



DEVELOPMENT OF STUDENTS' PROJECT TOPIC VERIFICATION AND APPROVAL SYSTEM.

AMANNAH, CONSTANE IZUCHUKWU
*Ignatius Ajuru University of Education,
Rumuolumeni, Port Harcourt, Nigeria.*

Abstract

Project topic allocation and approval system is the process of receiving project topics from students by the supervisor who carefully weighs the submitted topic for researchability before eventual approval. The key challenge associated with this method is frequent duplication of project

Keywords

Development, Verification, System, Approval, Project

topics by students, as well as copying of already existing project works in the department thereby rendering the entire research efforts unrewarding. This research work strives for a better means of

INTRODUCTION

In many higher institutions in Nigeria and beyond students carryout research in their given field of specialty (for example computer science, biology, chemistry, physics and so on) as a compulsory requisite for graduation. Usually, a project or research can be done by one or more students, though in some cases a project is suitable for more than one student to work on simultaneously. The student is allowed to choose the area which he wants to research on. Though, in some cases the lecturer will also offer a range of projects for students to choose from. Each student has preferences over the available projects

verifying, evaluating, achieve this aim, the developed using the allocating, and following objectives Android OS as the approving project were designed, to; platform and the topics for students at review the existing outcome of the their project year with method of project research was an a view to overcoming topic allocation, design android application the challenges of the a project allocation that was able to handle traditional means of system, and the shortcomings of approving project implement the project the existing system of work and its effect on allocation system. Test project work students. The aim of and deploy the system. allocation and this study was to The V-model approval. The develop students' methodology was developed android project topic adopted in the design application was tested verification and of the software. The and deployed. approval system. To software was

that he/she finds acceptable, whilst a lecturer will normally have preferences over the students that he/she is willing to supervise. There may also be upper bounds on the number of students that can be assigned to a particular project, and the number of students that a given lecturer is willing to supervise.

In Computer Science Department of Ignatius Ajuru University of Education, project work is an integral aspect of the completion of the undergraduate programme before the issuing of the Bachelor degree. According to Davidson (2002), every undergraduate research and creative project requires thoughtful and sustained collaboration between the student researcher and his/her supervisor. It is the responsibility of the student to seek out an appropriate research topic, which in some situations the project supervisor selects a topic for the supervisee. The essential responsibilities of the student researcher are as follows;

- Explore possible research topics that reflect his (researcher) interests. If he cares about the subject, he will pursue interesting questions because, he wants to know the answers to his

speculations. That will draw him into the subject, enrich his knowledge in that field.

- Identify the knowledge, experience, and skills that he will need to complete the project. The project supervisor may advise the researcher to enroll in an appropriate research methods course before embarking upon the project (for example if the supervisee does not have the sufficient knowledge to carry out the implementation of his research, the supervisor can advise him to enroll on an extra moral class to boost his capabilities).
- The researcher is expected to consult with his supervisor to determine if his project will be approved by the department and school at large.
- The researcher should ask his supervisor assigned to him by the department to review and critique his research topic.
- The researcher should develop a time frame and a system of communication with the project supervisor during the project period that will allow for an appropriate balance of mentor guidance and student independence. In many cases, regular weekly or bi-weekly meetings are optimal to keep the project on track.

Statement of the Problem

A project or research is a diligent inquiry or examination to seek or revise facts, principles, theories, applications, and so on; laborious or continued search after truth. It helps to expand the student's knowledge in that particular field and when the student wants to further his education in that he already has background knowledge of that field. The importance of project writing in the life of an undergraduate cannot be overemphasized or stressed because as students continue to grow in the field they need research. The traditional way of approving project works currently in Ignatius Ajuru University is unstable because it lacks the mechanism to check if two students have the same topics and if a student is copying his work from an already made source which is a very gigantic academic crime which is punishable by law. Also this current system of allocation of project makes project work look like an assignment and destroy the research spirit which the students were supposed to develop as undergraduates. This

leads to many unbaked graduates in the societies today. This project work tries to address the problems mentioned above and a system will be developed to handle those issues. A system will be developed which will act as a verifier, to verify if a project to has been done in the department before or elsewhere.

Aim and Objectives of the Study

The aim of this study is to develop students' project topic verification and approval system. To achieve this aim, the following objectives were frmed;

- I. Review the existing method of project topic allocation.
- II. Design a project allocation system.
- III. Implement the project allocation system.
- IV. Test and deploy the system.

Significance of the Study

The outcome of this study will be of great benefit to the students because it helps to develop their research skills in preparation for higher studies where they are required to carry out complex research works. It is of great importance to the staff of Computer Science Department of Ignatius Ajuru University of Education because they will not have to go through the stress of verifying project works. Also it will be of importance to other researchers in the field of computer science.

1.5 Scope of the Study

The computer science principle that will be applied in this project during the software develop will be bottom-up approach, which has to do with developing module by module until the full software is completed. The software environment in which the application will be deployed will comprise of:

- i. Web server to host application
- ii. Web browser (independent of OS) to access application
- iii. Mobile application for android devices

The application will be run on a local host. This is necessary because the administrative backend and the detailed application will be based on PHP. The administrative end will be accessed via a web browser on the PC. This

will not be dependent on the operating system since it is web based. The student end will run on Android and other operating system.

LITERATURE REVIEW

The literature was reviewed under the following subheadings;

- Theoretical Framework
- Related Literature
- Empirical Review

Theoretical Framework

In this section, we consider a theory that compares a particular group of strings (text) and checks for similarities, if it could be found with another string or set of strings.

Theory of Bloom Filter

Bloom filter was formulated by Burton H. Bloom in 1970 (Bloom, 1970) and is used widely today for different purposes including web caching, intrusion detection, content based routing. Given a string X , the Bloom filter computes k hash functions on it producing k hash values ranging from 1 to m . It then sets k bits in an m -bit long vector at the addresses corresponding to the k hash values. The same procedure is repeated for all the members of the set. This process is called "programming" of the filter (Dharmapurikar, Attig & Lockwood, 2005, Bloom, 1970). Below is a figure used to explain the theory. Two messages, X_1 and X_2 are being programmed in the Bloom filter which has $k = 4$ hash functions and $m = 13$ bits in the array. Note that different strings can have overlapping bit patterns as shown in figure 2.1. The query process is similar to programming, where a string whose membership is to be verified is input to the filter. The Bloom filter generates k hash values using same hash functions it used to program the filter. The bits in the m -bit long vector at the locations corresponding to the k hash values are looked up. If at least one of these k bits is found not set then the string is declared to be a non-member of the set. If all the bits are found to be set then the string is said to belong to the set with a certain probability. This uncertainty in the membership comes from the fact that those k bits in the m -bit vector can be set by any of the n members. Thus

finding a bit set does not necessarily imply that it was set by the particular string being queried. However, finding a bit not set certainly implies that the string does not belong to the set, since if it did then all the k bits would definitely have been set when the Bloom filter was programmed with that string. This explains the presence of false positives in this scheme, and the absence of any false negatives. The concept is illustrated in figures 2.1(b) and 2.1(c).

The relevance of this theory to this study is, a database will be setup to act like the Bloom filter, just like figure (a) above, some existing research works carried out by different scholars will be uploaded to the database then just like figure (b) above, Y is the research topic checking if it is the same with the one in the database, then we see an output as Z .

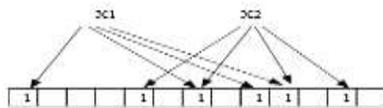


Figure 1(a): Programming multiple strings in the Bloom filter. (Strings X_1 and X_2 are being programmed. Here $k = 4$ and $m = 13$)

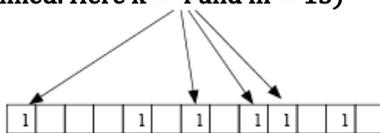


Figure 1(b): Querying a Bloom filter with a string. (Bloom filter gives a 'match' for string Y since all bits are set)

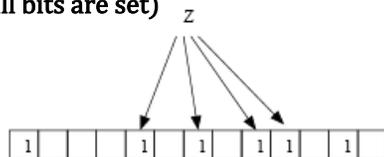


Figure 1(c): False positives. (Bloom filter gives a match for string Z though it is not programmed in it, since all the hash bits are set. This is a false positive)

Related Literature

In this section, we evaluate some works related to the research to give a scholarly basis for this project.

Student Project

A student project is a work that a student creates as part of a tertiary/high school, undergraduate, or graduate school program. Student projects are long academic documents that students write after they research a particular subject in depth. Therefore, student projects are usually assigned once per course, per semester, or only once as part of an academic program (David and Abraham, 2007). There are many different ways in which a lecturer or academic program will use student project assignments. Therefore, students may have to write projects for different reasons, depending on their level of study and academic institution. For example (Dye, 2001) "a secondary school student may have to write a year-end student project as part of an overall high school program. Usually, the project will be graded by a supervisor, but all students in a particular grade might have to work on a project in order to pass a grade. A junior project is a good example of this sort of assignment.

An undergraduate student project might be a large assignment that a student has to create as part of an academic course. In such a case, the undergraduate student project might be similar to a term paper. For an undergraduate student project, the student should research a particular subject in depth in order to create the paper. This paper would usually account for the majority of a semester grade. A student may also have to create a project for each course that he or she is taking.

A graduate student project has its own definition and set of requirements. In most cases, a graduate project is, according to Valter D.S. (2005), "one paper that a student works on for a large portion of the graduate program, especially in his or her final months of the program. This project requires a huge amount of research and may even be ground-breaking for a particular industry". Students will then have to defend their student project in front of a panel of judges that are familiar with the subject matter in the project. These panel members may ask the student questions related to his or her research or to the project itself. This sort of graduate project

is also often called a graduate school dissertation. While there are many definitions and uses for student project, the basic writing elements are the same. Therefore, students should always research a particular subject and write the academic document with an introduction, body, conclusion, resources, and appendices (Abraham, 2007).

Web Design

Web design is the “skill of creating presentations of content (usually hypertext or hypermedia) that is delivered to an end-user through the World Wide Web, by way of a Web browser or other Web-enabled software like Internet television clients, micro blogging clients and RSS readers” (Thorn, 2003). The intent of web design is to create a web site—collection of electronic documents and applications that reside on a web server/servers and present content and interactive features/interfaces to the end user in form of Web pages once requested. Such elements as text, bit-mapped images (GIFs, JPEGs) and forms can be placed on the page using HTML/XHTML/XML tags. Displaying more complex media (vector graphics, animations, videos, and sounds) requires plug-ins such as Adobe Flash, QuickTime, Java run-time environment, etc. Plug-ins are also embedded into web page by using HTML/XHTML tags (Thorn, 2003).

