MINI PROJECT(2020-21)

Build and deploy an e-Learning Android Application.

FINAL REPORT

Institute of Engineering & Technology

Submitted By:

- **Hrithik Bandil (181500274)**
- **Prashant Verma (181500493)**
- **Shashank Yadav (181500658)**

Supervised By:

Dr. Manoj Varshney Sir

(Assistant Professor)

(Department of Computer Engineering & Applications)

CONTENTS

Abstract

Paathshala is basically an e-learning application meant for students. There are many e-learning sites and applications running today. **Paathshala** is quite different from others.

This e-learning application provides valuable content to the students. It provides a better platform for learning. Apart from the content it also consists of Weekly Assessment, Quizzes, Questionnaires and Puzzles as well. It also has a Quora section in which students are free to ask doubts related to their course or subjects. It also has a section for placement related questions for different companies. This application provides mock test series for placement in different companies. Students can give feedback regarding the content delivered so that we can make the changes required according to the students.

Quizzes and Puzzles will help in increasing the interest of the student and will help in learning. We are also providing video lectures for different subjects and topics given by known Professors

INTRODUCTION

Present 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. The main objective of the E-Learning is 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 application 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.

Proposed System:

By this application it becomes easier way of communication between the student and the teacher.

This project used by two types of user:-

- Students
- Teachers

1.1 Overview and Motivation:

Time is one of the issue that learners and teachers both have to face in learning. E-learning, on the other hand, facilitates learning without having to organize when and where everyone who is interested in a course can be present. Designing a course in a way that makes it interactive and fun through the use of Quizzes and Puzzles.

It will help students to view the lectures as many times as the student want. Students can note down the important points which becomes difficult for some students in the class. It will also be helpful for the preparation of placement by solving mock test papers.

1.2 Objective:

The aim of PATHSHALA is to provide better platform for learning. It enhance the quality of learning and teaching. Improve the efficiency and effectiveness. Improve user- accessibility and time flexibility to engage **learners** in the **learning** process.

1.3 Organization of Project Report:

PHASES	TIME DURATION
Software requirement specification	2
weeks System design	3
weeks	
Coding	5 weeks
Testing	2 weeks
Documentation	2 weeks
Implementation	1 weeks

SOFTWARE REQUIREMENT ANALYSIS

System Analysis is a detailed study of the various operations performed by a system and their relationship within and outside the system .It is a systematic technique that defines goals and objectives the goal of the development is to deliver the system in the line with the user's requirements, and analysis is this process.

System study has been conducted with the following objectives in mind: -

- Identify the client's need.
- Evaluate the system concept for feasibility.
- Perform economical and technical analysis.
- Allocate functional to hardware, software, people, database and other system elements
- Establish cost and schedule constraints.
- Both hardware and software expertise is required to successfully attain the objectives.

2.1 Requirement Analysis

Information gathering is usually the first phase of the software development project. The purpose of this phase is to identify and document the exact requirements for the system. The user's request identifies the need for a new information system and on investigation re-defined the new problem to be based on MIS, which supports management. The objective is to determine whether the request is valid and feasible before a recommendation is made to build a new or existing manual system continue

The major steps are –

- Defining the user requirements.
- Studying the present system to verify the problem.
- Defining the performance expected by the candidate to use requirement

2.1.1 Hardware Requirements

Processor	:	Intel Core i5
Processor Speed	:	1.5
GHZ RAM	:	4 GB
Hard Disk	:	20 GB of free space

2.1.2 Software Requirements

Operating System	: Window 10
Front End	: XML
Back End	: Java

2.1.3 Tools and

Technology Tools:

- Windows 10
- Android Studio
- Google Firebase

Technology:

- **XML** is a Extensible Markup Language (**XML**) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

- **Java Script:** JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow to make dynamic pages.

2.2 Feasibility Study

Feasibility study is the process of determination of whether or not a project is worth doing. Feasibility studies are undertaken within tight time constraints and normally culminate in a written and oral feasibility report. I have taken a fixed time in feasibility study with my co-developer. The contents and recommendations of this feasibility study helped us as a sound basis for deciding how to precede the project. It helped in taking decisions such as which software to use, hardware combinations, etc.

2.2.1 Technical feasibility:

This is concerned with specifying equipment of software and hardware that will successfully satisfy the user requirements. The technical needs of the system may vary considerably, but might include:

- The facility to produce output in a given time.
- Response time under certain condition.
- Ability to produce a certain volume of transaction at a particular speed.
- In examining technical feasibility, configuration of the system is given more importance than the actual make of hardware. The configuration should give the complete picture about the system requirements. What speeds of input and output should be achieved at particular quality of printing.

According to the definition of technical feasibility the compatibility between front-end and back-end is very important. In our project the compatibility of both is very good. The degree of compatibility of PHP and SQL Server 2014 is very good. The speed of output is very good when we enter the data and click button then the response time is very fast and give result very quick. In ever find difficulty when we use complex query or heavy transaction. The speed of transaction is always smooth and constant. This software provides facility to communicate data to distant location.

We use Active Server Pages and JavaScript. The designing of front-end of any project is very important so we selected Active Server Pages, HTML & CSS as front-end duet following reason:

- Easy implementation of code.
- Well define interface and database.
- Well define hand shaking of SQL Server2014

At present scenario the no of backend are available but I have selected SQL Server 2008 because of the following number of reasons.

