

# Student Result Management System using PHP

**Dr. Anup Bhang<sup>1</sup>, Maithili Andankar<sup>2</sup>, Diksha Tiwari<sup>3</sup>**

Head of Department of Computer Application<sup>1</sup>,

Department of Master of Computer Application<sup>2,3</sup>

K. D. K College of Engineering, Nagpur, Maharashtra, India<sup>1,2,3</sup>

anupbhang@gmail.com<sup>1</sup>, maithiliandankar.mca23@kdkce.edu.in<sup>2</sup>, dikshatiwari.mca23@kdkce.edu.in<sup>3</sup>

**Abstract:** *Our educational institution maintains digital records of student data to ensure ease of access and management. However, these records are currently dispersed across various formats, posing challenges in maintenance, especially with the growing student population. This fragmentation often results in redundant data entry and inconsistencies, as students are required to repeatedly provide the same information. In response, we propose the development of a comprehensive web application designed to function as an integrated platform for aggregating and categorizing student data. This platform aims to consolidate all student-provided information into a centralized repository, which can then be categorized and selectively accessed by different user types according to their respective needs. Additionally, the web application will empower students to effortlessly update their information while facilitating efficient retrieval of selected student data by college staff. Moreover, it will streamline the online admission process by offering the option of online payment through a secure payment gateway. Furthermore, the application will include features for recording student marks in their respective courses, as well as tracking attendance. It will also inform students of new placement opportunities offered by the college and maintain records of faculty and departments. Additionally, it will log online transactions conducted by students during the admission process. The development of this application will utilize HTML, CSS, and JavaScript for the front-end, ensuring an intuitive and visually appealing user interface. On the backend, PHP and MySQL will be employed to provide robust data management and processing capabilities.*

**Keywords:** element, arrangement, design, presentation, embedding

## I. INTRODUCTION

Every educational institution necessitates a management system to uphold essential records of its students. As the institution expands, the student population burgeons, and consequently, the volume of student-related data escalates. Given the substantial daily influx of data, there arises a pressing need to enhance the efficiency and efficacy of student information management systems [1]. In today's interconnected world, the Internet has ingrained itself into the daily routines of a vast majority of individuals globally. Among the myriad activities undertaken online, one notable aspect is the storage of personal data within databases according to users' preferences. These databases facilitate users in submitting queries and retrieving pertinent information swiftly. Thus, there exists a demand for software systems dedicated to managing student information, which not only ensures time-efficient access to required data but also facilitates online accessibility [2]. Within any educational institution, numerous administrative departments are tasked with overseeing the institution's information and student databases. Each of these departments maintains diverse records pertaining to students, encompassing general particulars such as names, addresses, academic performance, and attendance, alongside specialized data pertinent to specific departments or courses. The records managed by these departments are indispensable for the seamless functioning of various modules within college administration. However, currently, these records are predominantly managed manually, necessitating a shift towards automation and centralization. Centralization is imperative as information from one module often proves indispensable for other modules, facilitating seamless sharing of course content and tracking of students' progress [3]. In summary, the Student Result Management System represents a paradigm shift in the way educational institutions manage student results. By embracing digital technology and automation, SRMS streamlines administrative processes, enhances transparency, and

empowers stakeholders with valuable insights. As educational institutions embark on the journey of digital transformation, SRMS emerges as an indispensable tool for fostering academic excellence and student success.

## II. OBJECTIVES

The objectives of the Student Result Management System (SRMS) project are as follows:

- Efficient Result Management: Optimal Result Administration
- Automation of Result Generation: Streamlining Result Production
- Secure Access Control: Robust Data Security Measures
- Real-Time Tracking of Student Performance: Immediate Monitoring of Academic Progress
- Comprehensive Analytical Insights: In-depth Data Analysis and Insights
- Enhanced Communication: Improved Stakeholder Interaction

