RESULT ANALYSIS SYSTEM

A project report submitted in partial fulfillment of the requirements for the award of the degree of

Integrated M.Sc. in Computer Science
with Specialization in
Artificial Intelligence and Machine Learning

Submitted by

Aiswarya V
Anjana A

Nehru Arts and Science College Kanhangad
Padnekkad P.O., Kasaragod Dt., Kerala - 67314

Affiliated to

Kannur University
Kannur

July 2023

CERTIFICATE

This is to certify that report entitled "**RESULT ANALYSIS SYSTEM**" is a bonafide report of the mini project (6B25ICSC - Project) presented during VIth semester by **Aiswarya V and Anjana A**, in partial fulfillment of the requirements for the award of the degree of Integrated M.Sc. in Computer Science with Specialization in Artificial Intelligence and Machine Learning.

Staff in Charge   Project Guide   Head of the Department

ABSTRACT

The Result Analysis System offers a streamlined and automated approach to address the essential task of monitoring and evaluating the academic performance of students within academic institutions. Manual methods for this purpose are not only time-consuming but also prone to human errors. This system provides an innovative solution that brings numerous advantages.

Enhanced Accuracy and Efficiency: By automating the result analysis process, this system significantly reduces the margin for errors that can occur during manual assessments. It ensures that academic evaluations are consistently accurate and efficient.

Improved Teacher-Student Communication: The system promotes better communication between teachers and students. This transparent communication fosters a constructive learning environment.

Empowering Educators: Teachers benefit from this system as well. They can conveniently upload university exam results, and the system promptly generates comprehensive reports. These reports allow teachers to evaluate their own teaching methods and better understand student outcomes. Armed with this data, educators can make informed decisions to refine their teaching strategies and enhance the learning experience.

Enhancing Academic Excellence: The Result Analysis System represents a significant stride towards enhancing academic excellence within educational institutions. By seamlessly automating the evaluation and monitoring of students' academic performance, it frees up valuable time and resources that can be redirected towards refining educational content and student support mechanisms. This system not only enhances the efficiency of administrative processes but also empowers educators to focus on optimizing teaching methods and providing students with the support they need to excel.

Streamlined Analysis through Simple Result Upload: With just a few simple steps for result upload, our Result Analysis System unlocks a world of analytical possibilities. Whether it's identifying top-performing students each semester, conducting detailed single-subject analyses, evaluating individual student progress, or examining results on a semester-by-semester basis, our system streamlines the process. Teachers and administrators can effortlessly upload results, and the system, in turn, generates a wealth of valuable insights. Furthermore, with an advanced search feature, users can easily access and retrieve various types of results, making the entire process user-friendly and efficient.

ACKNOWLEDGMENT

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Academic institutions face the critical task of assessing and monitoring their students academic progress, a process that, when performed manually, can be both time-consuming and prone to errors. To address these challenges, the Result Analysis System offers a centralized and automated solution.

In our system, when the University publishes results, it empowers departmental teachers to effortlessly upload students' corresponding results. Subsequently, the system generates a multitude of insightful analyses. These tasks involve identifying top-performing students each semester, conducting in-depth single-subject analyses, evaluating individual student performance, and examining results on a semester-by-semester basis.

Our Result Analysis System empowers educators and administrators by providing multifaceted capabilities, including student result analysis, semester-wise performance evaluation, departmental excellence assessment to identify the top students within a department, as well as single-subject analysis for an in-depth view of student performance in specific courses. Moreover, our advanced search functionality allows users to seamlessly retrieve various types of results, offering enhanced accessibility and data retrieval tailored to their specific requirements. These features collectively form a robust and versatile toolset, enabling educational institutions to make data-driven decisions, promote student success, and streamline their assessment and monitoring processes.

Any teacher or department within our college can access and benefit from these analyses. Additionally, our system facilitates access to the academic history of previous students who have studied at our institution, allowing retrieval of their results from any semester.

1.1.1 Modules

There are 3 modules in our system.

- Administrator

- Teacher

- Principal

1.1.2 Use Case Diagrams

Administrators are responsible for adding teachers, setting their passwords, and adding new courses(Figure1.1). Currently, the addition of students is not within the system, as it is handled by the SMS (Student Management System) of your college. Teachers are responsible for uploading the results of students, and both teachers and the principal can view different result analysis.(Figure1.2)
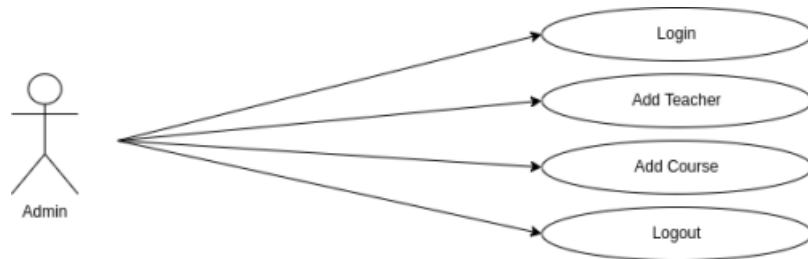
Figure 1.1: Use case diagram of admin
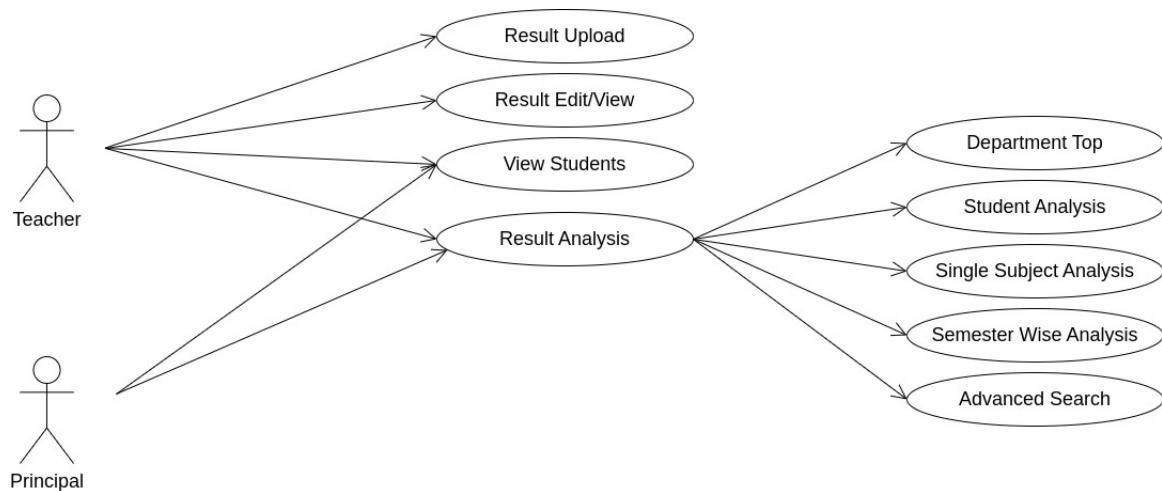
Figure 1.2: Use case diagram of teacher and principal

1.2 Feasibility Study

Feasibility study lets the developer for see the future of the project and the usefulness. Feasibility study is a test of system proposed regarding its workability, impact on the organization, ability to meet the needs and effective use of resources.