- Able to handle large data.
- Security.
- Robust RDBMS
- Backup &Recovery

Withthehelpofabovesupportweremovedefectofexistingsoftware.Infuturewe can easily switch over any platform. To ensure that system does not halt in case of undesired situation or events. Problem effected of any module does not affect any module of the system. A change of hardware does not produce problem.

2.2.2 Operational Feasibility:

It is mainly related to human organizational and political aspects. The points to be considered are:

- What changes will be brought with the system?
- What organization structures are distributed structures are distributed.
- What new skills will be required? Do the existing staff members have these skills? If not, can they be trained in due course of time?

At present stage all the work is done manually. So, throughput and response time is too much. Major problem is lack of security check that should have been applied.

Finding out the detail regarding user's request was very difficult, because data store was in different registers and different places. In case of any problem, no one can solve the problem until the person responsible is not present.

Current communication is entirely on telephonic conversation or personal meetings. Post computerization staff can interact using internet.

Now, we will explain the last point of operational feasibility i.e. handling and keeping of software , at every point of designing I will take care that menu options arenottoocomplexandcanbeeasilylearnedandrequiredleastamountoftechnical skills as operators are going to be from non-computers background.

2.2.3 Economic feasibility:

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as cost/benefit analysis: the procedure is to determine the benefits and saving that are expected from a proposed system and compare them with cost. If benefits out weighs cost, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have a chance of being approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle.

At present Company has ten systems with following configuration:

- Ram 4 GB or above for fast execution and reliability
- MOTHER Board x64 based PC
- Color Monitor 14” and17”
- Hard Disk 100GB
- Hence the economic feasibility is very good.

2.3 Analysis

System analysis is the first step towards the software building process. The purpose of system analysis is to understand the system requirements, identify the data, functional and behavioral requirements and building the models of the system for better understanding of the system.

In the process of system analysis one should first understand that, what the present system is, is how it works(i.e. processes).After analyzing these points we become able to identify the problems in the present system. Upon evaluating current problems and desired information (input and output to the system), the analyst looks towards one or more solutions. To begin with, the data objects, processing functions, and behavior of the system are defined in detail. After this models, from three different aspects of the system-data, function and behavior. The models created during the system analysis process helps in better understanding of data and control flow, functional processing, operational behavioral and information content.

2.4 Summary of Modules

a) Administrator

b) Login

Administrator

The administrator is responsible for maintaining Database of web portal. This module will update information of students.. Administrator will also manage the records for students. Administrator will provide the user-id and password for the students and teachers.

Login

Login module refers to authenticating the user, administrator and assignee and granting the access to their account. They can login with their registered username and password to do their work.

SOFTWARE DESIGN

A software design document (SDD) is a written description of a software product, that a software designer writes in order to give a software development team overall guidance to the architecture of the software project. An SDD usually accompanies an architecture diagram with pointers to detailed feature specifications of smaller pieces of the design. Practically, a design document is required to coordinate a large team under a single vision. A design document needs to be a stable reference, outlining all parts of the software and how they will work. The document is commanded to give a fairly complete description, while maintaining a high-level view of the software.

There are two kinds of design documents called HLDD (high-level design document) and LLDD (low-level design document).

The SDD contains the following documents:

1. The **data design** describes structures that reside within the software. Attributes and relationships between data objects dictate the choice of data structures.
2. The **architecture design** uses information flowing characteristics, and maps them into the program structure. The transformation mapping method is applied to exhibit distinct boundaries between incoming and outgoing data. The data flow diagrams allocate control input, processing and output along three separate modules.
3. The **interface design** describes internal and external programinter faces, as well as the design of human interface. Internal and external interface designs are based on the information obtained from the analysis model.
4. The **procedural design** describes structured programming concepts using graphical, tabular and textual notations. These design mediums enable the designer to represent procedural detail that facilitates translation to code. This blueprint for implementation forms the basis for all subsequent software engineering worked.

3.1 Data Flow Diagram(DFD)

The Data Flow Diagram (DFD)is a graphical representation of the flow of data through an information system. It enables you to represent the processes in your information system from the viewpoint of data. The DFD lets you visualize how the system operates, what the system accomplishes and how it will be implemented, when it is refined with further specification.

- Data flow diagrams are used by systems analysts to design information-processing systems but also as a way to model whole organizations. You build a DFD at the very beginning of your business process modeling in order to model the functions your system has to carry out and the interaction between those functions together with focusing on data exchanges between processes.You can associate data with conceptual, logical, and physical data models and object-oriented models.