## III. LITERATURE SURVEY

After conducting an extensive examination of literature regarding student outcome administration systems, it has become evident that there is a burgeoning requirement for effective and dependable systems to regulate student data within educational organizations. Traditional paper-based systems are deemed insufficient due to their inefficiency and proneness to errors, prompting the emergence of computer-based alternatives. Several inquiries have explored the utilization of various coding languages and database systems in the creation of these systems. Python has emerged as a prevalent and efficient language for constructing desktop applications, while the SQLite database is recognized for its reliability and efficacy in data storage. Significant attention has also been directed towards user interface design, with an emphasis on crafting interfaces that are intuitive and user-friendly. The tkinter library, a standard GUI toolkit for Python, has proven to be a valuable resource for creating user-friendly interfaces within Python-based applications. Furthermore, several studies have underscored the significance of security in student outcome administration systems, advocating for measures such as password protection to safeguard the confidentiality of student data. In essence, the literature review emphasizes the necessity of developing computerized systems that are both efficient and reliable for managing student data, with particular attention given to user interface design and security features. The Student Outcome Administration System project, leveraging Python GUI and SQLite database technologies, addresses these concerns and provides a robust solution for managing student data within educational organizations.

The proposed approach and its advantages over the current system are as follows:

- User-friendly (faculty members can effortlessly utilize the web-based application). Accessible at all times. (As long as the computer is connected to the network, the system remains accessible.) Simplified computation. Straightforward storage.
- Drawbacks of the current system:
- If a computer program is coded in the C programming language, it may rely on the operating system. The utilization of linear search in file handling could contribute to increased time complexity. Additionally, there is a lack of enhanced features commonly found in web applications.

Algorithms/programs in Use:

As mentioned earlier, various software applications exist today that aid users in discovering and storing fundamental information such as a student's name, grades, and seat number. The remaining computational tasks are either manually completed by faculty members at the university or necessitate separate software.

## IV. PROBLEM STATEMENT

The existing method of managing student information within the college is characterized by decentralized digital record-keeping across various platforms, including Excel spreadsheets and Google Drive. This fragmented system presents several challenges, notably the lack of cohesion in data interlinking, the repetitive nature of data entry, delays in accessing critical information, redundancy in stored data, and constraints in data manipulation capabilities. Moreover, as the student body continues to grow, the sheer volume of data compounds these issues, making it increasingly difficult to maintain, update, and efficiently retrieve specific information when needed.

## V. METHODOLOGY

To construct a student outcome management system employing PHP, js, css, html, Bootstrap and MySQLdb, the methodology encompasses multiple stages, including requirement elicitation, user interface design, database schema formulation, development, quality assurance, deployment, and upkeep. We use Css and Bootstrap for front end which can be leveraged to craft an intuitive user interface, while JS for validation of data orphp and MySQL can be logical and employed for streamlined storage of student data.

- *HTML*: The acronym for Hypertext Markup Language, serves as the primary markup language for crafting content intended for web browser display. It sets the standard for structuring textual information on the internet. Technologies such as Cascading Style Sheets (CSS) and scripting languages like JavaScript play vital roles in enhancing HTML-based content presentation.
- *CSS*: Cascading Style Sheets (CSS) functions as a language dedicated to defining the visual layout of documents composed in markup languages such as HTML. Together with HTML and JavaScript, CSS constitutes an integral component of the global network of interconnected documents, offering control over the appearance and formatting of web content.
- *PHP*: A versatile programming language initially conceptualized by Danish-Canadian programmer Rasmus Lerdorf in 1994, stands as a go-to tool for web development tasks. Its adaptability extends to a broad spectrum of general-purpose programming endeavours. Currently maintained by the PHP Group, PHP serves as a cornerstone for creating dynamic and interactive web-based applications.
- *MySQL*: An open-source relational database management system, distinguishes itself as a robust and freely accessible solution. The brainchild of co-founder Michael Widenius, who named it after his daughter My and combined it with "SQL" (Structured Query Language), MySQL empowers developers to efficiently store, retrieve, and manipulate structured data. Its widespread adoption underscores its significance in powering various web-based applications.
- *XAMPP*: Denoting Cross-Platform, Apache, MySQL, PHP, and Perl, offers a comprehensive suite of open-source web development tools. This versatile package simplifies the setup and management of local web server environments. Comprising essential components such as the Apache web server, MySQL or MariaDB database systems, and programming language interpreters for PHP and Perl, XAMPP facilitates seamless web application development and testing processes.
- *BOOTSTRAP*: A front-end framework, offers a robust toolkit for developing responsive and visually appealing web interfaces. Originally crafted by Twitter developers Mark Otto and Jacob Thornton, Bootstrap has evolved into an open-source project supported by a thriving community. Bootstrap facilitates rapid web development through its pre-designed components, such as buttons, forms, and navigation bars, alongside a flexible grid system for layout customization. Its integration with HTML streamlines the creation of mobile-first web designs, ensuring seamless rendering across various devices and screen sizes. Bootstrap's extensive CSS and JavaScript libraries enable developers to enhance user experience with interactive elements and dynamic content.
- *Features Of Bootstrap*: A distinguishes itself as a widely adopted and feature-rich front-end framework, renowned for its versatility and simplicity. Key attributes include its extensive library of pre-styled components and utilities, facilitating rapid prototyping and development. Bootstrap's grid system provides a responsive layout structure, ensuring consistent display across desktops, tablets, and smartphones. Its customizable themes and colour schemes offer flexibility in design customization, while robust documentation and community support simplify usage and troubleshooting. Additionally, Bootstrap's integration with popular JavaScript libraries enhances interactivity and functionality, empowering developers to create dynamic and engaging web experiences.

