

THAKUR DEGREE COLLEGE OF SCIENCE AND COMMERCE

KANDIVALI (EAST)

MUMBAI

A PROJECT REPORT ON

Cloud Based Free E-Learning Application

For

Thakur College of Science and Commerce

By

OJHA ABHISHEK DEVMANI

Submitted in partial fulfilment of
Master of Science (Computer Science)

[UNIVERSITY OF MUMBAI]

Thakur Degree College of Science and Commerce Kandivali (East), Mumbai.

ACADEMIC YEAR 2021-2022



DATE: 28-03-2022

COMPUTER SCIENCE DEPARTMENT (2021-2022)

CERTIFICATE OF APPROVAL

This is to certify that the project work entitled “**Cloud based Free E-Learning Application**” is prepared by **Ojha Abhishek Devmani** a student of “**Masters of Science (Computer Science) Part1**” course of University of Mumbai, which is conducted by our college.

This is the original study work and important sources used have been duly acknowledged in the report. The report is submitted in partial fulfilment of M.Sc. (Computer Science) course as per rules of University of Mumbai.

PROJECT GUIDE	HEAD OF DEPARTMENT	EXTERNAL EXAMINER

DECLARATION

I / We hereby declare that the project entitled, “**Cloud based Free E-Learning Application** ” done at THAKUR DEGREE COLLEGE OF SCIENCE AND COMMERCE , has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfilment of the requirement for the award of degree of MASTER OF SCIENCE (COMPUTER SCIENCE) to be submitted as semester II Mini project as part of our curriculum.

OJHA ABHISHEK DEVMANI

ACKNOWLEDGEMENT

Achievement is finding out what you would be doing rather than what you have to do. It is not until you undertake such a project that you realize how much effort and hard work it really is, what are your capabilities and how well you can present yourself or other things. It gives me immense pleasure to present this report towards the fulfillment of my project.

It has been rightly said that we are built on the shoulder of others. For everything I have achieved, the credit goes to all those who had helped me to complete this project successfully.

I take this opportunity to express my profound gratitude to management of Thakur Degree College of Science & Commerce for giving me this opportunity to accomplish this project work.

I am very much thankful to **Mrs. C. T. Chakraborty** - Principal of Thakur College for their kind co-operation in the completion of my project.

A special vote of thanks to our HOD **Mr. Ashish Trivedi** .Thanks to all our teachers for helping and guiding me throughout my project.

Finally, I would like to thank all my friends & entire Computer Science department who directly or indirectly helped me in completion of this project & to my family without whose support, motivation & encouragement this would not have been possible.

(OJHA ABHISHEK DEVMANI)

THAKUR DEGREE COLLEGE OF SCIENCE AND COMMERCE

Department of Computer Science

2021-2022

Students Name: OJHA ABHISHEK DEVMANI

Project Name: Cloud Based Free E-Learning Application.

College Name: Thakur College of Science and Commerce

PHASES	EXPECTED DATE OF COMPLETION	ACTUAL DATE OF COMPLETION	SIGNATURE
Preliminary Investigation	28/03/22	28/03/22	
System Analysis	28/03/22	28/03/22	
System Designing	28/03/22	28/03/22	
System Coding	28/03/22	28/03/22	
System Implementation	28/03/22	28/03/22	
Report Submission	28/03/22	28/03/22	

Organizational Overview

The **Thakur College of Science and Commerce (TCSC)** is a college in Kandivali in Mumbai of Maharashtra, India running by Thakur Educational Trust.

Thakur College was started in 1992 to serve the needs of students passing SSC examination from the schools around Kandivali area and Thakur Vidhya Mandir which has already established itself as one of the schools in the area. It offers courses at primarily the higher secondary and under-graduate levels. The courses at the undergraduate and post- graduate level are offered in affiliation with Mumbai University, Mumbai. An ISO 9001:2008 College with **A** grade as assessed by the National Assessment and Accreditation Council NAAC.

Name:	Thakur College of Science and Commerce
Founded:	1997
Address:	Thakur College of Science and Commerce, Thakur Village Kandivali(E), Mumbai – 400 001
Motto:	Journey towards Excellence
Total Staff:	200
Number of Students:	12500
Email:	Helpdesk@tcsc.org.in

INDEX

CHAPTER 01: INTRODUCTION	1
CHAPTER 02: PROBLEM STATEMENT	2
CHAPTER 03: THEORY, FIGURE, DIAGRAM	3
CHAPTER 04: LITERATURE REVIEW	7
CHAPTER 05: ANALYSIS AND DESIGN OF PROJECT	10
CHAPTER 06: IMPLEMENTATION OF PROJECT	32
CHAPTER 07: RESULT OF PROJECT	33
CHAPTER 08: CONCLUSION	53
CHAPTER 09: FUTURE SCOPE	53
CHAPTER 10: REFERENCES	55
LIST OF TABLES	
Activity Diagram Symbol	23
LIST OF FIGURES	
Fig 1: Architecture E- Learning Application	2
Fig 2: Connection Between E-Learning Application	3
Fig 3: Data Flow Diagram	14
Fig 4: Entity Relationship Diagram	15
Fig 5: ER Diagram ..	16
Fig 6: Use Case Diagram for User App of Free E-Learning Course Application	20
Fig 7: Use Case Diagram for User App of Free E-Learning Course Application ...	21
Fig 8: Activity Diagram Tables	24
Fig 9: Activity Diagram of Free E-Learning Application	24
Fig 10: Class Diagram of Free E-Learning Application	29
Fig 11: Sequence Diagram of Free E-Learning Application	30
Fig 12: Admin App Code Main.dart	31
Fig 13: Admin App Code Pubspec.yaml (App Configuration Files)	31
Fig 14: Snapshots of Admin Panels	32
Fig 15: Snapshots of User App	38
Fig 16: Overall User acquisitions Report	48

Introduction

Cloud based Free E-learning Application is an education via the Internet, network, or standalone computer. E-learning Application is basically the network-enabled convey of skills and knowledge. E-learning refers to using electronic applications and processes to learn. E-learning includes all forms of electronically supported learning and teaching. The information and communication systems, whether networked learning or not, serve as specific media to implement the learning process. This often involves both out-of-classroom and in-classroom educational experiences via technology, even as advances continue in regard to devices and curriculum. E-learning is the computer and network-enabled transfer of skills and knowledge. E-learning applications and processes include android-based learning, computer-based learning, virtual education opportunities and digital collaboration. Content is delivered via the Internet, intranet/extranet, audio or video tape, satellite TV, and CD-ROM. That is to say E-learning systems contain both Learning Management System and Course management system. It can be self-paced or instructor-led and includes media in the form of text, image, animation, streaming video and audio. It is commonly thought that new technologies can make a big difference in education. In young ages especially, children can use the huge interactivity of new media, and develop their skills, knowledge, and perception of the world, under their parents' monitoring, and totally free of course.

Many proponents of e-learning believe that everyone must be equipped with basic knowledge in technology, as well as use it as a medium to reach a particular goal and aim. In the 21st century, we have moved from the Industrial Age through the Information Age and now to the Knowledge Age. Knowledge and its efficient management constitute the key to success and survival for organizations in the highly dynamic and competitive world of today. Efficient acquisition, storage, transfer, retrieval, application, and visualization of knowledge often distinguish successful organizations from the unsuccessful ones. The ability to obtain, assimilate, and apply the right knowledge effectively will become a key skill in the next century. Learning is the key to achieving our full potential. Our survival in the 21st century as individuals, organizations, and nations will depend upon our capacity to learn and the application of what we learn to our daily lives. E-learning has the potential to transform how and when employees learn. Learning will become more integrated with work and will use shorter, more modular, just-in-time delivery systems. By leveraging workplace technologies, e-learning is bridging the gap between learning and work. Workers can integrate learning into work more effectively because they use the same tools and technology for learning as they use for work. Both employers and employees recognize that E-learning will diminish the narrowing gap between work and home, and between work and learning. E-learning is an option to any organization looking to improve the skills and capacity of its employees. With the rapid change in all types of working environments, especially medical and healthcare environments, there is a constant need to rapidly train and retrain people in new technologies, products, and services found within the environment. There is also a constant and unrelenting need for appropriate management and leveraging of the knowledge base so that it is readily available and accessible to all stakeholders within the workplace environment.

Problem Statement

With the advent in technology and with the perpetual increase in the strength of the students and the number of departments in the educational institutions, it is laborious to exchange the study materials between the students and the faculties.

To help the students get over the traditional methods of learning and make them accustomed to the internet where the notes for their respective subjects are easily available. It provides an automation procedure of studying the notes online. The implementation of this project helps both the students and the teachers. The teachers can upload their notes on to the website by using their unique ID and the students can gain access to these notes by searching for the name of the file under their respective department.

This project not only helps to facilitate easier access to notes for the students but also helps cutting down on expenditure for the universities as well. Students and Universities alike spend a considerable amount of money on printing costs which can be prevented.

This project is implemented on Android to facilitate easier access on a popular medium. The project uses Flutter Framework for Native App Development to write scripts which provide the app its functionality. Dart Programming Language uses to write the application logic and Google Firebase used for Cloud Storage . All contents are stored in an online database present on Google Firebase Firestore Cloud Storage Platform. The GUI components of the application are developed using Flutter Android based Framework . Digital Watermarking is used to identify original ownership of any document in the database and prevent any plagiarism from occurring.

A test on beta version of application was carried out by taking reviews of few students and teachers. The metrics of the application such as uploading rate, downloading rate, time taken to search for file, time taken for authentication, time taken for sign up have been measured and checked to ensure good performance. To prevent copyright infringement, a system has been implemented that imitates plagiarism checking within the Notehub server.

E-Learning is an inexpensive, efficient and comfortable way for students to easily access notes and an easier alternative to study new Technology around them.

Theory with figure / equation / Algorithm

Architecture of E-learning Application

The main requirement of the developers while developing any e-learning system is that it should be easy to use and safe. Students should be able to choose their course, find the resources easily, track their progress and reach their final goal on time. In the learning system students should be able to communicate and help each other, in other words it should be social and engaging. While developing the learning system the designer should focus on creating a perfect mix of traditional and virtual learning methods to help the students engage more and reach their goals

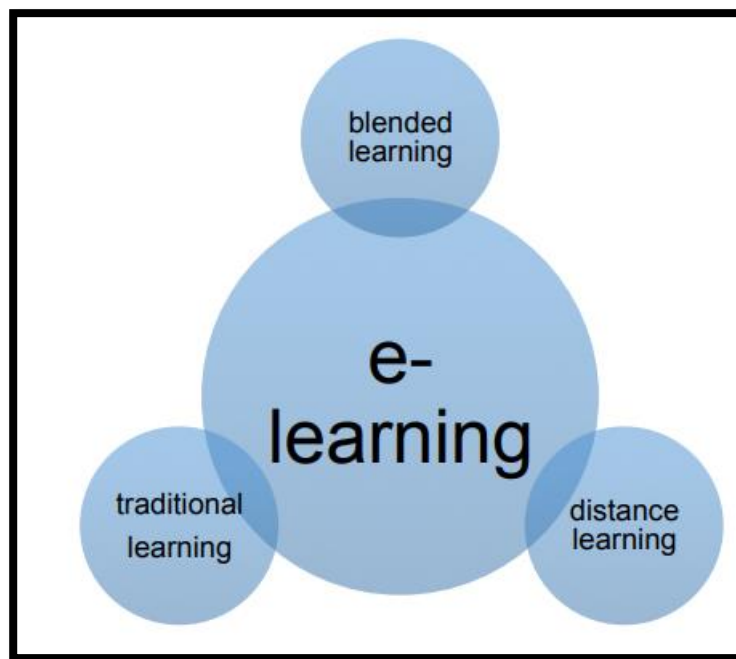


Fig 1: Architecture E- Learning Application.

A perfect e-learning system is a mix of all the learning systems shown in figure 2. In theory the perfect e-learning system will allow students to learn from their home while having some real-time support and interaction with other students who are taking the same courses however, it might be difficult to achieve in real cases.

The architecture of an e-learning system might be difficult to understand in technical term but it usually consists of three parts:

- User Interface
- Platform
- Database

A database is the place where the digital learning materials are stored and a learning platform is the media from which the students access the learning materials. A user interface is what

the student sees while accessing the platforms. The communication process between these three parts is very important for the whole system to work perfectly. For example, we can use MySQL to store the digital materials, use edX platform to access the materials and students can visit the edX platform using their web browser.

The connection between the three parts of the e-learning system is shown in figure

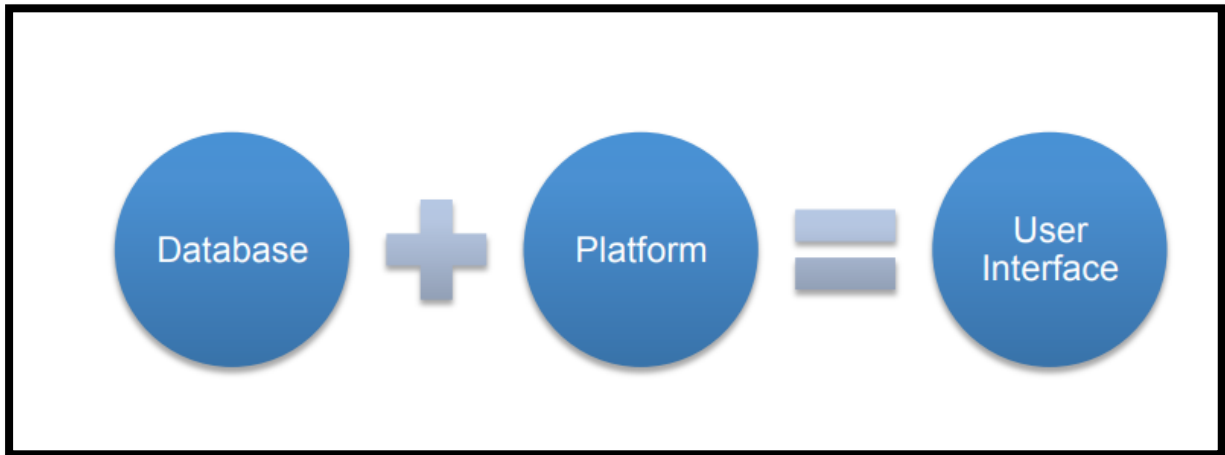


Fig 2: Connection Between E-Learning Application.

As we can see from the figure, the user interface is what the student sees while accessing the platform, which will have the access to the database.

The user interface is the means by which the student interacts with the system, including the input devices and the software. It allows user to input their command through input devices such as a mouse or touch screen and shows the effects of the user input. Applications and web browsers are some examples of the user interfaces.

A database is a structured set of information or data organised so that it can be accessed, managed and updated whenever necessary. In order to store, modify, extract

and administer the database a Application, database Application (DBMS) is required. There are many database Applications available in the market like Oracle, Microsoft SQL Server, and MySQL

The e-learning platform is the system that provides the access to the databases of E-learning materials or the online courses. The platforms are generally web-based. It enables the user to access the study materials, take tests, and track their progress whenever or wherever they want. There are many such platforms available in the market. Companies can either use such platforms or develop their own. Developing such platforms may be expensive, so companies generally use already available platforms. The platforms can be free (open source) or commercial. The commercial platforms will let the companies to modify them to fit the needs of the users. While choosing the platform it is necessary for the companies to understand what they want. It depends on a number of factors such as needs of students and technical skills of instructors.

As the world is moving towards the digital age and more and more universities are trying to maximize the use of technology in education, one must consider both the advantages and disadvantages of doing so. Although the use of technology in the field of education has revolutionised the way we learn there are still some downsides of relying completely on technology to learn. E-learning systems are the storehouses of information, trainings and knowledge but at the same time they could be the storehouses of misinformation, poorly guided training or bad knowledge. E-learning has many advantages. With the combination of a well-organised e-learning system and a highly motivated student, one can achieve great success in a short period of time. Some of the major advantages of e-learning are listed below:

- a) Convenient for students E-learning materials are self-placed and can be accessed any time the learner wants. They do not require the learner to be physically present in a classroom. Students can also download and save the learning materials for future purposes from the system.
- b) Lower cost E-learning is usually a cost-efficient way of learning for most students as they can choose from a large range of courses and make the selection depending on their needs. It can also be cost-efficient for many universities because once the learning platforms are set up, they can be reused for many sessions.
- c) Up-to-date learning materials The study materials in e-learning systems can be updated more frequently than in the classroom-based education systems. Once the study materials are placed in the system, they can be updated without changing the whole materials and the materials can be available and reused for longer times.
- d) Flexible way of learning E-learning is a flexible way of learning for many students. Most of the study materials are stored for the students to access whenever they want. Students can also choose between an instructor-led and a self-learning system. In e-learning systems it is also possible for students to skip over the study materials they already know and choose the ones they want to learn.
- e) World-wide learning society E-learning systems help in creating a worldwide learning society as anyone can access the study materials regardless of the geographical location. In the systems available now learners can also contribute to the study materials, which helps to keep the materials updated.
- f) Scalable e-learning systems The number of students in virtual classes or e-learning systems can be very few or really high without causing any significant difference in the total cost.
- g) Higher degree of freedom for students One may find it difficult at times to learn new ideas. E-learning systems provide the possibility for students to learn the same material repeatedly until they are satisfied.
- h) Better retention The video and audio materials used in e-learning make the whole learning process more fun. This will help students to remember the things they learn for a long period of time. E-learning materials can also be accessed whenever wanted, thus the repetition makes the retention easier

As any other systems e-learning also have some drawbacks. Being flexible is not always good as it may cause laziness and thus reduce efficiency. Some of the major disadvantages of e-learning include:

Cloud Based Free E-Learning Application

- a) **Low motivation** Students with low motivation may not achieve the set goals most of the time, as there is no one to look over. Students are themselves responsible for the routine and organisation of the course, thus leading to laziness and low motivation at times. Lack of a fixed schedule and deadlines may lead to students dropping out of the course prematurely
- b) **Technology-dependency** The study materials in e-learning are delivered using computer applications. For some people just to learn how to use those applications might take a long period of time. There are also other factors such as a poor Internet connection and machine malfunction, which may make the learning process tedious and time consuming.
- c) **Compatibility issues** As there are many learning systems available, sometimes the study materials prepared by using one system may not be compatible with another. For example mobile devices such as iPads block the flash videos from being played in their browsers. Access to websites may not be free in all places as some countries impose restrictions on the number of websites that can be accessed.
- d) **Reliability of the content** The content available on the Internet might not always be reliable. There are people who mislead the readers and feed wrong information. So the readers must be careful while searching for the information and check the reliability of the content before learning it.
- e) **Social isolation** Lack of a real classroom or classmates might not be good for all students. Students might feel socially isolated at times due to lack of real people around while learning.
- f) **Expenses management** In the long run e-learning is usually a cheaper option but still for the first time it might prove too expensive for some institutions. Buying new equipment like computers, projectors or new software all at once might not always be easy for new or small institutions. Also the cost of developing training materials is high compared to the traditional methods.
- g) **Disadvantages for disabled students** While developing new study materials or study environments companies usually target at large user groups. They might not always think about disabled students such as visually impaired students as they might not gain from it as the development costs may be high.
- h) **Not effective in all cases** In some cases face-to-face study materials might be more effective than learning online as e-learning lacks two-way communication

The objectives of the system are-

- To reduce the paperwork.
- Reduced manual effort.
- Reduced cost.
- Increased accuracy and reliability.