1.2.1 Technical Feasibility

The technical feasibility of Result Analysis System involves evaluating the system's hard- ware and software requirements.

Hardware Requirements

- Minimum RAM: 2GB

- Processor: Pentium or equivalent with similar performance

Software Requirements

- Operating System: Result Analysis System is designed to be platform-independent, compatible with various operating systems such as Windows, macOS, and Linux.

- Web Browser: The application runs on modern web browsers like Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.

- Programming Languages and Technologies:

  - Frontend: Specify the frontend programming languages and technologies:
    * HTML: For structuring web pages.
    * CSS: For styling and layout.
    * JavaScript: For interactive features and user interface enhancements.
  - Backend: Specify the backend programming languages and technologies:
    * PHP: As the primary backend scripting language.
    * MySQL: As the relational database management system.

This system is a platform-independent, use MySQL for database, PHP as the server-side scripting language and html as user interface. Thus Result Analysis System is technically feasible.

1.2.2 Economical Feasibility

The economic feasibility of the Result Analysis System is assessed by considering the financial aspects of the project. Our system proves to be a financially efficient choice for educational institutions and stakeholders, primarily due to the following factors:

- Cost-Effective Solution: The Result Analysis System is designed to provide a cost-effective solution. It eliminates the need for expensive, proprietary software licenses, as it is built using open-source technologies. Educational institutions can implement and utilize the system without incurring licensing costs, promoting financial sustainability.

- Scalability: Our system is engineered to scale seamlessly with the institution's needs. Whether there is an increase in the number of students or data volume, the system can adapt efficiently without demanding significant additional investments in infrastructure or operations.

In conclusion, the Result Analysis System offers economic feasibility by optimizing resources, reducing costs, and promoting efficient result analysis within educational institutions. It stands as a financially prudent choice for institutions seeking to enhance their academic performance evaluation processes while maintaining fiscal responsibility.

1.2.3 Behavioural Feasibility

The Result Analysis System demonstrates strong behavioral feasibility, as it encompasses various aspects that enhance its ease of use and efficiency in operation. Here are the behavioral feasibility considerations for the system:

- User-Friendly Interface: The system is designed with a user-friendly interface that ensures ease of use for teachers.

- Reduced Human Effort: The new system significantly reduces the manual effort required for result analysis and reporting. It automates the process, minimizing the need for extensive human intervention.

- Historical Student Data Access: One of the unique features of the system is its ability to access previous student details. This capability allows educators and administrators to retrieve and analyze the performance history of students who have graduated or completed previous academic terms, providing valuable insights for continuous improvement.

- Versatile Analysis: The system offers the flexibility to perform various types of result analysis. Whether it's departmental analysis, student-level analysis, semester analysis, or single-subject analysis, the system provides the tools and data needed for comprehensive assessment.

In summary, the Result Analysis System's user-friendly features, reduced manual effort, accessibility, and versatile analysis options make it a valuable tool for improving academic performance evaluation in educational institutions.

Chapter 2

Related Works

2.1 System Analysis

System analysis is a detailed study of the various operations performed by a system and their relationship within and outside of the system. One aspect of analysis is defining the boundaries of the system and determining whether or not a candidate system should consider other related systems. During analysis, data are collected on the valuable file, decision points,and transactions handled by the present system. Once the analysis is completed the system analyst has a firm understanding of what is to be done. System analysis and study has mainly two parts. They are Existing System and Proposed System.

2.1.1 Existing System

The current system operates in a manual mode, demanding considerable effort and consuming a substantial amount of time. In this existing setup, when results are published, teachers are required to manually transcribe or document the results of each student. When it comes to conducting analyses, these tasks are also carried out manually, adding to the overall time and effort required for academic assessments.

Disadvantages

- Time consuming

- High chance of error

- High manual effort

- Inadequate analysis

- Lack of different analyses

- Missing pre-student results

- Documentation is a challenging

2.1.2 Proposed System

The proposed system represents a significant departure from the current manual mode of operation. It introduces a streamlined and efficient approach to result analysis and academic assessments, leveraging modern technology to simplify processes.

In this proposed system, the need for manual transcription or documentation upon the publication of results is entirely eliminated. Teachers can effortlessly upload student marks and data directly into the system, greatly reducing the burden of manual data entry.

One of the primary advantages of the proposed system lies in its robust analytical capabilities. It offers a wide range of analytical tools and features that empower educators to conduct in-depth assessments with ease. These analyses encompass various aspects, including department-wise performance, student-specific insights, semester-wise trends, single-subject analyses, and advanced search functionality. This comprehensive suite of analytical components allows educators to gain deeper insights into student performance and academic trends.

Advantages

- Time-efficient

- Reduced chance of error

- Minimized manual effort

- Enhanced analysis capabilities

- Diverse analysis options

- Comprehensive student performance history

- Streamlined documentation

- Printing option for convenient physical copies

2.2 Requirement Elicitation and Analysis

The analysis of requirements for the Result Analysis System commenced with a comprehensive gathering of information from various stakeholders, including students and Teachers. This initial phase aimed to collect all relevant data that would serve as the foundation for defining the system's requirements. The collected information underwent a thorough analysis to establish a precise and unambiguous understanding of the project's objectives, while also addressing any ambiguities or inconsistencies in the initial problem statement.

2.2.1 Requirement Gathering

For the requirement gathering process, a combination of methods was employed, including personal interviews and group discussions. These methods allowed for a

holistic approach to gathering insights and needs for the project.
The following techniques were utilized:

- Personal Interviews : Interviews were conducted with key stakeholders to identify the activities involved in the result analysis process, comprehend the limitations of the existing system, and discern the types of reports required.

- Checklists : Checklists were employed to systematically identify the areas within the result analysis system that required thorough analysis. This approach helped prioritize aspects of the system based on their significance.

- Record Review : The examination of existing documentation played a pivotal role in understanding the standard operating procedures, forms, documents, and various reports utilized by the current system.