Improvements in browsers' compliance with W3C standards prompted a widespread acceptance and usage of XHTML/XML in conjunction with Cascading Style Sheets (CSS) to position and manipulate web page elements and objects. Latest standards, and proposals, aim at leading to browsers' ability to deliver a wide variety of content and accessibility options to the client possibly without employing plug-ins. Typically web pages are classified as static or dynamic:

Static pages don't change content and layout with every request unless a human (web master/programmer) manually updates the page. A simple HTML page is an example of static content. Dynamic pages adapt their content and/or appearance depending on end-users input/interaction or changes in the computing environment (user, time, database modifications, etc.) Content can be changed on the client side (end-user's computer) by using client-side scripting languages (JavaScript, JScript, Action script, etc.) to alter DOM elements (DHTML). Dynamic content is

often compiled on the server utilizing server-side scripting languages (Perl, PHP, ASP, JSP, ColdFusion, etc.). Both approaches are usually used in complex applications.

With growing specialization in the information technology field there is a strong tendency to draw a clear line between web design and web development. Web design is a kind of graphic design intended for development and styling of objects of the Internet's information environment to provide them with high-end consumer features and aesthetic qualities. The offered definition separates web design from web programming, emphasizing the functional features of a web site, as well as positioning web design as a kind of graphic design. The process of designing web pages, web sites, web applications or multimedia for the Web may utilize multiple disciplines, such as animation, authoring, communication design, corporate identity, graphic design, human-computer interaction, information architecture, interaction design, marketing, photography, search engine optimization and typography. Markup languages (such as HTML, XHTML and XML) Style sheet languages (such as CSS and XSL)

Client-side scripting (such as JavaScript) Server-side scripting (such as PHP and ASP)

Database technologies (such as MySQL and PostgreSQL) Multimedia technologies (such as Flash and Silverlight)

Web pages and web sites can be static pages, or can be programmed to be dynamic pages that automatically adapt content or visual appearance depending on a variety of factors, such as input from the end-user, input from the Webmaster or changes in the computing environment (such as the site's associated database having been modified).

With growing specialization within communication design and information technology fields, there is a strong tendency to draw a clear line between web design specifically for web pages and web development for the overall logistics of all web-based services.

Choosing the Web Design Software

The best web design software may depend on many factors. How advanced needs to be determined, as well as how easy to use. For a professional web

designer, a software with all the latest features will be needed. On the other hand, for an inexperienced designer with computers and wish only to create a site featuring pictures of family or a blog, a simpler program would be better suited to your needs (Mening, 2018). According to Severance and Lent (2004), there are many web design software programs in the market, and new ones are released all the time, as technology advances. For professional designer, you will need to find design software that supports complex computer programming language. Some may even allow you to code the language you are using directly into the software. These complex web design programs are necessary for larger sites which require sophisticated common gateway interface (CGI) scripts and other backend commands. More complicated scripts are important in order for the site to read and send forms, online payments, display graphics and videos, and provide other technologically advanced features. When picking out software for professional use, be sure it says on the site or packaging that it is suitable for professional designers.

For novice designer who needs web design software more for personal use than professional, a simple program is probably more suitable. There are many commercial software applications available that allow everyday consumers to create very attractive sites with little or no technical knowledge. Generally, these programs provide a simple layout that can be altered with text, graphics, and colored backgrounds.

WYSIWYG Web Builder and Web Design Software

WYSIWYG is acronym of What You See Is What You Get. It simply means that the software provides an editing interface which resembles how the page will be displayed in a web browser. And it does not require any HTML knowledge or experience for creating a web page or site. This makes the program simple and easy to use, especially for average computer users. And the program makes web design process faster and easier than using text editor or HTML editor (Bernheim, 2018). All of the WYSIWYG web design provides easy to use drag and drop feature which allows user to add image, text, link and other web elements by using mouse clicks (Pablo, 2003), The good program supports many web languages including HTML, XHTML, CSS, JavaScript, PHP, ASP.NET, ASP.NET and AJAX. You can also

find many useful tools within the program such as build-in FTP, templates, photo editors, HTML cleaner, HTML validate, spell check and more. This type of program offers complete web design solution for all types of web designer and web developer whether you are a novice or a seasoned designer, a hobbyist or professional designer. A good example of a WYSIWYG web design is

WYSIWYG Website Design Software Reviews

We have different WYSIWYG. Below are some of the WYSIWYG in the market today.

Adobe Dreamweaver is the most powerful web design software program on the market today. It is a product of Adobe Systems Inc. You may hear many people say that it is also a popular WYSIWYG website creator and editor or it is a full feature web page design software package. Yes, it is true. It is the top choice for professional web designers and web developers. It provides endless features ranging from standard to advanced functions. Microsoft Expression Web is a WYSIWYG website creator and editor by Microsoft Corporation. It is one of most popular web design software solution on the market today. It provides all the tools you need including HTML, XHTML, CSS, JavaScript, PHP, ASP.NET, ASP.NET AJAX, visual diagnostics and CSS design capability to produce high-quality and standards-based websites. The package includes Expression Web + Super Preview, Expression Design and Expression Encoder + IIS Smooth Streaming. Expression Design is a professional design tool for creating graphics content that can be used within Expression Web.

Review of Empirical Studies

In this section, empirical works done by other scholars will be reviewed critically.

Design and Implementation of Anti-Plagiarism System

The work Design and implementation of anti-plagiarism system, was carried out in 2017 by Raj and Srikantiah. They defined plagiarism as “the wrongful appropriation” and “purloining and publication” of another author’s language, thoughts, ideas or expressions and the representation

of them as one's own original work". They defined plagiarism detection as the process of locating instances of plagiarism within a work or document. The objectives of the work is as follows; to describe how plagiarism detection works, to highlight the various techniques of plagiarism detection, to highlight some of the advantages and disadvantages of plagiarism detection, and to design and implement a windows-based application that will help detect plagiarisms.

The Rapid Application Development (RAD) model was adopted for the development of the plagiarism detection system. The model views the process of software development in four stages. The phases are as follows;

- Requirements planning Phase
- User define phase
- Construction phase
- Cutover phase

Based on the analysis described in their work, they built a system to detect plagiarism in a document, and the document will be in the database before any operation can be done on it. The software was developed using C# for matching the document with that in the database. The experiments were performed on a Pentium Duo Core - 2.0 GHz with 1GB of RAM. To check for plagiarism, the original document uploaded into the database, then the copied document is used as the input data. Now the user would click on compare to compare both documents. The result is in percentage (%). It is very fast, if the document size is below 100kb and below but when exceeded it results to system error, so they recommended that the file size should not exceed that boundary. The file format to be uploaded must be in .docx format else it will report invalid file-type, then the output is in text (.txt) format making the system delay in the conversion process (Raj and Srikantaiah, 2017). After studying this system, some shortcomings were observed such as the system must convert docx to txt. Format which causes delay in the system run time. The system that will be develop at the end of this study would be able to take input of .docx format and give out output without conversion of the file type to another extension. The software was primarily designed for lecturers and must be used by the lecturer who owns the license; it cannot be used by another lecturer. The new system to

be develop at the end of this study can be used by anyone not restricted to any particular group of persons. The proposed software will be able to search its database and the internet.

Database Management System for Student Project Program (SPP)

The research Database Management System for Student Project Program (Spp), was carried out by Shubham Barsaiya, Tanmay and Arjun in 2016. The project was based on the Student Project Program (SPP) of the Karnataka State Council for Science and Technology (KSCST) receive their students proposals for projects from all over Karnataka, and the proposals are received in the form of hard copy and they are evaluated manually by the personnel at the SPP center. This process is very laborious and time consuming. This project was designed to meet the requirements of KSCST Student Project Program (SPP) for managing the data submitted by the students through online application form. According to Barsaiya et al (2016) the project is basically an integration of three modules namely. Front-end, which provides a Graphical User Interface through interactive and responsive web forms by which students can apply for SPP; Backend, which will manage all the data submitted by a student through online application form. The objectives of the study are as follows;

- To implement online proposal forms for the students to access, fill and submit them through the KSCST website.
- To implement a module that provides the administrators to view the proposals obtained and functions to filter and sort them based on the subject of interest accordingly so that they can be accessed by the subject Experts ,who evaluate them.
- To implement a module for the subject experts such that they can view the proposals pertaining to their subject-matter and evaluate them accordingly.
- To implement a module for users (Administrator and subject experts) to register with the system, so that they can carry out the authorized functions.

The methodology used was the iterative model. A web application was developed. In the online portal environment, they categorized users into

three classes: Student, SPP officials, and Professors. Each of these three classes will act as an individual entity in Relational Database (MySQL). Any access to this portal will be an authorized access, as each entity has to first sign-up in order to login into their account. As soon as any of the entities logins, the session will be started to reduce traffic on their portal. When that session expires, he/she will get automatically logged-out from the portal services. In order to provide security and privacy, the encryption and decryption of data is done. CLIENT will send requests to the SERVER through API'S and SERVER will send response to the CLIENTS in JSON (a markup language) format. The portal is dependent on single Server, this may lead to SINGLE POINT FAILURE. In order to protect their services and database from external attacks, SQL INJECTIONS were used. The research achieved its goal of allowing students to be able to submit their research proposal from anywhere and supervisors can also approve not necessarily in his office. But the research failed to address when two students submits the same project topic to different project supervisors how such problem could be tackled, also if the research was lifted somewhere from the internet or from the school library. This research will try to handle this shortcomings.

SYSTEM ANALYSIS AND DESIGN

Analysis of the existing System

The traditional way of allocating project to students in our higher institutions need to be reconsidered since project/research writing is sensitive aspect of student education in the higher institution. The traditional system create room for allocating a single topic to more than one students or group of student thereby creating inefficiency in the purpose of student project writing. Below is how project allocation and approval is done using the existing system. The students are shared among the lecturers of the department, each student submits a project topic to his/her supervisor for approval. The supervisor 'A' may not know what supervisor 'B' have approved, so during the defense of the project two students may end up defending similar or in some cases same topic just like in Computer Science department 2017/2018 where two students came up on stage to defend a similar topic. If there was a system to check

what other supervisors has approved and what has been done over the years, if the proposed system is already implemented, students doing similar works will be reduced to minimal.

Problems Associated with the Existing System

Although, the manual process of project management meet the requirement, in order to make the job more effective some development has to be involve. The problem, of this system however is in the speed of the process and the dissemination of information to users of the system. Sometimes, supervisees complain of not seeing their supervisors which may lead to delay in the course of writing the project report.

In the existing system, it will take a student to wait for lecturer to collect his/her project topic. Sometime, it takes days or even weeks to get a topic. Another problem with the existing system is the fact that some of these topics given to the students are brought forward by the student thereby creating a chance of copy and paste, which do not at the end of the day make the project work a research study. If the new system is implemented the above listed problems can be controlled.