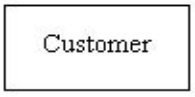
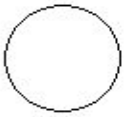
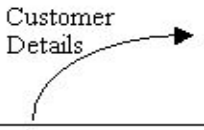
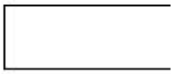
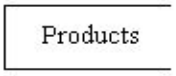
Name	Symbol	Description	Example
Entity		Used to represent people and organizations outside the system. They either input information to the system, accept output information from the system or both	
Process		These are actions that are carried out with the data that flows around the system. A process accepts input data and produces data that it passes on to another part of the DFD	
Data Flow		These represent the flow of data to or from a process	
Data Store		This is a place where data is stored either temporarily or permanently	

Fig 3.1: Data Flow Diagram Symbols

0-LEVEL DFD

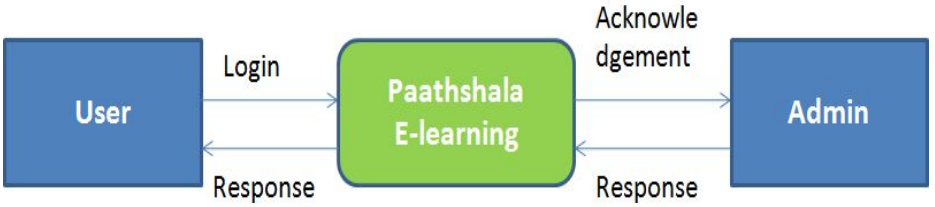
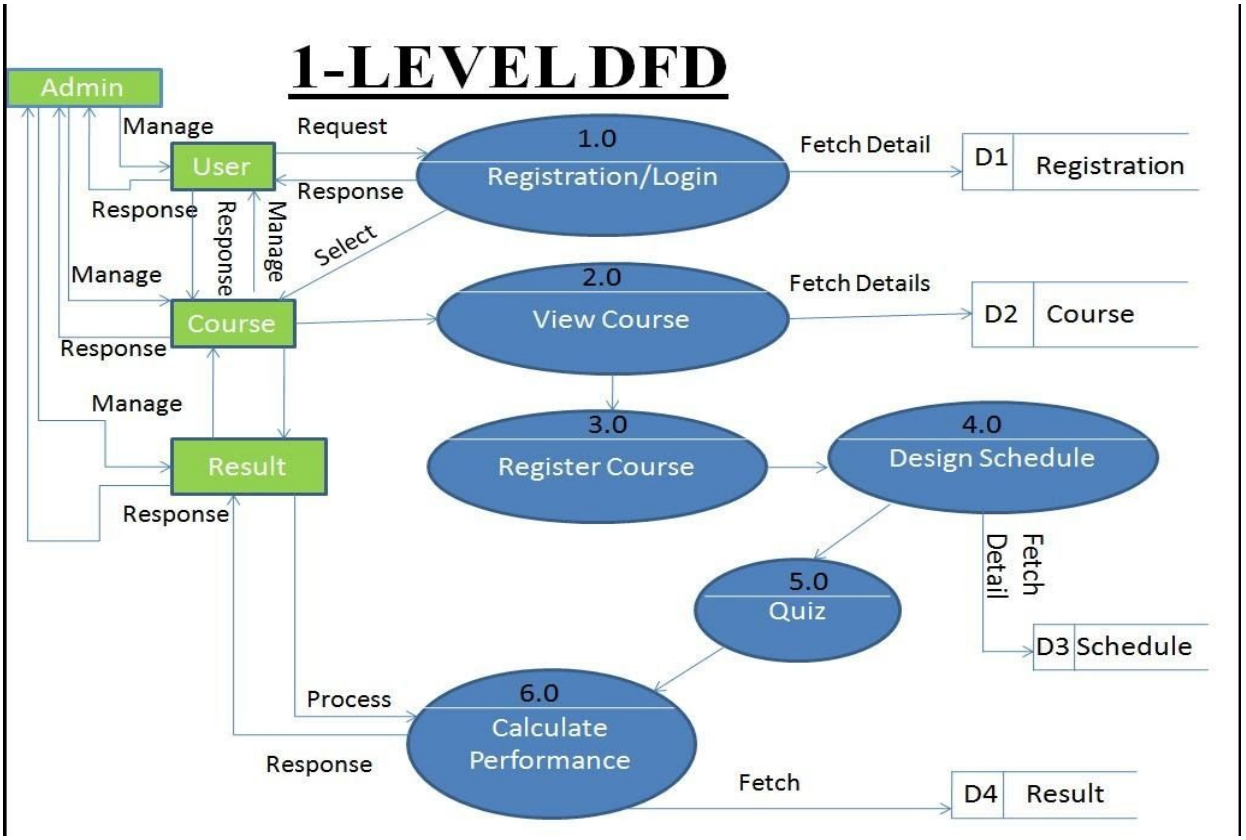


Fig 3.2

Fig 3.3


1-LEVEL DFD
Admin
Manage
User
Request
Response
1.0
Registration/Login
Fetch Detail
D1
Registration
Response
Response
Manage
Select
Manage
Course
2.0
View Course
Fetch Details
D2
Course
Response
Manage
Result
3.0
Register Course
4.0
Design Schedule
Response
Fetch
Detail
5.0
Quiz
D3
Schedule
Process
6.0
Calculate
Performance
Response
Fetch
D4
Result

3.2 Entity Relationship Diagram(ER-Diagram)

An entity-relationship diagram (ERD) is a graphical representation of an information system that shows the relationship between people, objects, places, concepts or events within that system. An ERD is a data modeling technique that can help define business processes and can be used as the foundation for a relational database.

While useful for organizing data that can be represented by a relational structure, an entity-relationship diagram can't sufficiently represent semi-structured or unstructured data, and an ERD is unlikely to be helpful on its own in integrating data into a preexisting information system.