This software package can be readily used by non-programming personnel thus avoiding human handled chance of error.

The students can create their accounts and study from the notes by linking their accounts with Application.

The application can be upgraded according to user's requirements with minor changes.

New features can be added in the future based on the requirements.

Literature Review

Education (training) is the process by which the wisdom, knowledge and skills of one generation are passed on to the next. This education process has been going on since time immemorial. It is the central process in the conservation and development of human culture. It began at the dawn of time and has continued till today. Eventually, the society developed for itself schools as the privileged places where education process takes place (Alonso & Norman, 1996). The move to wireless in telephony and computing is irreversible. There is no doubt that the World Wide Web is the most successful educational tool to have appeared in a long time. It combines and integrates text, audio and video with interaction amongst participants. It can be used on a global scale and is platform independent.

M-learning is a natural extension of E-learning and has the potential to make learning even more widely available and accessible than we are used to in existing e-learning environments. The role that communication and interaction play in the learning process is a critical success factor. It is within this context that m-learning can contribute to the quality of education. It offers opportunities for the optimization of interaction between lecturers and learners, among learners and among members of COPs (communities of practice) (Georgieva et al., 2004)

The primary differences between the two disciplines is that where learning previously occurred in front of a computer terminal, in the classroom, laboratory, or at home, it is now enabled to occur in the field, or at any location where the mobile device is fully functional (Sharma and Kitchens, 2004). With M-learning, instructors can incorporate multimedia demonstrations in their lectures and receive real-time feedback from their students using quizzes or surveys (Adewunmi et al., 2003), learning can be done anytime and anywhere and it provides support for continuous learning with the ability to collaborate with instructor's notebook during class thereby providing means of communication and teaching support while outside the classroom. Mobile learning is able to synchronous team member's appointments and schedules (Lehner et al., 2001); Classroom seating arrangement does not necessarily have to be fixed. Individual student activities such as web browsing, independently running example program or working through example problems in class can take place (Brown, 2001). An instructor can get immediate feedback on the lesson being taught. Student's can be assessed on multiple choices, true/false questions in the classroom. Real-time experiments can take place in classrooms. Instructors can provide examples such as simulations and web based documents that can be accessed at specific time to improve retention. Students do not have to waste time copying what the instructor writes on the whiteboard (Brown, 2001). Upadhyay (2007) summarizes the advantages of Mobile learning as seamless access to learning resource, freedom and power of choice, organized productivity and flexible, portable convenience.

I used the literature review to generate pertinent ideas that could be further refined. I also engaged in critical reflection on these ideas in order to evaluate their origins, meaning and status in the research. During the course of the study I continuously returned to the literature review to update it, refine it, and add new information and observations. In order to make sense of the literature I attempted to examine literature that was current, similar and relevant. I used the following criteria as filter to determine whether include or exclude text from the literature review process.

Is the material relevant in this study?

- Is the proposed text by an author who was suggested by other author's reputable authority?
- Will the material augment the intellectual matrix of this study?
- Will the proposed text shed light on the issues raised in this research?
- Does the date of the material suggest that it is an up to date contribution in this field?

There are two main aims to this literature review. The first aim is to establish the characteristics and importance of formative, coursework assessment. The second aim is to identify e-learning techniques, tools and approaches for this type of assessment, to discuss what is known about their effectiveness and to uncover factors influencing uptake.

For the first aim key articles and books that have shaped current theory on formative assessment of student work have been identified and analysed. For the second aim a comprehensive search of e-learning literature from 2001 to the present was conducted.

The classes taught by the E-learning Centre cover many computer and non computer oriented topics. The classes are presented as online seminars using video and flash media. Exercises are integrated into each class instruction segment. E-Learning Centre offers online web design tutorials on many aspects of web design, but also on general computer applications and even certifications.

According to some authors they say that LMS in universities and in different organizations is very wide spread because of the features they provide which simplify the management and delivery of course content to students. Regardless of their wide use, their focus is more on the delivery of content than on the learner. LMSs offer little or no opportunities for learners to interact and collaborate on different works and to author content which can contribute to the knowledge which is already available in the systems. LMS are mostly for people/students who are already computer literate, but for those who don't not have any computer skills have a problem to adapt to the new learning system.

Here are some of the negative aspects of the research area from different source:-

1. Bright and independent students (learners) benefits from constructivist approaches online.
2. Designing online courses may require more imagination and skill than traditional courses.
3. Too many new topics could unintentionally shift the attention from important topics in

online discussions.

4. Online interaction centered on information rather than constructing knowledge.
5. If collaboration serves no real purpose learners will end up learning in isolation.

In another researches it was found that to be effective, tertiary education should engage learners as active participants in their learning. Achieving this means offering learners opportunities for interaction in ways that can promote change and growth in the learner's conception of knowledge. Such pedagogies aim to encourage learners to become autonomous lifelong learners, capable of problem solving and critical thinking, and to move them from being passive recipients of information and knowledge to being active, enthusiastic learners and knowledge creators.

Here are some of the positive aspects of the research area from different source:-

1. Solving problems through online discussions is more rewarding than face to face discussions and fewer messages are generated.
2. The role for online lectures is critical to success.
3. Open-ended assignments tend to encourage deep thinking.
4. Knowledge acquired by means of repetition is transferred from short-term to long-term memory.
5. It helps enhance the skill of independent learning, social skills, teamwork, thinking skill and internet navigation skill.

The main method to identify relevant articles was reading titles and abstracts of selected journals articles and papers in conference proceedings. While more time consuming this method was chosen in preference to keyword search as it presents a more thorough approach to identifying relevant articles. Additionally, keyword search was carried out on several databases. This was done to cover some additional conference publications and to cross-check against the primary search method. Among the keywords used were: online marking, electronic submission, annotation, marking, assessment mark-up, assessment marking, e-learning formative essay, essay mark-up, formative assessment essay, and marking essay online.

Analysis and Design of the Project (with UML diagrams / Flow charts / algorithm)

In this section, I will explain and try to answer the research questions and determine the trends and relationship among the variables. It presents the findings of the study project, but also the general characteristic of the population. It shows the result of the survey conducted in the study of effectiveness of the E-Learning system also known as course repository.

This chapter also explains and shows the backbone of the system by showing you the database design, user case diagrams for administrators, lectures and students. It explain how each user of the system can use the system with easy. How they can start from logging in, what they can do in the system and to logging out.

Best Learning Application Defined

The best application solution is defined in this study as one in which all LMS components are considered within the total learning infrastructure of Zanzibar University such that maximum student success is ensured from both an institutional and System perspective. Aspects of these components within the frame work of student success were assessed by the following attributes:

- **Cost effectiveness:** - the total cost figures included in this report represent a current snapshot of the LMS expenditures excluding self-hosting and migration cost, is reported to the Assessment team. The benefits of the LMS is expected to be more than cost such as hosting maintenance and other cost that may be incurred
- **Support and Training:** - the system must have a virtual learning community provides students orientates templates and professionals, development resources for faculty by providing online help desk services for students and faculty that includes chat, email, telephone and a personalized support portal available 24/7/365.
- **Ease of Use:** - the LMS must have ease of use components and no additional instrument questions were developed the system should have a higher level of instructor and administrator perceived application functionality.
- **Scalability:** - the LMS must be able to report on the number of active course, users, and average course size and storage capacity on their LMS. It should be able to hold a lot of actives but still be able to use a less storage capacity.
- **Sustainability:** - The sustainability of an LMS is paramount to the future growth of distance learning in the university. The system-wide capability to support LMS-centric learning

technology is challenging in both the short and long terms. Information from the success NC listening tour notes indicate that: -

- (1) Colleges differ on LMS preferences but want continued support from the System Office.
- (2) Learning technology offers an effective and flexible means to facilitate learning
- (3) Increased enrolments have negatively affected instructors, strained facilities, and encouraged creative interventions to maximize resources.

System Requirements

System requirements are what is necessary for a client to install the web application in their system and be used hopefully without any difficulties. The aim of this is to help clients make sure they have all required tools or equipment.

With analysis there are some objectives for hardware, software and any other thing that would be the minimum requirements to install the system. The minimum requirements are as follows.

Hardware requirements

- 256 MB RAM but 1 Gig preferred
- Internet access at 56k or above
- 3 GB of hard disk space
- Screen resolution min 1024 x 768

Software requirements to build this application

Internet Explorer 7.0+, Safari 3.0+, Firefox 3.0+ (Browser must be Java and JavaScript enabled)

- Windows 7, Vista, or XP; Mac OSX 10.4+
- IE 7.0+, Safari 3.0+, Firefox 3.0+ (Browser must be Java and JavaScript enabled)
- System development software: - visual studio (asp.net and C#), SQL server.

System Evaluation

It is in the very hope that by the end of the implementation of this system 97 percent of the problems faced in the by the current system shall be resolved and it will meet all user requirement, user friendly as users were involved in all stages of the development system.

In this system there are three main users which are the administrator, the lecture and the student, each one of them has their specific task and roles they can perform within the system. The system is sensitive to privacy protection thereby the system has been designed to observe these matters.

Administrator

The system administrator will have full access privilege of the system which the other users cannot perform. Some of these include: - assigning roles to user (who is the Admin, lecture or student), deleting users, adding (department, faculties), and lastly creating users.

Application Users

The student will have less privileges, the student will be able to upload and download documents, comment on the blogs created by lectures, and view posts news by lectures, and administrators, and lastly students will be able to view their coursework.

Use cases: Elicit requirement from users in meaningful chunks. Construction planning is built around delivering some use cases n each interaction basis for system testing.

Class diagrams: shows static structure of concepts, types and class. Concepts how users think about the world; type shows interfaces of software components; classes shows implementation of software components.

Interaction diagrams: shows how several objects collaborate in single use case. Package diagram: show group of classes and dependencies among them. State diagram: show how single object behaves across many use cases.

Activity diagram: shows behavior with control structure. Can show many objects over many uses, many object in single use case, or implementations methods encourage parallel behavior, etc.

The end-product of this project is a comprehensive tool that can parse any vb.net program and extract most of the object oriented features inherent in the program such as polymorphism, inheritance, encapsulation and abstraction.

What is UML?

UML stands for Unified Modeling Language is the successor to the wave of Object Oriented Analysis and Design (OOA&D) methods that appeared in the late 80's. It most directly unifies the methods of Booch, Rumbaugh (OMT) and Jacobson. The UML is called a modeling language, not a method. Most methods consist at least in principle, of both a modeling language and a process. The Modeling language is that notation that methods used to express design.

Notations and meta-models:

The notation is the graphical stuff; it is the syntax of the modeling language. For instance, class diagram notation defines how items are concepts such as class, association, and multiplicity is represented. These are:

Class Diagram: The class diagram technique has become truly central within object-oriented methods. Virtually every method has included some variation on this technique. Class diagram is also subject to the greatest range of modeling concept. Although the basic elements are needed by everyone, advanced concepts are used less often. A class diagram describes the types of objects in the system and the various kinds of static relationship that exist among them. There are two principal kinds of static relationship:

- Association
- Subtype

Class diagram also show the attributes and operations of a class and the constraints that apply to the way objects are connected.

Association: Association represent between instances of class. From the conceptual perspective, association represents conceptual relations between classes. Each association has two roles. Each role is a direction on the association. A role also has multiplicity, which is a indication of how many object may participate in the given relationship.

Generalization: A typical example of generalization evolves the personal and corporate customer of a business. They have differences but also many similarity. The similarities can be placed in generalization with personal customer and corporate customer sub type.

Aggregation: aggregation is the part of relationship. It is like saying a car has engine and wheels as its parts. This sounds good, but difficult thing is considering, what is the difference is aggregation and association.

Interaction: interaction diagrams are models that describes how groups of objects collaboration in some behavior.

Typically, an interaction diagram captures the behavior a single use cases. The diagram shows a number of example objects and the messages that are passed between these objects in use cases. These are following approaches with simple use case that exhibits the following behavior.

Objects can send a message to another. Each message is checks with given stock item. There are two diagrams: Sequence and Collaboration diagram.

Package Diagram: One of the oldest questions in software methods is: how do you break down a large system into smaller systems? It becomes difficult to understand and the changes we make to them.

Structured methods used functional decomposition in which the overall system was mapped as a function broken down into sub function, which is further broken down into sub function and so forth. The separation of process data is gone, functional decomposition is gone, but the old question is still remains. One idea is to group the classes together into higher-level unit. This idea, applied very loosely, appears in many

objects. In UML, this grouping mechanism is package. The term package diagram for a diagram that shows packages of classes and the dependencies among them.

A dependency exists between two elements if changes to the definition of one element may cause to other. With classes, dependencies exist for various reasons: one class sends a message to another; one class has another as part of its data; one class mentions another as a parameter to an operation. A dependency between two packages exists; and any dependencies exist between any two classes in the package.

State diagram: State diagram are a familiar technique to describe the behavior of a system. They describe all the possible states a particular object can get into and how the objects state changes as a result of events that reach the objects. In most OO technique, state diagrams are drawn for a single class to show the lifetime behavior of a single object. There are many form of state diagram, each with slightly different semantics. The most popular one used in OO technique is based on David Harel's state chart.

Dataflow Diagram:

Data flow diagram symbols

A data flow diagram shows how data is processed within a system based on inputs and outputs. Visual symbols are used to represent the flow of information, data sources and destinations, and where data is stored. Data flow diagrams are often used as a first step toward redesigning a system. They provide a graphical representation of a system at any level of detail, creating an easy-to-understand picture of what the system does. A general overview of a system is represented with a context diagram, also known as a level 0 DFD, which shows a system as a single process. A level 1 diagram provides greater detail, focusing on a system's main functions. Diagrams that are level 2 or higher illustrate a system's functioning with increasing detail. It's rare for a DFD to go beyond level 2 because of the increasing complexity, which makes it less effective as a communication tool.

Data flow diagram notations

The two main types of notation used for data flow diagrams are Yourdon-Coad and Gane-Sarson, both named after their creators, all experts who helped develop DFD methodology: Ed Yourdon, Peter Coad, Chris Gane and Trish Sarson. There are some differences in style between the notation types. For example, Yourdon and Coad notation uses circles to represent processes, whereas Gane and Sarson notation use rectangles with rounded corners. Another variation is the symbol used for data stores—Yourdon and Coad uses parallel lines while Gane and Sarson notation uses an open-ended rectangle. Because DFD symbols vary, it's important to be consistent with whatever notation you choose in order to avoid confusion. If you're using DFD software, it will likely dictate which set of symbols are available to use.

All data flow diagrams include four main elements: entity, process, data store and data flow.

External Entity – Also known as actors, sources or sinks, and terminators, external entities produce and consume data that flows between the entity and the system being diagrammed. These data flows are the inputs and outputs of the DFD. Since they are external to the system being analyzed, these entities are typically placed at the boundaries of the diagram. They can represent another system or indicate a subsystem.

Process – An activity that changes or transforms data flows. Since they transform incoming data to outgoing data, all processes must have inputs and outputs on a DFD. This symbol is given a simple name based on its function, such as “Ship Order,” rather than being labeled “process” on a diagram. In Gane-Sarson notation, a rectangular box is used and may be labeled with a reference number, location of where in the system the process occurs and a short title that describes its function. Processes are typically oriented from top to bottom and left to right on a data flow diagram.

Data Store – A data store does not generate any operations but simply holds data for later access. Data stores could consist of files held long term or a batch of documents stored briefly while they wait to be processed. Input flows to a data store include information or operations that change the stored data. Output flows would be data retrieved from the store.

Data Flow – Movement of data between external entities, processes and data stores is represented with an arrow symbol, which indicates the direction of flow. This data could be electronic, written or verbal. Input and output data flows are labeled based on the type of data or its associated process or data store, and this name is written alongside the arrow.

