

**Tribhuvan University**

**Faculty of Humanities and Social Sciences**

**School Management System**

**A Proposal Report**

**Submitted to**

**Department of Bachelors in Computer Application**

**Jaya Multiple Campus**

**In partial fulfillment of the requirement for the Bachelor in Computer Application**

Submitted by:

Name: Dolma Lama

(6-2-721-10-2020)

Year/Semester:2nd/IV

August 2023

Under the Supervision of

Kumar Lamichhane

Table of Content

[Chapter 1: Introduction 8](#_Toc141949106)

[1.1 Background 8](#_Toc141949107)

[1.2 Problem Statement 8](#_Toc141949108)

[1.3 Objectives 8](#_Toc141949109)

[1.4 Scope and Limitation 8](#_Toc141949110)

[1.4.1 Scope 8](#_Toc141949111)

[1.4.2 Limitation 9](#_Toc141949112)

[Chapter 2: Literature Review 10](#_Toc141949113)

[2.1 Existing Projects 10](#_Toc141949114)

[2.1.1 Projectworlds (School Management System using python Django) 10](#_Toc141949115)

[2.1.2 CodeAstro 10](#_Toc141949116)

[Chapter 3: Requirement Analysis and Feasibility Study 11](#_Toc141949117)

[3.1 Requirement Identification 11](#_Toc141949118)

[3.1.1 Study of the existing system 11](#_Toc141949119)

[3.2 Feasibility Study 11](#_Toc141949120)

[3.2.1 Operational Feasibility 11](#_Toc141949121)

[3.2.2 Economic Feasibility 11](#_Toc141949122)

[3.2.3 Technical Feasibility 11](#_Toc141949123)

[3.2.4 Schedule Feasibility 11](#_Toc141949124)

[3.3 Requirement Specification 11](#_Toc141949125)

[3.3.1 Functional Requirements 11](#_Toc141949126)

[3.3.2 Non-Functional Requirements 12](#_Toc141949127)

[Chapter 4: System Planning 13](#_Toc141949128)

[4.1 Methodology 13](#_Toc141949129)

[4.1.1 Software development model 13](#_Toc141949130)

[4.2 Gantt Chart 14](#_Toc141949131)

[Chapter 5 : System Design 14](#_Toc141949132)

[5.1 System Architecture 15](#_Toc141949133)

[5.2 System Diagram 16](#_Toc141949134)

[5.2.1 Use Case Diagram](#_Toc141949135)

[16](#_Toc141949135)

[5.2.2 Data Flow Diagram 17](#_Toc141949136)

[5.2.3 ER Diagram 19](#_Toc141949137)

[Chapter 6: System Development and Testing 20](#_Toc141949138)

[6.1 Coding tools 20](#_Toc141949139)

[6.1.1 Front End 20](#_Toc141949140)

[6.1.2 Back End 20](#_Toc141949141)

[6.2 Software and Hardware Requirements 20](#_Toc141949142)

[6.3 Testing 21](#_Toc141949143)

[Chapter 7 : Lesson learnt/Outcome 22](#_Toc141949144)

[7.1 Expected Outcome 22](#_Toc141949145)

[7.2 Learnt Lessons 22](#_Toc141949146)

[Chapter 8 : Conclusion and Future Recommendation 23](#_Toc141949147)

[8.1 Conclusion 23](#_Toc141949148)

[8.2 Future recommendation 23](#_Toc141949149)

[References 23](#_Toc141949150)

[Appendix 25](#_Toc141949151)

**List Of Figures**

[Figure 1: Incremental Model 13](#_Toc141971916)

[Figure 2: Gantt Chart 14](#_Toc141971917)

[Figure 3: Three Tier Architecture 15](#_Toc141971918)

[Figure 4:Use Case Diagram 17](#_Toc141971919)

[Figure 5: Data Flow Diagram 18](#_Toc141971920)

[Figure 6: Data Flow Diagram 19](#_Toc141971921)

[Figure 7: ER Diagram 19](#_Toc141971922)

[Figure 8: Process Flow Diagram 20](#_Toc141971923)

**Acknowledgement**

First of all, I would like to express my sincere gratitude towards the Department of Humanities and Computer Science, Tribhuvan University for including the 4th Semester major project as part of the syllabus. Also, I extend my gratitude towards Mr. Subash Bista, Head of Department, Humanities and Computer Science, for assisting me in the project. Special thanks and gratitude to my project supervisor Mr. Kumar Lamichhane under whose guideline I was able to complete the project. I am whole heartedly thankful to him for giving me his valuable time and attention and for providing me a systematic way for completing my project in time.

Every attempt has been made to include each and every aspect of the project in this report so that the reader can clearly understand the project. I would be pleased to get the feedback on it.

Sincerely,

Dolma Lama

**Abstract**

A school management system is necessary in schools nowadays because it efficiently automates administrative tasks, improves communication between stakeholders, organizes and analyzes crucial data, enhances student performance tracking, optimizes resource management, ensures data security and privacy, supports remote learning, facilitates financial management, and aids in the institution's growth, ultimately leading to a more effective, well-organized, and successful educational environment.

