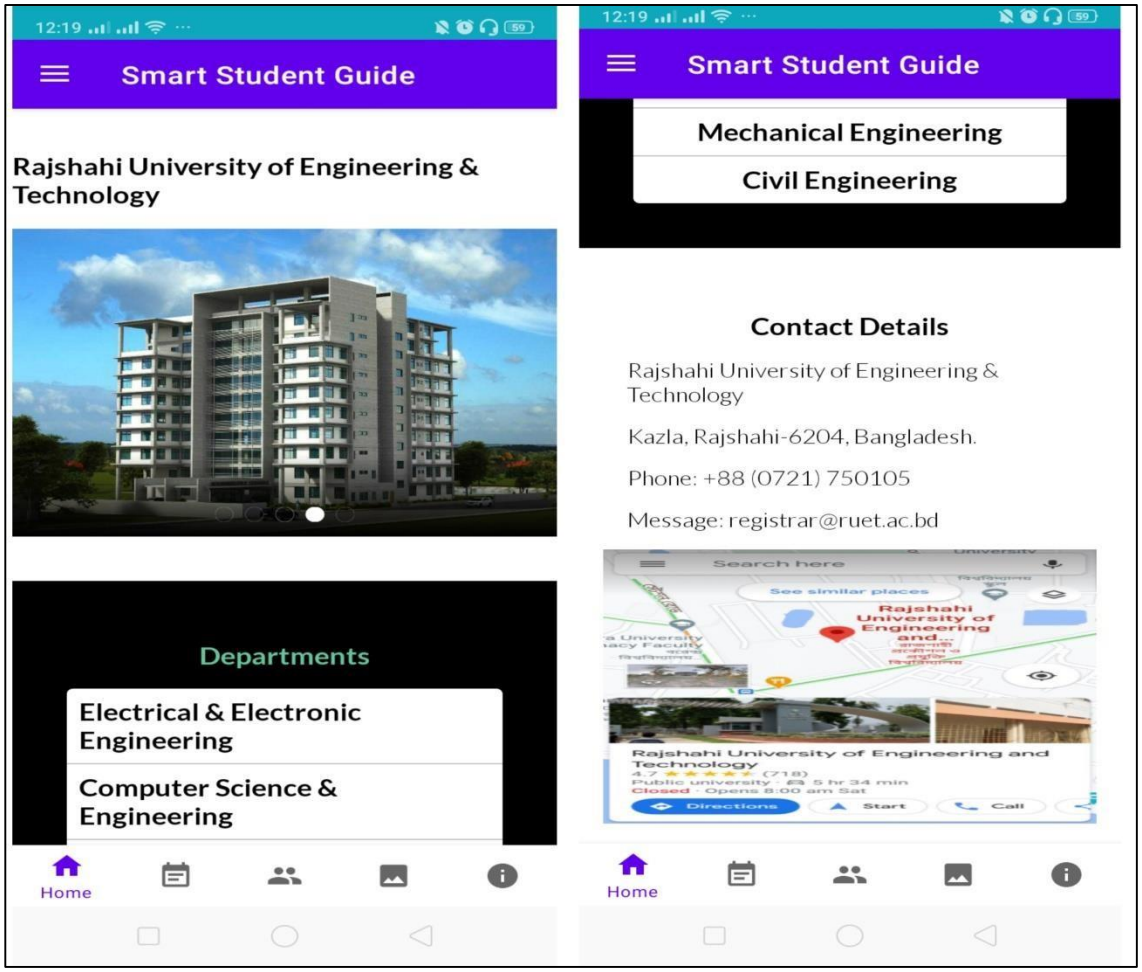

Figure 6: Student-view Homepage

3. Homepage

This is the page the user ID holder will see at the very beginning.

The left part is shown first and as the user scrolls down, he/she will see the second part as well. In the first part, here is a changing picture of RUET's buildings, it shows different pictures with the moving of the white dot from first blank to last blank space, which is called Auto Image Slider.

After adding dependencies in app level and project level gradle and syncing, in homefragment's xml file, the layout is designed. In sliderlayout, the slider is added to the id. Normal simple transformation is added here as

SliderTransformAtion. There are five image changes, so a loop is to rotate five times, the image URL is set which is uploaded is firebase storage. The image link is passed from the main app connected to the admin app. Imagescale is centercrop. Glide dependencies are added. After syncing internet permission was got.