Three main components of an ERD are the entities, which are objects or concepts that can have data stored about them, the relationship between those entities, and the cardinality, which defines that relationship in terms of numbers.

- **Components of the ER Model**

The three main components of the ER Model are **entities**, **attributes** and **relationships**.

- In ERM terms, an entity is a "thing" within the organization that we want to keep information about, such as a customer, employee or course. In other words, an entity in an ERM actually refers to a table, and rows within the table are referred to as entity occurrences. Entities are represented by rectangles containing the name of the entity. Entity names must be singular and in capital letters.
- Each entity has attributes which are the properties of each entity. Attributes will be implemented as columns in the tables. Each attribute has a domain which specifies the set of possible values an attribute can have. For instance, the range of values for a telephone extension may be specified as a set of integer numbers between 4000 and 4999. An attributes domain is not displayed in ER diagrams, but is recorded in the data dictionary.

- Attributes can be of various types. A composite attribute can be subdivided into smaller parts. For example, an attribute Name can be subdivided into First Name and Last Name. Attributes that cannot be subdivided are called simple attributes.FirstNameandLastNamearenowsimpleattributes.Mostattributes have only a single value and as such are called single valued attributes. For example, a Teacher can have only one Last Name or a Subject can have only one Subject Code. Multi valued attributes can have more than one value. For example, a Student could have more than one Certificate or a Department may have several Extensions.
- A key attribute is an attribute that has a unique value for each entity occurrence. In other words, a key attribute is used to identify each row uniquely. For example, a Subject Code will uniquely identify each subject as no two subjects can have the same SubjectCode. Key attributes are represented by underlining its name.

- A relationship is the association between entities or entity occurrences

3.2.1 ER Diagram of PAATHSHALA:

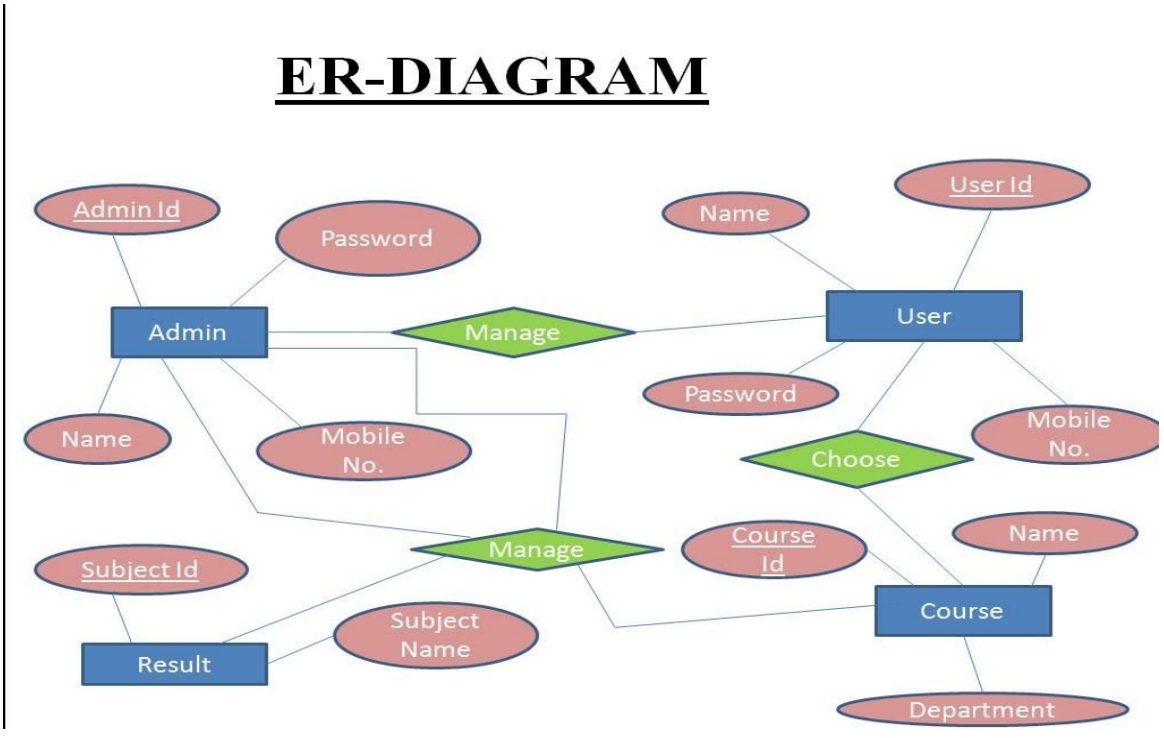


Fig 3.4

3.3 Database Design

A good database design is crucial for a high-performance application, just as an Aerodynamic body is important to a race car. If the car doesn't have smooth lines, it will produce drag and go slower. Without optimized relationships, your database won't perform as efficiently as possible. Thinking about relationships and database efficiency is part of normalization.

Beyond the issue of performance is the issue of maintenance—your database should be easy to maintain. This includes storing only a limited amount (if any) of repetitive data. If you have a lot of repetitive data and one instance of that data undergoes a change (such as a name change), that change has to be made for all occurrences of the data. To eliminate duplication and enhance your ability to maintain the data, you might create a table of possible values and use a key to refer to the value. That way, if the value changes names, the change occurs only once in the master table. The reference remains the same throughout other tables.