Analysis of the Proposed System

In the Proposed System, we have three roles in this system, an administrator, a supervisor/lecturer and a student. An administrator logs into this system. Students register in this system and get userid. A student should register, provide his information and also provide information about his team members (if team project is allowed). This will be saved in a database. After students are done with the registration process, then the department shortlists students for supervisors. Students can then submit their project topics to their supervisors and if approved it should appear in the homepage of the application software. For instance, if Design and Implementation of Student project allocation and approval system was submitted by Egbirika Gladys. If it is then approved and allocated to her it should appear in the homepage like this:

Design and Implementation of Students project Allocation and Approval System by Egbirika Gladys DOCS, IAUE

As such other lecturers and students can see it and no one will approve the topic for anyone or send the topic for approval again. An auto-generated

email will be sent to students telling them that their supervisor/lecturer has approved their project. This email will be responded by the student in a week time. There will be Facility for password changing. There is facility for changing the password for the student as well as the lecturer.

Merits of the Proposed System

To join other standard educational institutions in the world, it became necessary to develop a computerize students project management system, where students can get there project work approved with less stress and time. With the help of the proposed system (an android software), students will not only prevent the problem of project work duplication but will also reduce or stop the habit of students copying project works done years ago. All the works done in the department will be in the website's database. Once a similar work is sent to a lecture he searches the project work in the department online project library to find if they are similar.

Methodology of the Proposed System

The research method that will be applied in this work will be the V-model (Verification and Validation) method. The method is chosen because of its advantages in the research namely;

- i. Simple and easy to use
- ii. Testing activities and design happens before the actual coding
- iii. Proactive defect tracking: defects are found at early stage.

In using this model, the researcher gathers data related to the proposed system after the conceptualization of the idea, she tries to analyze it and understand the unit which will come first and its position in the web application. Then designing the modules and testing them will be the next. Finally, implementation of the design, testing and integration of all the modules.

Design of the Proposed System

In this section, the functionality of the proposed system will be described in details, how the overall system will look, how to query the system via input and how the system will respond or display its output.

Architecture of the Proposed System

The architecture of the proposed system is illustrated in figure 2.

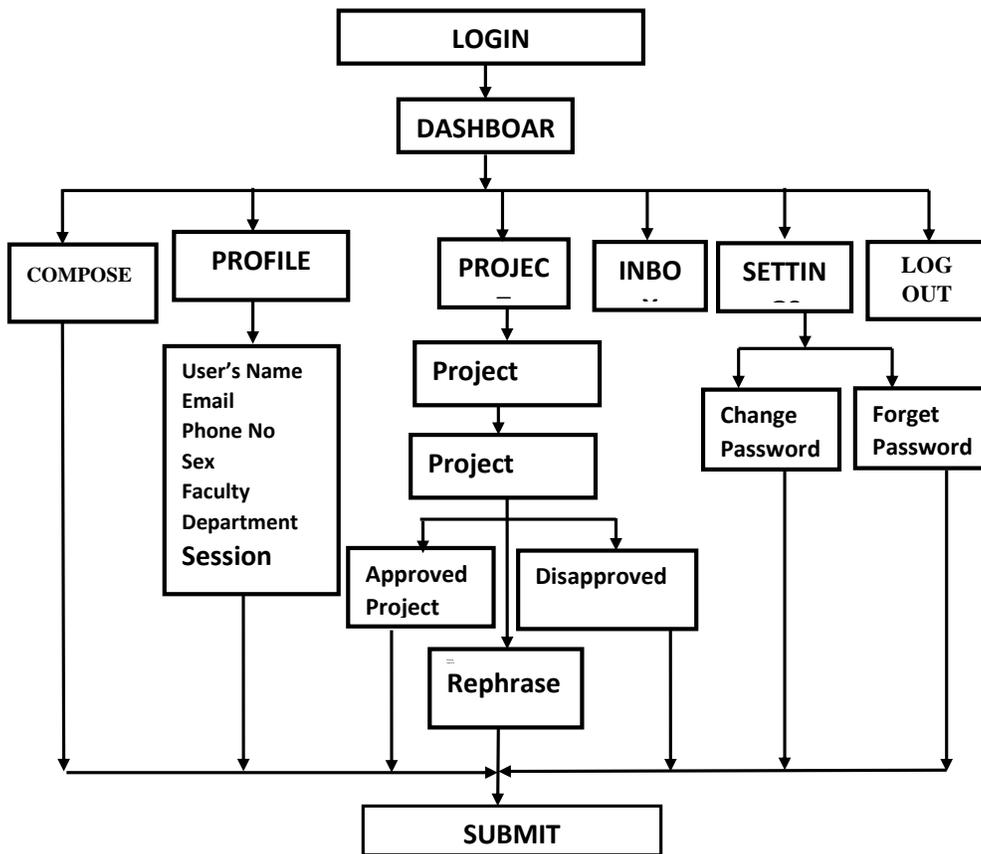


Figure 2: Architectural Design of the proposed

Once the software is launched, the user is asked to login with his details which allows him/her to gain access to the dashboard, once the login details matches the one in the database access is granted to him. In the dashboard the user can navigate to different modules of the software like visit his 'profile' which contains name, email, phone number (if there is need to call the supervisee or supervisor), sex, Faculty, department and the

Academic session. Also he/she can have access to the view the “Project” if it has been approval or declined. The supervisor in his dashboard when logged in sees the list of projects that are waiting for approval and those he approved. The inbox is just a message box were either supervisor or supervisee can send a message to the supervisor/supervisee. A user may feel his password is not strong or not secure he will navigate to the “SETTINGS” and change his password to his desired integrity status. After a user’s activity in the software is completed he clicks on the “LOGOUT” button and the application will close successfully.

Input Design of the Proposed System

For a user to be able to use the software he must have an account in the database, which correct details are expected to be entered during login, the user is expected to enter his email address or matriculation number and password to gain access to the figure 3.4a Dashboard.

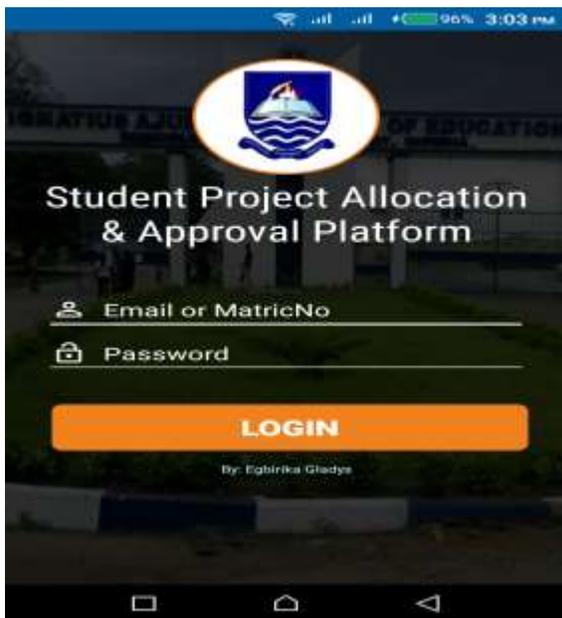


Figure 3(a): User Login Page

Sometimes users feels that their login details have been compromised and keens to be changed, in such situation the user just clicks on the “SETTING” in figure 3.1 and “Password reset” page will pop up, where the user can change his password to his desired new password and click on the “confirm” button will be stored in the database.



Figure 3(b): Password Reset Page

The main aim of this research is for the students to be able to submit their project topic for approval to their supervisor, figure 3.2c is the project submission page. The supervisee is expected to write the project topic and write a brief summary about the topic which will include the problem the student intend solving.

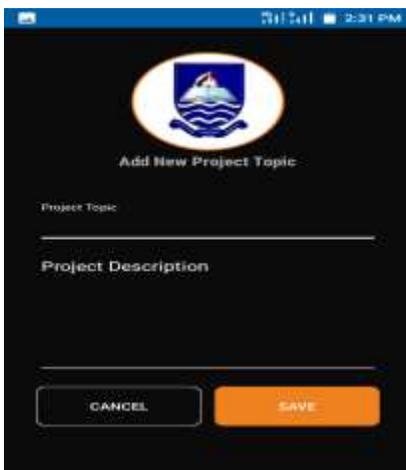


Figure 3(c): Project submission Page
3.5.3 Process Design of the Proposed System

The data acquired from the users will be used for displaying and generating reports where necessary, authenticating users' login and for other processing operations. The data are stored, sorted and arranged in a tabular form in the database and are also kept in a meaningful & clearer format for easy use and access. Below is a user activity in the

software, in the page labelled '1' in figure 3.4, the user is at the Login page after login details has been accepted the user moves to label '2' which is the dashboard, label '3' is showing the user profile while label '4' is showing awaiting topics for approval, label '5' the user is sending a project topic for approval and '6' the user is changing his/her password.

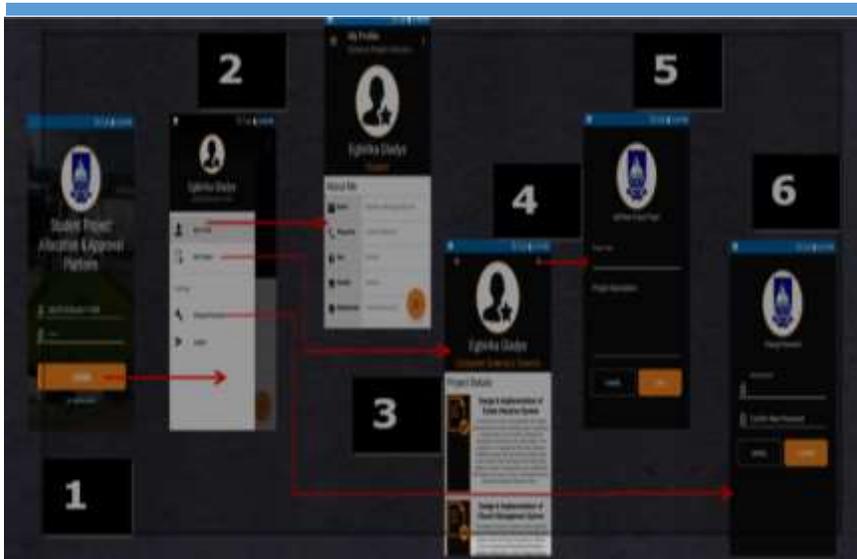


Figure 4: The process view of the Software

In the software, the supervisee is not given the permission to access everything in the software except what is displayed in his dashboard.

Output Design of the Proposed System

The output design displays manipulation done by the user. The output design allows the system to indicate the effect of the users' manipulations. Below are some of the output interfaces of the software. Figure 5(a) is the software's dashboard after a user has successfully keyed in his username or matriculation number as the case maybe, the dashboard opens for the user for navigating to either part of the software.

From figure 5 when the profile button is clicked it opens the user profile showing the user's basic details. Below is the user's profile. When the 'Project' button is click from figure 5(a) the projects the user submitted will be enlisted, showing which is approved or waiting approval or rejected. Figure 5(b) below shows list of project(s) waiting approval. After students have been registered into the system, the profile will be displayed as in figure 5(a). Only registered users are able to view profile of the software. Figure 5(b) is showing name of students, department and the project topic for the supervisor to approve. The supervisor either "Accepts" or "Rejects".