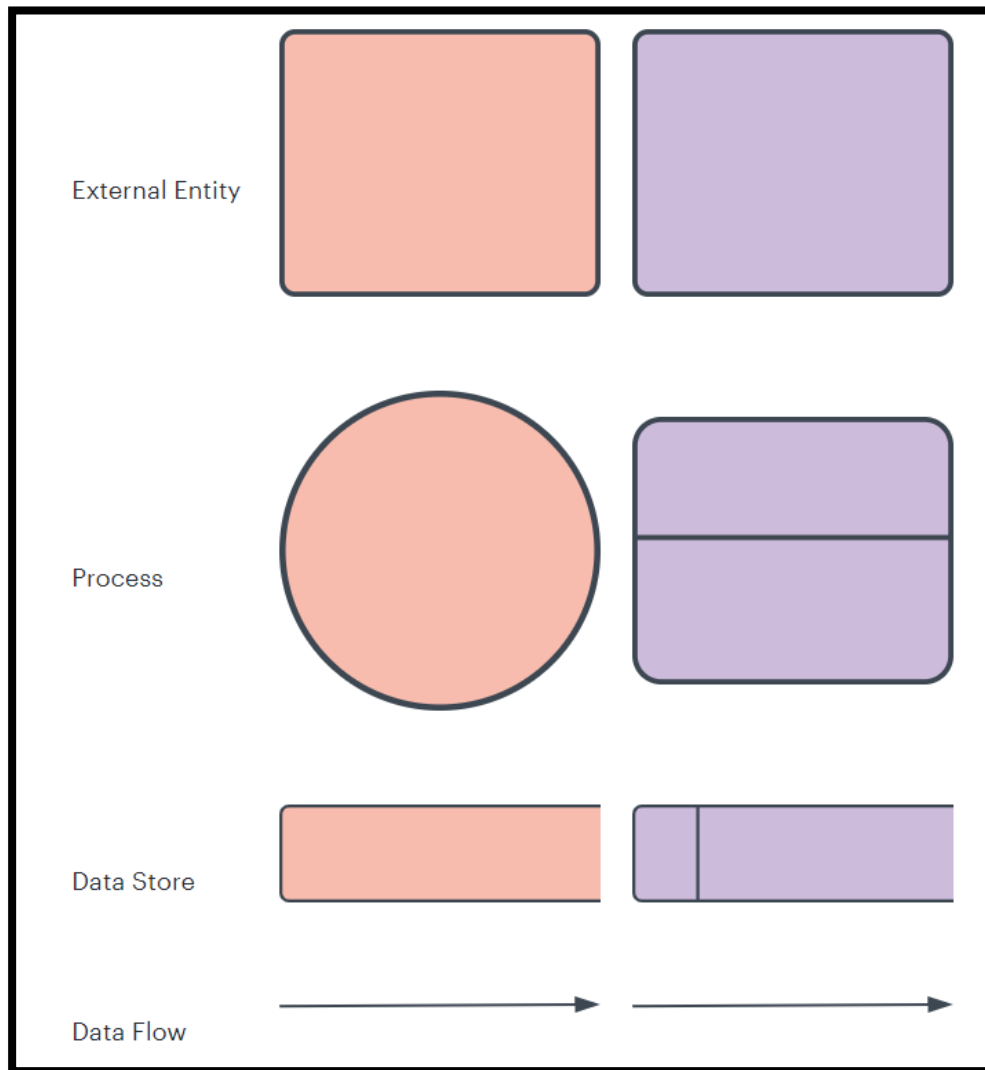


Fig 3: Data Flow Diagram

Data flow diagram is the starting point of the design phase that functionally decomposes the requirements specification. A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flows in the system. A DFD describes what data flow rather than how they are processed, so it does not hardware, software and data structure.

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing (structured design). A data flow diagram (DFD) is a significant modeling technique for analyzing and constructing information processes. DFD literally means an illustration that explains the course or movement of information in a process. DFD illustrates this flow of information in a process based on the inputs and outputs. A DFD can be referred to as a Process Model.

The data flow diagram is a graphical description of a system's data and how to

Process transform the data is known as Data Flow Diagram (DFD).

Unlike details flow chart, DFDs don't supply detailed descriptions of modules that graphically describe a system's data and how the data interact with the system. Data flow diagram number of symbols and the following symbols are of by DeMarco.

Entity Relationship (ER) Diagram

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how "entities" such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

Cloud Based Free E-Learning Application

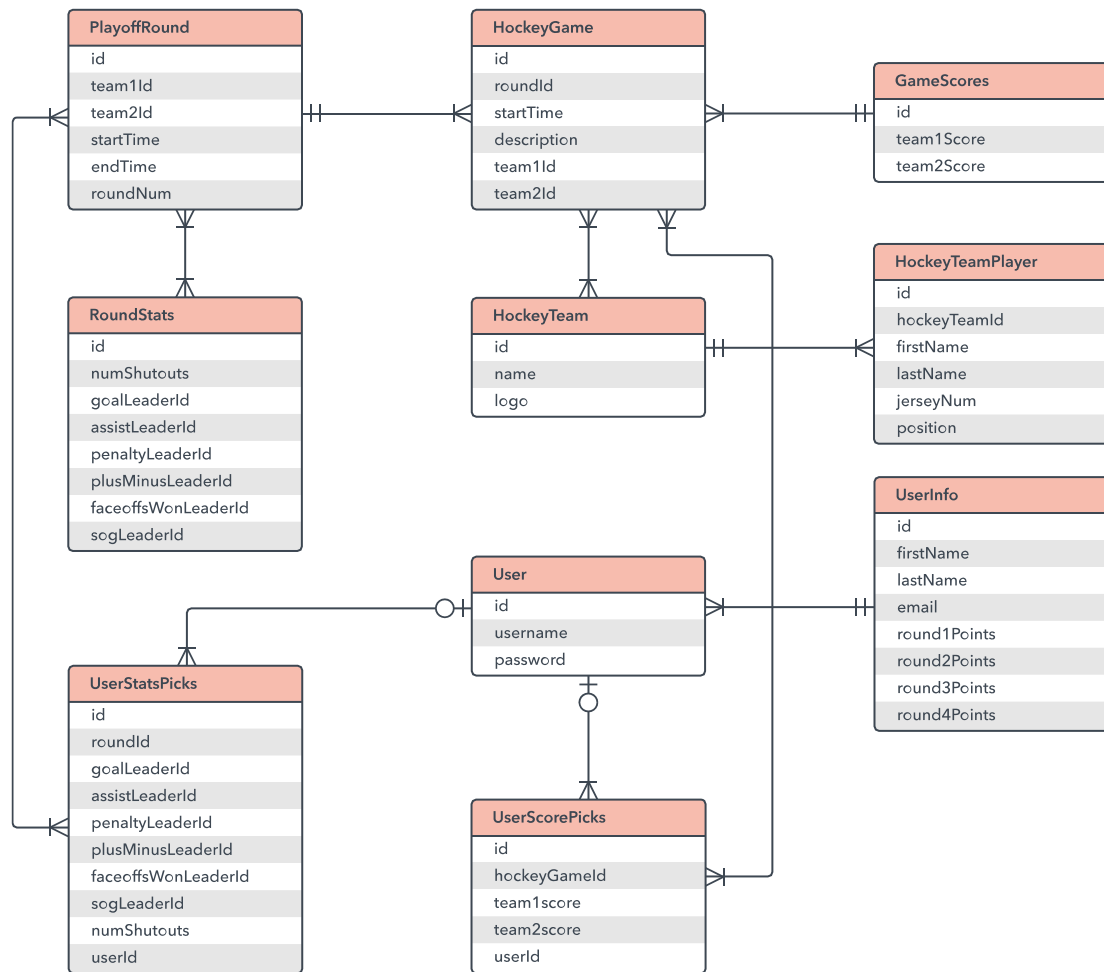


Fig 4: Entity Relationship Diagram.

ER diagrams are related to data structure diagrams (DSDs), which focus on the relationships of elements within entities instead of relationships between entities themselves. ER diagrams also are often used in conjunction with data flow diagrams (DFDs), which map out the flow of information for processes or systems.

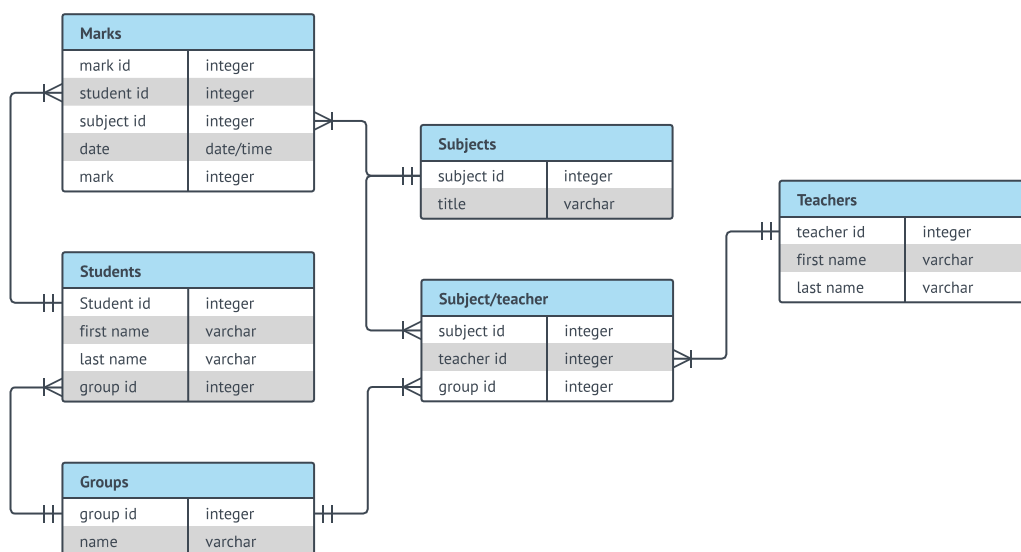


Fig 5: ER Diagram.

Uses of entity relationship diagrams

- **Database design:** ER diagrams are used to model and design relational databases, in terms of logic and business rules (in a logical data model) and in terms of the specific technology to be implemented (in a physical data model.) In software engineering, an ER diagram is often an initial step in determining requirements for an information systems project. It's also later used to model a particular database or databases. A relational database has an equivalent relational table and can potentially be expressed that way as needed.
- **Database troubleshooting:** ER diagrams are used to analyze existing databases to find and resolve problems in logic or deployment. Drawing the diagram should reveal where it's going wrong.
- **Business information systems:** The diagrams are used to design or analyze relational databases used in business processes. Any business process that uses fielded data involving entities, actions and interplay can potentially benefit from a relational database. It can streamline processes, uncover information more easily and improve results.
- **Business process re-engineering (BPR):** ER diagrams help in analyzing databases used in business process re-engineering and in modeling a new database setup.
- **Education:** Databases are today's method of storing relational information for educational purposes and later retrieval, so ER Diagrams can be valuable in planning those data structures.
- **Research:** Since so much research focuses on structured data, ER diagrams can play a key role in setting up useful databases to analyze the data.

The components and features of an ER diagram

ER Diagrams are composed of entities, relationships and attributes. They also depict cardinality, which defines relationships in terms of numbers. Here's a glossary:

Entity

A definable thing—such as a person, object, concept or event—that can have data stored about it. Think of entities as nouns. Examples: a customer, student, car or product. Typically shown as a rectangle.



Entity


Entity type: A group of definable things, such as students or athletes, whereas the entity would be the specific student or athlete. Other examples: customers, cars or products.

Entity set: Same as an entity type, but defined at a particular point in time, such as students enrolled in a class on the first day. Other examples: Customers who purchased last month, cars currently registered in Florida. A related term is instance, in which the specific person or car would be an instance of the entity set.

Entity categories: Entities are categorized as strong, weak or associative. A **strong entity** can be defined solely by its own attributes, while a **weak entity** cannot. An associative entity associates entities (or elements) within an entity set.



Weak Entity




Associative Entity

Entity keys: Refers to an attribute that uniquely defines an entity in an entity set. Entity keys can be super, candidate or primary. **Super key:** A set of attributes (one or more) that together define an entity in an entity set. **Candidate key:** A minimal super key, meaning it has the least possible number of attributes to still be a super key. An entity set may have more than one candidate key. **Primary key:** A candidate key chosen by the database designer to uniquely identify the entity set. **Foreign key:** Identifies the relationship between entities.

Relationship

How entities act upon each other or are associated with each other. Think of relationships as verbs. For example, the named student might register for a course. The two entities would be the student and the course, and the relationship depicted is the act of enrolling, connecting the two entities in that way. Relationships are typically shown as diamonds or labels directly on the connecting lines.



Relationship



Weak Relationship

Recursive relationship: The same entity participates more than once in the relationship.

Attribute

A property or characteristic of an entity. Often shown as an oval or circle.

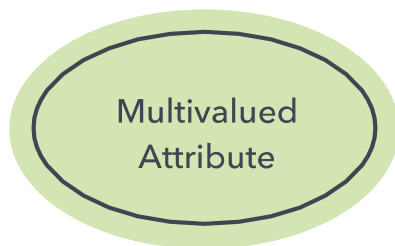


Descriptive attribute: A property or characteristic of a relationship (versus of an entity.)

Attribute categories: Attributes are categorized as simple, composite, derived, as well as single-value or multi-value. **Simple:** Means the attribute value is atomic and can't be further divided, such as a phone number. **Composite:** Sub-attributes spring from an attribute. **Derived:** Attributed is calculated or otherwise derived from another attribute, such as age from a birthdate.



Multi-value: More than one attribute value is denoted, such as multiple phone numbers for a person.

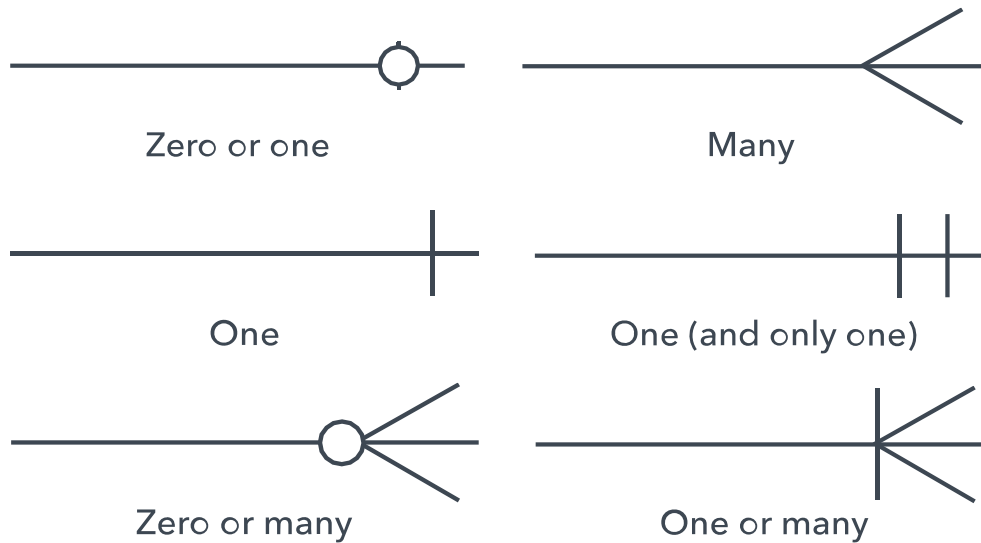


Single-value: Just one attribute value. The types can be combined, such as: simple single-value attributes or composite multi-value attributes.

Cardinality

Defines the numerical attributes of the relationship between two entities or entity sets. The three main cardinal relationships are one-to-one, one-to-many, and many-many. A **one-to-one example** would be one student associated with one mailing address. A **one-to-many example (or many-to-one, depending on the relationship direction)**: One student registers for multiple courses, but all those courses have a single line back to that one student. **Many-**

to-many example: Students as a group are associated with multiple faculty members, and faculty members in turn are associated with multiple students.



Cardinality views: Cardinality can be shown as look-across or same-side, depending on where the symbols are shown.

Cardinality constraints: The minimum or maximum numbers that apply to a relationship.

Use Case Model of the Project

The use case model for any system consists of “use cases”. Use cases represent different ways in which the system can be used by the user. A simple way to find all the use case of a system is to ask the questions “What the user can do using the application”. The use cases partition the system behavior into transactions such that each transaction perform some useful action from the users point of view.

This Use Case Diagram is a graphic depiction of the interactions among the elements of E-Learning Application. It represents the methodology used in system analysis to identify, clarify, and organize system requirements of E-Learning Application. The main actors of E-Learning Application in this Use Case Diagram are: Super Admin, System User, Teacher, Student, who perform the different type of use cases such as Manage Course, Manage Students, Manage Schedules, Manage Fees, Manage Durations, Manage Training, Manage Subject, Manage Users and Full E-Learning Application Operations. Major elements of the UML use case diagram of E-Learning Application are shown on the picture below.