**The Classic Waterfall Development Model**

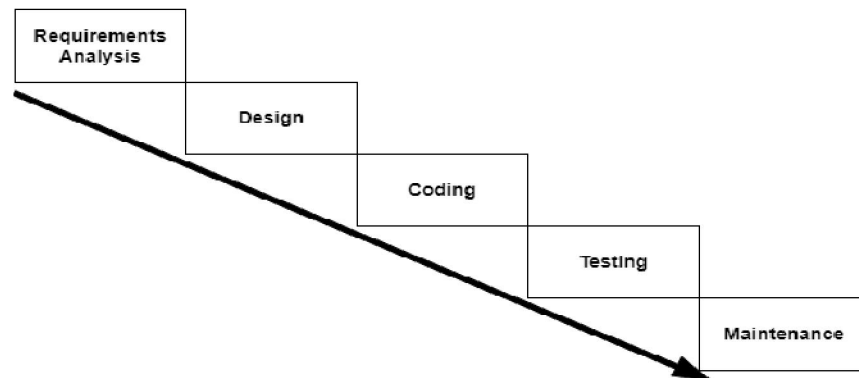


Fig.1 illustrates the waterfall model.

Originally derived from manufacturing and construction industries, the waterfall model is tailored for highly structured physical environments where retroactive alterations are exceptionally costly, if not unfeasible. Adapted for software development due to the absence of formal software development processes at the time, this hardware-centric model served as a foundation.

We employ this model owing to well-defined, clear, and immutable requirements, rendering it optimal for our software development endeavours. Our project definition remains stable. The simplicity and comprehensibility of this model make it straightforward and hassle-free to comprehend and employ. With its sequential nature, where phases are executed and completed one after the other without overlap, the waterfall model is suitable for projects where a predetermined sequence is well-established.

**VI. MODULES AND IMPLEMENTS**

The Registration Module facilitates the enrolment of new users by collecting their particulars. The Login Module oversees the administration of access credentials. The Course Module maintains records pertaining to the courses available. The Student Module oversees the management of student details and enrolment. The Result Module handles the organization of student performance outcomes. The View Result Module permits users to access their outcomes by inputting their credentials. These components offer a streamlined and structured approach to handling student information within an educational setting.

The Student Result Management System divided into two modules–

- Student
- Admin

Admin module allows administrators to manage users, access control, and system configurations, while the student module enables students to view their results, course schedules.

**Admin Features**

- Admin Dashboard
- Admin can add/update/ Class
- Admin can add/update/ Subjects
- Admin can add/update/ Active/Inactive Subject combination with class
- Admin can register a new student and also edit info of the student
- admin can declare/ edit the result of a student.
- Admin can change own password

**Student Features**

- Students can search their results using a valid roll-id.
- Student can view their result
- Student can print the result