Below is the 'Departments' option.

At the downmost part of the page, there is the bottom nevigation bar* which shows different activities after clicking the options. Home, Notice, Faculty, gallery and about are the five options the navigation bar shows, no custom icon is used, all are built in icons.

Adding dependencies in the gradle (material's one and navigation's two) is the first step. The five fragments were kept in the packages after creating for the five options. After creating a directory with navigation resourse type, the navigations are designed. For this, firstly, some fragments are created for the options, which name will be displayed after clicking the option is decided.

Home fragment is created and rest are copy-pasted and edited. These five are the five screens to be displayed. Resourse file is created and icons added for different options, framelayout is created in main activity. Labels are made selected, when an option is selected, its name is shown. With clicking in every option, the background is changed so that the blank fragment can be differentiated. Then every fragments are designed one by one.

For now, there are four different departments in the application.With scrolling down, there are some contact details available. The registrar's email address is also provided. At the downmost part over the bottom-nevigations, there is a map sign which points to the google map. Google map is a very important part for user whoever wants to know RUET better. With a massive squad of vehicles dispersed around the planet, google repeatedly drive around every accessible road they can find, all the while taking 360-degree photos

everywhere they go. Based on the GPS coordinates of those vehicles, google overlays its Street View images on top of its basemap.
The URL of google map was passed and it shows the location of RUET comparing to the surroundings. If the user turns on his/her device location, he/she can also get the distance from his/her current location to RUET and maybe the time required to go there.

In the home fragment, in fragmenthome.xml, the layout was designed is split mode. Another xml file of type values was created to store all the sizes(title size, text size , margin etc). If some size needs to be changes, it can be modified just creating values and changing here would be enough. In the imageview, a single picture of RUET's location in google map was added as the button which takes to the map with clicking it. Just google map was redirected because every phone contains google application. The properties were customed.

Google map can be a good option for the new students.

*Bottom Navigation bars display three to five destinations at the bottom of a screen. Each destination is represented by an icon and an optional text label. When a bottom navigation icon is tapped, the user is taken to the top-level navigation destination associated with that icon.

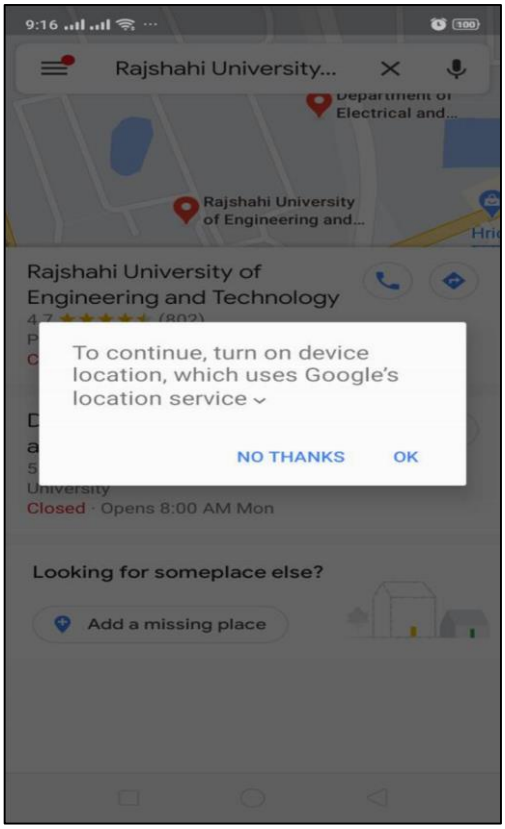

Figure 7: Google Map

In the homepage, below there is the navigation bar which contains the different screens containing the different parts.

Navigation Bar

1. Notice

The notice upload option is provided in the admin dashboard which is only controlled by the admin. Those uploaded notices are manifested for the users. The users with student accounts will only be able to see the notices uploaded by the admin.

The process is almost same as the image-uploading process. The notices will be organized as 'newest first' order. A caption can also be given and the date-time will be there.