The requirement for the best school management system arises because it addresses the limitations of existing systems by offering comprehensive features, user-friendliness, customizability, integration capabilities, security, customer support, updates, and cost-effectiveness, ensuring an optimal and future-proof solution for school administration and management.

*Keywords:[**School Management System,Management,Administrative task,Crucial,Enhances,Data, Optimizes, Integration,User-friendliness, Customizability,Institution]*

**List Of Acronyms**

AI Artificial Intelligence

CSS Cascading Style Sheets

DFD Data Flow Diagram

DRY Don't Repeat Yourself"

ER Entity-Relationship

HTML Hyper Text Markup Language

LMS Learning Management Systems

SDLC Software Development Life Cycle

UAT User Acceptance Testing

UI User Interface

UML Unified Modeling Language

VS Visual Studio

## 

# **Chapter 1: Introduction**

## **1.1** **Background**

A School Management System is a comprehensive software solution designed to streamline and automate various administrative and operational tasks within an educational institution. It provides a centralized platform that enables efficient management of student information, academic records, attendance, and other essential aspects of school administration.

By incorporating a School Management System, educational institutions can improve their overall efficiency, reduce paperwork, and enhance communication between teachers, students, and administrators. The system typically includes modules for admissions, student data management, attendance tracking, and generating various reports. Security and data privacy are paramount concerns in any educational institution, and the Django framework addresses these issues through its built-in security features. With robust authentication mechanisms and data encryption protocols, the School Management System ensures that sensitive information remains safeguarded from unauthorized access.

In summary, a School Management System is a powerful tool that optimizes the administrative processes of educational institutions, promotes effective communication, and enhances the overall productivity and effectiveness of the school ecosystem.

## **1.2 Problem Statement**

* Manual and time-consuming administrative tasks: Traditional paper-based systems for managing student information, attendance and other administrative tasks can be labor-intensive, error-prone, and time-consuming. This can lead to inefficiencies, delays, and inaccuracies in data management, resulting in administrative burden and reduced productivity.
* Limited accessibility and communication gaps: Inadequate communication and limited accessibility to information can hinder effective collaboration among teachers, students, and administrators. The lack of a centralized platform can make it challenging to share important updates, progress reports, and other relevant information, leading to miscommunication and a lack of transparency.
* Data security and privacy concerns: Schools handle sensitive student information, including personal details and academic records. Without a robust School Management System, data security and privacy can be compromised.

## **1.3 Objectives**

a. To Streamline administrative processes

b. To Improve data management and reporting

c. To maintain data security and privacy

## **1.4 Scope and Limitation**

### **1.4.1 Scope**

1) Student Information Management

3) Teacher Information Management

3) Admissions and Enrollment

4) Attendance and Leave Management

5) Fee Management

### **1.4.2 Limitation**

1) Initial setup and implementation

2) Connectivity and accessibility

3) Cost considerations

# 

# 

# **Chapter 2: Literature Review**

## **2.1 Existing Projects**

### **2.1.1 Projectworlds (School Management System using python Django)**

Django School Management System by Yugesh Verma[1], is used by school manager to manage their school records: student data staff results and finances4. Technologies used in this project are HTML, CSS, JavaScript, Python, SQLite and Django. Solely, it’s expected to be used on a single machine or online for managers only. It has an interactive dashboard built in for admin using which admin can manage student, manage fees, manage employees, manage results. The results have improved as it made easier for school manager but it currently doesn’t allow students/staff to login. Ultimately, I would make the system in such a way that, the student/staff can login to the system and do different academic activities.

### **2.1.2 CodeAstro**

A project by Sumit Kumar[2] based on School Management System which used Python with Django Web Framework. This system as well as the web application’s concept was all clear, it was the same as real-life scenarios and well-implemented on it. In particular, this school management system project in Python Django focused mainly on dealing with student-teacher records. Also, the system displayed all the available dues amounts. In addition, the system allowed managing attendance records. Evidently, this project was divided into three categories: Student, Teacher, and Admin Panel. In an overview of this web application, a student can simply register and start using it. Student /Teacher has to apply for it. After confirmation of a student’s account, the user can view his/her details, public notice from admin and teachers. The result of this project is improved as it includes almost all the features.

### **2.1.3 VetBosSel**

In this project by J. Vetrivel Pandian[3], it explained how to create college management system Python using Django framework. It included the features like add student or staff, manage student/staff, notify them, about subject and courses. It explained about how to use SQLite in project. This project helped me to use SQLite in my project.

**2.1.4 Ijraset**

Student Record Management System using Django by Vivek K. Patil[4], is a software helpful for student as well as department staff. The Student Record Management System (SRMS) is a web-based application designed to manage student records efficiently. Using the Django framework, SRMS included features such as registration, enrolments, grading, photos, remarks, and record tracking. The system focused on student-related data, including their CGPA, address, phone number, fees, and other details. This project helped to know how to manage students’ records.