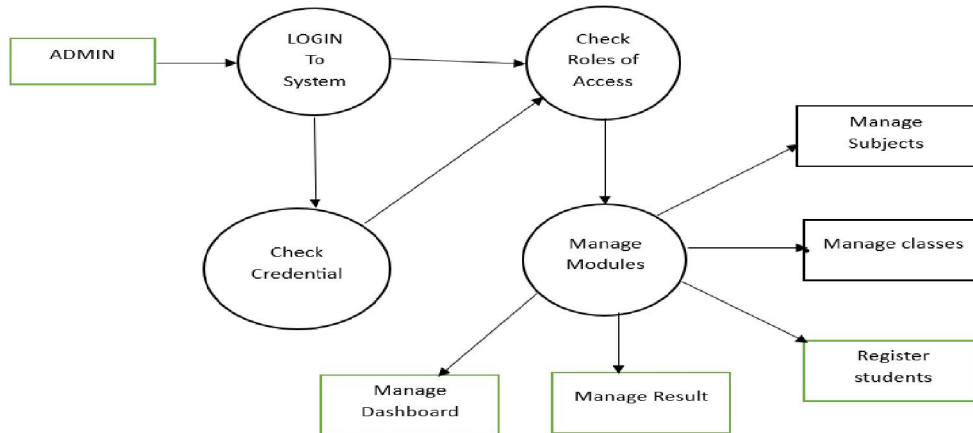


Fig 2: Data Flow Diagram

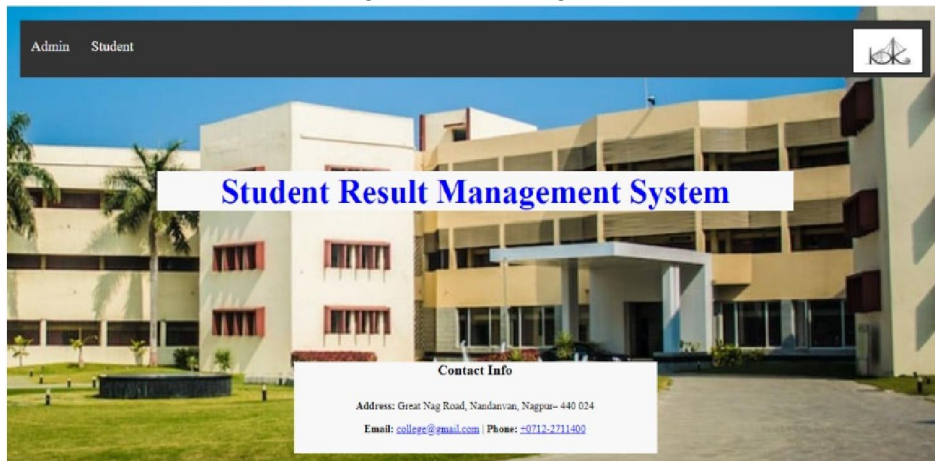
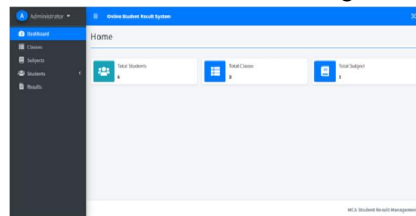