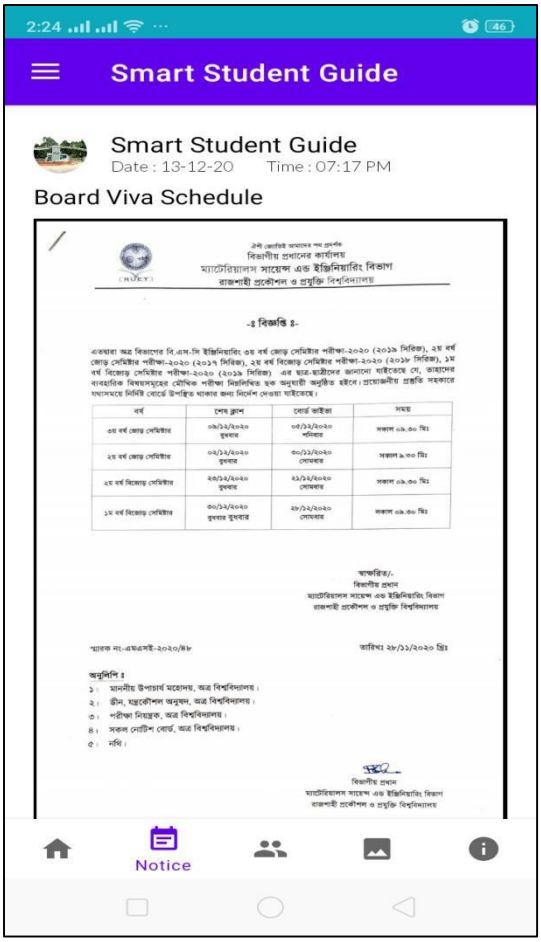

Figure 8: Notice Board

2. Faculty:

The faculty option is for the information of the faculties. This is department-wise organized. For now, there are 4 departments' information- CSE, EEE, ME and CE which are alphabetically organized. For now, there are only the teachers' information.

The name, designation, email and a copy of theirs picture is provided here.

The information are not changeable by the user, just the admin can update infos. It's comparedly smaller but can be expanded later. The figure is not given because it's the same without the 'update info' option which is a part of 'update faculty' option in the admin dashboard.

3. Gallery: The gallery option gives the images uploaded by the admin. It could be added with the notice board but it is more unofficial which may include the suggestions from teachers, class test marks, important topic lists etc. It's also alphabetically ordered.

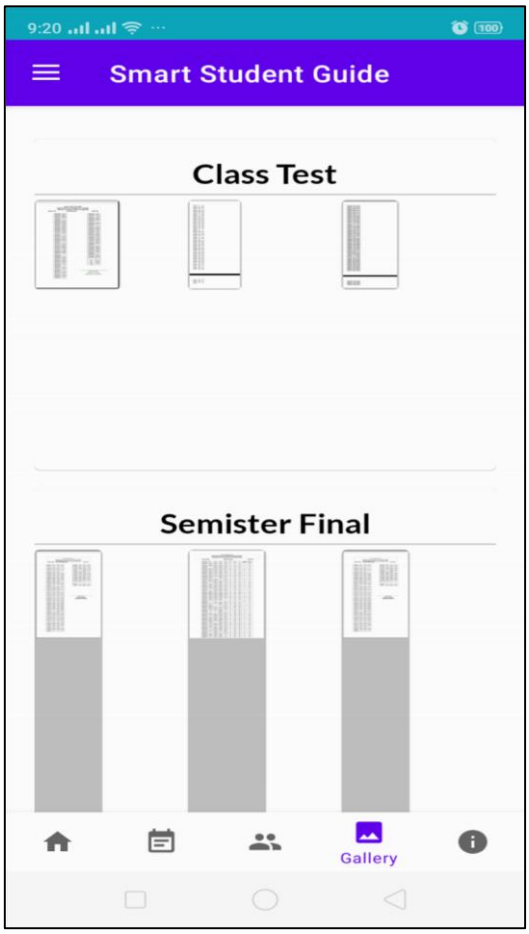

Figure 9: Gallery

Students can check necessary information from here.

4. About:

The next option in bottom navigation bar contains information about RUET for now. A picture of the university and in the information part there are history, vision, Mission, campus map, location, weather,

accommodation, food, market and hospital. It can be updated by the admin account holder.