Chapter 3

Methodology

3.1 System Development

The development of the Result Analysis System was executed using PHP, a server-side scripting language known for its versatility and web development capabilities. The user interface was crafted using HTML, CSS, and JavaScript to create an interactive and user-friendly front end. PHP provided the foundation for the back-end logic. The development environment was established using tools like Visual Studio Code.

3.2 Database Design and Data Integration

In our project, we recognized the importance of leveraging existing student data, which was already stored in the database of the college's Student Management System. This integration allowed us to build upon the foundation of pre-existing student details, eliminating the need to re-enter redundant data.To seamlessly integrate the Result Analysis System with the Student Management System's database, we extended the database schema to accommodate additional information related to courses and teachers.Admin of our project is responsible for entering course data including course name,course code,credit etc and teacher data include teacher name ,department into the database. By integrating with the existing Student Management System's database, we were able to build a seamless and comprehensive Result Analysis System while minimizing data duplication and maintaining data consistency across the college's information systems. This approach not only saved time and effort but also ensured that student, course, and teacher data remained accurate and synchronized.

We introduced an additional database table specifically designed to capture and manage mark data.It includes fields such course_id and stud_id, ce, ese marks. By creating this separate table, we ensured that teacher-entered marks were appropriately organized, securely stored, and easily accessible for analysis and reporting.

3.3 User Feedback

User feedback was an integral part of our methodology. We actively sought input from users, including teachers and administrators, at various stages of development.

Feedback sessions were conducted to gather insights into system usability and functionality. This feedback guided the refinement of system features and the user interface.

3.4   Framework and Tool Selection

In the process of developing a web app, we rely on a range of essential tools. Such as:

• **Visual Studio Code** : It is an advanced code editor that offers features like debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and integrated Git support. .

• **phpMyAdmin** : phpMyAdmin is a web-based graphical interface used for managing MySQL. It simplifies database administration by providing a user-friendly interface accessible through web browsers. With phpMyAdmin, users can create, modify, and delete databases and tables, execute SQL queries, import and export data, and manage user privileges.

Chapter 4

Design and Implementation

4.1 Architecture

4.1.1 Data Flow Diagram

A data-flow diagram is a way of representing a flow of data through a process or a system.

- Level 0 - DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled (Figure 4.1)

- Level 1 - Flow of data in admin module is shown in Figure 4.2.

- Level 2 - Flow of data in teacher module is shown in Figure 4.3.

- Level 2 of process 4 - Flow of data in result analysis is shown in Figure 4.4.

- Level 3 - Flow of data in principal module is shown in Figure 4.5.
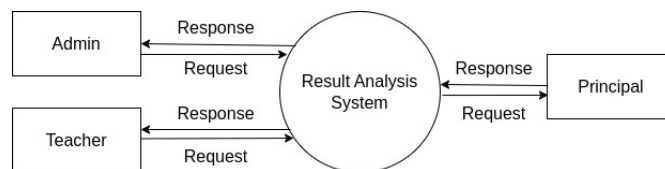
Figure 4.1: Context Level DFD
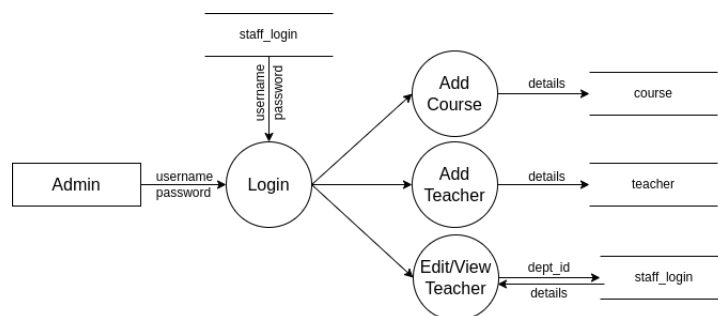
Figure 4.2: Level 1: DFD
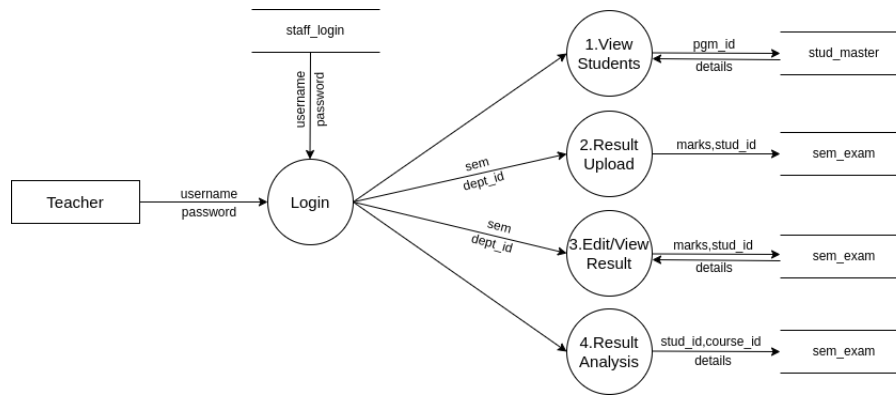
Figure 4.3: Level 2: DFD
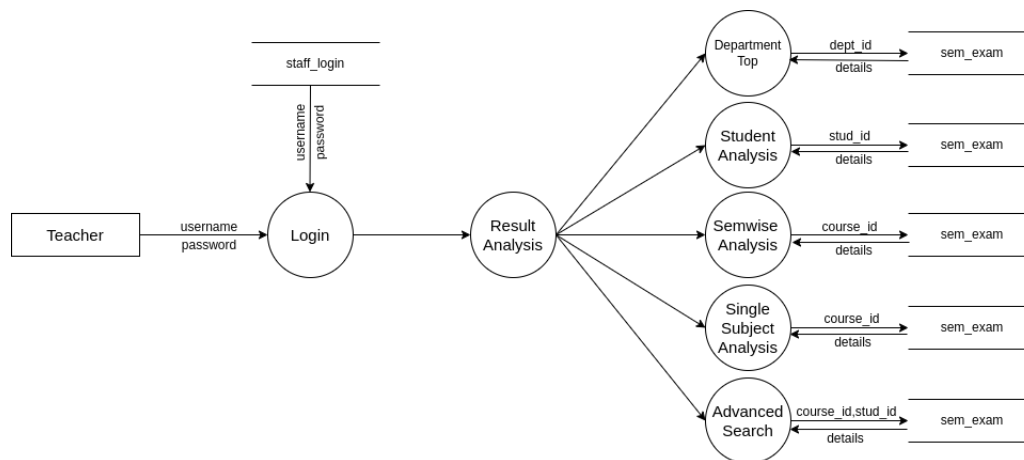
Figure 4.4: Level 2 of process 5: DFD
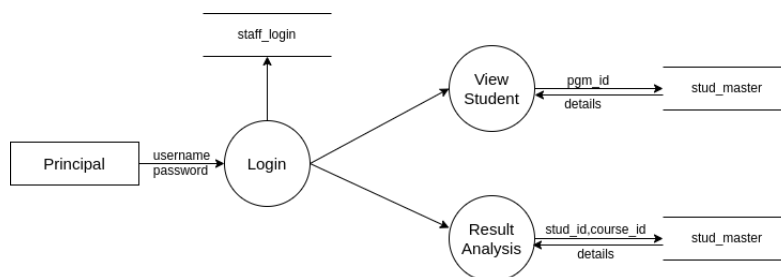
Figure 4.5: Level 3: DFD

