STUDENT RESULT MANAGEMENT SYSTEM

A PROJECT REPORT
For
Mini Project (KCA353)
Session (2023-24)

Submitted by

KAJAL TRIVEDI
(2200290140078)

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of
Dr. Amit Kumar
Assistant Professor

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206

(MARCH 2024)

CERTIFICATE

Certified that **KAJAL TRIVED 2200290140078** has carried out the project work having "**Student Result Management System**" (**Mini Project-KCA353**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU**)** (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date: **KAJAL TRIVEDI** (**2200290140078**)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Dr. Amit Kumar   **Dr. Arun Tripathi**
Assistant Professor   **Head**
Department of Computer Applications   **Department of Computer Applications**
KIET Group of Institutions, Ghaziabad   **KIET Group of Institutions, Ghaziabad**

KAJAL TRIVEDI

ABSTRACT

Student result management system is a comprehensive project aimed at developing a robust and scalable platform, closely resembling the popular accountability service. This project, undertaken with the goal of providing users with a familiar and feature-rich experience, encompasses the entire software development lifecycle from conceptualization to deployment. The project incorporates key functionalities of Student result management system, including user registration, profile creation, student Add/Remove, Result Generate. Leveraging modern web technologies and frameworks, Key features of Student result management system include a user-friendly interface, a secure authentication system, efficient data storage utilizing a relational database.

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

Sr.no	Abbreviation	Definition
1	HTML	Hyper Text Mark-up Language
2	CSS	Cascading Style Sheets
3	PHP	Hypertext Pre-processor
4	DFD	Data Flow Diagram
5	XAMPP	Cross Platform Apache MySQL PHP Perl
6	MYSQL	My Structured Query Language

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The Student Result Management System, is a comprehensive endeavour to replicate the functionality and user experience of the renowned microblogging platform. Motivated by the desire to understand and implement key features of a modern Result Management System, this project encompasses a range of technical aspects spanning front-end development, back-end infrastructure, and real-time communication.

1.2 PROJECT DESCRIPTION

In today's fast-paced world, instant Result and Generating of ideas are paramount. Student Result Management System platforms have become a dominant force in this landscape, offering a familiar yet innovative experience for Developers.

1.2.1 PROJECT SCOPE:

This will encompass the core functionalities of Result Management System, including:

- User registration and profiles: Create accounts, edit profiles, and manage users.
- Interactions: add, remove, and update the records.

1.2.2 TARGET AUDIENCE:

This Platform targets individuals and communities seeking a dynamic platform for:

- Sharing quick updates.
- Connecting with the Collaborators.
- Engaging in quality based Record System.

1.3   HARDWARE/SOFTWARE

1.3.1   HARDWARE REQUIREMENTS

Hardware-   **:** Processor i3 or above

Clock speed-   **:** 3.0 GHz

RAM size-   **:** 4 GB or above

Hard Disk capacity   **:** 500 GB or above

1.3.2   SOFTWARE REQUIREMENTS

Operating System-   **:** Windows 10

Browser-   **:** Google chrome or any other

Application software-   **:** Visual Studio Code

Technology   **:** PHP

Server Required   **:** XAMPP

Language Required   **:** HTML, CSS, JAVASCRIPT

Documentation   **:** MS Word

CHAPTER 2

FEASIBILITY STUDY

A feasibility study analyses the viability of a project to determine whether the project or venture is likely to succeed. The study is also designed to identify potential issues and problems that could arise while pursuing the project. A feasibility study evaluates a project's or system's practicality. As part of a feasibility study, the objective and rational analysis of a potential business or venture is conducted to determine its strengths and weaknesses, potential opportunities and threats, resources required to carry out, and ultimate success prospects. Two criteria should be considered when judging feasibility: the required cost and expected value. A feasibility study is a comprehensive evaluation of a proposed project that evaluates all factors critical to its success in order to assess its likelihood of success. Business success can be defined primarily in terms of ROI, which is the amount of profits that will be generated by the project.

2.1 TECHNICAL FEASIBILITY

Technical feasibility study is concerned with specifying equipment and software that will successfully satisfy the user requirement; the technical needs of the system may vary considerably. The facility to produce outputs in a given time. A technical feasibility study reviews the technical resources available for your project. This study determines if you have the right equipment, enough equipment, and the right technical knowledge to complete your project objectives.