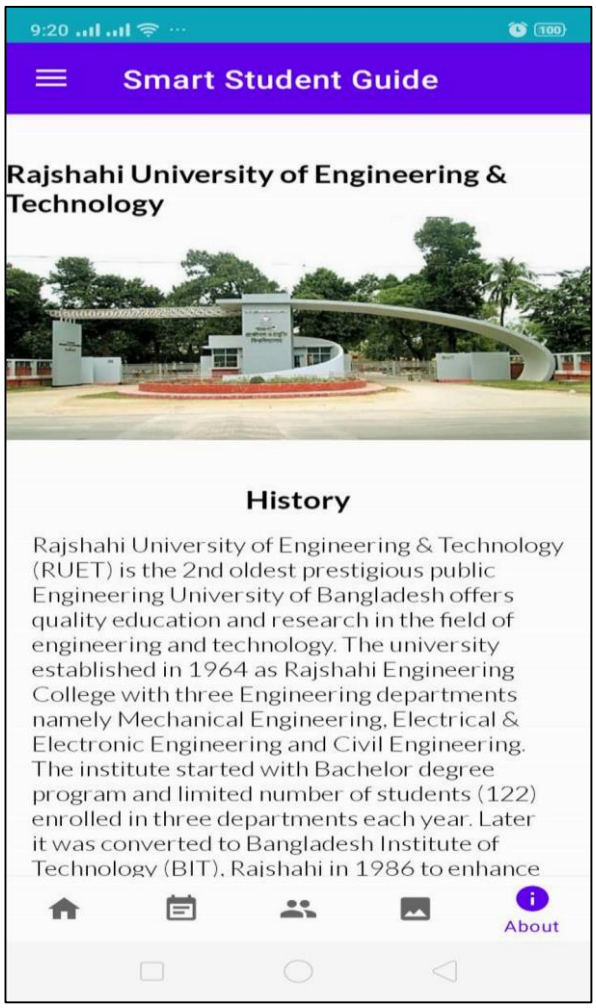

Figure 10: About

Those are the options here. The homepage containing the navigation options, all of them are changeable by the admin. By default, entering the application, the user will see the homepage while he/she has to select options to go to the other pages.

In textview of about frafment's fragment_about.xml, along with the headline with customed height, weight, color, size and gravity, the about part was written.

Navigation Drawer

The next thing here is the navigation drawer*. At the topmost of every pages, the three lines points to the navigation drawer to come after select. It can also be created with the bottom navigation. The drawer icon provides the options with clicking.

If the icons are downloaded or custom, and not built in, then we have to rename them without capital letter or numbers. In the xml file, items were used for the options Video lectures, Ebooks, Website and Developer with their icons, after creating one, rest were copy-pasted. Then the navigation header was created. A separate layout resourse file was created and the layout was designed with all properties. Scaletype was fitxy. Here, relative layout or constant layout could be used instead of linear layout. But we needed a simple image. In activitymain.xml, a drawerLayout was created which contains the constant layout and the navigation. ID was given, all those was shown in the palate.*

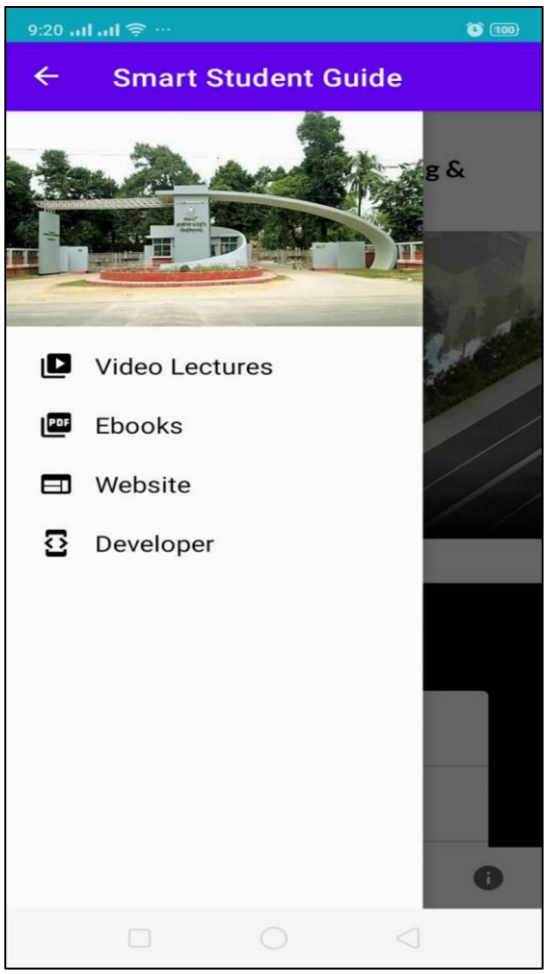

Figure 11: Navigation Drawer

*Navigation drawers provide access to destinations and app functionality, such as switching accounts. It is a UI panel that shows an app's main navigation menu. The drawer appears when the user touches the drawer icon in the app bar or when the user swipes a finger from the left edge of the screen.