4.1.2 Database Design

- Table Name : stud_master (Table 4.1)

  Description : Details of students

Column name	Data Type	Description
stud_id	int	id of student
roll_no	int	roll no of student
uty_reg_no	int	register number of student
name	varchar(50)	name of student
year_of_admin	int	year of admission
sex	char(1)	sex
pgm_id	int	programme
language_id	int	second language
quota	varchar(10)	quota of student

Table 4.1: Student Details

- Table Name : course (Table 4.2)

  Description : Details of papers

Column name	Data Type	Description
course_id	int	id of course
course_title	varchar(100)	course name
course_code	varchar(12)	course code
lab_theory	char(1)	lab/theory
course_type_id	int	course type
dept_id	int	department
semester	int	semester
syllabus_intro_year	int	year of syllabus
credits	int	credit
total_internal	int	total internal
total_external	int	total external

Table 4.2: Course Details

- Table Name : sem_exam (Table 4.3)

  Description : Result of students

Column name	Data Type	Description
course_id	int	course id
stud_id	int	student id
ce	int	internal mark obtain
ese	int	external mark obtain

Table 4.3: Result of Students

4.1.3 ER Diagram

An entity-relationship diagram is a data modeling technique that creates a graphical representation of the entities, and the relationships between entities, within an information system. Figure 4.6 represents the ER diagram of the result analysis system.
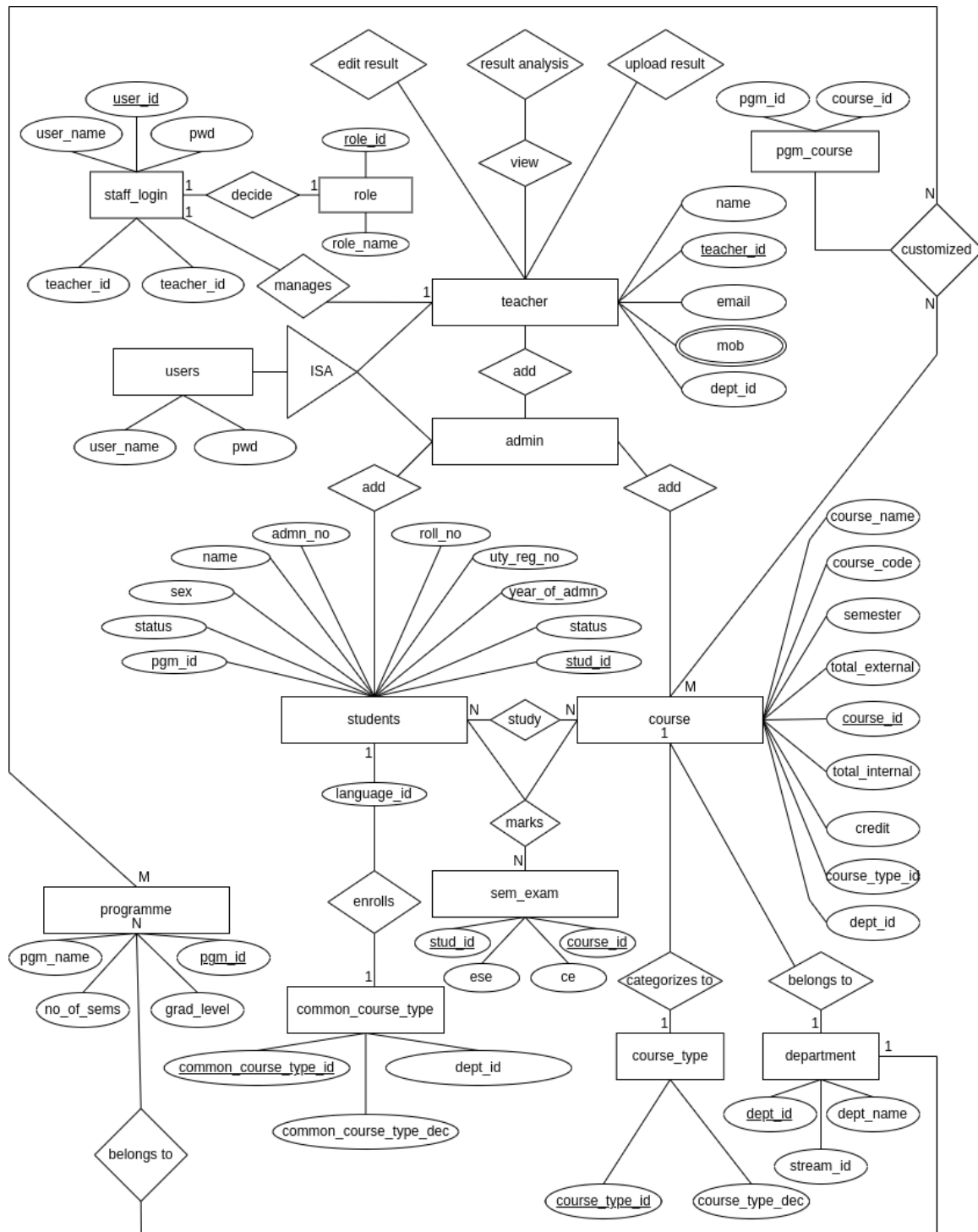
Figure 4.6: ER daigram

4.2 Implementation

We gathered information for our project through personal interviews with teachers, checklists, previous analyses, and record reviews. During our research, we identified that manual analysis processes were time-consuming and prone to errors. As a result, we concluded that implementing our system within the college context would be a sensible solution. The proposed system has been developed using PHP for the front-end, with HTML, CSS, and JavaScript utilized for design and user interface components. MySQL is employed as the back-end database. We provided sample data for testing our code and identified an issue where marks can be uploaded exceeding the maximum allowed mark. This testing process has been crucial in validating our project and finding a solution to the problem. Testing is indeed a vital step in the development process to ensure the correctness and reliability of our system.