For example, if your project plan proposes creating 50,000 products per month, but you can only produce 30,000 products per month in your factories, this project isn't technically feasible. This assessment focuses on the technical resources available

To the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves the evaluation of the hardware,

Software, and other technical requirements of the proposed system. As an exaggerated example, an organization wouldn't want to try to put Star Trek's transporters in their building—currently, this project is not technically feasible.

2.2 BEHAVIOURAL FEASIBILITY

Behavioural feasibility is a critical aspect to consider when developing a Twitter clone project. This feasibility study assesses whether users and stakeholders are likely to accept and adapt to the new microblogging platform based on their behavioural patterns, preferences, and expectations. The literature on behavioural feasibility for social media platforms, including microblogging services, can provide valuable insights into user behaviour and acceptance. Behavioural feasibility for a Twitter clone project involves a comprehensive analysis of user behaviour, preferences, and cultural considerations. Drawing upon existing literature on user behaviour in social media, UX design, feature acceptance, community building, adoption patterns, and user feedback can provide a solid foundation for developing a microblogging platform that aligns with user expectations and encourages widespread adoption.

2.3 OPERATIONAL FEASIBILTY

Operation feasibility is used to check whether the project is operationally feasible or not. Our project is mainly different from the other system because of its web-support feature. So the measure for operational feasibility is something different from other system.

Generally the operational feasibility is related to organization aspects. The change determination is as such that early product were either a man or group of men or the jobs

Based manual but now a day with the advent of Internet technology. This assessment involves undertaking a study to analyse and determine whether—and how well—the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development. This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility—helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

CHAPTER 3

DESIGN

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement have been specified and analysed, system design is the first of the three technical activities - design, code and test that is required to build and verify software. The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design. System Design is the process of designing the architecture, components, and interfaces for a system so that it meets the end-user requirements.

System Design for tech interviews is something that can't be ignored! Almost every IT giant whether it be Facebook, Amazon, Google, Apple or any other ask various questions based on System Design concepts such as scalability, load-balancing, caching, etc. in the interview. This specifically designed System Design tutorial will help you to learn and master System Design concepts in the most efficient way from basics to advanced level.

3.1 USE CASE DIAGRAM

In the Unified Modelling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. A use case diagram doesn't go into a lot of detail—for example, don't expect it to model the order in which steps are performed. Instead, a proper use case diagram depicts a high-level overview of the relationship between use cases, actors, and systems. Experts recommend that use case diagrams be used to supplement a more descriptive textual use case.

UML is the modelling toolkit that you can use to build your diagrams. Use cases are represented with a labelled oval shape. Stick figures represent actors in the process, and the actor's participation in the system is modelled with a line between the actor and use case. To depict the system boundary, draw a box around the use case itself.

These diagrams are used at a very high level of design. This high level design is refined again and again to get a complete and practical picture of the system. A well-structured use case also describes the pre-condition, post condition, and exceptions. These extra elements are used to make test cases when performing the testing.