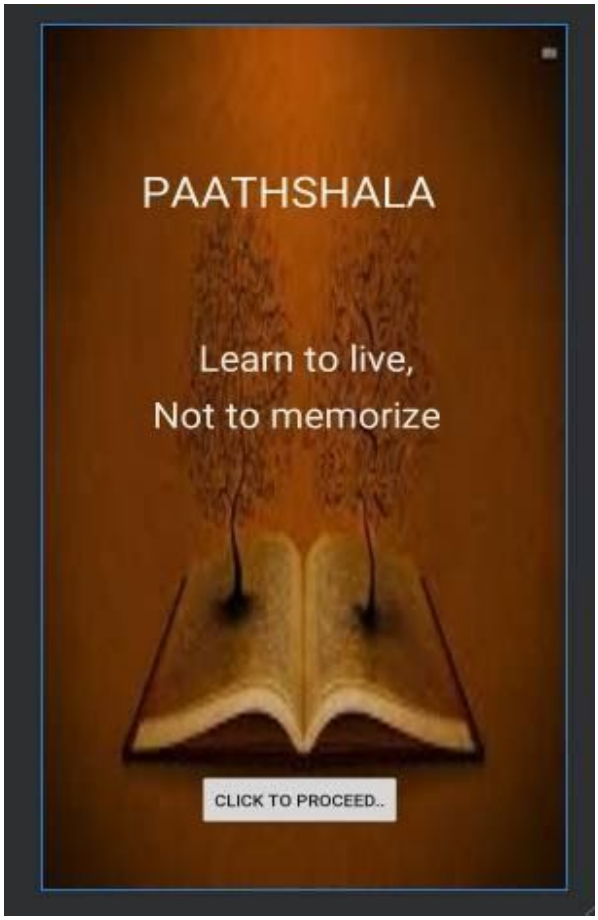
Attribute	Type	Description	Constraints
Username	Varchar(30)	Username of Admin	Primary Key
Password	Varchar(30)	Password of Admin	Not Null
Id	Int(255)	Id of admin	Not Null
Mobile no	Int(255)	Contact of admin	Not Null

Table 3.2: Admin login

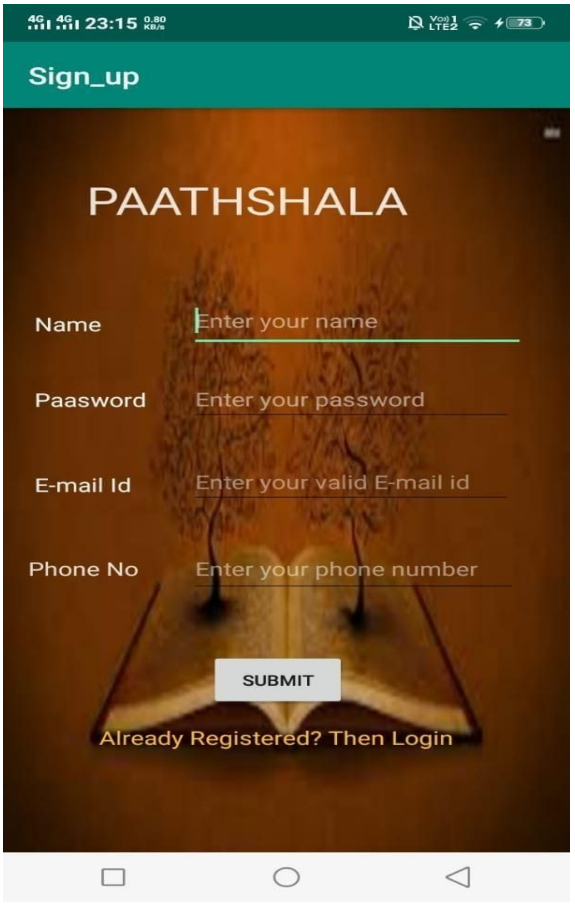
Attribute	Type	Description	Constraints
ID	Int(255)	ID of user	Primary Key
Name	varchar(255)	Name of user	Not Null
mobile	Int(255)	Contact of user	Not Null
Pwsd	varchar(255)	Password of user	Not Null