In figure 4.4, we can observe the comprehensive depiction of our project's growth and timeline chart, providing a visual representation of its evolution over time.

	Mar	Jun	Jul	Aug	Sep
Planning					
Design					
Coding					
Testing					

Table 4.4: TimeLine Chart

Chapter 5

Testing

Testing is a critical phase to ensure that the system functions correctly and meets its intended objectives.

5.1 Testing Strategy

5.1.1 System Testing

System testing evaluates the entire system as a whole, including end-to-end functionality. It aims to assess the system's overall functionality, performance, security, and reliability in a real-world or simulated environment.
This testing includes Data insert and validation test and User interface test.

5.1.2 Unit Testing

Unit testing involves testing individual components and functions of the application to ensure they perform as expected.
This stage includes checking the accuracy of mark uploads and data processing and Login and Authentication test.

5.1.3 Integration Testing

Integration testing assesses the interactions between different modules and components of the system. We validate that data flows seamlessly between the teacher's mark upload interface and the analytical components of the web application.
This includes verifying that data is correctly captured, processed, and delivered to the appropriate modules for analysis.Test how the system responds when there are data format errors, missing information, or other issues during data upload. The system should provide meaningful error messages and gracefully handle these situations.
Integration testing should include checks for data integrity. Verify that data is not corrupted during transmission between the interface and the analytical components.

5.2 Test Cases

In this section, we outline the test cases designed to evaluate the functionality of our Result Analysis System. Each test case focuses on specific aspects of the system, ensuring comprehensive coverage.

5.2.1 Mark Upload Testing

Test Case : Uploading Marks for Sample Students
Objective: To verify the accuracy and integrity of mark uploads.
Steps: Teachers upload marks for a sample set of students.
Expected Result: Marks are successfully uploaded, and the data is correctly processed.

Sl.no	Programmes	CE	ESE
1	Communicative English	10	36
2	Readings on Kerala	11	33
3	Naya Sahithya	10	41

Figure 5.1: Mark exceeds the maximum allowed limit

5.2.2 Department Top Analysis Testing

Test Case : Department Top Analysis
Objective: To confirm that the system accurately identifies top-performing students within a department.
Steps: Analyze results to find the top students in a selected department.
Expected Result: The system correctly identifies the top students based on their marks.

5.2.3 Student wise Analysis Testing

Test Case : Student-Specific Analysis
Objective: To validate the system's ability to provide individual student mark status analysis.
Steps: Access the individual student's profile for the chosen semester.
Expected Result: Accurate display of the student's analysis for the specified semester.

5.2.4 Semester-wise Analysis Testing

Test Case : Semester-wise Analysis
Objective: To ensure that the system can perform semester-wise analysis.

Steps: Select a specific semester and examine the performance data.
Expected Result: Accurate presentation of performance data for the chosen semester.

5.2.5   Single Subject Analysis Testing

Test Case : Single-Subject Analysis
Objective: To verify that the system provides in-depth analysis of a single subject's performance.
Steps: Select a subject and analyze the results.
Expected Result: Accurate presentation of performance insights for the chosen subject.

5.2.6   Advanced Search Testing

Test Case : Advanced Search Functionality
Objective: To confirm the effectiveness of the advanced search feature.
Steps: Perform advanced searches using various criteria.
Expected Result: Return of relevant and accurate search results.

In the initial stage of the uploading process, the system inadvertently permitted the uploading of marks that exceeded the maximum allowable mark for a particular paper. This oversight had the potential to significantly distort the subsequent analysis. Recognizing this issue, we promptly addressed it by implementing a code modification. This modification now prevents the addition of marks that exceed the established maximum mark, ensuring the integrity and accuracy of the analysis process.That is when a user attempts to upload a mark above the maximum allowed mark, the system immediately provides a visual indication by highlighting the input box with a distinct red border (Figure5.1). This ensures that the marks stay within the specified limits and maintains data accuracy.

Chapter 6

Results and Discussion

The Result Analysis System is a comprehensive software solution designed to streamline and enhance the academic performance evaluation process within educational institutions. The automation of result analysis has not only reduced manual effort but also improved the quality of evaluations. With consistent and reliable results, educators can make data-informed decisions to enhance the learning experience. The system's ability to conduct various types of analyses, such as departmental top analysis, semester wise analysis etc. and historical data retrieval, offers a deeper understanding of academic performance trends. This information is invaluable for academic planning and decision-making.

6.1 Results

6.1.1 Uploading result of students

We've effectively incorporated a robust system for uploading student results corresponding to their academic papers in each semester, a visual representation of which can be found in Figure 6.1. This system boasts several key features:

1. Maximum Mark Validation: If a teacher inadvertently enters marks exceeding the predetermined maximum score for an assessment, a prominent indicator will appear around the input box, clearly signaling the error. In this situation, the system will intelligently prevent submission until the erroneous marks are rectified.

2. Submission Success Page: Upon successfully uploading and submitting the marks, the system will gracefully transition to a dedicated success message page. This page provides immediate confirmation that the data has been securely saved. Furthermore, it furnishes a 'Next' button, strategically positioned for seamless navigation to the next student, streamlining the grading process with efficiency.

3. Duplicate Entry Alert: To prevent redundancy and errors, when a teacher inadvertently attempts to upload marks for a student for the second time, a discerning notification will promptly appear. This alert serves as a safeguard, informing the teacher that marks for the specific student have already been successfully uploaded. This redundancy check ensures data integrity and minimizes potential inaccuracies in the grading records.

Sl.no	Programmes	CE	ESE
1	Computer Organization		
2	Database Management System		
3	Object Oriented Programming using Java		
4	Lab-4 : Object Oriented Programming using Java		
5	Lab-5: Database Management System		
6	Statistical Inference		
7	Foundation Mathematics for Machine Learning II		

Name : NIHAL NARAYANAN
Register No : NA20PICS01
Programme : Integrated M.Sc. Computer Science
Semester : 4

Figure 6.1: Uploading result

In summary, our system ensures accuracy, ease of use, and effective data management for the task of uploading and managing student results during each semester, making the entire process a user-friendly and error-free experience.

6.1.2   Edit/view result

In addition to the primary result upload functionality, we have also seamlessly integrated the capability to edit and update student marks, particularly when supplementary or improvement exam results are published by the university. Teachers can easily edit and update student marks, as shown in Figure 6.2. After editing, clicking 'Submit' triggers a successful submission message, along with a 'Next' button for seamless navigation to the next student.

Figure 6.2: Edit result

6.1.3 Different types of analyses

We have conducted various types of analysis, including departmental top analysis and semester-wise analysis, student analysis, single subject analysis etc.