Figure 5(a): Software Dashboard

LIST OF PROJECT(S) WAITING APPROVAL	
<input type="checkbox"/>	NAME: F-NAME L-NAME <input type="text"/> <input type="button" value="ACCEPT/REJECT"/>
	DEPT: DEPARTMENT NAME <input type="text"/>
	PROJECT: PROJECT TITLE <input type="text"/>
<input type="checkbox"/>	NAME: F-NAME L-NAME <input type="text"/> <input type="button" value="ACCEPT/REJECT"/>
	DEPT: DEPARTMENT NAME <input type="text"/>
	PROJECT: PROJECT TITLE <input type="text"/>
<input type="checkbox"/>	NAME: F-NAME L-NAME <input type="text"/> <input type="button" value="ACCEPT/REJECT"/>
	DEPT: DEPARTMENT NAME <input type="text"/>
	PROJECT: PROJECT TITLE <input type="text"/>
SUPERVISOR'S REMARK: <input type="text"/>	

Figure 5(b): Students' projects in Staff dashboard waiting approval

Database Design of the Proposed System

Database is where all the data (password and username, existing project topics and so on) are stored in the software. The structure of relational database shows the different tables that make up the database and links among the fields.

Table 1: Users Database Table design (User)

S/N	Column Name	Data Type	Size	Constraint
1	Username	VARCHAR	50	Primary Key
2	Surname	VARCHAR	60	
3	Middle Name	VARCHAR	60	
4	First Name	VARCHAR	60	
5	Sex	VARCHAR	6	
6	Department	VARCHAR	50	
7	Password	VARCHAR	40	

Security Measures of the Proposed System

The importance of security in any software cannot be overemphasized due to hackers' threat on mobile, desktop and web applications, with the emergence of new hacking tools on daily bases such as KeyLogger (which can be used to steal cookies from mobile device or desktop), Brute Force (this method does dictionary check of the user password until it gets a match, that's why users are advised not to use only alphabets as their password), Grease Monkey and so on. Software developer, webmasters and designers are supposed to be at the alert. Below are some of the secure measures that will be applied in the proposed software development.

- a. Access:** Level of permission given to the supervisee will be limited. A supervisee should be able to see his own profile, his dashboard, see his inbox and send message that is all he can do but changing any feature in the software like changing the changing the design of the application, editing another student profile is not possible but the software

administrator who may be the Head of Department (H.O.D) can perform all this actions and he alone can register students and staffs to the Software's database.

- b. Password strength:** there are some applications that can hack passwords, if the password is weak for example "prince". Maybe the owner of the password is "Prince Charles". It's easy to guess, which is one skill hackers use to get password, also the hacking software called "Dictionary Checker" the password is vulnerable to it because software tries the victim's username with all the words in the dictionary until it gets a match. To avoid being victims to hacking attacks, during registration the password must contain alphanumeric characters. For example "pr1nc3", Gladys can be "G|_4DY5" these passwords are acceptable and are strong.

SYSTEM IMPLEMENTATION

Choice of Platform

The platform which the design was implemented was the Android Operating System, because of the flexibility of the android software, portability and so on.

System Requirements

In this subsection, the system requirement for effective performance will be discussed, which include both the hardware and software requirements.

Minimum Hardware Requirements

- A computer machine with Pentium IV Microprocessor
- Minimum memory (RAM) of 512MB and above
- Network Interface Card (NIC)
- Modem
- Mouse and keyboard
- An android phone

Minimum Software Requirement

- MySQL server
- Operating System: Window OS, Mac OS, DOS etc.

- Apache server
- Android Studio

Software Testing

Before the deployment of the system, the following testing methods were used to test the software, the bottom-up and top-down testing techniques were adopted.

Documentation

In order to use the software in your phone, first it must be installed in the android phone. Once it has been installed the user clicks on the software and it starts running and the Login page pops up. After typing the correct password and username it opens the user dashboard for the user to perform any operation of his choice. For those who are interested in using their computer below are the steps for running the software in your system;

- Copy the raw file of the propose software to the system, save it in a particular folder and give the folder a name (for example “Project file”).
- Download Bluestack and install in your system.
- Run the bluestack and locate the file that was copied earlier to project file.
- Once the file is located, double click on it and it will install in the system.
- After it has been installed, you can follow the steps outlined for mobile device users but note you will be running the software through bluestack always.

RESULTS

The output of the proposed system on implementation is highlighted in figures 6 to 9. Figure 6 is the login platform, figure 7 is the project topic submission and description platform, figure 8 is the project topic’s status sent to the student’s dashboard for his attention after the decision of the student’s supervisor on the submitted topics. while figure 9 is the project topic evaluation platform.

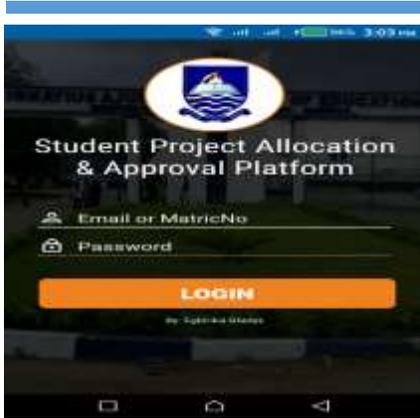


Figure 6: Password Reset Page



Figure 7: Project submission Page

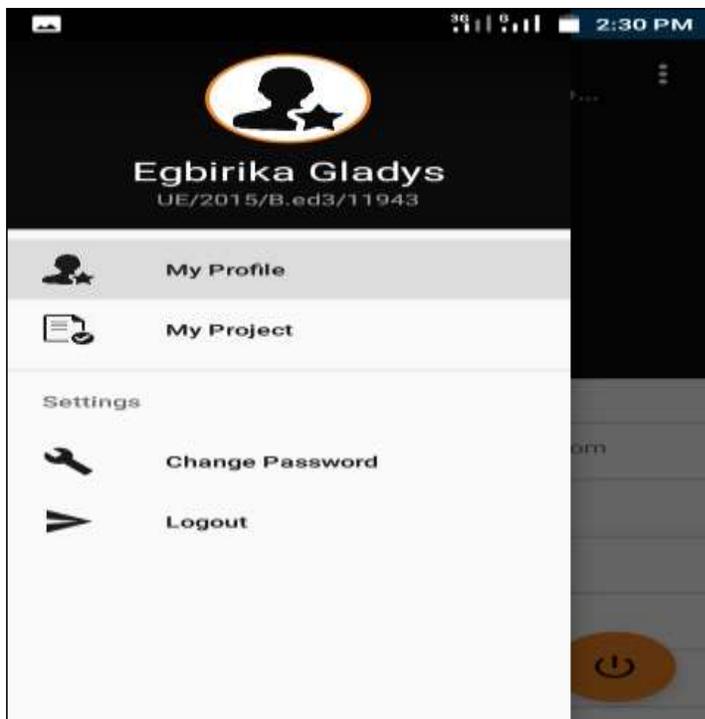


Figure 8: Student Dashboard

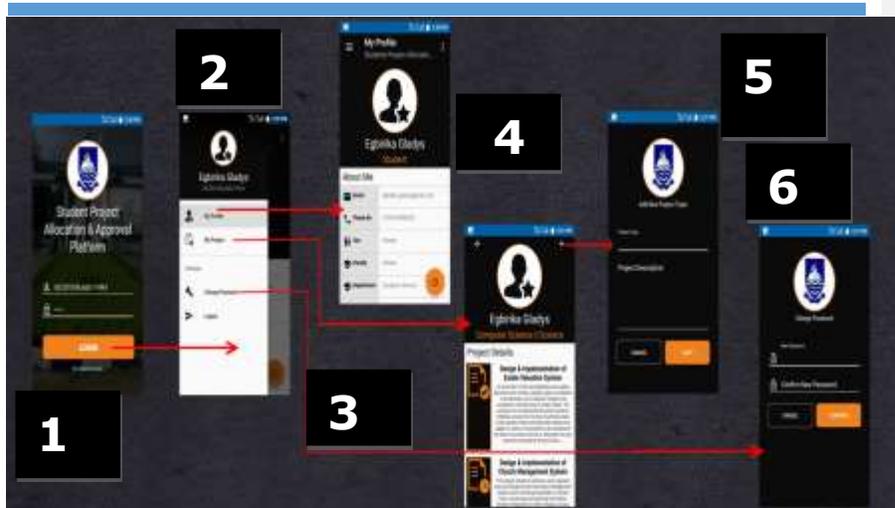


Figure 9: The process view of the Software

SUMMARY AND CONCLUSION

Summary

Student Project work is an essential part of their stay in school and it is a period when students are allowed to choose any topic or branch in their discipline to research on their own under the auspices of a lecturer as a supervisor. In this work, the current means of project allocation and approval system was review, how it works and its shortcomings and so on. A new system was proposed to solve this shortcoming and enhance the performance of both supervisors and supervisees. An android software was develop using android programming language and MySQL was used for the database which will enable students and supervisors to submit and approve a project work respectively, from anywhere without both parties seeing each other.

Conclusion

In this work, I tried to show how the existing method of student's project work allocation and approve system has failed in the long run. For instance, sometimes students complain of not seeing their supervisors to approval their project work and also during project defense two or more students

may end up carrying out research about a single topic or a student can go to the department library to pick a project work and duplicate it which is a great academic theft unknown to the project supervisor he/she will just approve the topic. A Software was developed which students which are registered by the HOD can have access to this software and also supervisor whose students can submit their project for approval via the software, thereby eradicating the hitch of not seeing supervisor in the office. The supervisor reviews the topic and has the right to turn the topic down (reject) or approve it (accept). Various security measures were put into consideration for the security system of the software.

Recommendations

Upon completion of this project, I recommend the followings;

- The department should adopt the application, because it is more efficient and effective compared to the traditional way of project allocation and approval system.
- HOD can appoint someone to take administrative oversight of the system, one who will be responsible for effective functioning of the system.
- If eventually the system is adopted, it should be integrated with the existing school management system instead of being run as an independent application.