Figure 1: Home Page



**Admin Dashboard**

**View Result Page**

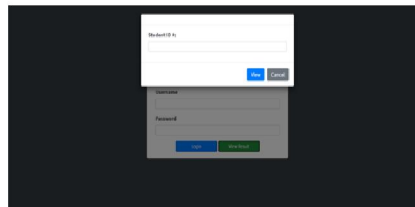


Figure 2: admin-Dashboard and view-result page



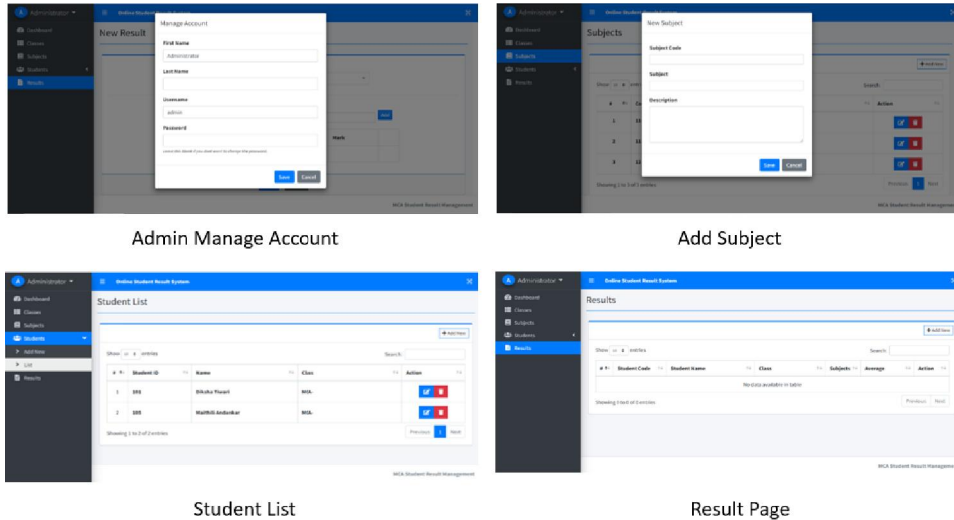


Figure 3: Modules

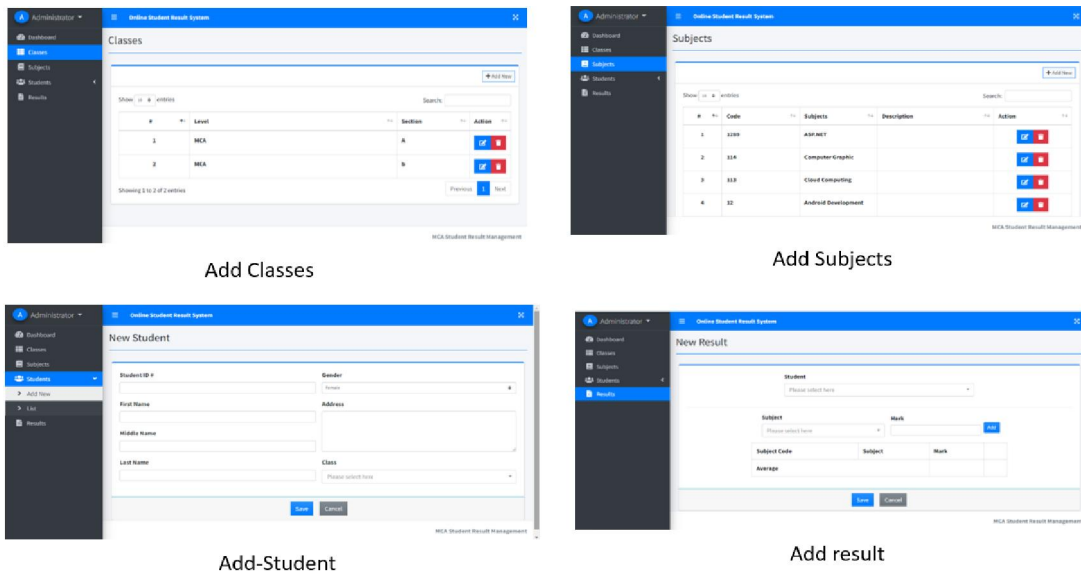


Figure 4: Modules

## VII. PROPOSED WORK

The proposed system operates with two primary roles: the Administrator and the Student. Administration duties are overseen by the Administrator role, who manages access to the system's database. Individuals with authorized access can retrieve stored information from the database. The Administrator holds complete control over the system, while Students are granted access to their profiles and semester results.

Data operations within the system are governed by three core actions: Read (r), Write (w), and Execute (x). The Administrator has access to six modules, each with specific permissions for various actions: Register (rwx), Login (wx), Profile (rwx), Setting (wx), Upload (rwx), and Logout (rwx). Conversely, Students have access to five modules, including Login (wx), Profile (rwx), Setting (wx), Result (rx), and Download Marks Sheet (Dx).

Authentication is mandatory for both roles prior to accessing any data, requiring login credentials. The encryption of the authentication method ensures that even in the event of a Man-In-The-Middle Attack (MIMA), attackers cannot decipher the process. Upon successful authentication, users are logged in, while unsuccessful attempts lead to

redirection to the homepage with a notification of "Mismatch username/password." A session between the server and the user is established post-login, allowing access to the user's designated modules. Termination of tasks is facilitated through a logout button, which ends the session.