1. Department Top - In this, users can specify the desired number of top students for each semester, as shown in Figure 6.3, for each class. Additionally, we provide a printing option for generating hard copies of the results.

Figure 6.3: Department top searching

2. Student Analysis - In student analysis, we can retrieve the marks from multiple semesters of a student at once, and the overall result is obtained only when marks from all semesters are uploaded. Additionally, it offer bar diagram Figure 6.4 showing the percentage for each semester, facilitating communication between teachers and students. It also offer a printing option, which we believe is valuable for meetings with parents and other documentation purposes.
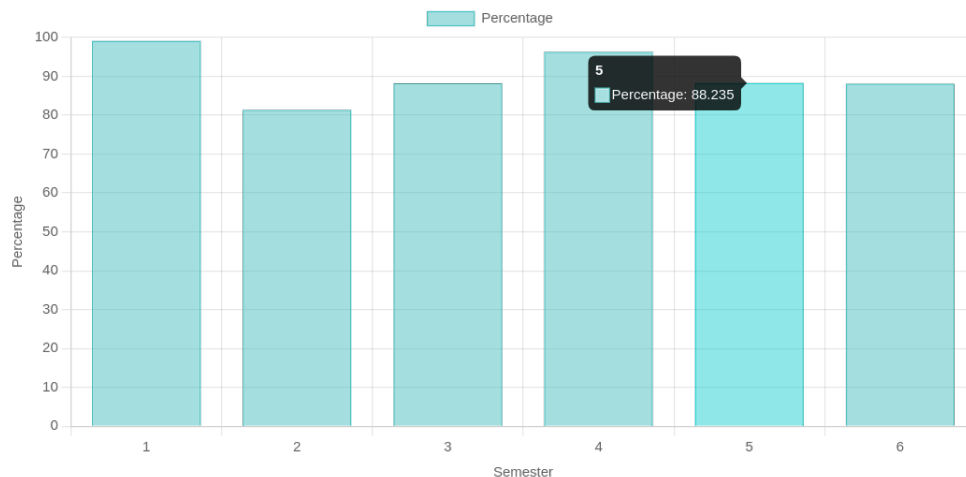
Figure 6.4: Bar diagram of a student based on semesters

3. Single Subject Analysis - In this, we display students marks for a specific paper. Additionally, we provide a pie chart illustrating the distribution of grades, Figure 6.5.
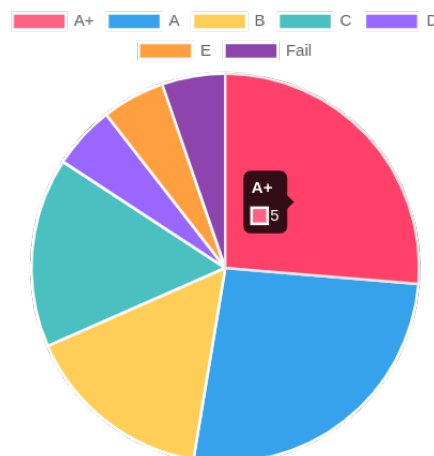
Figure 6.5: Pie chart depicting the distribution of grades for a specific paper.

4. Semester Wise Analysis - In this, we display the complete results of all students in each semester. Additionally, we provide a separate table that shows the number of grades in each paper. This feature helps identify the number of students who pass and fail.

5. Advanced Search - In the advanced search feature, you have the flexibility to search for results in various ways. Semester is a mandatory input, and you can further refine your search using options such as name, registration number, program, year of admission, quota, status (pass or fail), and gender. This enables you to perform diverse searches.

   eg: finding all male students who passed in BSc Plant Science and are also sports quota students.

6.2   Discussion

The Result Analysis System is a development, in the field of evaluating performance in educational institutions. This system offers educators an efficient platform equipping them with tools to analyze student results comprehensively. Our Result Analysis System introduces an automated approach significantly enhancing the accuracy and effectiveness of result assessments. By eliminating the possibility of errors it establishes a foundation, for consistently dependable academic evaluations.

While the Result Analysis System has made strides in addressing challenges there is still plenty of room, for improvement and further development. One area that holds promise for work is incorporating features that allow for reading and analyzing PDF result documents. This would greatly expand the systems capabilities making it more versatile in handling a range of document formats. Additionally integrating PG result analysis into the system would cater to the needs of institutions. Furthermore introducing the option to include open course names would enhance user friendliness and adaptability ensuring integration with academic programs. Lastly implementing security measures is crucial to fortify data protection and safeguard sensitive academic information. Measures such as role based access control, data encryption and intrusion detection systems can be employed to preserve the integrity and confidentiality of data. All these potential enhancements would make the Result Analysis System even more robust, flexible and secure. Ultimately improving its usefulness, for educational institutions.

Chapter 7

Contributions

In this project, we collaborated to develop the Result Analysis System. We divided the work into phases, including planning, design, implementation, testing, and evaluation. Here, we'll outline each team member's contributions, the challenges we encountered, and the lessons we took from the project.

Aiswarya V

- **Technical Contributions:**

  - Developed PHP files designed to facilitate the seamless uploading of academic results into the database.

  - Conducted various analyses of the results, each focusing on different aspects and variables to gain comprehensive insights.

  - Worked on integrating front end with the backend, ensuring smooth data flow and functionality.

- Design Contributions

  - Played an important role in designing the login page, analysis page, and other user interfaces, ensuring a user-friendly and visually appealing experience for teachers.

  - Made substantial contributions to the overall styling and design, utilizing HTML, CSS, JavaScript, and Bootstrap to create a cohesive and user-friendly interface that enhances the system's usability.

  - Worked on integrating front end with the backend, ensuring smooth data flow and functionality.

- Overcoming Challenges

  - Addressed the issue of uploading marks that exceeded the maximum allowable mark for a specific paper.

  - Assisted in resolving frontend integration issues, ensuring seamless communication between frontend and backend components.

- Reflection and Growth

  - Acknowledge in PHP programming skills during the development of the Result Analysis System.

  - Gained a deeper understanding of security practices and measures in web application development.

– The project enriched our practical skills in crafting web pages that are both visually appealing and user-friendly.

Anjana A

- **Technical Contributions:**

  – Developed PHP files designed to facilitate the seamless editing of academic results into the database.

  – Conducted various analyses of the results, each focusing on different aspects and variables to gain comprehensive insights.

  – Worked on integrating front end with the backend, ensuring smooth data flow and functionality.