Reference

- Baeza-Yates, R. A. (2010). String Searching Algorithms. Retrieved 15-06-2018 from <http://orion.lcg.ufrj.br/Dr.Dobbs/books/book5/chap10.htm>
- Barsaiya, B.T. & Misra, A.(2016). Database Management System for Student Project Program (Spp). Retrieved on 31/07/2018 from http://www.kscst.iisc.ernet.in/spp/39_series/SPP39S/01_Seminar%20Projects/049_39S_BE_2268.pdf

- Bernheim, L. (2018). 7 Best WYSIWYG Web Builder Reviews. Retrieved 10-05-2018 from <https://www.hostingadvice.com/how-to/wysiwyg-web-builder/>
- Bloom, B. H. (1970). Space/Time Trade-offs in Hash Coding with Allowable Errors. Retrieved 21-06-2018 from http://dmod.eu/deca/ft_gateway.cfm.pdf.
- Dharmapurikar, S., Attig, M. & Lockwood J. (2005). Design and Implementation of a String Matching System for Network Intrusion Detection Using FPGA-Based Bloom Filters. Retrieved 17/07/2018 from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.61.6479&rep=rep1&type=pdf>
- Griffiths, R. (2004). Knowledge Production and the Research-Teaching Nexus: The Case of the Built Environment Disciplines. Retrieved 16-05-2018 from <https://www.tandfonline.com/doi/abs/10.1080/0307507042000287212>
- Healey, M. (2005). Linking Research and Teaching: Exploring Disciplinary Spaces and the Role of Inquiry-Based Learning.
- Homden, B. (2017). Why is University Research Important for Students? Retrieve on 16-05-2018 from <https://www.kaplanpathways.com/about/news/university-research-important-students/>
- Huffington, J. (2011). The Role of Academic Research. Retrieved 16-05-2018 from <http://www.huffingtonpost.com/dr-john-story/the-role-of-academic-research-2011-05-16.html>
- Mening, R. (2018). Essential Web Design Software. Retrieved 10/06/2018 from <https://websitesetup.org/web-design-software/>

Moussa, M. I. & El-Atta A. H. A. (2011). A Visual Implementation of Student Project Allocation. Retrieved 17/06/2018 from <https://pdfs.semanticscholar.org/0dc8/a616dc1824c1059686e880e9bdc633fa957f.pdf>

Nikitin, F. (2006). Bloom Filters and their Applications. Retrieved 19/5/2018 from <http://cs.joensuu.fi/pages/whamalai/sciwri/fedor.pdf>

Pablo, S. (2003). WYSIWYG Web Builder. Retrieved 10/06/2018 from https://www.pablosoftwaresolutions.com/html/wysiwyg_web_builder.html

Walkington, H. (2004). Students as Researchers: Supporting Undergraduate Research in the Disciplines in Higher Education. Retrieved on 16-05-2018 from https://www.heacademy.ac.uk/system/files/resources/Students%20as%20researchers_1.pdf

Wan-hong, L. (2016). A Web-Based Project Allocation System. Retrieved on 10-06-2018 from <https://i.cs.hku.hk/fyp/2016/fyp16022/files/project-plan.pdf>
Retrieved on 16-05-2018 from <https://delta.wisc.edu/Events/BBB%20Balance%20Healey.pdf>

Appendix 1 : SOURCE CODE

AndroidManifest.xml file

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
package="samzytechsolutions.com.studentsprojectallocationapproval">
  <uses-permission android:name="android.permission.INTERNET" />
  <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
```

```
<application
  android:allowBackup="true"
  android:icon="@mipmap/ic_launcher"
  android:label="@string/app_name"
  android:roundIcon="@mipmap/ic_launcher_round"
  android:supportsRtl="true"
  android:theme="@style/AppTheme">
  <activity
    android:name=".MainActivity"
    android:label="Allocation      &      Approval"
    android:theme="@style/AppTheme.NoActionBar" />
  <activity
    android:name=".LoginActivity">
    <intent-filter>
      <action android:name="android.intent.action.MAIN" />
      <category android:name="android.intent.category.LAUNCHER"
    />
  />
  </intent-filter>
</activity>
<activity android:name=".ViewProjectDetails" />
<activity android:name=".ViewProjectDetails_Options" />
<activity android:name=".ChangePassword" />
<activity android:name=".ForgottenPassword" />
<activity android:name=".AddProject" />
<activity android:name=".EditProject" />
<activity android:name=".AddUser" />
<activity android:name=".AddStudent" />
<activity android:name=".EditUser" />
<activity android:name=".EditStudent" />
<activity android:name=".config.VolleyActivity"></activity>
</application>
```

</manifest>

build.gradle file

apply plugin: 'com.android.application'

```
android {
  compileSdkVersion 27
  defaultConfig {
    applicationId
```

```
"samzytechsolutions.com.studentsprojectallocationapproval"
  minSdkVersion          15
  targetSdkVersion       27
  versionCode            1
  versionName            "1.0"
  testInstrumentationRunner
"android.support.test.runner.AndroidJUnitRunner"
}
buildTypes                {
  release                  {
    minifyEnabled          false
    proguardFiles          getDefaultProguardFile('proguard-android.txt'),
'proguard-rules.pro'
  }
}
}

dependencies               {
  implementation          fileTree(dir: 'libs', include: ['*.jar'])
  implementation          'com.android.support:appcompat-v7:27.1.1'
  implementation          'com.android.support:design:27.1.1'
  implementation          'com.android.support.constraint:constraint-
layout:1.1.3'
  testImplementation      'junit:junit:4.12'
  implementation          'com.android.support:support-v4:27.0.2'
  implementation          'com.android.volley:volley:1.0.0'
  androidTestImplementation 'com.android.support.test:runner:1.0.2'
  androidTestImplementation
'com.android.support.test.espresso:espresso-core:3.0.2'
}
Login Activity Layout XML
<?xml                      version="1.0"                      encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:fitsSystemWindows="true"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:background="@drawable/school"
```

```
android:gravity="center"
android:orientation="vertical"

android:id="@+id/login_relative_layout"
tools:ignore="ExtraText"
tools:context=".LoginActivity">

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#bf000000"
    >
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:orientation="vertical"
    android:paddingLeft="25dp"
    android:paddingRight="25dp">

<ImageView
    android:layout_width="120dp"
    android:layout_height="120dp"
    android:layout_gravity="center"
    android:layout_marginBottom="2dp"
    android:layout_marginTop="30dp"
    android:background="@drawable/circle"
    android:padding="5dp"
    android:src="@drawable/iaue_logo" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginBottom="40dp"
    android:fontFamily="sans-serif-condensed"
    android:text="Student Project Allocation & Approval Platform"
    android:textAlignment="center"
    android:textColor="@android:color/white"
    android:textSize="30dp"
```

```
        android:textStyle="normal" />

<android.support.design.widget.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/TextLabel"
    app:hintEnabled="false">

    <AutoCompleteTextView
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:drawableLeft="@drawable/ic_username"
        android:drawablePadding="10dp"
        android:hint="Email" or MatricNo"
        android:imeOptions="actionNext"
        android:inputType="textEmailAddress"
        android:maxLines="1"
        android:singleLine="true"
        android:textColor="@color/white" />

</android.support.design.widget.TextInputLayout>

<android.support.design.widget.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/TextLabel"
    app:hintEnabled="false">

    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:drawableLeft="@drawable/ic_password"
        android:drawablePadding="10dp"
        android:hint="Password"
        android:imeActionId="6"
        android:imeActionLabel="@string/action_sign_in_short"
        android:imeOptions="actionGo"
        android:inputType="textPassword"
```

```
        android:maxLines="1"
        android:singleLine="true"
        android:textColor="@color/white"
        android:theme="@style/TextLabel" />

</android.support.design.widget.TextInputLayout>

<Button
    android:id="@+id/btnLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="2dp"
    android:layout_marginLeft="5dp"
    android:layout_marginTop="30dp"

android:background="@drawable/custom_stylish_neutral_selector"
    android:clickable="true"
    android:paddingLeft="@dimen/desc_padding"
    android:paddingRight="@dimen/desc_padding"
    android:text="LOGIN"
    android:textColor="@color/white"
    android:textSize="22sp"
    android:textStyle="bold" />

<!--      Link      to      Login      Screen      -->
<Button
    android:id="@+id/bt_forgottenPassword"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="0dip"
    android:background="@null"
    android:visibility="gone"
    android:paddingLeft="@dimen/desc_padding"
    android:paddingRight="@dimen/desc_padding"
    android:text="Forgotten Password?"
    android:textAllCaps="false"
    android:textColor="#bdfff"
    android:textSize="10dp" />

<TextView
```

```
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dip"
        android:background="@null"
        android:gravity="center_horizontal"
        android:layout_gravity="center_horizontal"
        android:paddingLeft="@dimen/desc_padding"
        android:paddingRight="@dimen/desc_padding"
        android:text="By:          Egbirika          Gladys"
        android:textAllCaps="false"
        android:textColor="#bdfff"
        android:textSize="10dp"
    </LinearLayout>
</RelativeLayout>
</RelativeLayout>
```

Main Activity Layout XML

```
<?xml          version="1.0"          encoding="utf-8"?>
<android.support.v4.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/drawer_layout"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:fitsSystemWindows="true"
tools:openDrawer="start">

    <include
        layout="@layout/app_bar_main"
        android:layout_width="match_parent"
        android:layout_height="match_parent"

    <android.support.design.widget.NavigationView
        android:id="@+id/nav_view"
        android:layout_width="wrap_content"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:fitsSystemWindows="true"
        app:itemIconTint="#1a1a1a"
```

```
app:itemTextColor="#1a1a1a"  
app:headerLayout="@layout/nav_header_main"  
app:menu="@menu/activity_main_drawer" />
```

```
</android.support.v4.widget.DrawerLayout>
```

Add	New	User	Activity	Layout	XML
<?xml		version="1.0"		encoding="utf-8"?	>

```
<ScrollView
```

```
xmlns:android="http://schemas.android.com/apk/res/android"  
xmlns:app="http://schemas.android.com/apk/res-auto"  
xmlns:tools="http://schemas.android.com/tools"  
android:layout_width="fill_parent"  
android:layout_height="fill_parent"  
android:background="@color/colorPrimary"  
android:orientation="vertical"  
android:id="@+id/login_relative_layout"  
tools:ignore="ExtraText"  
android:focusableInTouchMode="true"  
tools:context=".AddUser">
```

```
<ScrollView
```

```
android:id="@+id/scrollpane"  
android:layout_width="fill_parent"  
android:layout_height="fill_parent">
```

```
<LinearLayout
```

```
android:layout_width="fill_parent"  
android:layout_height="wrap_content"  
android:layout_gravity="center"  
android:orientation="vertical"  
android:paddingLeft="25dp"  
android:paddingRight="25dp"
```

```
>
```

```
<ImageView
```

```
android:layout_width="120dp"  
android:layout_height="120dp"  
android:layout_gravity="center"  
android:layout_marginBottom="2dp"  
android:layout_marginTop="30dp"
```

```
        android:background="@drawable/circle"  
        android:padding="5dp"  
        android:src="@drawable/iaue_logo" />  
  
    <TextView  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:text="Register Admin / Lecturer"  
        android:layout_gravity="center"  
        android:fontFamily="sans-serif-condensed"  
        android:layout_marginBottom="5dp"  
        android:textColor="@android:color/darker_gray"  
        android:textSize="20dp"  
        android:textStyle="bold" />  
  
    <android.support.design.widget.TextInputLayout  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:theme="@style/TextLabel"  
        app:hintEnabled="true">  
  