The relationships between and among the actors and the use cases of E-Learning Application:

- Super Admin Entity : Use cases of Super Admin are Manage Course, Manage Students, Manage Schedules, Manage Fees, Manage Durations, Manage Training, Manage Subject, Manage Users and Full E-Learning Application Operations
- System User Entity : Use cases of System User are Manage Course, Manage Students, Manage Schedules, Manage Fees, Manage Durations, Manage Training, Manage Subject

Cloud Based Free E-Learning Application

- Teacher Entity : Use cases of Teacher are Create Courses, Invite Students, Create Papers, Check Results, Publish Results
- Student Entity : Use cases of Student are Check Courses, Apply for Course, Give Tests, View Results

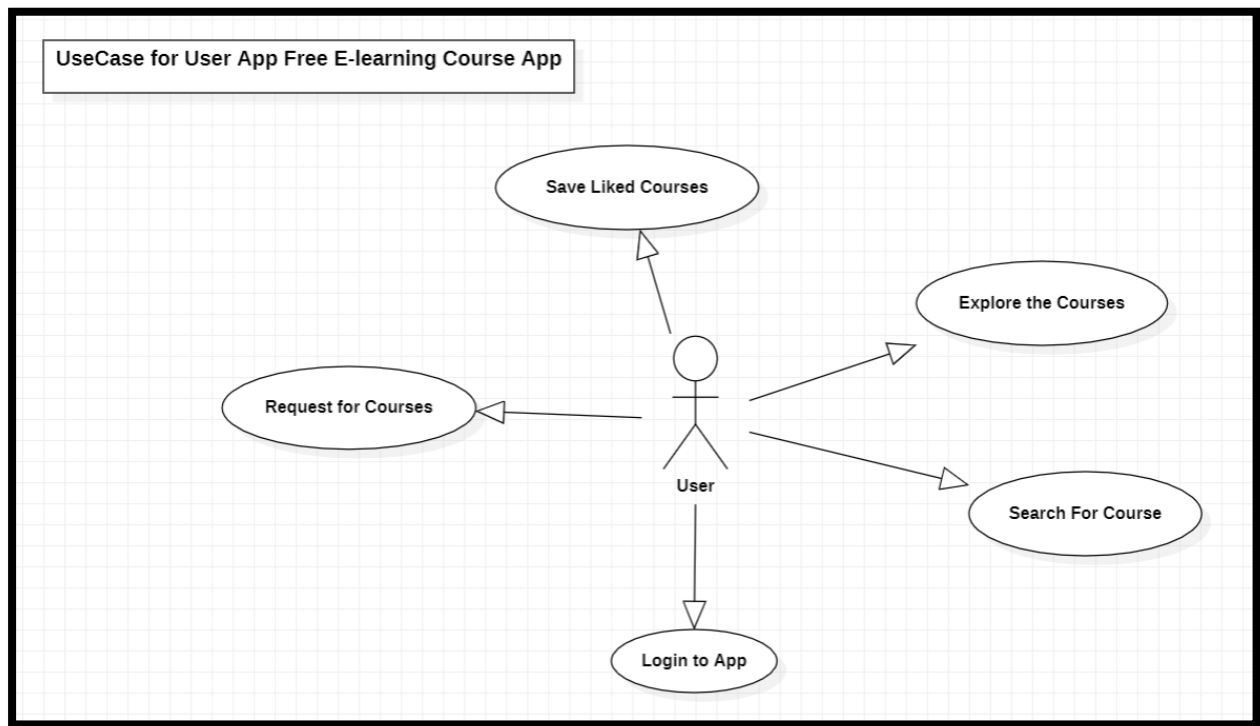


Fig 6: Use Case Diagram for User App of Free E-Learning Course Application.

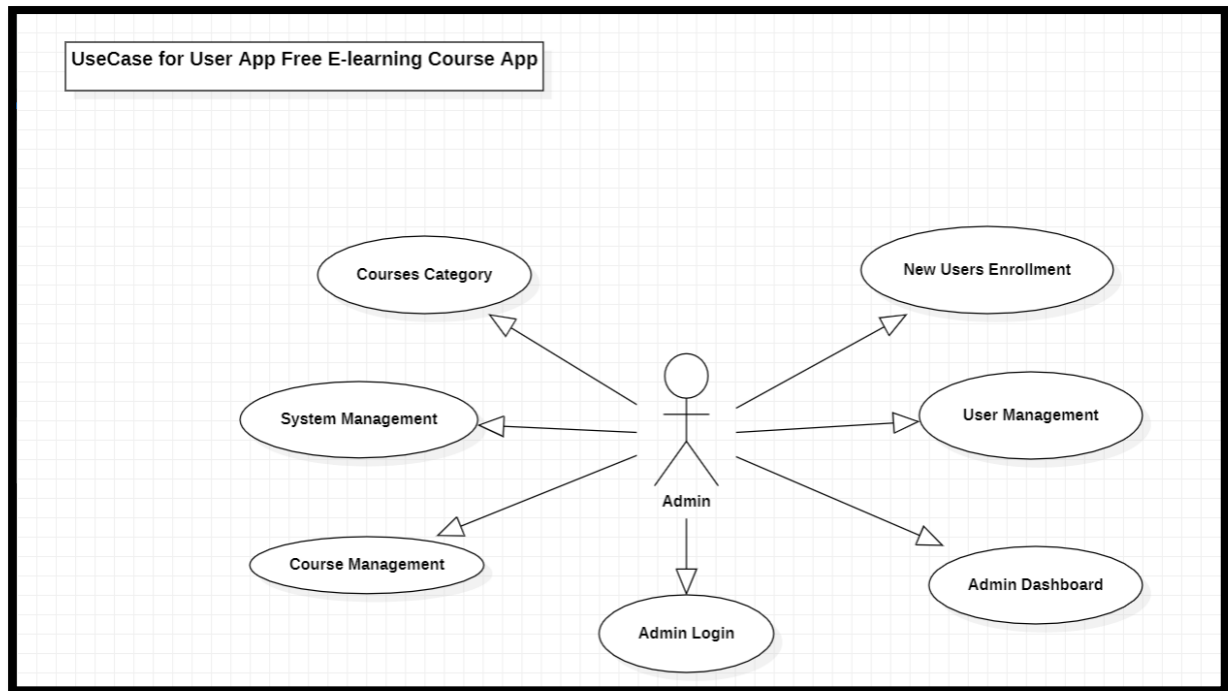


Fig 6: Use Case Diagram for Admin App of Free E-Learning Course Application.

Activity diagram

The [Unified Modeling Language](#) includes several subsets of diagrams, including structure diagrams, interaction diagrams, and behavior diagrams. Activity diagrams, along with [use case](#) and [state machine diagrams](#), are considered behavior diagrams because they describe what must happen in the system being modeled.

Stakeholders have many issues to manage, so it's important to communicate with clarity and brevity. Activity diagrams help people on the business and development sides of an organization come together to understand the same process and behavior. You'll use a set of specialized symbols—including those used for starting, ending, merging, or receiving steps in the flow—to make an activity diagram, which we'll cover in more depth within this activity diagram guide.

Basic components of an activity diagram


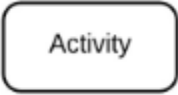

Before you begin making an activity diagram, you should first understand its makeup. Some of the most common components of an activity diagram include:

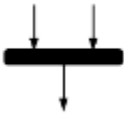







- **Action:** A step in the activity wherein the users or software perform a given task. In Lucidchart, actions are symbolized with round-edged rectangles.

- **Decision node:** A conditional branch in the flow that is represented by a diamond. It includes a single input and two or more outputs.
- **Control flows:** Another name for the connectors that show the flow between steps in the diagram.
- **Start node:** Symbolizes the beginning of the activity. The start node is represented by a black circle.
- **End node:** Represents the final step in the activity. The end node is represented by an outlined black circle.

Activity diagram symbols

These activity diagram shapes and symbols are some of the most common types you'll find in UML diagrams.

Symbol	Name	Description
	Start symbol	Represents the beginning of a process or workflow in an activity diagram. It can be used by itself or with a note symbol that explains the starting point.
	Activity symbol	Indicates the activities that make up a modeled process. These symbols, which include short descriptions within the shape, are the main building blocks of an activity diagram.
	Connector symbol	Shows the directional flow, or control flow, of the activity. An incoming arrow starts a step of an activity; once the step is completed, the flow continues with the outgoing arrow.

Symbol	Name	Description
	Joint symbol/ Synchronization bar	Combines two concurrent activities and re-introduces them to a flow where only one activity occurs at a time. Represented with a thick vertical or horizontal line.
	Fork symbol	Splits a single activity flow into two concurrent activities. Symbolized with multiple arrowed lines from a join.
	Decision symbol	Represents a decision and always has at least two paths branching out with condition text to allow users to view options. This symbol represents the branching or merging of various flows with the symbol acting as a frame or container.
	Note symbol	Allows the diagram creators or collaborators to communicate additional messages that don't fit within the diagram itself. Leave notes for added clarity and specification.
	Send signal symbol	Indicates that a signal is being sent to a receiving activity.
	Receive signal symbol	Demonstrates the acceptance of an event. After the event is received, the flow that comes from this action is completed.
	Shallow history pseudostate symbol	Represents a transition that invokes the last active state.
	Option loop symbol	Allows the creator to model a repetitive sequence within the option loop symbol.



Symbol	Name	Description
	Flow final symbol	Represents the end of a specific process flow. This symbol shouldn't represent the end of all flows in an activity; in that instance, you would use the end symbol. The flow final symbol should be placed at the end of a process in a single activity flow.
[Condition]	Condition text	Placed next to a decision marker to let you know under what condition an activity flow should split off in that direction.
	End symbol	Marks the end state of an activity and represents the completion of all flows of a process.

Fig 8: Activity Diagram Tables

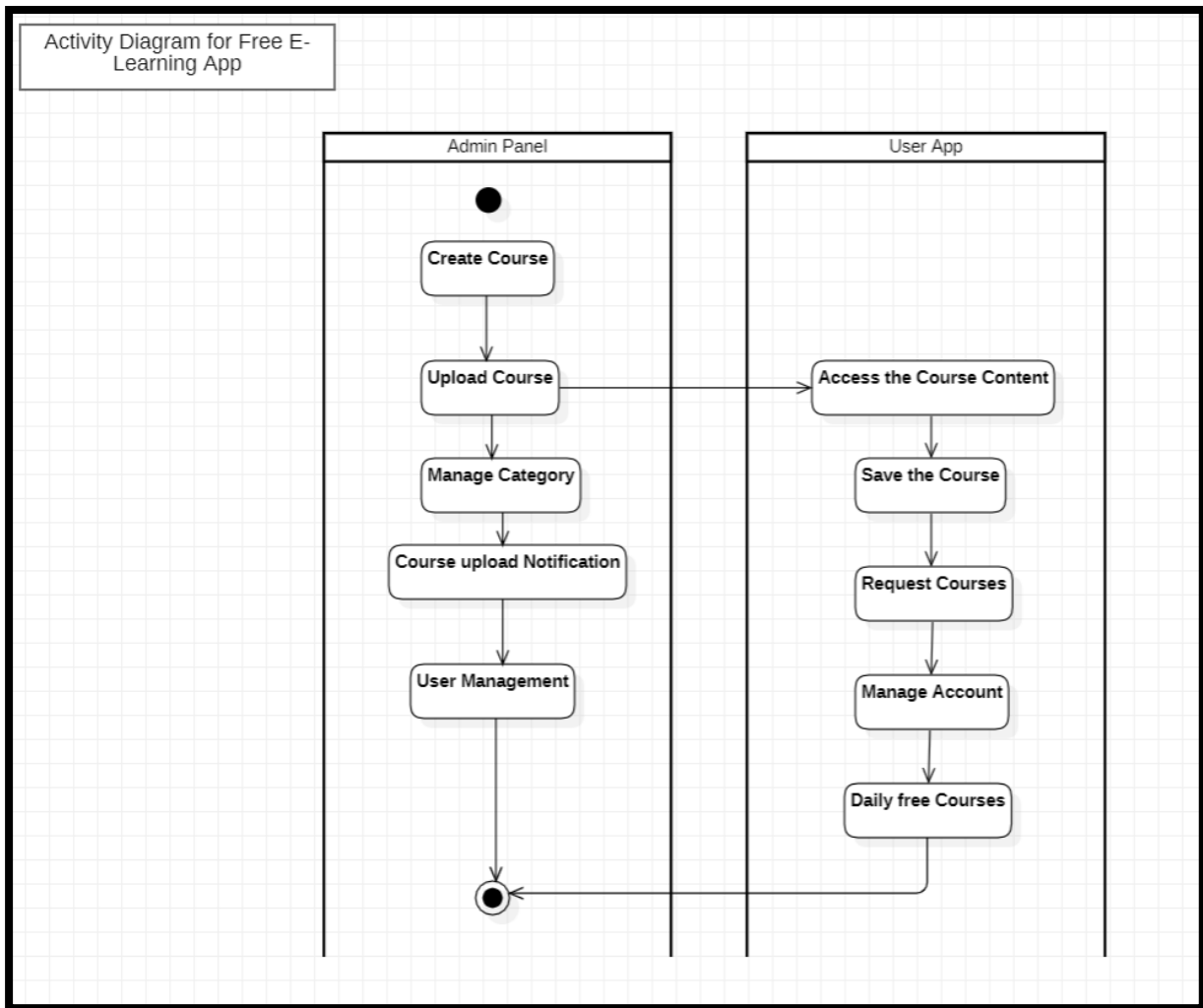


Fig 9: Activity Diagram of Free E-Learning Application

Class Diagram

The [Unified Modeling Language](#) (UML) can help you model systems in various ways. One of the more popular types in UML is the class diagram. Popular among software engineers to document software architecture, class diagrams are a type of structure diagram because they describe what must be present in the system being modeled. No matter your level of familiarity with UML or class diagrams, our [UML software](#) is designed to be simple and easy to use.

UML was set up as a standardized model to describe an object-oriented programming approach. Since classes are the building block of objects, class diagrams are the building blocks of UML. The various components in a class diagram can represent the classes that will actually be programmed, the main objects, or the interactions between classes and objects.

The class shape itself consists of a rectangle with three rows. The top row contains the name of the class, the middle row contains the attributes of the class, and the bottom section

expresses the methods or operations that the class may use. Classes and subclasses are grouped together to show the static relationship between each object.

The UML shape library in Lucidchart can help you create nearly any custom class diagram using our UML diagram tool.

Benefits of class diagrams

Class diagrams offer a number of benefits for any organization. Use UML class diagrams to:

- Illustrate data models for information systems, no matter how simple or complex.
- Better understand the general overview of the schematics of an application.
- Visually express any specific needs of a system and disseminate that information throughout the business.
- Create detailed charts that highlight any specific code needed to be programmed and implemented to the described structure.
- Provide an implementation-independent description of types used in a system that are later passed between its components.

Basic components of a class diagram

The standard class diagram is composed of three sections:

- **Upper section:** Contains the name of the class. This section is always required, whether you are talking about the classifier or an object.
- **Middle section:** Contains the attributes of the class. Use this section to describe the qualities of the class. This is only required when describing a specific instance of a class.
- **Bottom section:** Includes class operations (methods). Displayed in list format, each operation takes up its own line. The operations describe how a class interacts with data.

Member access modifiers

All classes have different access levels depending on the access modifier (visibility). Here are the access levels with their corresponding symbols:

- Public (+)
- Private (-)
- Protected (#)

- Package (~)
- Derived (/)
- Static (underlined)

Member scopes

There are two scopes for members: classifiers and instances.

Classifiers are static members while instances are the specific instances of the class. If you are familiar with basic OO theory, this isn't anything groundbreaking.

Additional class diagram components

Depending on the context, classes in a class diagram can represent the main objects, interactions in the application, or classes to be programmed. To answer the question "What is a class diagram in UML?" you should first understand its basic makeup.

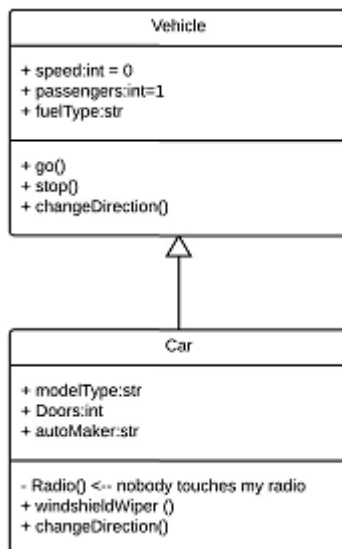
- **Classes:** A template for creating objects and implementing behavior in a system. In UML, a class represents an object or a set of objects that share a common structure and behavior. They're represented by a rectangle that includes rows of the class name, its attributes, and its operations. When you draw a class in a class diagram, you're only required to fill out the top row—the others are optional if you'd like to provide more detail.
 - **Name:** The first row in a class shape.
 - **Attributes:** The second row in a class shape. Each attribute of the class is displayed on a separate line.
 - **Methods:** The third row in a class shape. Also known as operations, methods are displayed in list format with each operation on its own line.
- **Signals:** Symbols that represent one-way, asynchronous communications between active objects.
- **Data types:** Classifiers that define data values. Data types can model both primitive types and enumerations.
- **Packages:** Shapes designed to organize related classifiers in a diagram. They are symbolized with a large tabbed rectangle shape.
- **Interfaces:** A collection of operation signatures and/or attribute definitions that define a cohesive set of behaviors. Interfaces are similar to classes, except that a class can have an instance of its type, and an interface must have at least one class to implement it.

- **Enumerations:** Representations of user-defined data types. An enumeration includes groups of identifiers that represent values of the enumeration.
- **Objects:** Instances of a class or classes. Objects can be added to a class diagram to represent either concrete or prototypical instances.
- **Artifacts:** Model elements that represent the concrete entities in a software system, such as documents, databases, executable files, software components, etc.

Interactions

The term "interactions" refers to the various relationships and links that can exist in class and object diagrams. Some of the most common interactions include:

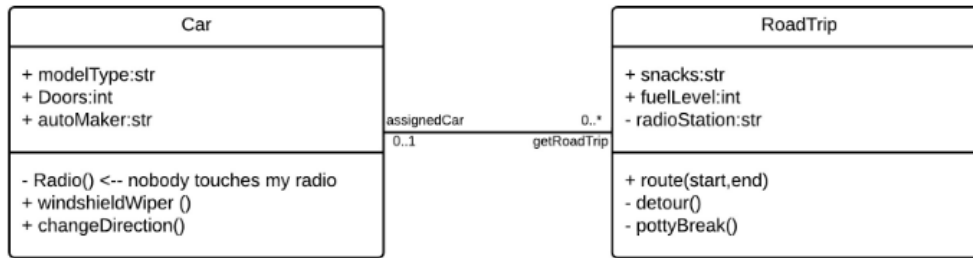
- **Inheritance:** The process of a child or sub-class taking on the functionality of a parent or superclass, also known as generalization. It's symbolized with a straight connected line with a closed arrowhead pointing towards the superclass.