Although use case is not a good candidate for forward and reverse engineering, still they are used in a slightly different way to make forward and reverse engineering. The same is true for reverse engineering. Use case diagram is used differently to make it suitable for reverse engineering.
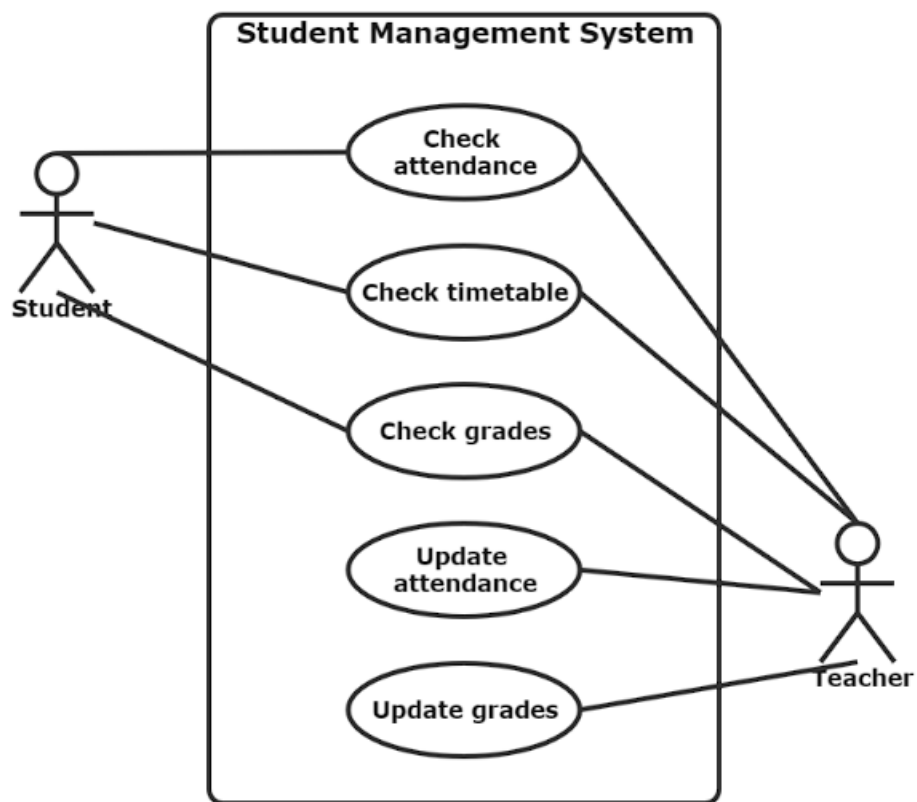
Use case diagrams can be used for –
• Requirement analysis and high level design.
• Model the context of a system.
• Reverse engineering.
• Forward engineering.

STUDENT MANAGEMENT SYSTEM
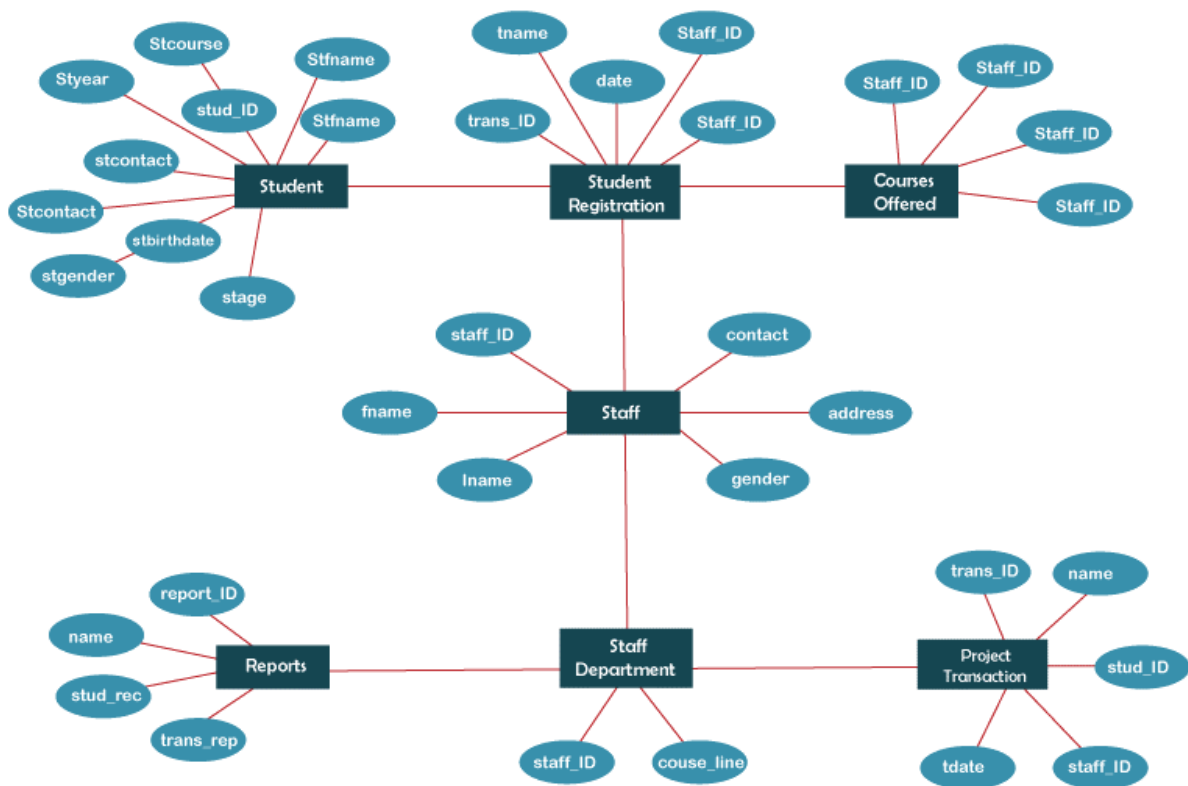
ER-Diagram (3.1)