        <AutoCompleteTextView  
            android:id="@+id/firstname"  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:hint="First Name"  
            android:imeOptions="actionNext"  
            android:textColor="@color/white"  
            android:drawableLeft="@drawable/ic_check_circle"  
            android:drawablePadding="10dp"  
            android:inputType="text"  
            android:maxLines="1"  
            android:singleLine="true" />  
  
    </android.support.design.widget.TextInputLayout>  
  
    <android.support.design.widget.TextInputLayout  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:theme="@style/TextLabel"
```

```
app:hintEnabled="true">
```

```
<AutoCompleteTextView  
  android:id="@+id/middle"  
  android:layout_width="match_parent"  
  android:layout_height="wrap_content"  
  android:hint="Middle" Name"  
  android:imeOptions="actionNext"  
  android:textColor="@color/white"  
  android:drawableLeft="@drawable/ic_check_circle"  
  android:drawablePadding="10dp"  
  android:inputType="text"  
  android:maxLines="1"  
  android:singleLine="true" />
```

```
</android.support.design.widget.TextInputLayout>
```

```
<android.support.design.widget.TextInputLayout  
  android:layout_width="match_parent"  
  android:layout_height="wrap_content"  
  android:theme="@style/TextLabel"  
  app:hintEnabled="true">
```

```
<AutoCompleteTextView  
  android:id="@+id/lastname"  
  android:layout_width="match_parent"  
  android:layout_height="wrap_content"  
  android:hint="Last" Name"  
  android:imeOptions="actionNext"  
  android:textColor="@color/white"  
  android:drawableLeft="@drawable/ic_check_circle"  
  android:drawablePadding="10dp"  
  android:inputType="text"  
  android:maxLines="1"  
  android:singleLine="true" />
```

```
</android.support.design.widget.TextInputLayout>
```

```
<Spinner  
  android:id="@+id/rank"  
  android:paddingTop="1dp"
```

```
    android:layout_weight="1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textColor="@color/white"
    android:layout_marginLeft="3dp"
    android:layout_marginRight="3dp"
    android:hint="Rank"
    android:entries="@array/Rank"
    android:popupBackground="@color/colorPrimary"
    android:paddingBottom="13dp"
    android:spinnerMode="dialog"
    android:drawSelectorOnTop="true"
    android:background="@drawable/spinner_selector" />
```

```
<android.support.design.widget.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/TextLabel"
    app:hintEnabled="true">
```

```
<AutoCompleteTextView
    android:id="@+id/email"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Email"
    android:imeOptions="actionNext"
    android:textColor="@color/white"
    android:drawableLeft="@drawable/ic_check_circle"
    android:drawablePadding="10dp"
    android:inputType="textEmailAddress"
    android:maxLines="1"
    android:singleLine="true" />
```

```
</android.support.design.widget.TextInputLayout>
```

```
<Spinner
    android:id="@+id/sex"
    android:paddingTop="1dp"
    android:layout_weight="1"
    android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"  
android:textColor="@color/white"  
android:layout_marginLeft="3dp"  
android:layout_marginRight="3dp"  
android:hint="Sex"  
android:entries="@array/Gender"  
android:popupBackground="@color/colorPrimary"  
android:paddingBottom="13dp"  
android:spinnerMode="dialog"  
android:drawSelectorOnTop="true"  
android:background="@drawable/spinner_selector" />
```

```
<android.support.design.widget.TextInputLayout  
android:layout_width="match_parent"  
android:layout_height="wrap_content"  
android:theme="@style/TextLabel"  
app:hintEnabled="true">
```

```
<AutoCompleteTextView  
android:id="@+id/phoneno"  
android:layout_width="match_parent"  
android:layout_height="wrap_content"  
android:hint="Phone Number"  
android:imeOptions="actionNext"  
android:textColor="@color/white"  
android:drawableLeft="@drawable/ic_check_circle"  
android:drawablePadding="10dp"  
android:inputType="phone"  
android:maxLines="1"  
android:singleLine="true" />
```

```
</android.support.design.widget.TextInputLayout>
```

```
<android.support.design.widget.TextInputLayout  
android:layout_width="match_parent"  
android:layout_height="wrap_content"  
android:theme="@style/TextLabel"  
app:hintEnabled="true">
```

```
<AutoCompleteTextView
```

```
        android:id="@+id/department"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Department"
        android:imeOptions="actionNext"
        android:textColor="@color/white"
        android:drawableLeft="@drawable/ic_check_circle"
        android:drawablePadding="10dp"
        android:inputType="text"
        android:maxLines="1"
        android:singleLine="true" />

</android.support.design.widget.TextInputLayout>

<android.support.design.widget.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/TextLabel"
    app:hintEnabled="true">

    <AutoCompleteTextView
        android:id="@+id/faculty"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Faculty"
        android:imeOptions="actionNext"
        android:textColor="@color/white"
        android:drawableLeft="@drawable/ic_check_circle"
        android:drawablePadding="10dp"
        android:inputType="text"
        android:maxLines="1"
        android:singleLine="true" />

</android.support.design.widget.TextInputLayout>

<android.support.design.widget.TextInputLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/TextLabel"
    app:hintEnabled="true">
```

```
<AutoCompleteTextView
  android:id="@+id/password"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:hint="Password"
  android:imeOptions="actionNext"
  android:textColor="@color/white"
  android:drawableLeft="@drawable/ic_check_circle"
  android:drawablePadding="10dp"
  android:inputType="textPassword"
  android:maxLines="1"
  android:singleLine="true" />

</android.support.design.widget.TextInputLayout>

<!-- Login Button -->
<LinearLayout
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:orientation="horizontal"
  android:paddingBottom="5dp"
  android:weightSum="2">
  <Button
    android:id="@+id/bt_user_cancel"
    android:layout_width="fill_parent"
    android:layout_weight="1"
    android:layout_height="wrap_content"
    android:paddingLeft="30dp"
    android:paddingRight="30dp"
    android:background="@drawable/custom_stylish_white_inverse_selector"
    android:layout_marginTop="10dp"
    android:layout_marginRight="5dp"
    android:text="Cancel"
    android:textColor="@color/white"
    android:textSize="14sp" />
  <Button
    android:id="@+id/bt_user_save"
```

```
        android:layout_marginLeft="5dp"
        android:layout_weight="1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dip"

android:background="@drawable/custom_stylish_neutral_selector"
        android:text="Save"
        android:paddingLeft="30dp"
        android:paddingRight="30dp"
        android:textSize="14sp"
        android:textColor="@color/white"
    />
</LinearLayout>

<!--      Link      to      Login      Screen      -->
</LinearLayout>

</ScrollView>

</ScrollView>

Students List Fragment Layout XML
<?xml      version="1.0"      encoding="utf-8"?>
<FrameLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:id="@+id/members_tab"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="5dp"
        android:layout_marginTop="5dp"
        android:layout_height="match_parent"
        tools:context=".fragments.StudentsFragment">

<android.support.v4.widget.SwipeRefreshLayout
        android:id="@+id/swipe_refresh_layout_students"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
```

```
<!-- TODO: Update blank fragment layout -->
<ListView
  android:id="@+id/list_students"
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  tools:listitem="@layout/list_row"
  android:divider="#b5b5b5"
  android:footerDividersEnabled="true"
  android:dividerHeight="1.5dp"
  android:saveEnabled="true"
 />

<!--android:divider="#b5b5b5"-->
<!--android:dividerHeight="1.5dp"-->
<!--android:footerDividersEnabled="true"-->

</android.support.v4.widget.SwipeRefreshLayout>
</FrameLayout>
User Profile Fragment Layout XML
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:fitsSystemWindows="true"
app:expandedTitleMarginStart="70dp"
>
<android.support.design.widget.AppBarLayout
  android:id="@+id/app_bar_layout_"
  android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
  android:layout_width="fill_parent"
  android:layout_height="215dp">

  <android.support.design.widget.CollapsingToolbarLayout
    android:id="@+id/CollapsingToolbarLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:titleEnabled="false"
```

```
app:layout_scrollFlags="scroll|exitUntilCollapsed"
app:contentScrim="?attr/colorPrimary"
app:expandedTitleMarginStart="48dp"
app:expandedTitleMarginEnd="64dp">

<android.support.v7.widget.Toolbar
    android:id="@+id/toolbar1"
    android:layout_width="match_parent"
    android:layout_height="170dp"
    app:popupTheme="@style/ThemeOverlay.AppCompat.Light"
    app:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
    app:layout_collapseMode="pin"
    android:minHeight="?attr/actionBarSize"/>

<include
    layout="@layout/content_view_profile_header"
    app:layout_collapseMode="parallax"/>

</android.support.design.widget.CollapsingToolbarLayout>

</android.support.design.widget.AppBarLayout>

<android.support.v4.widget.NestedScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:fillViewport="true"
    android:layout_below="@+id/About"
    app:layout_behavior="@string/appbar_scrolling_view_behavior">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:id="@+id/gg"
    android:weightSum="1">

<TextView
    android:layout_width="wrap_content"
    android:layout_below="@+id/app_bar_layout_"
```

```
        android:layout_above="@+id/gg"
        android:paddingTop="3dp"
        android:paddingBottom="3dp"
        android:paddingLeft="8dp"
        android:textColor="@color/colorPrimary"
        android:layout_gravity="left"
        android:gravity="center_horizontal"
        android:textStyle="normal"
        android:text="About"
    Me"

    android:textAppearance="@style/TextAppearance.AppCompat.Headline"
"
        android:layout_height="wrap_content"
    />

<TableLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:divider="@drawable/horizontal_divider"
    android:showDividers="middle"
    android:layout_weight="0.67">

    <TableRow        android:divider="@drawable/verticle_divider"
        android:showDividers="middle">

    </TableRow>

    <TableRow        android:divider="@drawable/verticle_divider"
        android:showDividers="none">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="10dp"
        android:drawableLeft="@drawable/ic_email"
        android:drawableTint="@color/colorPrimary"
        android:drawablePadding="5dp"
        android:background="@color/row_activated"
        android:textColor="@color/black"
        android:text="Email"
        android:textStyle="bold"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small" />
```

```
<TextView  
    android:id="@+id/p_email"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:text="Egbirikagladys@yahoo.com"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small" />
```

```
</TableRow>
```

```
<TableRow    android:divider="@drawable/verticle_divider"  
    android:showDividers="none">
```

```
<TextView  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:drawableLeft="@drawable/ic_phone"  
    android:drawableTint="@color/colorPrimary"  
    android:drawablePadding="5dp"  
    android:background="@color/row_activated"  
    android:textColor="@color/black"  
    android:text="Phone" No"  
    android:textStyle="bold"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small" />
```

```
<TextView  
    android:id="@+id/p_phoneno"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:text="+2347014083522"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small"
```