In this example, the object "Car" would inherit all of the attributes (speed, numbers of passengers, fuel) and methods (go(), stop(), changeDirection()) of the parent class ("Vehicle") in addition to the specific attributes (model type, number of doors, auto maker) and methods of its own class (Radio(), windshieldWiper(), ac/heat()). Inheritance is shown in a class diagram by using a solid line with a closed, hollow arrow.

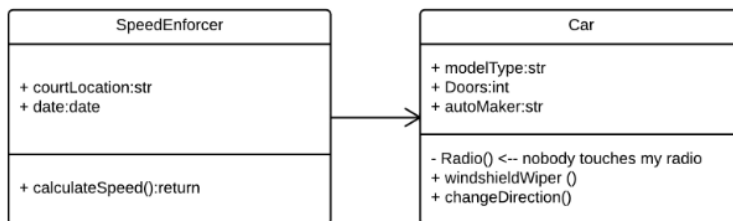
- **Bidirectional association:** The default relationship between two classes. Both classes are aware of each other and their relationship with the other. This association is represented by a straight line between two classes.

Cloud Based Free E-Learning Application



In the example above, the Car class and RoadTrip class are interrelated. At one end of the line, the Car takes on the association of "assignedCar" with the multiplicity value of 0..1, so when the instance of RoadTrip exists, it can either have one instance of Car associated with it or no Cars associated with it. In this case, a separate Caravan class with a multiplicity value of 0..* is needed to demonstrate that a RoadTrip could have multiple instances of Cars associated with it. Since one Car instance could have multiple "getRoadTrip" associations—in other words, one car could go on multiple road trips—the multiplicity value is set to 0..*

- **Unidirectional association:** A slightly less common relationship between two classes. One class is aware of the other and interacts with it. Unidirectional association is modeled with a straight connecting line that points an open arrowhead from the knowing class to the known class.



Class Diagram for E- Learning App Model

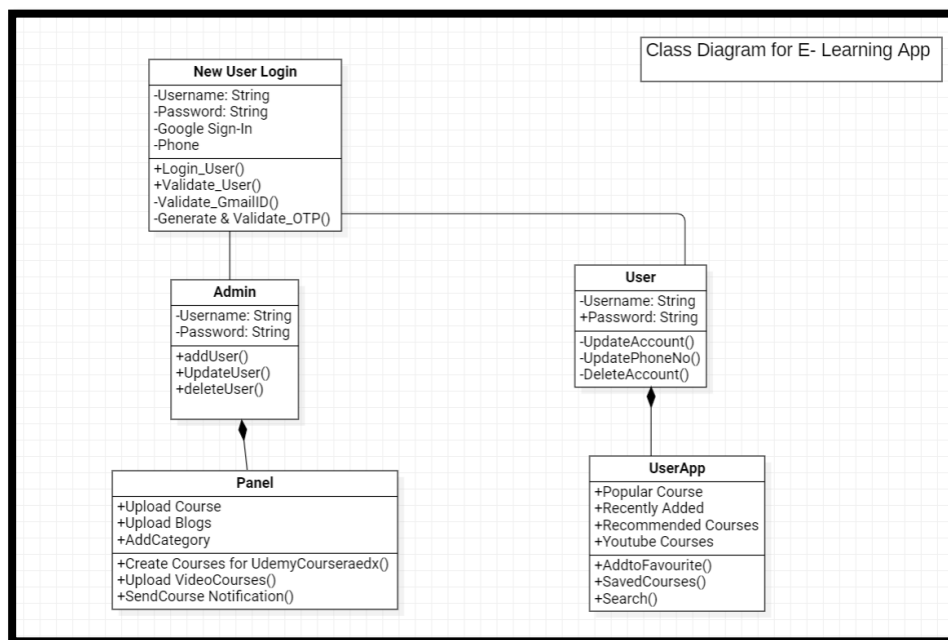


Fig 10: Class Diagram of Free E-Learning Application.

Sequence diagram

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

Sequence Diagram for E- Learning App Model

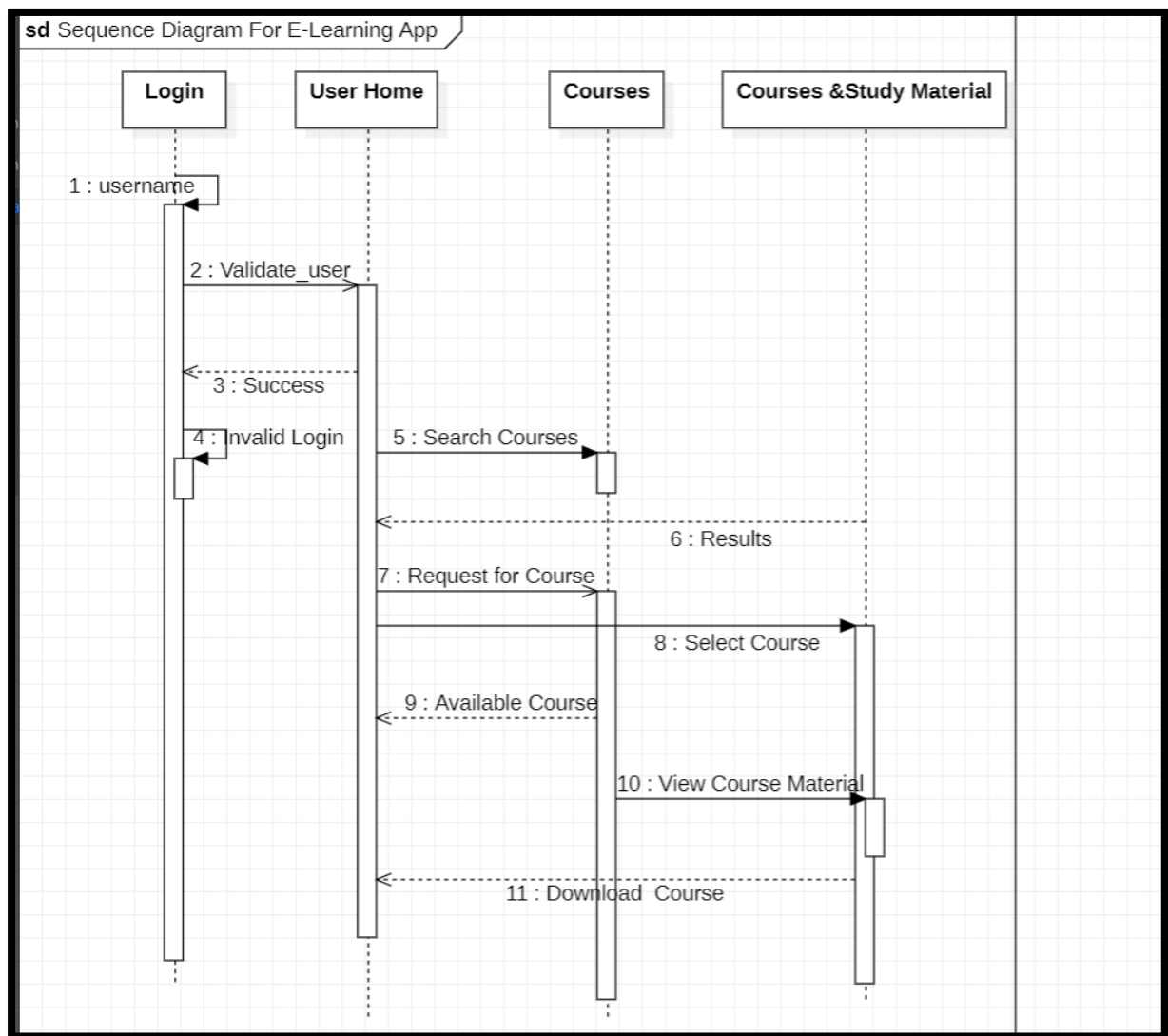


Fig 11: Sequence Diagram of Free E-Learning Application.

Implementation of Project

Admin Panel Project Structure

Folder: lib -> Main.dart

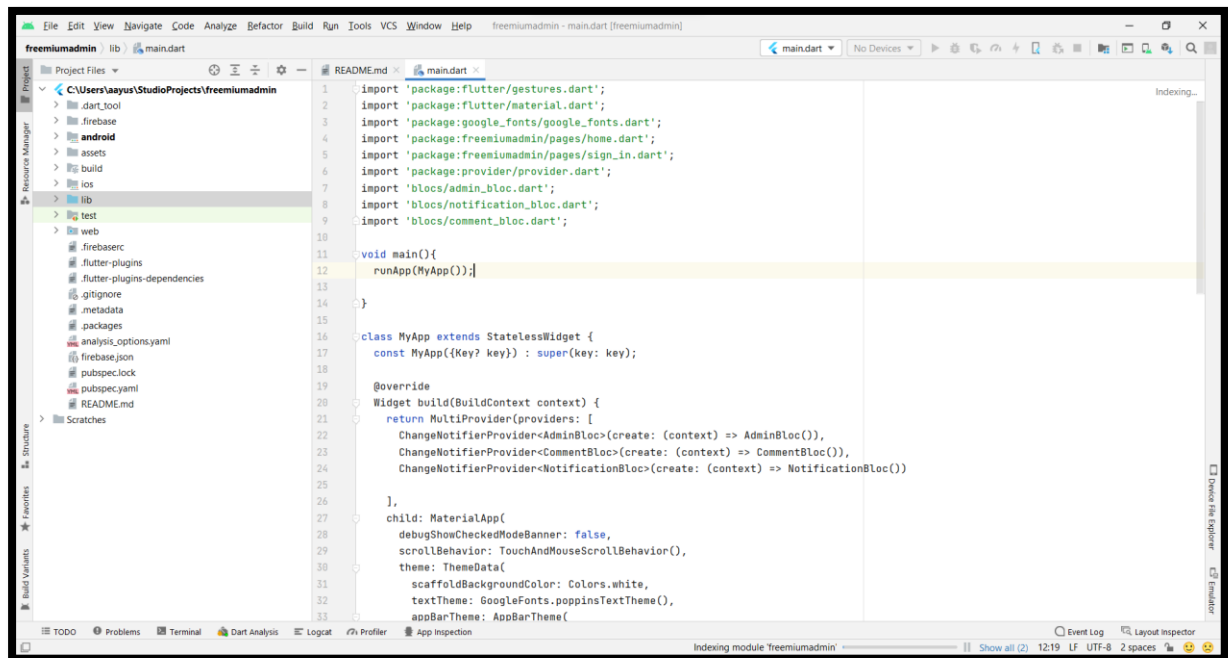


Fig 12: Admin App Code Main.dart

Pubspec.yaml (App Configuration Files)

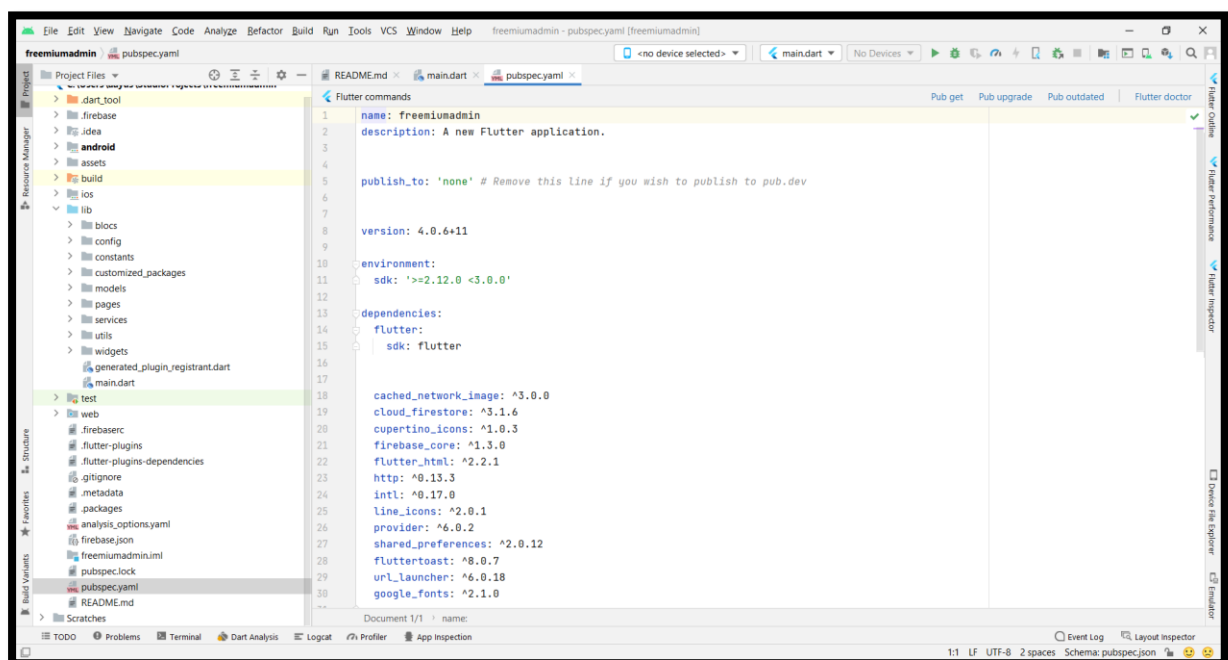


Fig 13: Admin App Code Pubspec.yaml (App Configuration File)

For Security purpose of Realtime application the overall code for Admin Panel and User App is not available in this documentation.