Use Case Diagram (3.1)

3.1 DATABASE TABLES

3.1.1 ADMIN TABLE

FIELD	TYPE	CONSTRAINTS
username	Varchar2(255)	-
Password	Varchar2(255)	-

3.1.2 TABLE CLASSES

FIELD	TYPE	CONSTRAINTS
Id	Int	PRIMARY KEY
CLASSNAME	VARCHAR	NOT NULL
SECTION	CHAR	NOT NULL

3.1.3 TABLE RESULT

FIELD	TYPE	CONSTRAINTS
Id	NUMBR	PRIMARY KEY
STUDENT_ID	NUMBER	NOT NULL
CLASS_ID	NUMBER	NOT NULL
SUBJECT_ID	NUMBER	NOT NULL
MARKS	NUMBER	NOT NULL , DEFAULT

3.1.4 STUDENT TABLE

FIELD	TYPE	CONSTRAINTS
STUDENT_ID	Int(11)	NOT NULL
STUDENT_NAME	Varchar2(225)	DEFAULT NULL
EMAIL	VARCHAR2	NOT NULL
ROLL NUMBER	NUMBER	-
GENDER	CHAR	-
DOB	DATE	-

3.1.5 SUBJECT TABLE

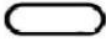| FIELD	TYPE	CONSTRAINTS
Id	Int(11)	NOT NULL
SUBJECT_NAME	Int(11)	NOT NULL
SUBJECT_CODE	Int(11)	NOT NULL

3.2 FLOWCHART DIAGRAM

Flowcharts are nothing but the graphical representation of the data or the algorithm for a better understanding of the code visually. It displays step-by-step solutions to a problem, algorithm, or process. It is a pictorial way of representing steps that are preferred by most beginner-level programmers to understand algorithms of computer science, thus it contributes to troubleshooting the issues in the algorithm. A flowchart is a picture of boxes that indicates the process flow in a sequential manner. Since a flowchart is a pictorial representation of a process or algorithm, it's easy to interpret and understand the process. To draw a flowchart, certain rules need to be followed which are followed by all professionals to draw a flowchart and is widely accepted all over the countries.

Process flowchart: This type of flowchart shows all the activities that are involved in making a product. It basically provides a pathway to analyse the product to be built. A process flowchart is most commonly used in process engineering to illustrate the relation between the major as well as minor components present in the product. It is used in business product modelling to help understand employees about the project requirements and gain some insight about the project.

Data flowchart: As the name suggests, the data flowchart is used to analyse the data, specifically it helps in analysing the structural details related to the project. Using this flowchart, one can easily understand the data inflow and outflow from the system. It is most commonly used to manage data or to analyse information to and fro from the system.

Business Process Modelling Diagram: Using this flowchart or diagram, one can analytically represent the business process and help simplify the concepts needed to understand business activities and the flow of information. This flowchart illustrates the business process and models graphically which paves a way for process improvement.