```
ll" />
```

```
</TableRow>
```

```
<TableRow android:divider="@drawable/verticle_divider"  
android:showDividers="none">
```

```
<TextView  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:padding="10dp"  
android:drawableLeft="@drawable/ic_gender"  
android:drawableTint="@color/colorPrimary"  
android:drawablePadding="5dp"  
android:background="@color/row_activated"  
android:textColor="@color/black"  
android:text="Sex"  
android:textStyle="bold"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Sma  
ll" />
```

```
<TextView  
android:id="@+id/p_gender"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:padding="10dp"  
android:text="Female"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Sma  
ll" />
```

```
</TableRow>
```

```
<TableRow android:divider="@drawable/verticle_divider"  
android:showDividers="none">
```

```
<TextView  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"
```

```
android:padding="10dp"  
android:text="Faculty"  
android:drawableLeft="@drawable/ic_school"  
android:drawableTint="@color/colorPrimary"  
android:drawablePadding="5dp"  
android:background="@color/row_activated"  
android:textColor="@color/black"  
android:textStyle="bold"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small" />
```

```
<TextView  
    android:id="@+id/p_faculty"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:text="Science"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small" />
```

```
</TableRow>
```

```
<TableRow    android:divider="@drawable/verticle_divider"  
    android:showDividers="none">
```

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="10dp"  
    android:text="Department"  
    android:drawableLeft="@drawable/ic_school"  
    android:drawableTint="@color/colorPrimary"  
    android:drawablePadding="5dp"  
    android:background="@color/row_activated"  
    android:textColor="@color/black"  
    android:textStyle="bold"
```

```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small"
```



```
android:textAppearance="@style/Base.TextAppearance.AppCompat.Small"
    android:visibility="invisible" />

</TableRow>
<TableRow android:divider="@drawable/verticle_divider"
    android:showDividers="middle">
</TableRow>

</TableLayout>
</LinearLayout>

</android.support.v4.widget.NestedScrollView>

</android.support.design.widget.CoordinatorLayout>
```

LoginActivity Java Class

```
package samzytechsolutions.com.studentsprojectallocationapproval;

import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.KeyEvent;
import android.view.View;
import android.view.inputmethod.EditorInfo;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RelativeLayout;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.JsonObjectRequest;
```

```
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.HashMap;
import java.util.Map;

import samzytechsolutions.com.studentsprojectallocationapproval.config.AppCo
nfig;
import samzytechsolutions.com.studentsprojectallocationapproval.config.Genera
lController;
import samzytechsolutions.com.studentsprojectallocationapproval.config.PrefMa
nager;
import samzytechsolutions.com.studentsprojectallocationapproval.config.Reques
tQueueSingleton;
import samzytechsolutions.com.studentsprojectallocationapproval.config.SQLite
Handler;
import samzytechsolutions.com.studentsprojectallocationapproval.model.User;

public class LoginActivity extends Activity {
    RelativeLayout LoginLayout;
    private Button btnLogin;
    private Button btnForgottenPassword,btnProgressDialogTwo;
    private EditText inputUsername;
    private EditText inputPassword;
    private SQLiteHandler db;
    private PrefManager session;
    //////////////////////////////////////
    private ProgressDialog pDialog;
    RequestQueue requestQueue;
    static final int INTERNET_REQ = 23;
```

```
static final String REQ_TAG = "VACTIVITY";
////////////////////////////////////
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_login);
    inputUsername = (EditText) findViewById(R.id.username);
    inputPassword = (EditText) findViewById(R.id.password);
    btnLogin = (Button) findViewById(R.id.btnLogin);
    btnForgottenPassword = (Button)
findViewById(R.id.bt_forgottenPassword);
    //////////////////////////////////////
    // Progress dialog
    progressDialog = new ProgressDialog(this);
    progressDialog.setCancelable(false);
    requestQueue =
RequestQueueSingleton.getInstance(this.getApplicationContext()).getReq
uestQueue();
    //////////////////////////////////////
    btnLogin.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if
            (GeneralController.isNetworkAvailable(getApplicationContext())){
                STUDENT_LOGIN(getApplicationContext());
            }
        }
    });
    btnForgottenPassword.setOnClickListener(new
View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent(getApplicationContext(),
ForgottenPassword.class));
        }
    });
    inputPassword.setOnEditorActionListener(new
```

```
TextView.OnEditorActionListener() {  
  
    @Override  
    public boolean onEditorAction(TextView v, int actionId, KeyEvent  
event) {  
        if(actionId == EditorInfo.IME_ACTION_GO){  
  
            btnLogin.performClick();  
  
        }  
  
        return false;  
    }  
};  
  
}  
  
public void STUDENT_LOGIN(final Context context) {  
    // SQLite database handler  
    db = new SQLiteHandler(context);  
    // Session manager  
    session = new PrefManager(context);  
  
    GeneralController.ShowDialog(pDialog, "Please Wait", "Logging in...");  
  
    String URL =  
(inputUsername.getText().toString().trim().contains("@")) ?  
AppConfig.URL_STAFF_LOGIN : AppConfig.URL_STUDENT_LOGIN;  
  
    //Toast.makeText(context, URL, Toast.LENGTH_SHORT).show();  
    StringRequest stringRequest = new  
StringRequest(Request.Method.POST, URL,  
    new Response.Listener<String>() {  
        @Override  
        public void onResponse(String response) {  
            GeneralController.hideDialog(pDialog);  
            //Toast.makeText(context, response,  
Toast.LENGTH_SHORT).show();  
            if(response.isEmpty()){
```

```
Toast.makeText(context, "Incorrect Username or  
Password!", Toast.LENGTH_SHORT).show();  
}else {  
  
try {  
JSONArray obj = new JSONArray(response);  
JSONObject userJson = obj.getJSONObject(0);  
  
User u = new User();  
u.setFirstName(userJson.getString("firstname"));  
u.setMiddleName(userJson.getString("middlename"));  
u.setLastName(userJson.getString("lastname"));  
u.setEmail(userJson.getString("email"));  
u.setSex(userJson.getString("sex"));  
u.setPhoneNo(userJson.getString("phone"));  
u.setDepartment(userJson.getString("department"));  
u.setFaculty(userJson.getString("faculty"));  
u.setPassword(userJson.getString("password"));  
try {  
if(userJson.getString("matricno")==null){  
u.setRank(userJson.getString("rank"));  
}else{  
u.setSession(userJson.getString("session"));  
u.setSupervisor(userJson.getString("supervisor"));  
u.setMatricNo(userJson.getString("matricno"));  
u.setRank("Student");  
}  
}catch (JSONException e){  
u.setRank(userJson.getString("rank"));  
}  
  
if (session.MemberLogin(u)){  
  
startActivity(new Intent(getApplicationContext(),  
MainActivity.class));  
  
Toast.makeText(context, "Logged in as  
"+session.getMemberRole(), Toast.LENGTH_SHORT).show();  
}
```

```
        } catch (JSONException e) {
            Toast.makeText(context, "Incorrect Username or
Password!", Toast.LENGTH_SHORT).show();
            e.printStackTrace();
        }
    }

    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            GeneralController.hideDialog(pDialog);
            Toast.makeText(context, "Network Error... Try Again
"+error.getMessage(), Toast.LENGTH_SHORT).show();
        }
    });
    protected Map<String, String> getParams() throws
com.android.volley.AuthFailureError {
        Map<String, String> params = new HashMap<String, String>();

        params.put("password",inputPassword.getText().toString().trim());

        params.put("matricno",inputUsername.getText().toString().trim());
        return params;
    };
    stringRequest.setTag(REQ_TAG);
    requestQueue.add(stringRequest);
}
}
```

MainActivity Java Class

package samzytechsolutions.com.studentsprojectallocationapproval;