Results of Project

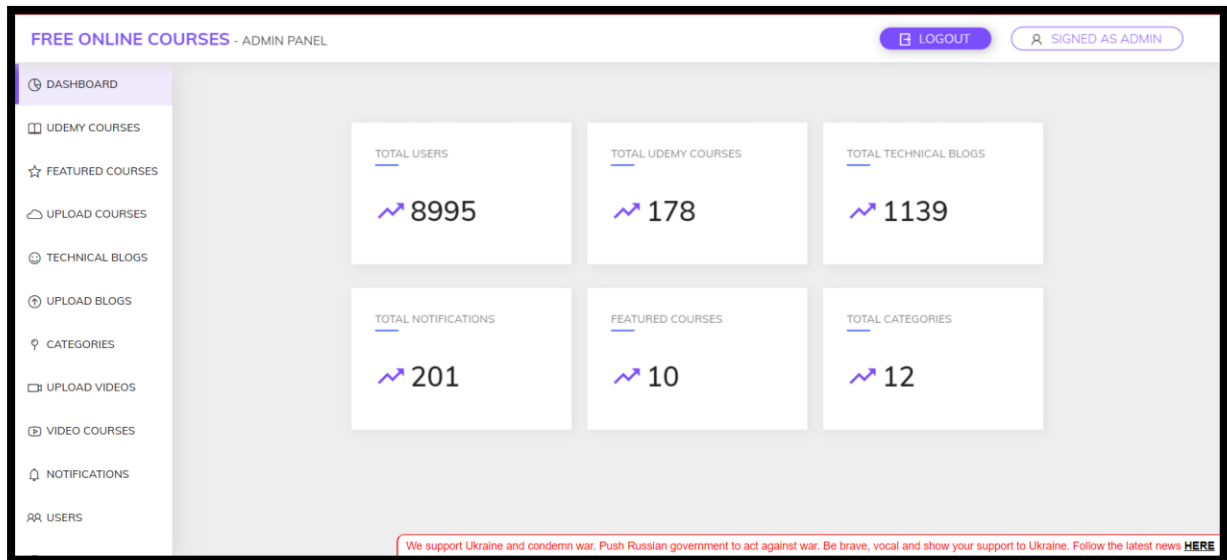
Result of TCSCCS Admin Panel

Link for Admin Panel :-

<https://freemiumadmin.firebaseio.com/>

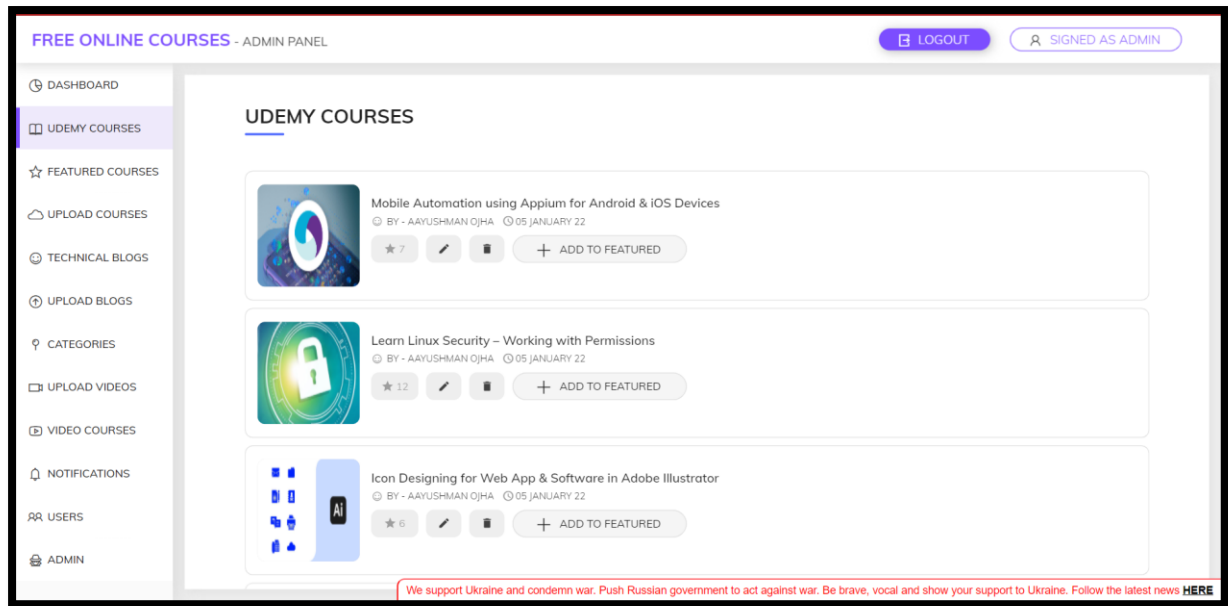
Fig 14: Snapshots of Admin Panels

1. Dashboard

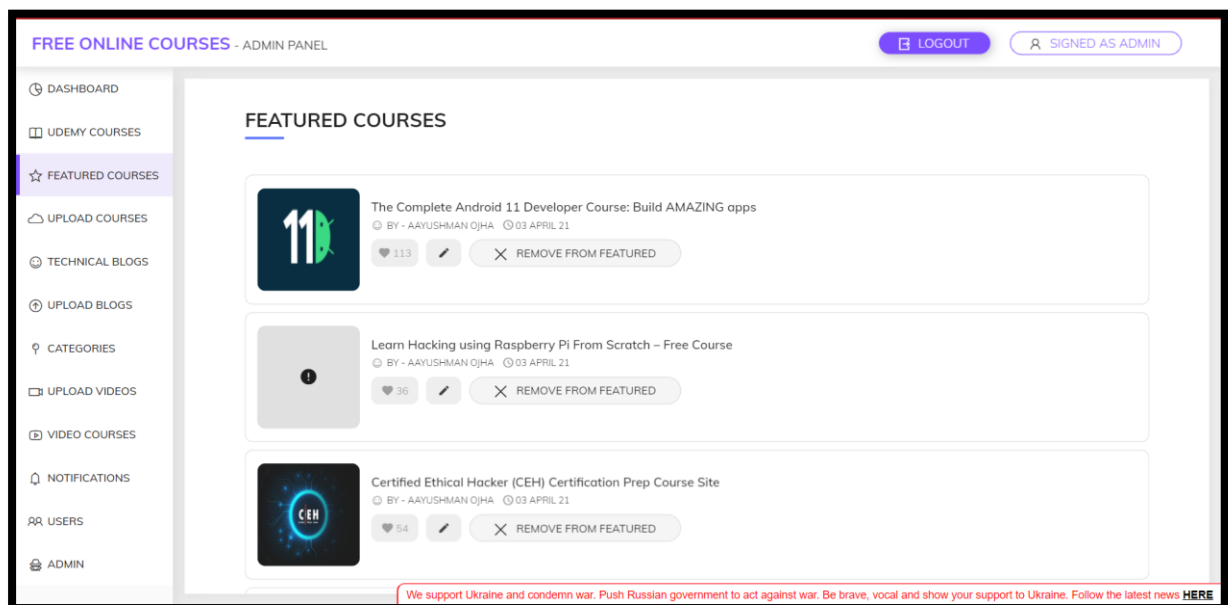


2. Udemy, Coursera, edx, Udacity Courses

Cloud Based Free E-Learning Application



3. Featured Courses.



4. Upload Courses

Cloud Based Free E-Learning Application

FREE ONLINE COURSES - ADMIN PANEL

LOGOUT

SIGNED AS ADMIN

DASHBOARD

UDEMY COURSES

FEATURED COURSES

UPLOAD COURSES

TECHNICAL BLOGS

UPLOAD BLOGS

CATEGORIES

UPLOAD VIDEOS

VIDEO COURSES

NOTIFICATIONS

USERS

ADMIN

UPLOAD UDEMY COURSE

COURSE CATEGORY

PROGRAMMING

COURSE NAME

ENTER COURSE NAME

COURSE DRIVE LINK

IMAGE 01 (THUMBNAIL)

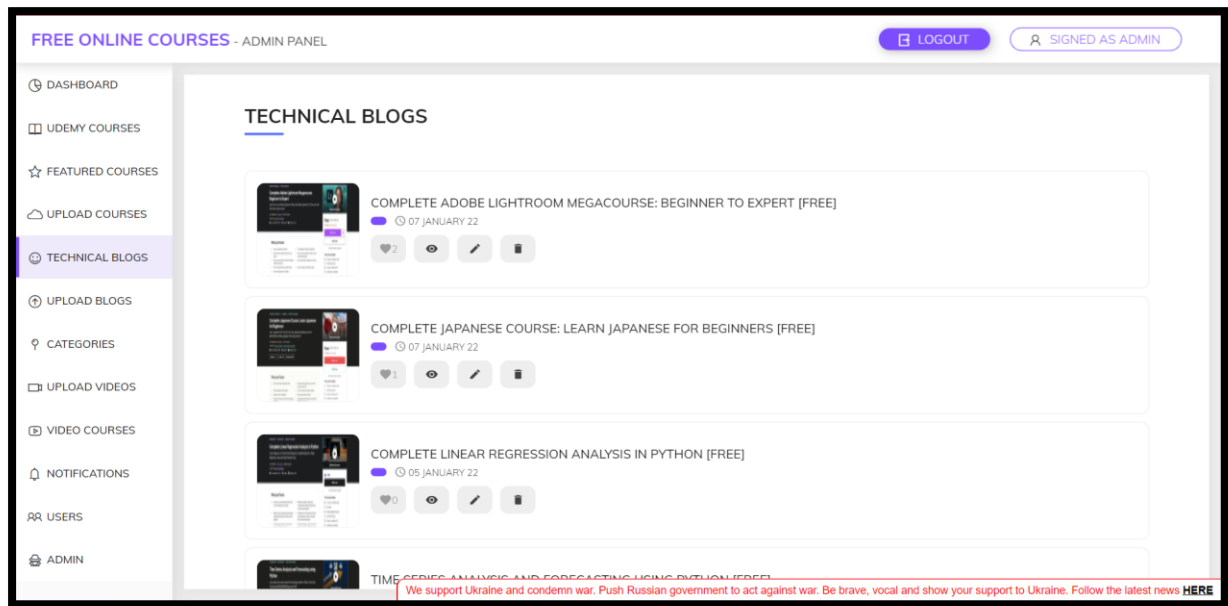
IMAGE 02

IMAGE 03

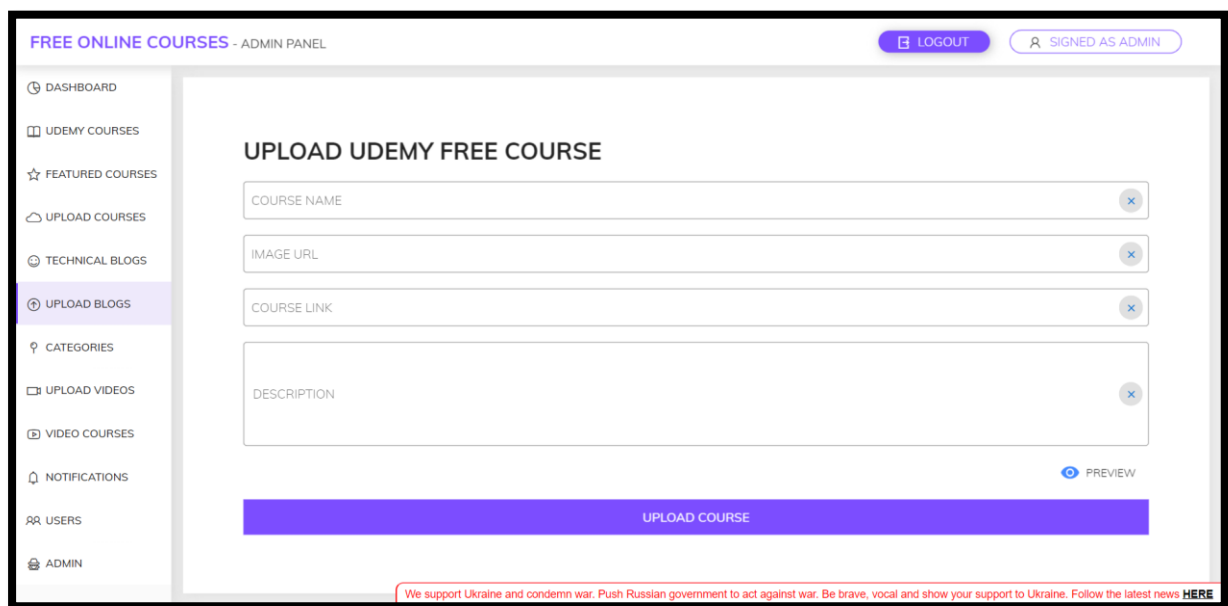
COURSE DETAILS

We support Ukraine and condemn war. Push Russian government to act against war. Be brave, vocal and show your support to Ukraine. Follow the latest news [HERE](#)

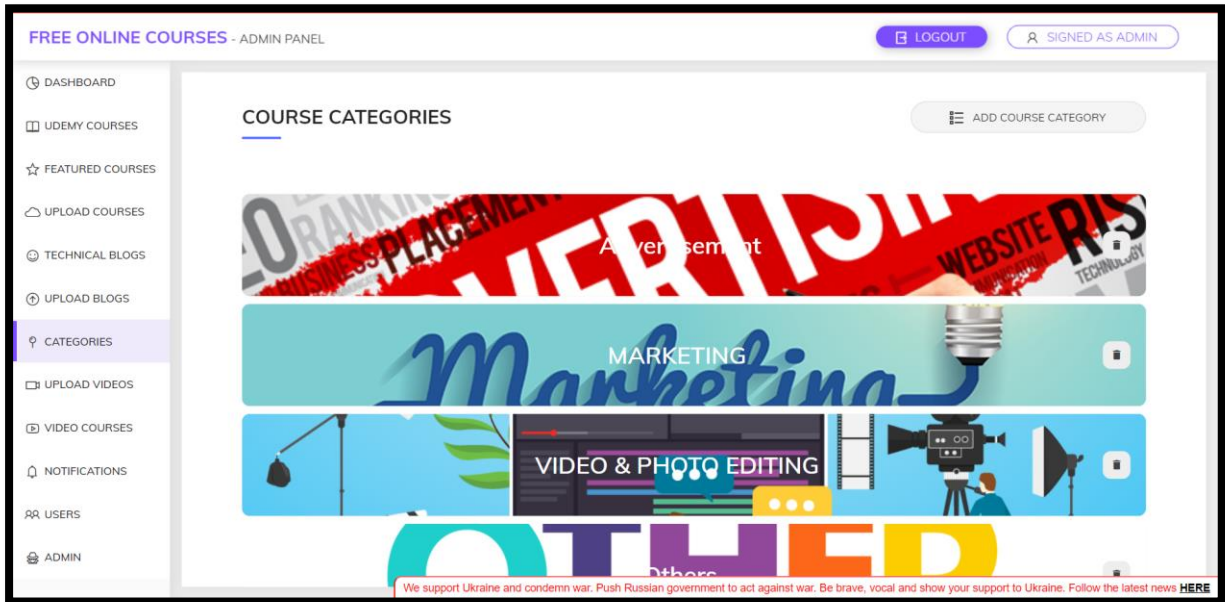
5. Technical Blogs



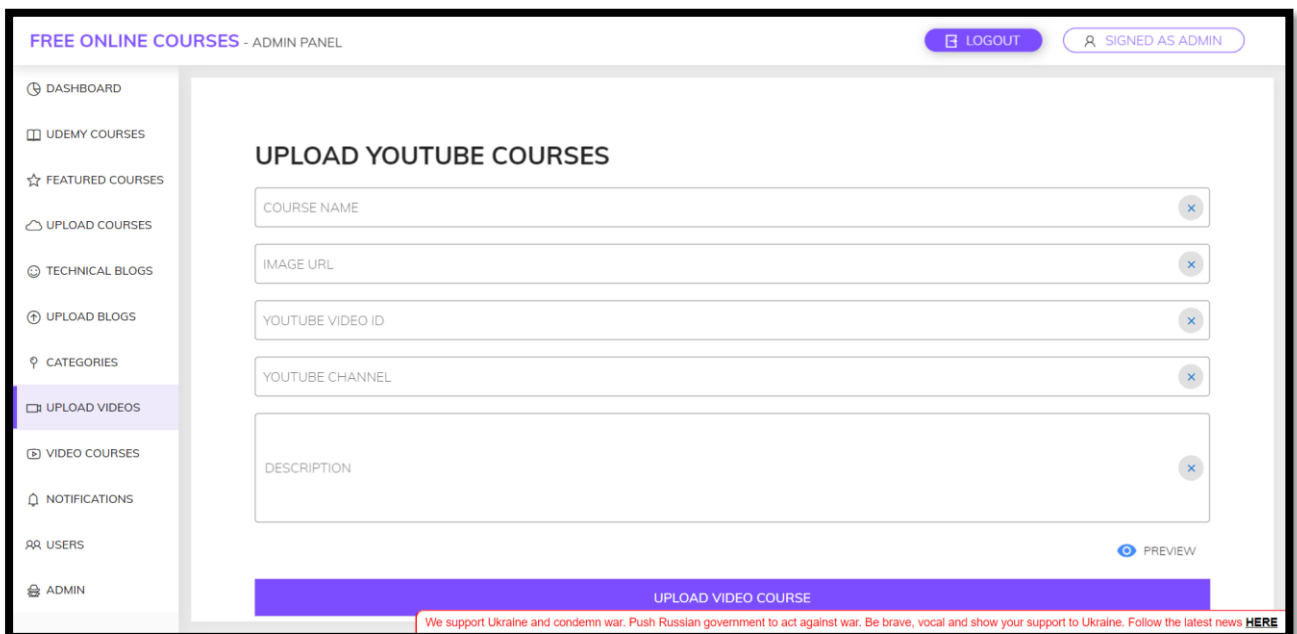
6. Uploaded Blogs



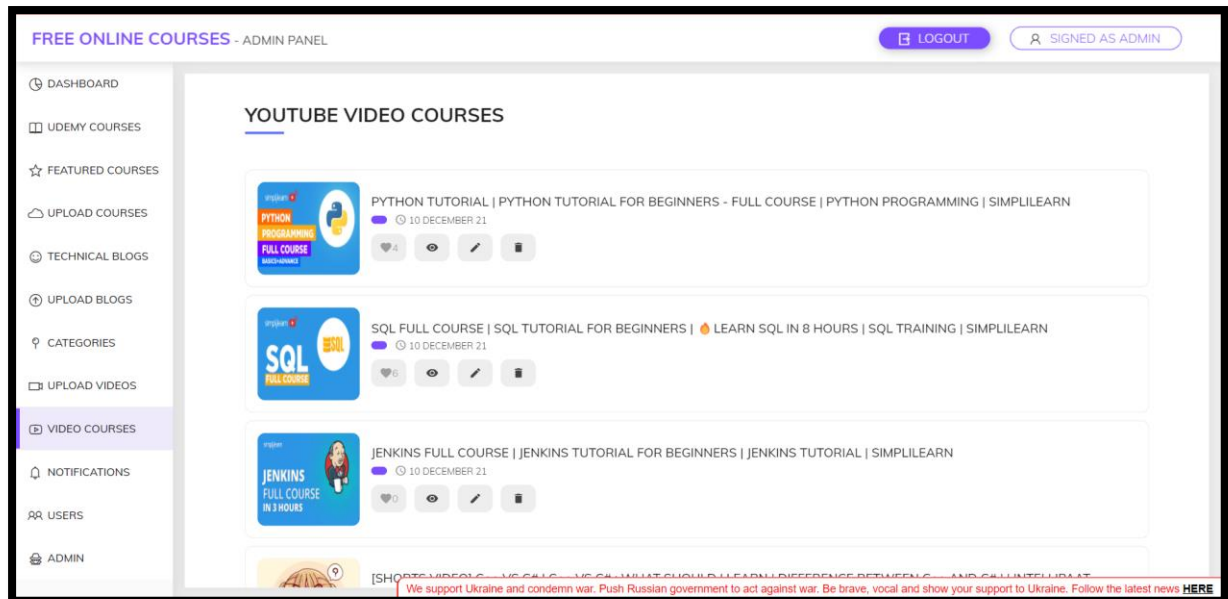
7. Category



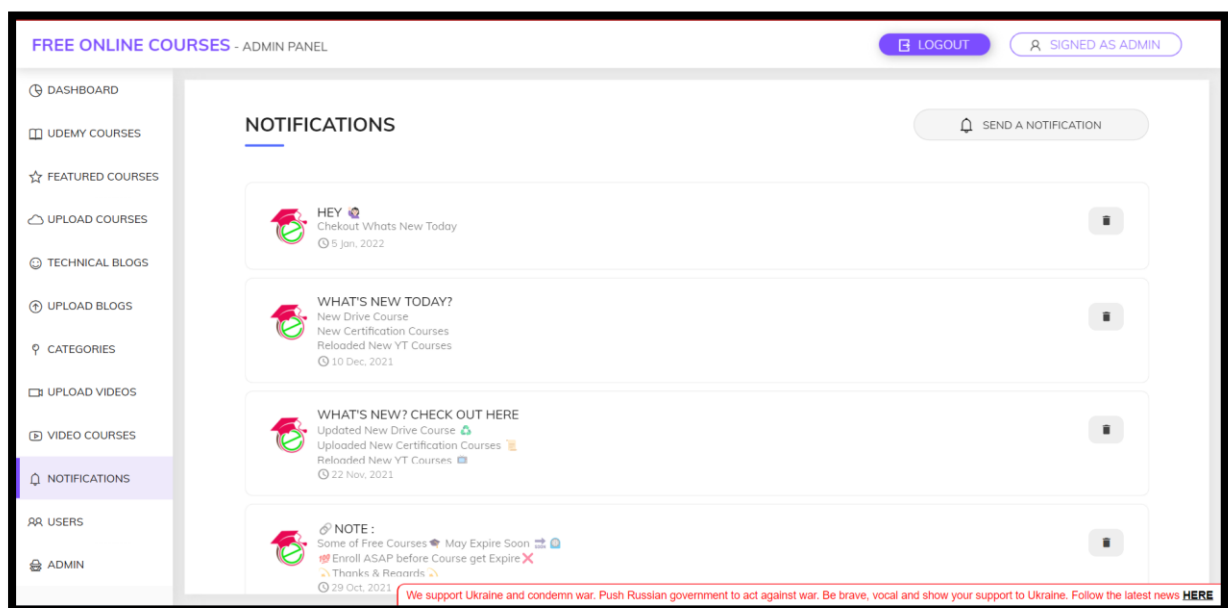
8. Upload Video Courses (Youtube).