Symbol	Symbol Name	Description
→ ←	Flow Lines	Used to connect symbols
⬭	Terminal	Used to start, pause or halt in the program logic
▱	Input/output	Represents the information entering or leaving the system
▭	Processing	Represents arithmetic and logical instructions
◇	Decision	Represents a decision to be made

3.3 DATA FLOW DIAGRAM

DFD is the abbreviation for **Data Flow Diagram**. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. It is a graphical tool, useful for communicating with users, managers and other personnel. It is useful for analysing existing as well as proposed system.
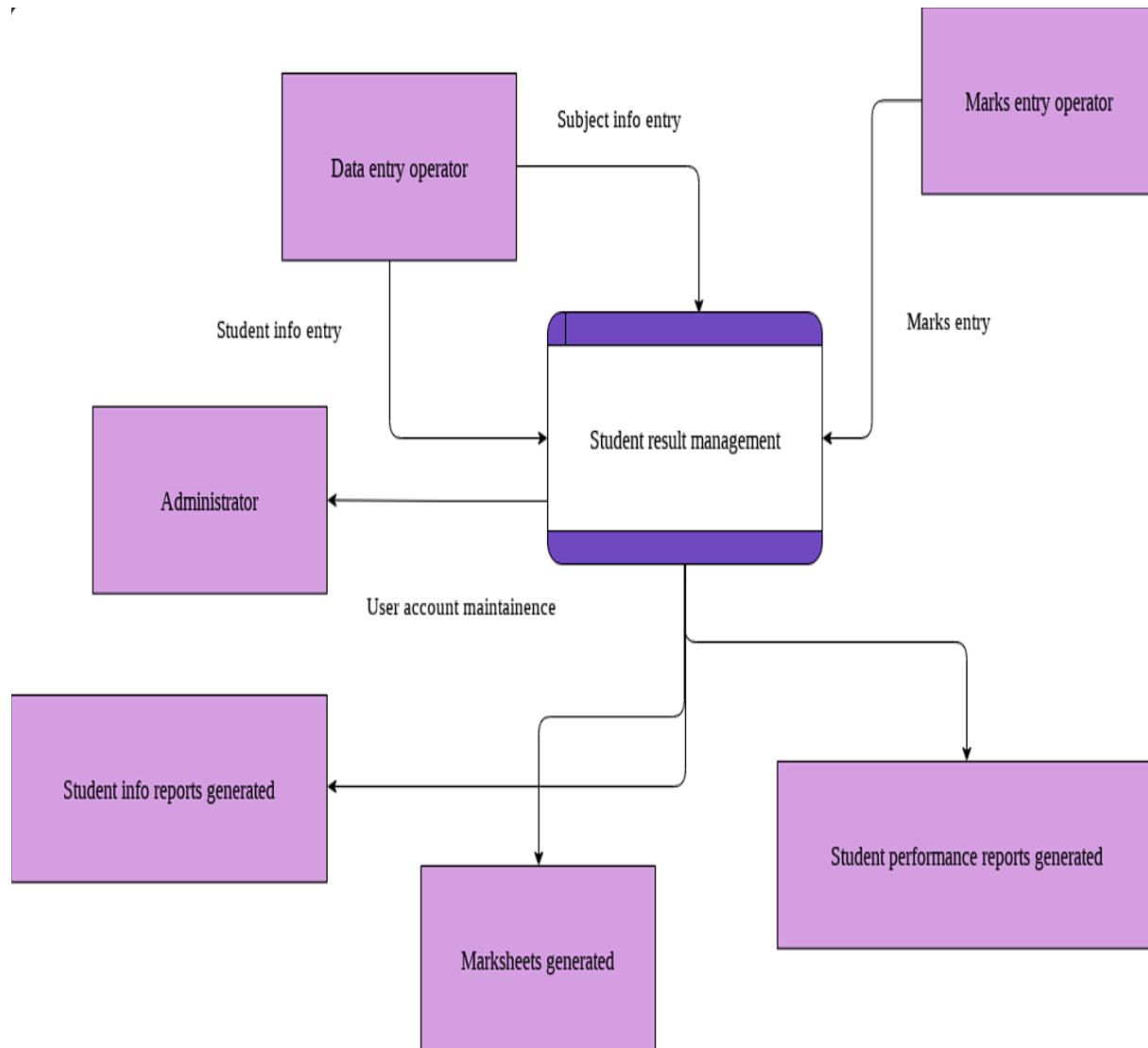
It provides an overview of
- What data is system processes.
- What transformation are performed.
- What data are stored?
- What results are produced etc.?