Student information is securely stored within a relational database, constructed using MySQL server. Retrieval and processing of data are conducted using logical gates aligned with specific queries. The database architecture adheres to standardized practices to meet system requirements. Successful transactions are committed upon completion, ensuring data integrity. In the event of any issues, a rollback mechanism is executed, restoring data to its original committed state. Overall, the database design maintains the integrity of all ACID (Atomicity, Consistency, Isolation, Durability) characteristics.

### VIII. FUTURE SCOPE

In brief, the upcoming objectives of the project encompass:

Issuing student certificates

Enhancing the Student Result Management System with advanced features

Global accessibility by hosting the platform online

Historically, result analysis relied on manual data entry, but now utilizes SQL files for extraction. In the future, data retrieval and parsing could support diverse formats such as DOC, CSV, and ODT. Furthermore, there's potential for data visualization through various methods including pie charts and graphs. To enhance user experience, interactive elements like a student group chat can be integrated, fostering discussions on engineering topics to meet user needs effectively.

In the near future, the system interface is set to undergo enhancements, featuring more appealing, interactive, and meaningful imagery. Additionally, system functionality will be enriched with email and SMS notifications. Furthermore, the institution's services will be computerized extensively, transforming it into a comprehensive Learning Management System (LMS). System evolution will involve iterative development based on user feedback.

### IX. CONCLUSION

The Student Result Management System is an online platform accessible to individuals worldwide, anytime, and from any location. This application streamlines the calculation and presentation of results, benefiting both students and faculty members. Our project endeavors to meet the demands of project management with straightforward programming. This software package is anticipated to meet all organizational requirements and serve as a robust solution. The goal of software planning is to establish a structure enabling managers to generate accurate estimates at the project's outset and continually update them as the project advances. The document discusses the Student Result Management System (SRMS), which aims to address the challenges encountered by students in managing their academic records. The SRMS is developed using PHP, MySQL, HTML, CSS, and JavaScript, and it is locally hosted using an Apache web server. The software development approach is based on the Participatory Iterative Process (PIP) Model. A detailed breakdown of the system architecture and its fundamental components is provided to elucidate the main functionalities of the framework. Additionally, a use case diagram is included to illustrate the different types of system users and the diverse functionalities associated with each user category.

### X. ACKNOWLEDGMENT

The development of the M.C.A project necessitates the collaboration and guidance of numerous individuals. Therefore, we deem it our utmost responsibility to extend our gratitude to all those who aided us throughout this endeavour. We take immense pleasure in expressing our appreciation to Dr. Anup Bhangе, the Head of the Master of Computer Application Department, for his invaluable guidance and constructive feedback throughout the preparation of this project. We extend our sincere thanks for his unwavering support and coordination. Additionally, we would like to express our gratitude to Dr. Valsson Varghese, the Principal, for his generous cooperation, guidance, and provision of necessary facilities during the execution of this project. We also extend our thanks to all other faculty members of the Master of Computer Application Department for their valuable insights and cooperation.

**REFERENCES**

- [1] S.R.Bharamagoudar, Geeta R.B &S.G.Totad, "Web service Api for student information and course management systems". International Journal of AdvancedResearch in Computer and Communication Engineering Vol. June 2013.
- [2] Almahdi Alshareef, Ahmed Alkilany, "Toward a Student Information System for Sebha University, Libya", Fifth international conference on InnovativeComputing Technology (INTECH 2015)-p 34-39
- [3] Prabhu T Kannan, Srividya K Bansal, "Unimate: A Student Information System", International Conference on Advances in Computing, Communications andInformatics (ICACCI 2013)-p-1251-1256
- [4] Potula, H., SriChand, M., Kumar, P., Reddy, N. V. H., & Kumar, I. (2022). Student Result Management System Using Web Technologies. Journal of Emerging Technologies and Innovative Research (JETIR), 9(4), h182.
- [5] Singh, M., & Dev, P. (2023). Student Result Management System. International Journal of Scientific Research in Technology and Science, 5(4), 5174.
- [6] Bhanushali, R., Agarwal, C., Dongare, T., & Sharma, S. (2022). Student Management System. International Journal for Research in Applied Science & Engineering Technology (IJRASET), 10(7), 422.
- [7] Rainy, & Sharma, D., & Singh, H. (2023). Student Result Management System. Journal of Software Engineering and Simulation, 9(4), 53-61.