import android.annotation.SuppressLint;

```
import android.content.Intent;
import android.graphics.Typeface;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentTransaction;
import android.support.v4.view.MenuItemCompat;
import android.view.Gravity;
import android.view.View;
import android.support.design.widget.NavigationView;
import android.support.v4.view.GravityCompat;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuItem;
import android.view.ViewGroup;
import android.widget.LinearLayout;
import android.widget.TextView;
```

```
import
samzytechsolutions.com.studentsprojectallocationapproval.config.PrefMa
nager;
```

```
import
samzytechsolutions.com.studentsprojectallocationapproval.config.SQLite
Handler;
```

```
import
samzytechsolutions.com.studentsprojectallocationapproval.fragments.Ad
minsFragment;
```

```
import
samzytechsolutions.com.studentsprojectallocationapproval.fragments.Ap
provedProjectsFragment;
```

```
import
```

```
samzytechsolutions.com.studentsprojectallocationapproval.fragments.LecturersFragment;
```

```
import
```

```
samzytechsolutions.com.studentsprojectallocationapproval.fragments.MyProfile;
```

```
import
```

```
samzytechsolutions.com.studentsprojectallocationapproval.fragments.PendingProjectsFragment;
```

```
import
```

```
samzytechsolutions.com.studentsprojectallocationapproval.fragments.StudentsFragment;
```

```
public class MainActivity extends AppCompatActivity  
    implements NavigationView.OnNavigationItemSelectedListener {  
    TextView admins_count, lecturers_count, students_count,  
approved_projects_count, pending_projects_count, Nav_Name,  
Nav_Regnumber;  
    private boolean viewIsAtHome;  
    public static int navItemIndex = 0;  
    FloatingActionButton fab;  
    NavigationView navigationView;  
    private View navHeader;  
  
    private SQLiteHandler db;  
    private PrefManager session;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
        setSupportActionBar(toolbar);  
  
        db = new SQLiteHandler(getApplicationContext());  
        // Session manager
```

```
session = new PrefManager(getApplicationContext());

toolbar.setSubtitle("Students Project Allocation & Approval");

NavigationView navigationView = (NavigationView)
findViewById(R.id.nav_view);
navHeader = navigationView.getHeaderView(0);
navigationView.setNavigationItemSelectedListener(this);

Nav_Name = (TextView) navHeader.findViewById(R.id.nav_name);
Nav_Regnumber = (TextView)
navHeader.findViewById(R.id.textView);

try {
    Nav_Name.setText(session.getMemberFullName());
    Nav_Regnumber.setText(session.getRegNumber());
} catch (Exception e){
    e.printStackTrace();
}

fab = (FloatingActionButton) findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        Snackbar snack1 = Snackbar.make(view, "Do you want to
logout?",
Snackbar.LENGTH_LONG)
.setAction("Yes", new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(getApplicationContext(),
LoginActivity.class);
startActivity(intent);
```

```
        finish();
    }
    });
    snack1.show();

}
});

DrawerLayout drawer = (DrawerLayout)
findViewById(R.id.drawer_layout);
ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
    this, drawer, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
drawer.addDrawerListener(toggle);
toggle.syncState();

//NAV BADGES
admins_count=(TextView)
MenuItemCompat.getActionView(navigationView.getMenu().findItem(R.i
d.nav_admins));
lecturers_count=(TextView)
MenuItemCompat.getActionView(navigationView.getMenu().findItem(R.i
d.nav_lecturers));
students_count=(TextView)
MenuItemCompat.getActionView(navigationView.getMenu().findItem(R.i
d.nav_students));
approved_projects_count=(TextView)
MenuItemCompat.getActionView(navigationView.getMenu().findItem(R.i
d.nav_approved_projects));
pending_projects_count=(TextView)
MenuItemCompat.getActionView(navigationView.getMenu().findItem(R.i
d.nav_pending_projects));

InitializeBadges(admins_count);
InitializeBadges(lecturers_count);
```

```
InitializeBadges(students_count);
InitializeBadges(approved_projects_count);
InitializeBadges(pending_projects_count);

DisplayBadges();

displayView(R.id.nav_profile);
navigationView.getMenu().getItem(0).setChecked(true);

hideItem();
}

@Override
public void onBackPressed() {
    DrawerLayout drawer = (DrawerLayout)
findViewById(R.id.drawer_layout);
    if (drawer.isDrawerOpen(GravityCompat.START)) {
        drawer.closeDrawer(GravityCompat.START);
    } else {
        //super.onBackPressed();
    }

    /*
    if (!viewIsAtHome) { //if the current view is not the News fragment
        displayView(R.id.nav_profile); //display the News fragment
    } else {
        moveTaskToBack(true); //If view is in News fragment, exit
application
    }
    */
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
```

```
if (viewIsAtHome) {
    getMenuInflater().inflate(R.menu.main, menu);
}

// if (navItemIndex==1) {
//     getMenuInflater().inflate(R.menu.chemicals_menu_search, menu);
// }

return true;

}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_change_password) {
        startActivity(new Intent(MainActivity.this,
ChangePassword.class));
    }

    if (id == R.id.action_logout) {
        Intent intent = new Intent(getBaseContext(), LoginActivity.class);
        startActivity(intent);
        finish();
    }
    return super.onOptionsItemSelected(item);
}
```

```
@SuppressWarnings("StatementWithEmptyBody")
@Override
public boolean onNavigationItemSelected(MenuItem item) {
    int id = item.getItemId();
    displayView(item.getItemId());
    toggleFab();
    return true;
}
@SuppressLint("NewApi")
private void InitializeBadges(Textview t) {
    LinearLayout.LayoutParams lp = new
LinearLayout.LayoutParams(ViewGroup.LayoutParams.WRAP_CONTENT,
    ViewGroup.LayoutParams.WRAP_CONTENT);
    //lp.weight = 1.0f;
    lp.gravity = Gravity.CENTER_VERTICAL;
    lp.topMargin=17;
    lp.width=40;
    lp.height=40;
    //Gravity property aligns the text
    t.setGravity(Gravity.CENTER);
    t.setTypeface(null, Typeface.BOLD);
    t.setTextSize(10);
    t.setTextColor(getResources().getColor(R.color.black));

t.setBackground(getResources().getDrawable(R.drawable.item_count));
    t.setLayoutParams(lp);
}
public void displayView(int viewId) {
    Fragment fragment = null;
    String title = getString(R.string.app_name);
    switch (viewId) {
        case R.id.nav_profile:
            fragment = new MyProfile();
            title = "My Profile";
            viewsAtHome = true;
    }
}
```

```
        navItemIndex=0;
        break;
    case
        fragment          =          new          R.id.nav_admins:
        title              =              "Administrators";
        viewIsAtHome      =              false;
        navItemIndex=1;
        break;
    case
        fragment          =          new          R.id.nav_lecturers:
        title              =              "Lecturers";
        viewIsAtHome      =              false;
        navItemIndex=2;
        break;
    case
        fragment          =          new          R.id.nav_students:
        title              =              "Students";
        viewIsAtHome      =              false;
        navItemIndex=3;
        break;
    case
        fragment          =          new          R.id.nav_approved_projects:
        title              =          "Approved          Projects";
        viewIsAtHome      =              false;
        navItemIndex=4;
        break;
    case
        fragment          =          new          R.id.nav_pending_projects:
        title              =          "Pending          Projects";
        viewIsAtHome      =              false;
        navItemIndex=5;
        break;
    case
        startActivity(new          R.id.nav_my_project:
        ViewProjectDetails.class));          Intent(MainActivity.this,
```

```
//drawer.closeDrawers();
break;
case R.id.nav_change_password:
startActivity(new Intent(MainActivity.this,
ChangePassword.class));
//drawer.closeDrawers();
break;
case R.id.nav_logout:
Intent intent = new Intent(getBaseContext(), LoginActivity.class);
startActivity(intent);
finish();
break;
}

if (fragment != null) {
FragmentTransaction ft =
getSupportFragmentManager().beginTransaction();
ft.replace(R.id.content_frame, fragment);
ft.commit();
}

// set the toolbar title
if (getSupportActionBar() != null) {
getSupportActionBar().setTitle(title);
}

DisplayBadges();

DrawerLayout drawer = (DrawerLayout)
findViewById(R.id.drawer_layout);
drawer.closeDrawer(GravityCompat.START);
}

// show or hide the fab
private void toggleFab() {
```

```
if (viewIsAtHome)
    fab.show();
else
    fab.hide();
}

private void hideItem()
{
    navigationView = (NavigationView) findViewById(R.id.nav_view);
    Menu nav_Menu = navigationView.getMenu();

    String User = session.getMemberRole();
    if(User.equalsIgnoreCase("Student")){
        nav_Menu.findItem(R.id.nav_admins).setVisible(false);
        nav_Menu.findItem(R.id.nav_students).setVisible(false);
        nav_Menu.findItem(R.id.nav_lecturers).setVisible(false);
        nav_Menu.findItem(R.id.nav_pending_projects).setVisible(false);
        nav_Menu.findItem(R.id.nav_approved_projects).setVisible(false);
    }else if(User.equalsIgnoreCase("Lecturer")){
        nav_Menu.findItem(R.id.nav_admins).setVisible(false);
        nav_Menu.findItem(R.id.nav_students).setVisible(false);
        nav_Menu.findItem(R.id.nav_lecturers).setVisible(false);
        nav_Menu.findItem(R.id.nav_my_project).setVisible(false);
    }else if(User.equalsIgnoreCase("Administrator")){
        nav_Menu.findItem(R.id.nav_pending_projects).setVisible(false);
        nav_Menu.findItem(R.id.nav_approved_projects).setVisible(false);
        nav_Menu.findItem(R.id.nav_my_project).setVisible(false);
    }else{
        nav_Menu.findItem(R.id.nav_admins).setVisible(false);
        nav_Menu.findItem(R.id.nav_students).setVisible(false);
        nav_Menu.findItem(R.id.nav_lecturers).setVisible(false);
        nav_Menu.findItem(R.id.nav_pending_projects).setVisible(false);
        nav_Menu.findItem(R.id.nav_approved_projects).setVisible(false);
    }
}
```

```
        nav_Menu.findItem(R.id.nav_my_project).setVisible(false);

    }

}

void DisplayBadges(){
    try {

admins_count.setText(String.valueOf(session.getAdministratorsCount()));
;

lecturers_count.setText(String.valueOf(session.getLecturersCount()));

students_count.setText(String.valueOf(session.getStudentsCount()));

    }catch (Exception e){
        e.printStackTrace();
    }

    try {

approved_projects_count.setText(session.getApprovedProjectCount());

pending_projects_count.setText(session.getPendingProjectCount());

    }catch (Exception e){
        e.printStackTrace();
    }

}

}
```

Students ListView Adapter Java Class

```
package
samzytechsolutions.com.studentsprojectallocationapproval.config;

import android.content.Context;
import android.util.SparseBooleanArray;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.ImageView;
import android.widget.TextView;

import java.util.List;
import samzytechsolutions.com.studentsprojectallocationapproval.R;
import samzytechsolutions.com.studentsprojectallocationapproval.model.User;
public class LazyAdapter extends ArrayAdapter<User> {
    Context context;
    LayoutInflater inflater;
    List<User> worldpopulationlist;
    private SparseBooleanArray mSelectedItemsIds;
    // ImageLoader imageLoader;
    public LazyAdapter(Context context, int resourceId,
        List<User> worldpopulationlist) {
        super(context, resourceId, worldpopulationlist);
        mSelectedItemsIds = new SparseBooleanArray();
        this.context = context;
        this.worldpopulationlist = worldpopulationlist;
        inflater = LayoutInflater.from(context);
        //imageLoader = new ImageLoader(context);
    }
    private class ViewHolder {
        TextView title;
        TextView artist;
        TextView duration;
        TextView phone;
    }
}
```

```
        ImageView                                thumb_image;
    }
    public View getView(int position, View view, ViewGroup parent) {
        final ViewHolder holder;
        if (view == null) {
            holder = new ViewHolder();

            view = inflater.inflate(R.layout.list_row, null);
            // Locate the TextViews in listview_item.xml
            holder.title = (TextView)view.findViewById(R.id.title); // title
            holder.artist = (TextView)view.findViewById(R.id.artist); // artist
            name-> Dept || Email
            holder.duration = (TextView)view.findViewById(R.id.duration); //
            duration ->Gender
            holder.phone = (TextView)view.findViewById(R.id.phone); //
            phone

            holder.thumb_image=(ImageView)view.findViewById(R.id.list_image);
            // thumb image
            view.setTag(holder);
        } else {
            holder = (ViewHolder) view.getTag();
        }
        // Capture position and set to the TextView
        User song = new User();
        song = worldpopulationlist.get(position);
        // Setting all values in listview
        holder.title.setText(song.getFirstName()+"
"+song.getMiddleName()+" "+song.getLastName());
        holder.artist.setText(song.getFaculty()+" ||
"+song.getDepartment());
        holder.duration.setText(song.getPhoneNo());
        holder.phone.setText(song.getSupervisor()+" -->
"+song.getProject());

        // circleImage(holder.thumb_image, song.getImagePath(), false);
        //imageLoader.DisplayImage(song.getImagePath(),
holder.thumb_image);
        return holder;
    }
}
```

```
@Override
public void remove(User object) {
    worldpopulationlist.remove(object);
    notifyDataSetChanged();
}

public List<User> getWorldPopulation() {
    return worldpopulationlist;
}

public void toggleSelection(int position) {
    selectView(position, !mSelectedItemsIds.get(position));
}

public void removeSelection() {
    mSelectedItemsIds = new SparseBooleanArray();
    notifyDataSetChanged();
}

public void selectView(int position, boolean value) {
    if (value)
        mSelectedItemsIds.put(position, value);
    else
        mSelectedItemsIds.delete(position);
    notifyDataSetChanged();
}

public int getSelectedCount() {
    return mSelectedItemsIds.size();
}

public SparseBooleanArray getSelectedIds() {
    return mSelectedItemsIds;
}
}
```