Data Flow Diagram can be represented in several ways. The DFD belongs to structured-analysis Modelling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes.

The Data Flow Diagram has 4 components:

- **Process** Input to output transformation in a system takes place because of process function. The symbols of a process are rectangular with rounded corners, oval, rectangle or a circle. The process is named a short sentence, in one word or a phrase to express its essence
- **Data Flow** Data flow describes the information transferring between different parts of the systems. The arrow symbol is the symbol of data flow. A relatable name should be given to the flow to determine the information which is being moved. Data flow also represents material along with information that is being moved. Material shifts are modelled in systems that are not merely informative. A given Flow should only transfer a single type of information. The direction of flow is represented by the arrow which can also be bi-directional.
- **Warehouse** The data is stored in the warehouse for later use. Two horizontal lines represent the symbol of the store. The warehouse is simply not restricted to being a data file rather it can be anything like a folder with documents, an optical disc, a filing cabinet. The data warehouse can be viewed independent of its implementation. When the data flow from the warehouse it is considered as data reading and when data flows to the warehouse it is called data entry or data updating.
- **Terminator** The Terminator is an external entity that stands outside of the system and communicates with the system. It can be, for example, organizations like banks, groups of people like customers or different departments of the same organization, which is not a part of the model system and is an external entity. Modelled systems also communicate with terminator.

Data entry operator

Subject info entry

Marks entry operator

Student info entry

Student result management

Marks entry

Administrator

User account maintainence

Student info reports generated

Marksheets generated

Student performance reports generated

DFD 1(3.6)

CHAPTER 4

PROJECT SCREENSHOTS

4.1 Login Page

A login form utilizes the credentials of a user, in order to authenticate their access. It generally consists of the typical username or email and password. But more fields can be added to improve the site's security
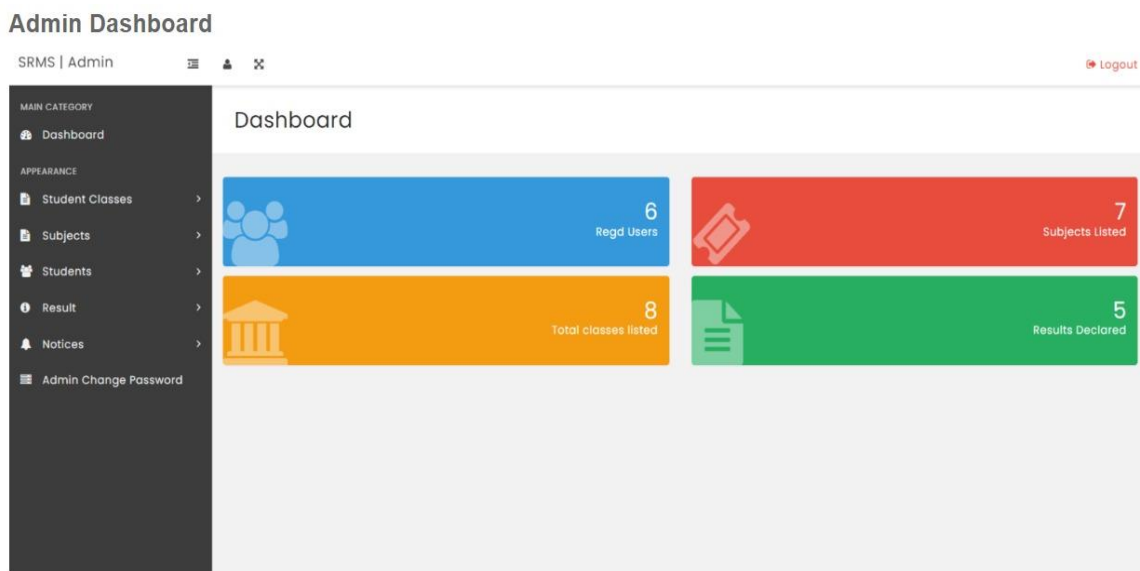
4.2 Result Page

This module seamlessly takes care of Result, discharge and the transfer processes of Student Result. It enables the search on availability and manages the allocation of Result.