*Navigation header is an image which generally is placed in top of the navigation drawer, it's not a mandatory but it beautifies the navigation header.

Navigation Drawer Options

There are four options in the navigation drawer. The first one is-

Video Lectures

There , the URL of a google drive link is provided which is provided by one of our classmates. The link contains all the necessary stuffs fro online class such as-online class recording, teacher-provided slides, sheets, downloaded pdfs, any kinds of suggestion from teachers, class routine etc. Foe now online-class materials are added. But it can be expanded later.

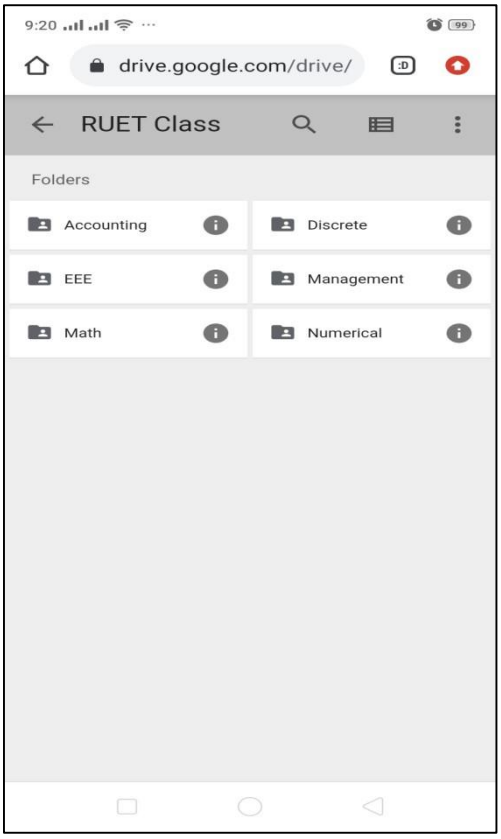

Figure 12: Video Lectures

All those data are stored in firebase storage. The next option is-

Ebooks

The admin app-part contains the pdf uploading part, those uploaded ebooks are provided here. The user account holder can directly download the pdf from here. The upload option was designed in the same way using the custom text-view, textInputLAyout and input talking buttons.

Ebooks option is directly controlled. The admin uploads the pdf files after downloading them himself. All those uploaded pdf are stored in firebase while in the application. These can be used from any device, no fear of having lost devices of admin and having to upload the pdfs again.
With clicking the names, the ebooks are directly downloaded, for

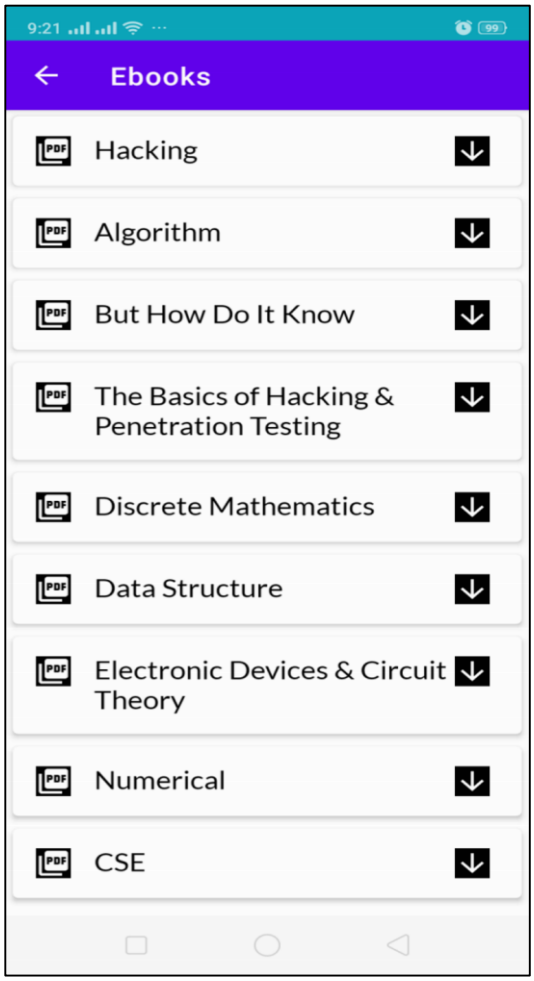

Figure 13: Ebook

now, they can't be read online. The names can be renamed as with the choice of the admin.