Cloud Based Free E-Learning Application



9. Notifications



10. Users

FREE ONLINE COURSES - ADMIN PANEL LOGOUT SIGNED AS ADMIN

USERS

Profile Picture	Name	Email	UID
	18-404 nikhil	[redacted]@gmail.com	UID : ycyCDKqFvRZ4pWbZAw2KnzSElt2
	1814O13 Bhuvaneshwari M C	[redacted]@gmail.com	UID : V6yKvRjyWwZdgSP1wryKnONd2
	18MEC018 Sivasankaran	[redacted]@gmail.com	UID : VV2dcWfVZVHwza9ik19yJL9ne02
	18UGPH009 Pranav K K	[redacted]@gmail.com	UID : UcsjMYN9bR2o1yI9nUNeGEsirt2
	19-205 lavakushareddy	[redacted]@gmail.com	UID : zgSiu3mQybM3tejux9KwDLfjG6x2
	19-4C2 Swapna patnam	[redacted]@gmail.com	UID : WPhRlJTZokcbTZkn4gYaupQllxu2
	19131A0496 KANDREGULA YAMINI	[redacted]@gmail.com	UID : 4cpyBOoY5hO0ic4VqIBOoZQO33

We support Ukraine and condemn war, Push Russian government to act against war. Be brave, vocal and show your support to Ukraine. Follow the latest news [HERE](#)

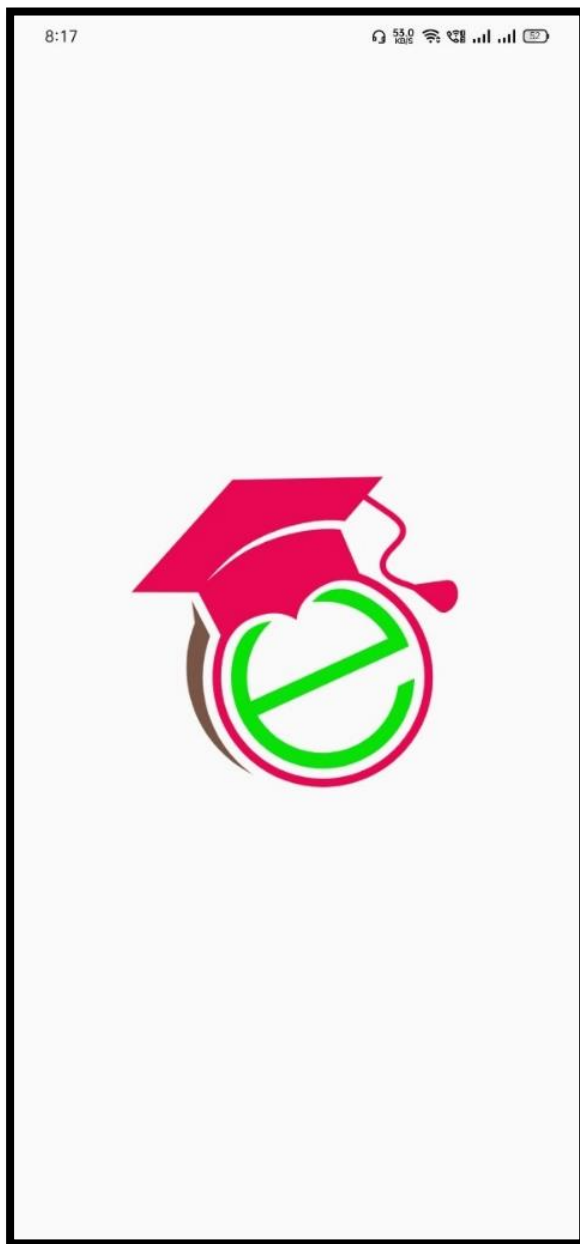
Result of TCSCCS Android Application

Link for Android Application :-

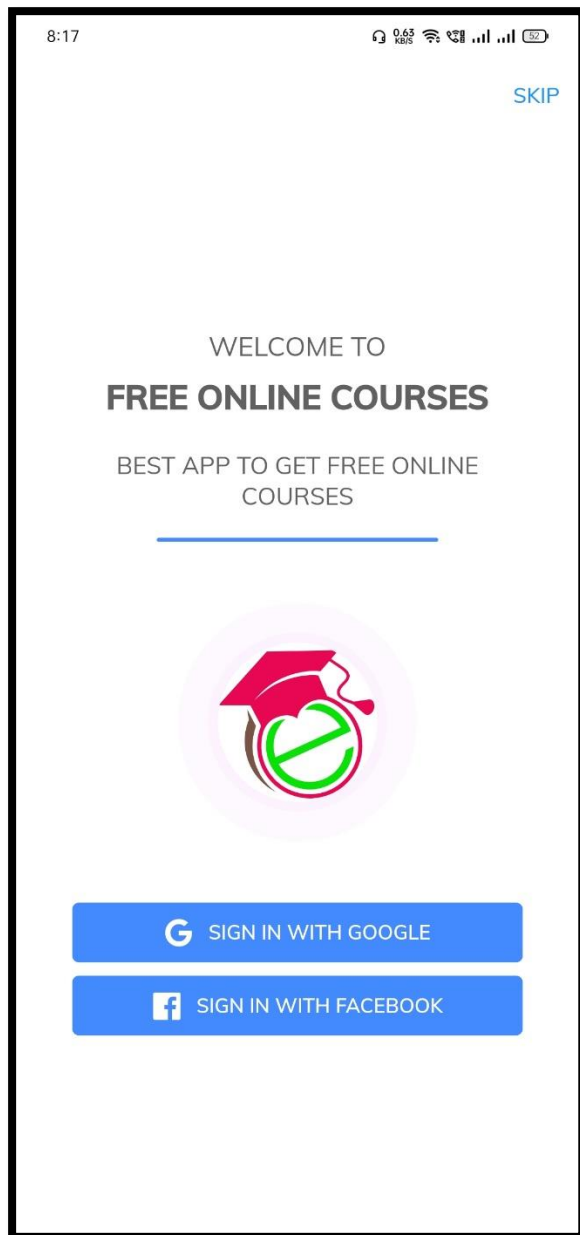
<https://play.google.com/store/apps/details?id=net.manish.blog>

Fig 15: Snapshots of User Application.

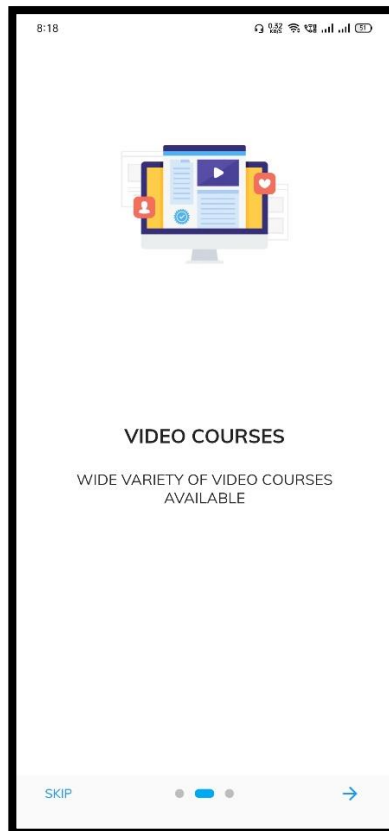
1. App Free Udemy Course



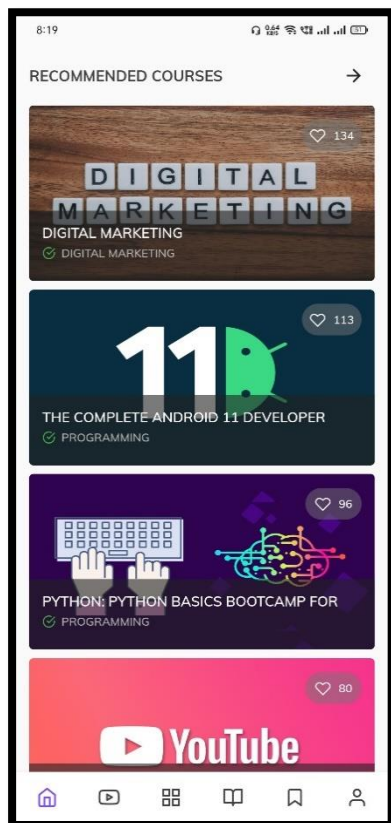
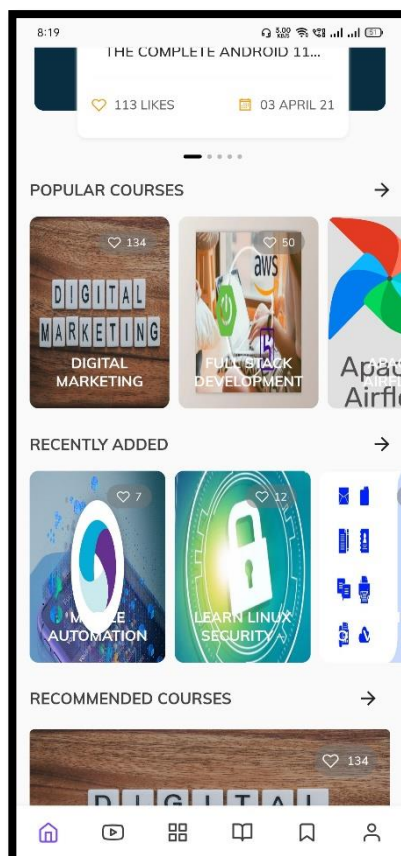
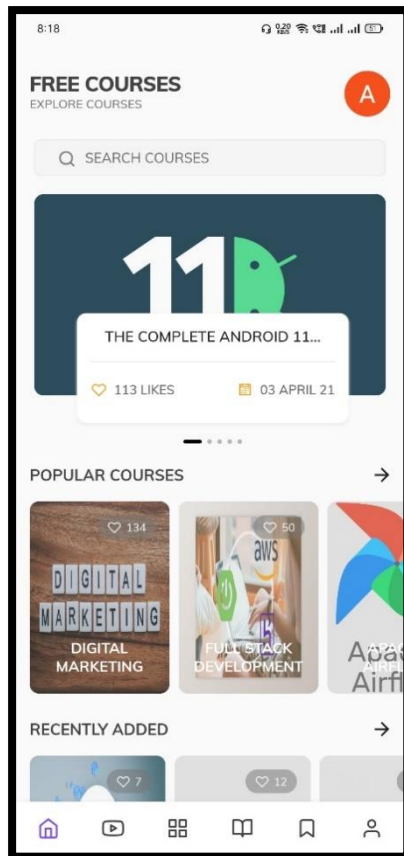
2. Google and Facebook Login/Signup



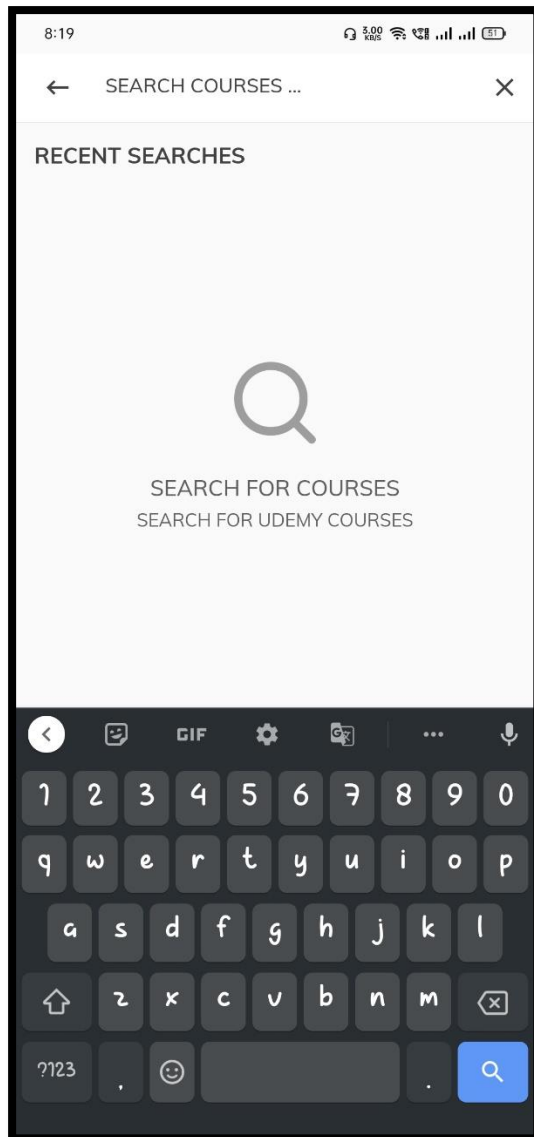
3. Splash Screen.