Result Record Management System

Student Name : kajal trivedi

Student Roll Id : 1116

Student Class: Fourth(C)

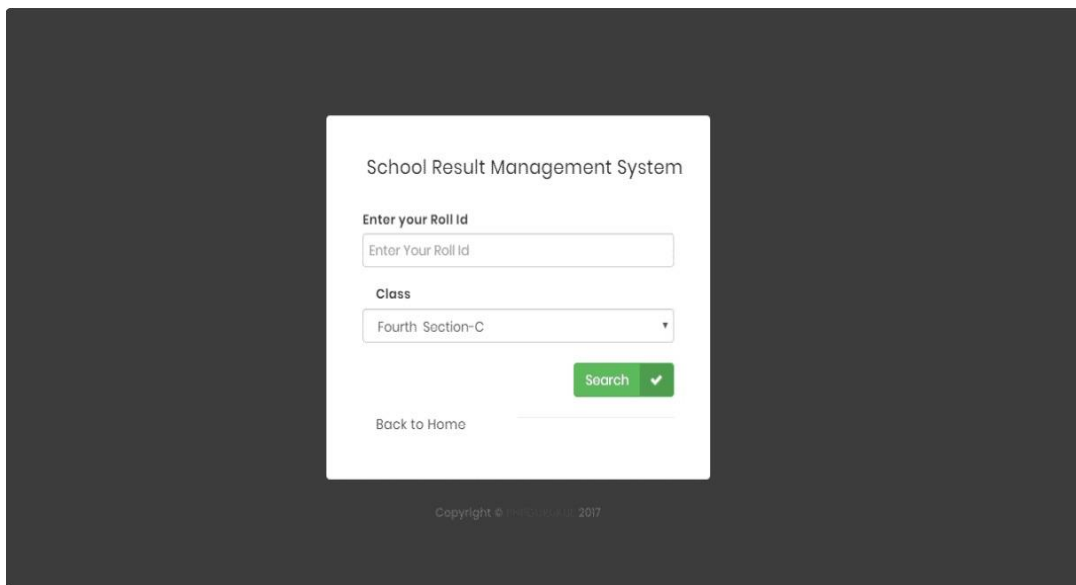| #	Subject	Marks
1	English	67
2	Maths	65
3	Music	50
4	Science	65
Total Marks		247 out of 400
Percntage		61.75 %

Back to Home

4.3 Dashboard

This module seamlessly takes care of Result Managing, Subject Managing, Student Record Managing processes of Student Result. It enables the search on availability and manages the allocation of Result.

4.4 Subject Page

This module seamlessly takes care of Result Managing, Subject Managing, Student Record Managing processes of Student Result. It enables the search on availability and manages the allocation of Result.

CHAPTER 5

TESTING

Testing is a process of executing a program with the intent of finding bugs that makes the application fail to meet the expected behaviour. System Analysis and Design process including Requirement Analysis, Business Solution Options, Feasibility Study, Architectural Design was discussed in previous chapter. Generally Software bugs will almost always exist in any software module. But it is not because of the carelessness or irresponsibility of programmer but because of the complexity. Humans have only limited ability to manage complexity. This chapter discusses about the testing of the solution and implementation methodologies. Regardless of the development methodology, the ultimate goal of testing is to make sure that what is created does what it is supposed to do. Testing plays a critical role for assuring quality and reliability of the software. I have included testing as a part of development process. The test cases should be designed with maximum possibilities of finding the errors or bugs. Software Testing is the process of executing a program or system with the intent of finding errors. The scope of software testing often includes examination of code as well as execution of that code in various environments and conditions. Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements. Testing stage of

The project can be explained as below and system was tested for all these stages. Various level of testing are as follows

5.1 TESTING LEVELS

Unit testing: Unit testing tests the functionality of individual units of source code. It is the smallest component of a testable software that works in isolation with other parts of the code. I have done unit testing for various individual components of the source code to uncover errors within the boundary of the application.

Integration testing: Integration testing focuses on the design and construction of the software. Here the individual components that are tested using unit tests are combined and tested as a group. Its primary purpose is to expose the defects associated with the interfacing of modules. It checks if the modules perform the desired functionality when integrated together

System testing: System testing is performed on a completely integrated system to see if it meets the requirements, System Testing is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system with the corresponding requirements. In system testing, integration testing passed components are taken as input.