Table 3.3: User table

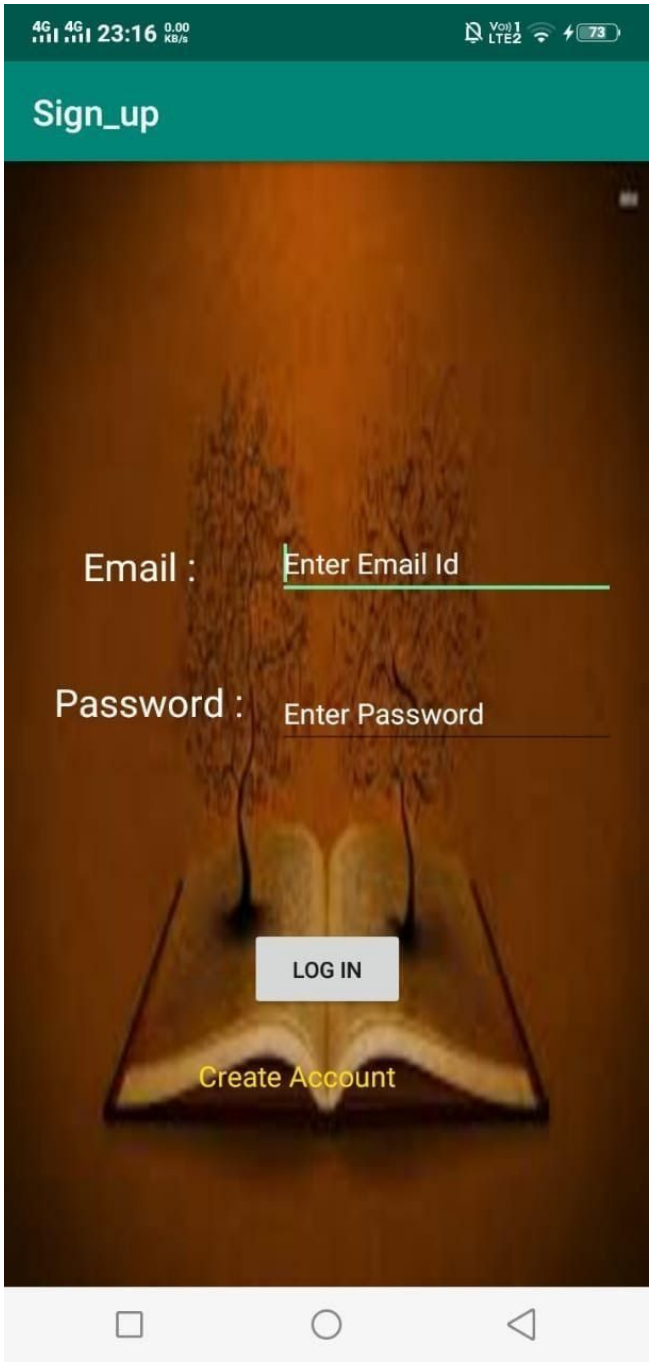
IMPLEMENTATION &USER INTERFACE

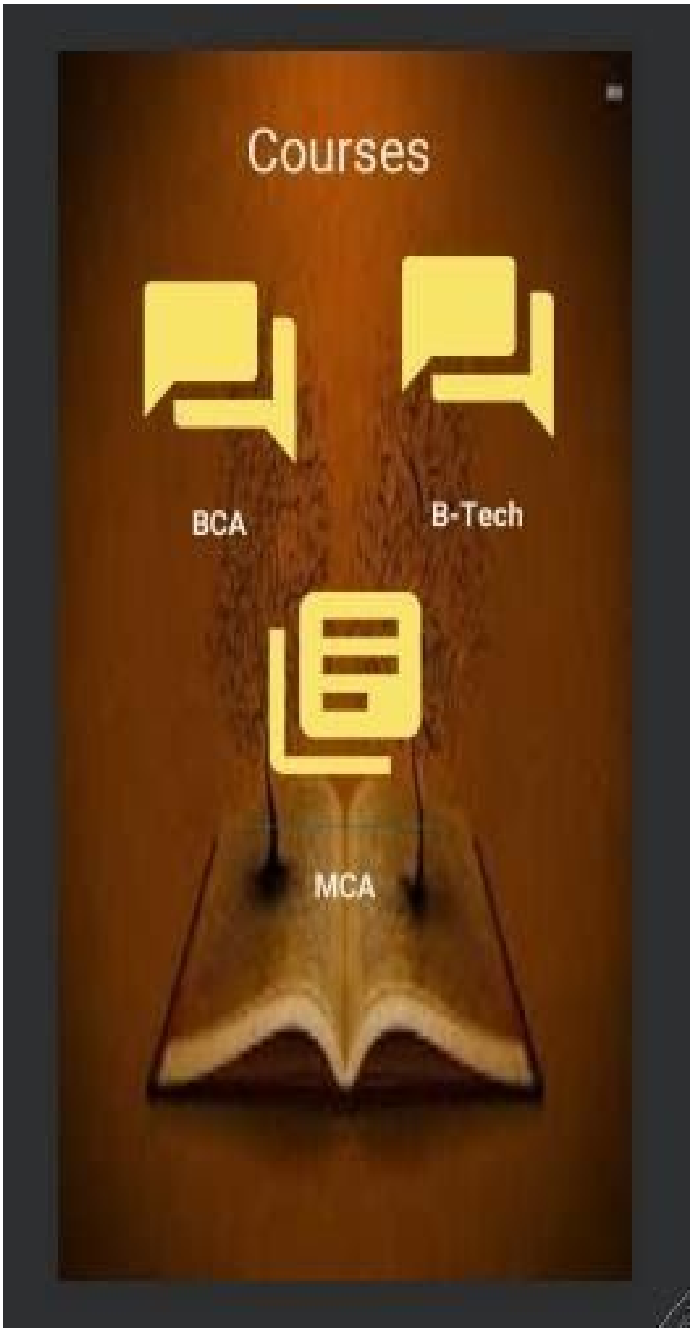


Landing Page

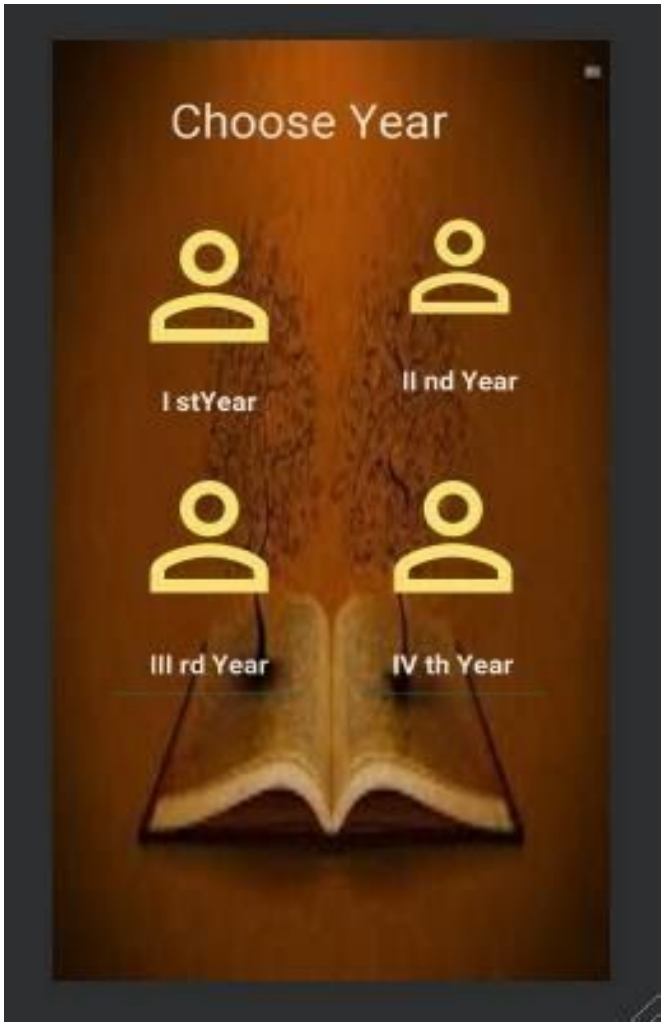


SignUp Page

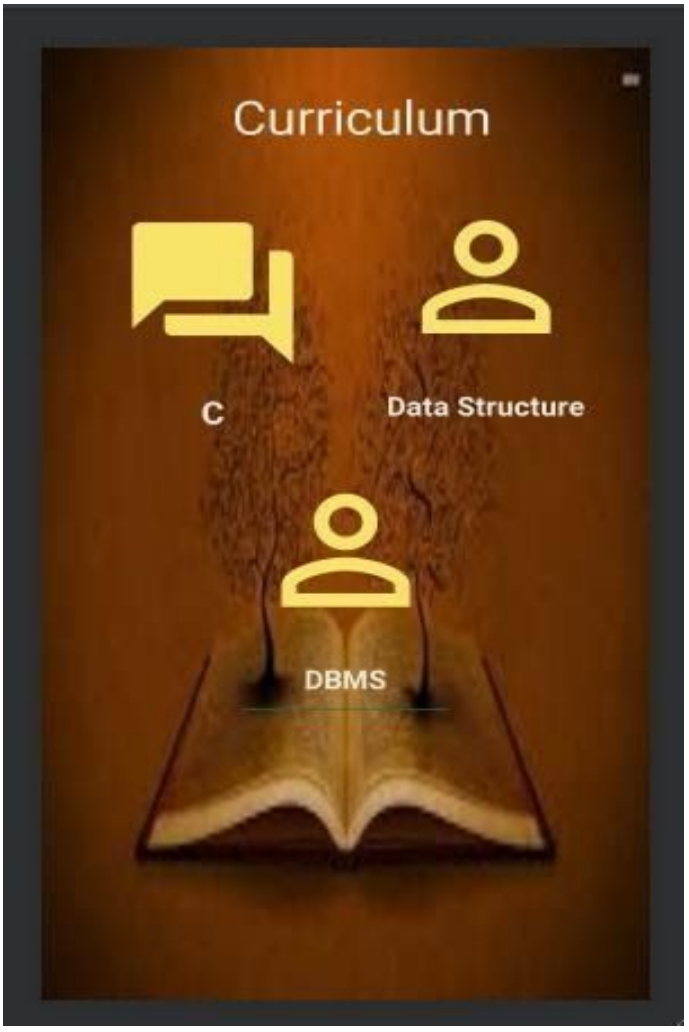




Select Course

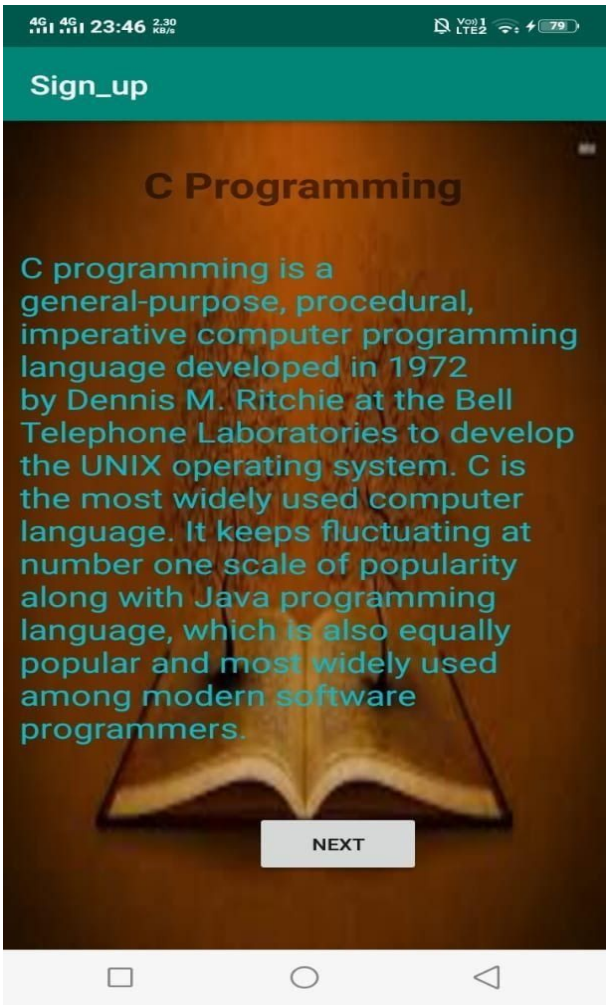


Select Year



Select Subject


What do you want?
Notes
Assessment
Quizzes


23:46
Sign_up
C Programming
C programming is a general-purpose, procedural, imperative computer programming language developed in 1972 by Dennis M. Ritchie at the Bell Telephone Laboratories to develop the UNIX operating system. C is the most widely used computer language. It keeps fluctuating at number one scale of popularity along with Java programming language, which is also equally popular and most widely used among modern software programmers.
NEXT

SOFTWARE TESTING

5.1 Testing

Software testing is the process of executing a program with intension of finding errors in the code. It is a process of evolution of system or its parts by manual or automatic means to verify that it is satisfying specified or requirements or not.

Generally, no system is perfect due to communication problems between user and developer, time constraints, or conceptual mistakes by developer.

To purpose of system testing is to check and find out these errors or faults as early as possible so losses due to it can be saved.

Testing is the fundamental process of software success.

Testing is not a distinct phase in system development life cycle but should be applicable throughout all phases i.e. design development and maintenance phase.

Testing is used to show incorrectness and considered to success when an error is detected.

5.2 Objectives of Software Testing

Software Quality Improvement: The computer and the software are mainly used for complex and critical applications and a bug or fault in software causes severe losses. So a great consideration is required for checking for quality of software.

Verification And Validation:

- Verification means to test that we are building the product in right way .i.e. are we using the correct procedure for the development of software so that it can meet the user requirements.
- Validation means to check whether we are building the right product or not.

Software Reliability Estimation: The objective is to discover the residual designing errors before delivery to the customer. The failure data during process are taken down in order to estimate the software reliability

5.3 Principles of Software Testing

All tests should be traceable to end user requirements.

Tests should be planned long before testing begins

Testing should begin on a small scale and progress towards testing in large

To be most effective testing should be conducted by an independent third party

The primary objective for test case design is to derive a set of tests that has the highest livelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used. They are

White box testing.

Black box testing.

5.3.1 White-box testing:

White box testing focus on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

5.3.2 Block-box testing:

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides through test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.

5.4 Testing fundamentals

Testing is a process of executing program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. If testing is conducted successfully it uncovers the errors in the software. Testing cannot show the absence of defects, it can only show that software defects present.

5.5 Testing Information flow:

Information flow for testing flows the pattern. Two class of input provided to test the process. The software configuration includes a software requirements specification, a design specification and source code.

Test configuration includes test plan and test cases and test tools. Tests are conducted and all there sults are evaluated.That is test results are compared with expected results. When erroneous data are uncovered, an error is implied and debugging commences.

CONCLUSION

This was the first considerably large and important project undertaken by us during our BCA course. It was an experience that changed the way we perceived project development. The coding could not be started before the whole system was completely finalized. Even then there were so many changes required and the coding needed to be changed. We attribute this to inadequate information gathering from the user. Though there were many meetings with the user and most of the requirements were gathered, a few misinterpretations of the requirements still crept in. It made us realize how important the systems analysis phase is. The project is a classic example, that learning of concepts needs to be supplemented with application of that knowledge.

On the whole it was a wonderful experience developing **PAATHSHALA** and we would have considered my education incomplete without undertaking such a project which allowed us to apply all that we have learnt and tried to develop a project that can be useful for students as well as teachers exchanging of content and other study material. It is developed using Android so that it can be accessed very easily and at any time. The application is developed with an aim of usability so that it is an easy to use system that requires the least amount of user input possible. For using this application general computer knowledge is enough. Users will be authenticated to ensure that no unauthorized users gain access to private information.

BIBLIOGRAPHY & REFERENCES

To develop this web application of PAATHSHALA, we used Android Studio for Front End and Google Firebase for Back End (Database). We take some knowledge towards automation system from some books that are given below:

The Joy of PHP Programming: A Beginner's Guide – by Alan Forbes

PHP & MySQL Novice to Ninja – by Kevin Yank

References:

[1]. www.w3schools.com/php.net
[2]. www.stackoverflow.com
[3]. www.php.net-tutorial.com
[4]. www.w3schools.com/css