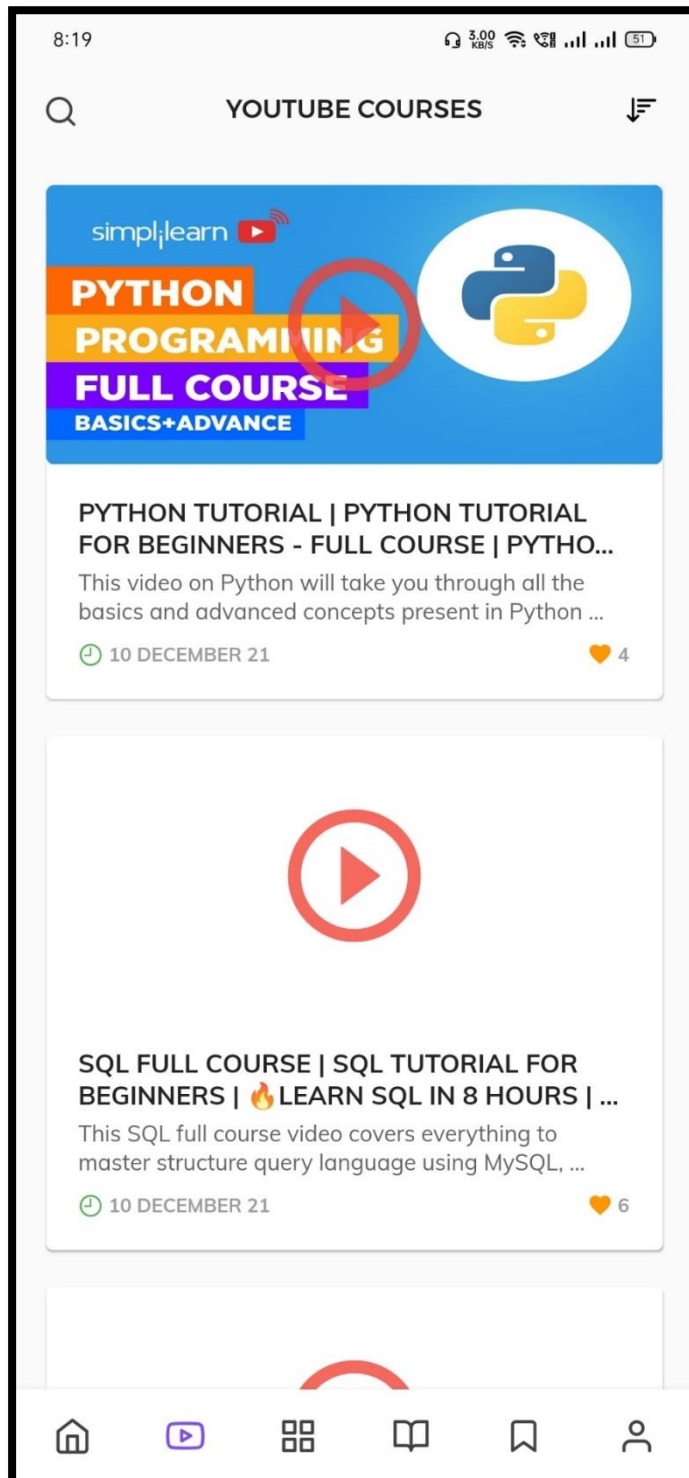
4. Home Page



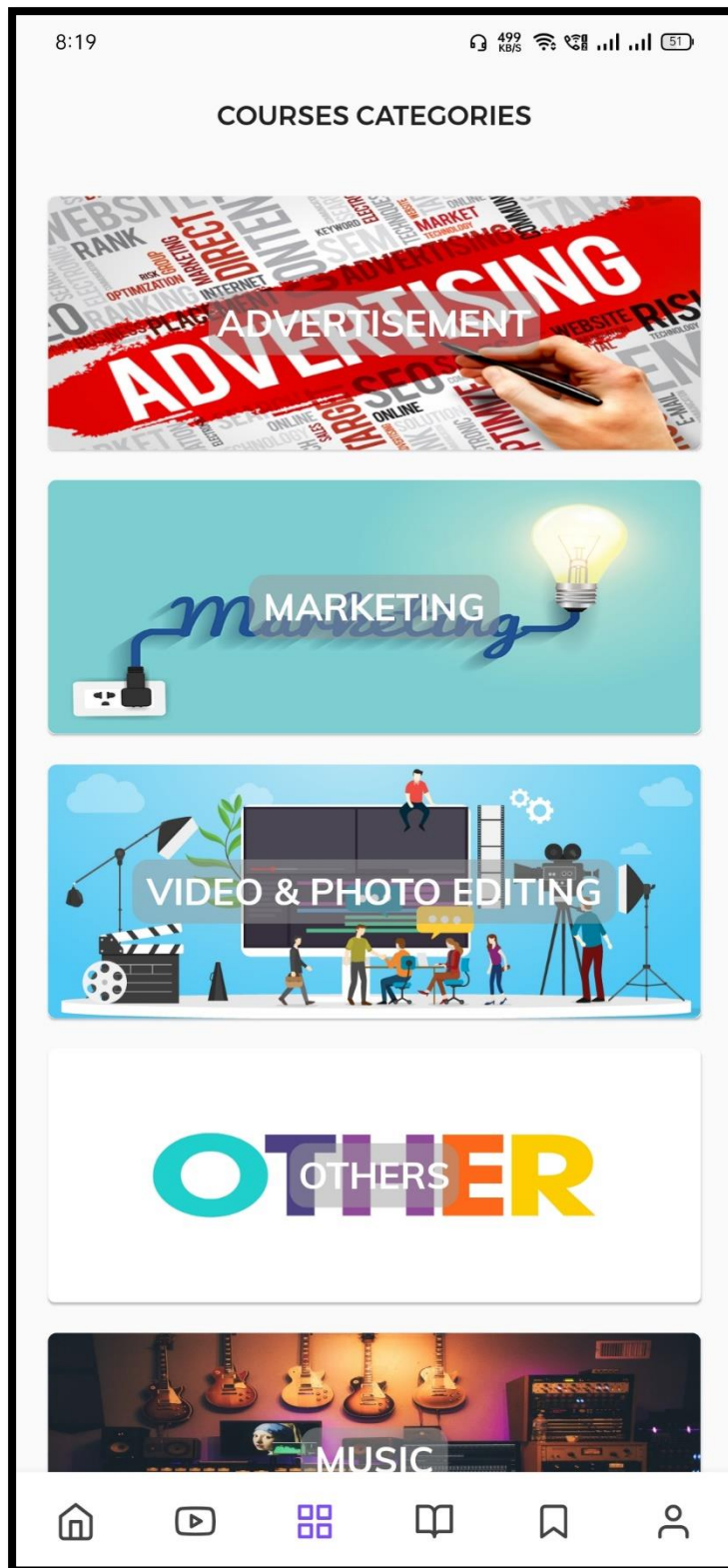
5. Search Courses



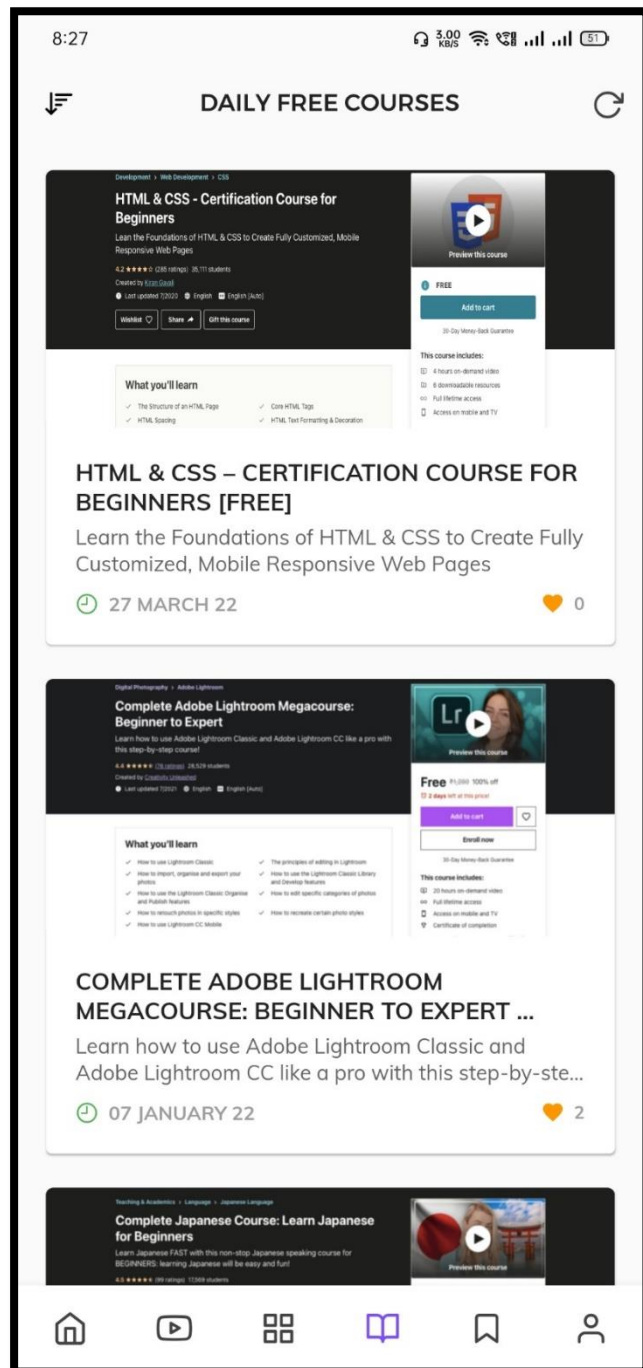
6. YouTube Courses



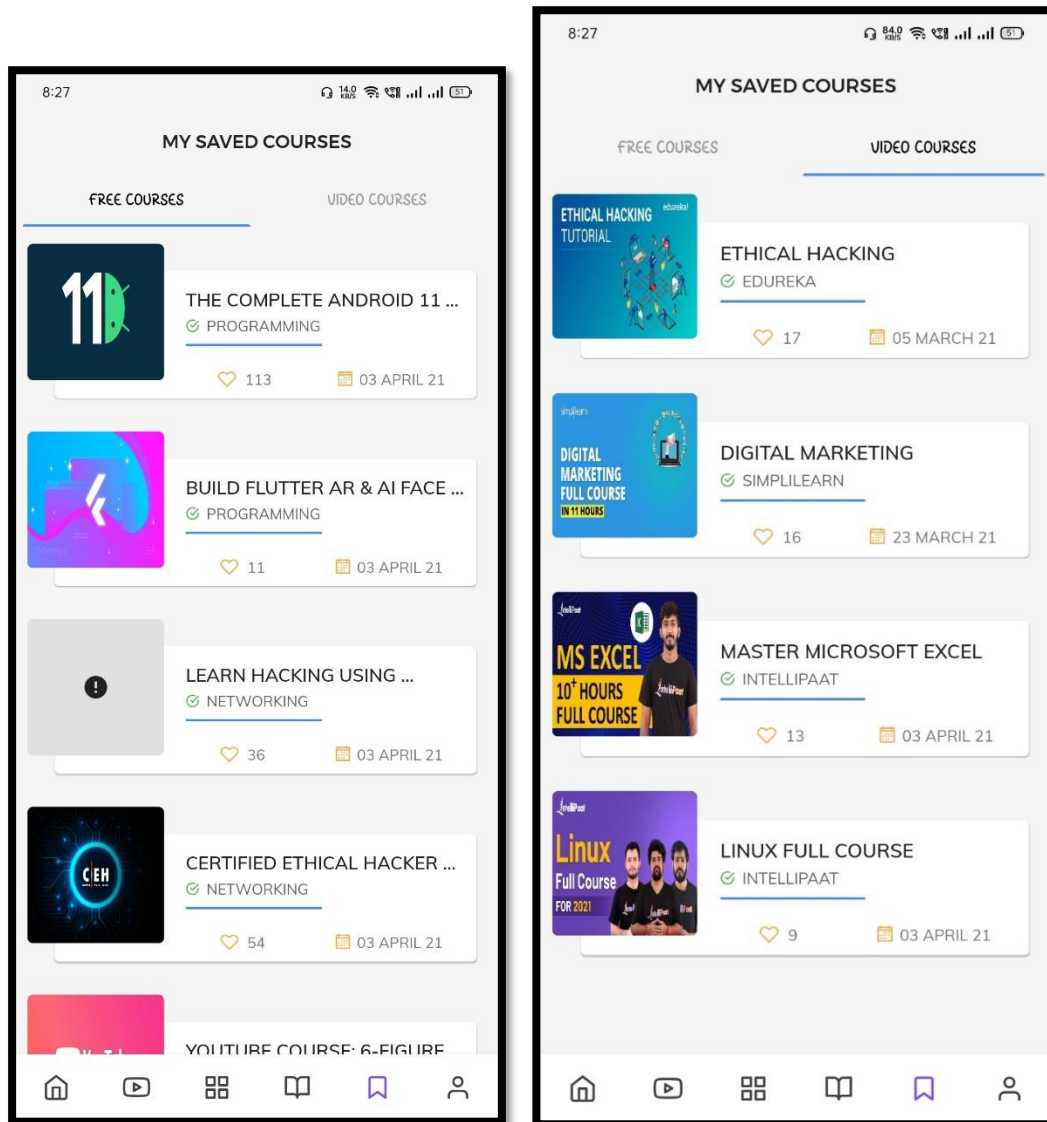
7. Courses Category



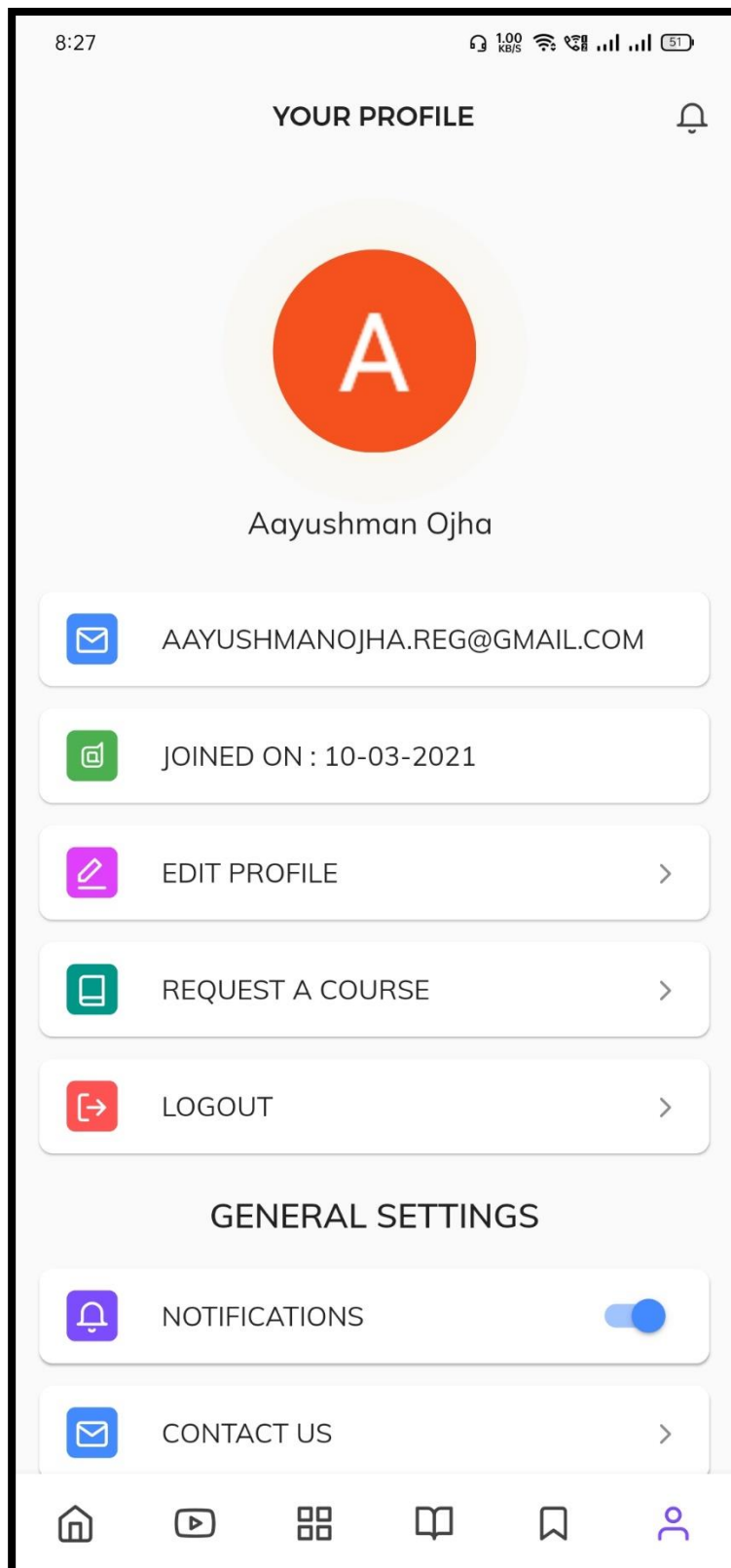
8. Daily Free UdemY Course Page



9. Saved Courses and videos



10. User Profile and Setting Page



Conclusion

Our project is only a humble venture to satisfy the needs to access free Course from. Several Platform like Udemy, Coursera, edx, Udacity. This Application shall prove to be a powerful Application in satisfying all the requirements of the Courses. The objective of Application is to provide a Platform that enables to Learn new technology and software for their define purposes. This application made within a limited time frame at the beginning of the software project and should be updated regularly as the project progresses.

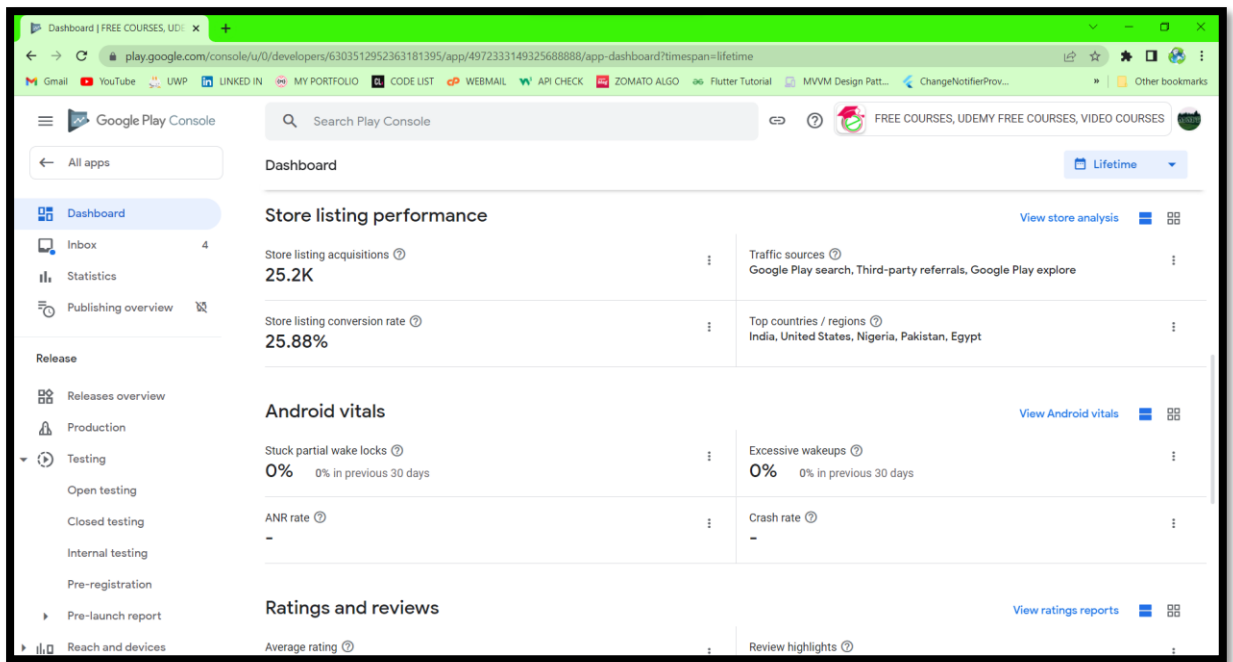


Fig 16: Overall User acquisitions Report.

Future Scope of Project

In a nutshell, it can be summarized that the future scope of the project circles around maintaining information regarding:

We can add following features in future.

- We can give more advance Web based Application for Free E-Learning Application including Website, Progressive Web App, Windows Desktop Support and Mac Os.
- We will make Website and host on online servers to make it accessible worldwide
- Integrate multiple load balancers to distribute the loads of the system

Cloud Based Free E-Learning Application

- Create the master and slave database structure to reduce the overload of the database queries
- Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers

At the very Beginning Stage of Development Area Android application is Developed Later on the Web Version and Desktop version will be going to released Soon.

The above mentioned points are the enhancements which can be done to increase the applicability and usage of this project. Here we can maintain the records of Assignment and Student. Also, as it can be seen that now-a-days the players are versatile, i.e. so there is a scope for introducing a method to maintain the E-learning Application.

Enhancements can be done to maintain all the Courses, Accounts.

We have left all the options open so that if there is any other future requirement in the system by the user for the enhancement of the system then it is possible to implement them In the last we would like to thanks all the rating and reviews received from existing users in the development of the system directly or indirectly. We hope that the project will serve its purpose for which it is develop there by underlining success of process.

Although I have put my best efforts to make the software flexible, easy to operate but limitations cannot be ruled out even by me. Though the software presents a broad range of options to its users some intricate options could not be covered into it; partly because of logistic and partly due to lack of sophistication. Paucity of time was also major constraint, thus it was not possible to make the software fool proof and dynamic. Lack of time also compelled me to ignore some part such as storing old result of the candidate etc.

Considerable efforts have made the software easy to operate even for the people not related to the field of computers but it is acknowledged that a layman may find it a bit problematic at the first instance. The user is provided help at each step for his convenience in working with the Application

References and Bibliography:

DOCUMENTATION	https://flutter.dev/ https://dart.dev/ https://developer.android.com/ https://firebase.google.com/ http://cloud.google.com/ https://play.google.com/store/apps/details?id=net.manish.blog
ERROR AND DOUBTS	https://github.com/ https://stackoverflow.com/ https://www.quora.com/