Regression testing: Regression testing aims at verifying the functionality of the software that is previously tested and to which changes are made. It is to ensure the old software still works with new changes.

Acceptance testing: Acceptance testing is conducted to verify if the system compliance the business requirements. Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

Adhering to the levels of testing, Unit testing is performed on individual components of the system ensuring the expected behaviour. Later, I have integrated various components together and performed Integration testing. Once the integration testing is done, I have

Performed System 30 testing and ensured the application works as per the requirements. Finally, acceptance testing is performed to check if the client accepts the system

• All its module are working properly and the project is best viewed both in Computer.
• The main requirement for this project that you should have good internet connection.

Performance Testing: Performance testing is performed to determine how well the system can perform in terms of responsiveness under all kinds of load. The web application is tested to see if it can sustain huge amount of requests providing higher throughput under different loads. I have simulated multiple hits on various pages of the application to evaluate the overall performance.

5.2 TEST CASES

5.2.1 FOR USER SIGNUP

FIELD	DETAILS
Email	Example123@gmail.com
Username	Test
Password	******

Record entered successfully

5.2.2 LOGIN CASE

FIELD	DETAILS
Email	Example123@gmail.com
Password	******

Logged in successfully

CHAPTER 6

FUTURE SCOPE AND CONCLUSION

Future Scope:

The future of the Twitter clone holds immense potential for growth and innovation. Here are some key areas for future development:

1. Enhanced User Experience:
 - Implementing advanced algorithms for personalized content delivery based on user preferences   and behaviour.
 - Integration of augmented reality (AR) and virtual reality (VR) elements to create a more immersive user experience.

2. Advanced Content Moderation:
  - Leveraging machine learning and artificial intelligence for more effective and nuanced content moderation, reducing the prevalence of harmful or inappropriate content.
  - Incorporating blockchain technology to ensure transparency and accountability in content moderation decisions.

3. Monetization Strategies:
  - Introduction of innovative monetization features, such as exclusive content subscriptions, premium user accounts, and a decentralized tipping system.
  - Exploring partnerships with businesses for targeted advertising and sponsored content.

4. Global Expansion:
  - Focusing on expanding user bases in untapped markets and regions to create a truly global social media platform.
  - Localization efforts to cater to diverse linguistic and cultural preferences.

5. Interoperability and Integration:
  - Facilitating seamless integration with other popular social media platforms and communication tools.
  - Exploring interoperability with emerging technologies, such as the integration of decentralized identity systems.

6. Accessibility and Inclusivity:
  - Continuous improvements to make the platform more accessible for users with disabilities.
  - Implementing features that promote inclusivity and diversity, fostering a safe and welcoming online environment.

7. Environmental Sustainability:
  - Adopting eco-friendly practices and technologies to reduce the platform's carbon footprint.
  - Encouraging and supporting environmental initiatives within the user community.

8. Open Source Collaboration:
  - Opening up certain aspects of the platform for community-driven development through open source initiatives.
  - Collaborating with developers to enhance features, security, and overall platform performance.

Conclusion:

In conclusion, the Twitter clone stands at the forefront of social media innovation, providing users with a platform for real-time communication and information sharing. As we look to the future, the emphasis will be on continuous improvement, technological advancements, and user-centric innovations. By addressing issues related to content moderation, embracing new monetization strategies, expanding globally, and staying at the forefront of technological trends, the Twitter clone is poised to remain a dynamic and influential player in the social media landscape. The commitment to user experience, accessibility, and environmental sustainability ensures that the platform evolves in a responsible and inclusive manner, fostering a global community that thrives on meaningful connections and shared experiences.

BIBLIOGRAPHY

[1] Introduction To PHP Retrieved on 10/10/2023 from
https://www.tutorialspoint.com/php/index.html

[2] https://www.udemy.com/course/create-a-twitter-clone-with-php-mysql-and-js-from-scratch/

[3] https://www.geeksforgeeks.org/php-tutorial/

[4] https://www.javatpoint.com/php-tutorial

[5] https://www.javatpoint.com/javascript-tutorial