- Design Contributions

  – Took a lead role in conceptualizing and designing the uploading result page, analysis page, and editing page, prioritizing a cohesive visual identity, ease of use, and accessibility to ensure a seamless interaction for users across the system.

  – Made substantial contributions to the overall styling and design, utilizing HTML, CSS, JavaScript, and Bootstrap to create a cohesive and user-friendly interface that enhances the system's usability.

  – Worked on integrating front end with the backend, ensuring smooth data flow and functionality.

- Overcoming Challenges

  – Addressed security vulnerabilities identified during testing, implementing measures like tokens to secure user actions and data.

  – Worked on optimizing database queries to enhance system performance, overcoming performance-related challenges.

- Reflection and Growth

  – Acknowledge in PHP programming skills during the development of the Result Analysis System.

  – Acknowledge the acquisition of proficiency in MySQL database management during the project.

  – The project enriched our practical skills in crafting web pages that are both visually appealing and user-friendly.

Chapter 8

Conclusion

The Result Analysis System represents a transformative innovation in the realm of academic performance evaluation within educational institutions. This system provides an efficient and user-friendly interface for teachers to analyze students' results. Our Result Analysis System offers a streamlined and automated approach that significantly enhances the accuracy and efficiency of result analysis. By eliminating the margin for human errors, it ensures that academic evaluations are consistently reliable.

By allowing teachers to conveniently upload exam results and promptly generate comprehensive reports, our system equips educators with valuable insights into their teaching methods and student outcomes. Armed with this data, educators can make informed decisions to refine their teaching strategies, ultimately enhancing the overall learning experience. In addition to its user-friendly interface and efficiency, the system incorporates various analysis components, such as department top analysis, student analysis, semester analysis, and single subject analysis etc.

Furthermore, our system extends its utility beyond current students. It also enables the retrieval of results for previously studied students. This flexibility ensures that historical academic data remains accessible and useful, aiding educators in making data-driven decisions over time. Additionally, the principal of the college can use the system to analyze results, gaining valuable insights into overall academic performance.

While the Result Analysis System has addressed many existing challenges, there is room for further enhancement. Future work could focus on incorporating additional features, such as the ability to read and analyze PDF result documents, PG result analysis, adding open course names and implementing more advanced security measures to safeguard sensitive academic data.

In conclusion, the Result Analysis System web application has proven to be a valuable tool for optimizing the course selection process and improving the overall experience for teachers. The successful development and implementation of the Result Analysis System mark a significant step towards modernizing and streamlining administrative procedures within the college. This system not only enhances the efficiency and accuracy of academic result analysis but also empowers educators to make data-driven decisions, ultimately contributing to a more effective and student-centric learning environment.

Bibliography

[1] Lynn Beighley and Michael Morrison. *Head First PHP & MySQL: A Brain-Friendly Guide*. O'Reilly Media, 2008.

[2] Tim Berners-Lee. Html tutorial. `https://www.w3.org/TR/html401/`.

[3] David Flanagan and Gregor M Novak. Java-script: The definitive guide, 1998.

[4] Steven Holzner. *PHP: The Complete Reference*. McGraw-Hill, Inc., 2007.

[5] Pankaj Jalote. *An Integrated Approach to Software Engineering*. Springer Science & Business Media, 2012.

[6] Julie C Meloni. *Sams Teach Yourself HTML, CSS, and JavaScript All in One*. Pearson Education India, 2016.

[7] Abraham Silberschatz, Henry F. Korth, and S. Sudarshan. *Database Systems Concepts*. McGraw-Hill Higher Education, 7th edition, 2019.

[8] Ian Sommerville. *Software Engineering*. Pearson, 10th edition, 2015.

[9] W3Schools. Php tutorial. `https://www.w3schools.com/php/`.

Appendices

Appendix A

Sample Code

resultupload.php

```php
if (isset($_GET['pgm_id']))
{
    $semester = $_GET['sem'];
    $stud_id = $_GET['stud_id'];
    $pgm_id = $_GET['pgm_id'];
    $roll_no = $_GET['roll_no'];
    $year_of_admn = $_GET['year_of_admn'];
    $query = "SELECT pgm_id FROM stud_master WHERE stud_id=" .
    $stud_id;

    $pgm_ids = mysqli_query($dbc, $query);
    foreach ($pgm_ids as $a) {
        $pgm_id = $a['pgm_id'];
    }
    $query = "SELECT pgm_name FROM programme WHERE pgm_id=" .
    $pgm_id;
    $pgm_names = mysqli_query($dbc, $query);
    foreach ($pgm_names as $a) {
        $pgm_name = $a['pgm_name'];
    }
    $query = "SELECT name FROM stud_master WHERE stud_id=" .
    $stud_id;
    $names = mysqli_query($dbc, $query);
    foreach ($names as $a) {
        $name = $a['name'];
    }
    $query = "SELECT uty_reg_no FROM stud_master WHERE stud_id=" .
    $stud_id;
    $utyregs = mysqli_query($dbc, $query);
```

```php
    foreach ($utyregs as $a) {
        $utyreg = $a['uty_reg_no'];
    }
}


if (isset($_SESSION['username']))
{
    if($_SESSION['role_id'] == 2)
    {
    if (isset($_POST['submit']))
    {
        $stud_id = $_POST['stud_id'];
        $semester = $_POST['semester'];
        $roll_no = $_POST['roll_no'];
        $year_of_admn = $_POST['year_of_admn'];

        foreach ($_POST['ce'] as $course_id => $ce_value)
        {
            $ese_value = $_POST['ese'][$course_id];
            $query = "INSERT INTO sem_exam (course_id,stud_id,ce,
    ese) VALUES ('$course_id','$stud_id','$ce_value', '$ese_value')";
            mysqli_query($dbc, $query);
        }
        $success_url = 'http://' . $_SERVER['HTTP_HOST'].dirname(
    $_SERVER['PHP_SELF']) . '/resultuploadsuccess.php?'.'stud_id='.
    $stud_id.'&roll_no='.$roll_no.'&sem='.$semester.
        '&year_of_admn='.$year_of_admn;
        header('Location: ' . $success_url);
    }
    }
}
?>
```