Website

The next option is the website. There is the URL of RUET official website is passed, and the website comes with clicking the icon. Even if a student may not know much from the application's others parts, visiting the website can be more helpful, all the official news is provided in the website. So it's a very essential option.

Figure 14: Website

Developer

In the developer option, both of our information is provided. It includes our name, roll number, email address and a picture.

Part 3-Using Process

The using process of the application is given below-

1. Firstly, one have to register him/herself in the application if he/she doesn't already have an account. In the login screen, the user will have to input his/her email and create a strong password. The his/her account will be enabled. After filling these two fields, the submit button will be enabled. If the user keeps any of these fields empty, the submit button will not be enabled.
2. After having an account, he/she has to login by entering the email and password, the application will detect the account as admin or only user.
3. After entering, the homescreen will be shown with bottom navigation bar, navigation drawer and google map. Just by clicking every option, he/ she can go to the pages or screens; and get advantages.
4. The admin will update the information as described before, while the user's job is to just getting benefits.

   This is how the application is to be used.

Part 4-Tools

1. Windows OS
2. Android studio as IDE
3. Languages-Java, XML(Extensive markup Language)
4. Firebase for database
5. Google Location service

Part 5-Business Model

Basic Cost of Using 'Smart Student Guide' : As stated before, this application only requires internet connection and not anything else. So

internet cost is the only cost. To access the current location, it costs 5MB per hour. To go to the pages and websites, a user needs 3 MB if he/she needs some urgent information. A pdf of university book is mostly 50 MB, so he/she can download it at once. He/she needs space in the phone storage. The admin needs a little bit more amount of internet, so Banglalink or Graminphone's cheap but efficient packages can do good. For worst case, let's assume, the user uses 1GB at BDT 189 30 days pack of Grameenphone operator. So, 189 BDT can be monthly cost(Just an example).

Business Plan

Being well-known among people is one of the most important need for a successful business. So to make business with this 'Smart Student Guide' application, at first this application needs to be popular to the students of our country.

Initially we can work with the students of RUET. For the interest of advertisement, we can collaborate with the community of the popular clubs of our university like RCF, RUET DC, Onuronon, Radio RUET etc. With the help of these communities, we can make our application familiar in our campus. Further the popularity can be spread in other university students and finally all over Bangladesh. It can be used for school, collages and even pre-primary students also.

Secondly, after the successful advertisement, we can collaborate with the SIM operator companies (Grameenphone, Banglalink, Airtel, Robi etc.). Suppose, we can make a deal with GP on the condition that they will offer the users a special package like, using only 'Smart Student Guide' application, the users can buy some efficient internet packages. On the other hand, the number of users of their application or SIM will be increased because of this special offer. Then we can demand a percentage of their profit against this special package. In this way we can make business with other SIM companies.

Part 6-Future Scope

1. We can add more detail information about everything such that all departments, teachers etc.
2. Youtube lectures form important topics can be provided by passing the link of the videos.
3. We can collaborate with Foodpanda, RUET's nearly food-stalls, every men-women halls to organize and centerize every single information needed for a RUET student. He/she won't need to go to stalls themselves for checking if some specific foof is there or not, returning hall after going somewhere will be easier too knowing every timetable.
4. If some facility of communicating with other students can be added like discord, messenger, zoom etc. it could be a great opportunity of accumulating everything, students won't have to search through different places randomly and just can use this single application.

Part 7-Result and Conclusion

This report describes the android application 'Smart student Guide'. This application can be a very essential tool for a student.

The main purpose of making this application is to make our student life easier. A connection between so many students in easiest way possible. So many important information that make a student go through so many different activities, can be gained from here pretty easily.

We tried our best to develop this as much as possible. So, we hope this application will be helpful for the students in future.

Github Link: https://github.com/mondolmridul007/Smart_Student_Guide