   dept_top.php


```php
<?php
        if (isset($_POST['submit']))
        {   ?>
            <table id=printTable align="center" border="solid">
            <tr><th>UTY Reg No</th><th>Name</th><th>Percentage</th
```

```php
></tr>
        <?php
        $all_per=[];
        $top_students = [];
        foreach ($studs as $a) {
        $TotalCreditPoint=0;
        $SGPA = 0;
        $totalCredit =0;
        $percentage =0;
        $name = $a['name'];
        $uty = $a['uty_reg_no'];
        $stud_id =$a['stud_id'];
        $unique_course_titles = [];
        foreach ($course_ids as $a) {
        $course_id = $a['course_id'];
        $query = "SELECT course_title, total_internal,
total_external, credits FROM course WHERE course_id = " .
$course_id . " AND semester = " .$semester." credits <> " .
$credit . " AND  syllabus_intro_year IN (" . $year . ", " .
$year2 . ")";
                $courses = mysqli_query($dbc, $query);
                while ($row = mysqli_fetch_assoc($courses)) {
                    $course_title = $row['course_title'];
                    $total_internal = $row['total_internal'];
                    $total_external = $row['total_external'];
                    $credits = $row['credits'];
                    $unique_course_titles[] = $course_title;
                    $query3 = "SELECT ce,ese FROM sem_exam WHERE
  stud_id = " . $stud_id . " AND course_id = " . $course_id;
                    $marks = mysqli_query($dbc, $query3);
                    $ce_values = [];
                    while ($row = mysqli_fetch_assoc($marks)) {
                        $ce = $row['ce'];
                        $ce_values[] = $ce;
                        $ese = $row['ese'];
                        $ese_values[] = $ese;
                    }
                    if (empty($ce_values)) {
                    continue;
                    } else {
```

```php
                    foreach ($ce_values as $ce) {
                        $mark = $ese + $ce;
                        $GP = ($mark/($total_external+
$total_internal)) * 10 ;
                        $CreditPoint=$credits * $GP;
                        $TotalCreditPoint = $TotalCreditPoint+
$CreditPoint;           }
                    }
                    $totalCredit = $totalCredit + $credits;
 }

            }
            $SGPA= $TotalCreditPoint/ $totalCredit;
            $percentage = $SGPA *10;
            $all_per[] = $percentage;
            $top_students[] = ['name' => $name, 'uty_reg_no' =>
$uty, 'percentage' => $percentage];
        }
        usort($top_students, function ($a, $b) {
            return $b['percentage'] - $a['percentage'];
        });
        for ($i = 0; $i < $top && $i < count($all_per); $i++) {
            $student_details = $top_students[$i];
            $name = $student_details['name'];
            $uty_reg_no = $student_details['uty_reg_no'];
            $percentage = $student_details['percentage'];
            $formatted_percentage = number_format($percentage,
2);
            echo "<tr><td>{$uty_reg_no}</td><td>{$name}</td><td
>{$formatted_percentage }</td></tr>";
        }
        ?>
        </table>
        <br>
        <div align=center>
            <button class="print" onclick=print()>Print</button>
        </div>
        <?php
        }
    ?>
```

semwiseanalysis.php

```php
<?php
        if (isset($_POST['submit']))
        {
            <br><br><br>
            <table id="printTable2" align="center" border="solid"
    width="50%">
            <?php $tables[]="printTable2"; ?>
            <tr><th>Papers</th>
            <th>A+</th>
            <th>A</th>
            <th>B</th>
            <th>C</th>
            <th>D</th>
            <th>E</th>
            <th>Fail</th></tr>
            <?php
            $unique_course_titles=[];
            $total_externals=[];
            $total_internals=[];
            $i=null;
            $course_ids2=[];
            foreach ($course_ids as $a)
            {
                $Aplus=$A=$B=$C=$D=$E=$Fail=0;
                $course_id = $a['course_id'];
                $query = "SELECT course_title,total_external,
    total_internal FROM course WHERE course_id = " . $course_id . "
    AND semester = " . $semester."  AND credits <> " . $credit . "
    AND  syllabus_intro_year
                IN (" . $year . ", " . $year2 . ")";
                $courses = mysqli_query($dbc, $query);
                foreach($courses as $a)
                {
                    $course_title = $a['course_title'];
                    $total_internal = $a['total_internal'];
                    $total_external = $a['total_external'];
```

```
                if($course_title != $i)
                {
                     $total_externals[]=$total_external;
                     $total_internals[]=$total_internal;
                     $course_ids2[]=$course_id;
                     $unique_course_titles[]=$course_title;
                     $i=$course_title;
                }
           }
      }
      $i=0;
      foreach ($course_ids2 as $course_id)
      {
           $total_external=$total_externals[$i];
           $total_internal=$total_internals[$i];
           $course_title=$unique_course_titles[$i];
           $i++;
           foreach ($studs as $a)
           {
                $stud_id = $a['stud_id'];
                $query = "SELECT ce,ese FROM sem_exam WHERE
course_id = ".$course_id." AND stud_id = ".$stud_id."";
                $semmarks = mysqli_query($dbc,$query);
                foreach($semmarks as $a)
                {
                     $ese = $a['ese'];
                     $ce = $a['ce'];
                     $mark=$ese + $ce;
                     $GP = ($mark/($total_external+
$total_internal)) * 10 ;
                     if($GP >= 9.00)
                     {
                          $Aplus=$Aplus+1;
                     }
                     else if($GP >= 8.00 && $GP <= 8.99)
                     {
                          $A=$A+1;
                     }
                     else if($GP >= 7.00 && $GP <= 7.99)
                     {
```

```
                    $B=$B+1;
              }
              else if($GP >= 6.00 && $GP <= 6.99)
              {
                    $C=$C+1;
              }
              else if($GP >= 5.00 && $GP <= 5.99)
              {
                    $D=$D+1;
              }
              else
              {
                    if($total_external == 40)
                    {
                        if(($ese >= 16) && ($ce + $ese >=20)
    )

                        {
                              $E=$E+1;
                        }
                        else
                        {
                              $Fail=$Fail+1;
                        }
                    }
                    if($total_external == 20)
                    {
                        if(($ese >= 8) && ($ce + $ese >=10))
                        {
                              $E=$E+1;
                        }
                        else
                        {
                              $Fail=$Fail+1;
                        }
                    }
              }
          }
      }
      if($previous_course != $course_title)
      {
```

```php
            $previous_course=$course_title;
            echo '<tr>
                    <th>' . $course_title . '</th>
                    <td>' . $Aplus . '</td>
                    <td>' . $A . '</td>
                    <td>' . $B . '</td>
                    <td>' . $C . '</td>
                    <td>' . $D . '</td>
                    <td>' . $E . '</td>
                    <td>' . $Fail . '</td>
                  </tr>';
            $Aplus=$A=$B=$C=$D=$E=$Fail=0;
        }
        else
        {
            $Aplus=$A=$B=$C=$D=$E=$Fail=0;
            continue;
        }
    }?>
    </table>
    <div align=center>
        <br>
        <button class="print" onclick=print()>Print</button>
    </div>
    <?php
    }
?>
```