# **Chapter 3: Requirement Analysis and Feasibility Study**

## **3.1 Requirement Identification**

### **3.1.1 Study of the existing system**

There are a lot of existing systems in the School Management System. In some system it doesn’t allow Teacher/Student to login, in some system there is no features to take an attendance and in another one there is no feature to assign class teacher, subject teacher. Although there are different kinds of systems, I have created them in such a way that it includes above all features and in order to provide an easy interface to the user. The main purpose for creating this system is to provide easy working platform for Teacher, Admin and Student.

## **3.2 Feasibility Study**

A feasibility study is an evaluation and analysis of a project or system that somebody has proposed.

### **3.2.1 Operational Feasibility**

This system uses simple technologies to design. So, it is user friendly.

### **3.2.2 Economic Feasibility**

The system does not require extra software and hardware i.e., it uses open-source technologies. So, there is no recurring cost than just the internet connection.

### **3.2.3 Technical Feasibility**

In order to design this system, it uses off-self and existing technologies, software and hardware so there is no technological hurdle to build this system. The following software are used for the development of the system.

1. VS Code

### **3.2.4 Schedule Feasibility**

It involves evaluating the project's timeline, milestones, and tasks to determine if they are achievable within the desired schedule constraints. Schedule feasibility is essential for project planning and management as it helps in setting realistic expectations, identifying potential scheduling issues, and ensuring that the project can be completed on time.

## **3.3 Requirement Specification**

### **3.3.1 Functional Requirements**

1. **User Module**
2. User can login the system
3. User can go to their respective dashboard
4. User can view their profile, attendance
5. Users can post notice
6. User can logout from the system
7. **Admin Module**
8. Admin can login the system
9. Admin can add, delete, update, view and take attendance of users
10. Admin can view the list of users
11. Admin can post notice
12. Admin can logout from the system

### **3.3.2 Non-Functional Requirements**

1. Usability and User Interface: The School Management System should have an intuitive and user-friendly interface that is easy to navigate for administrators, teachers and students.

2. Performance and Scalability: It should be able to handle peak loads during busy periods, such as admissions or examination times. Additionally, the system should be scalable to accommodate future growth in student enrollment and increased usage.

3. Security and Privacy: The system should comply with relevant data protection regulations, ensuring the privacy and confidentiality of student and teacher information.

4. Reliability and Availability: It should have measures in place to prevent data loss, such as regular backups and disaster recovery plans. The system should also have failover capabilities to ensure uninterrupted access in the event of hardware or network failures.

5. Accessibility: It should provide features such as screen reader compatibility, keyboard navigation, and adjustable font sizes to accommodate different accessibility needs.

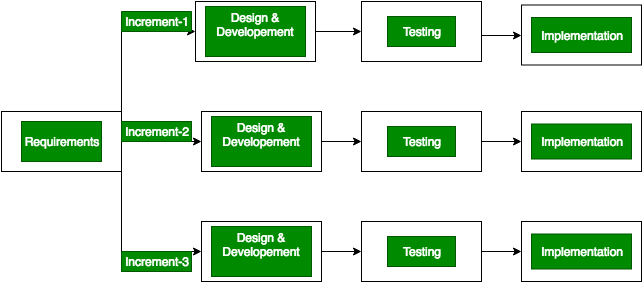
# 

# **Chapter 4: System Planning**

## **4.1 Methodology**

### **4.1.1 Software development model**

In this model, each module goes through certain requirements, design, implementation and testing phases. Every subsequent release the module adds function to previous release. The process continues until the complete system is achieved. I have used this model due to superior requirements. Also, my project has a lengthy development schedule. I am not fully well skilled or trained. I have applied learning by doing concept.

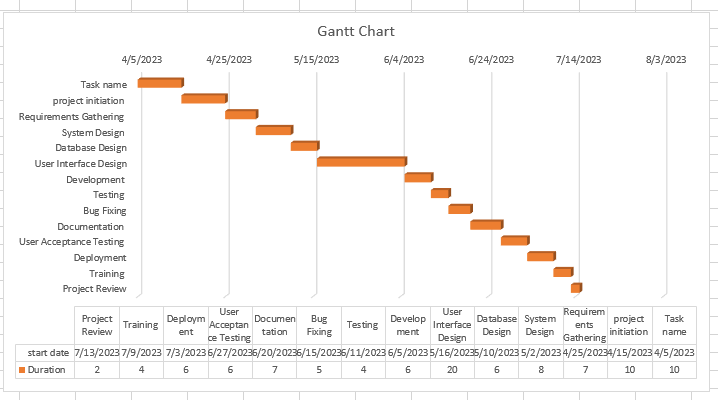


**Figure 1: Incremental Model**

1. **Requirement analysis:** In the first phase the system functional requirements are understood with the help of existing system. To develop the software under the incremental model, this phase performs a crucial role.
2. **Design & Development:** In this phase of the Incremental model of SDLC, the design of the system functionality and the development method are finished with success. When software develops new practicality, the incremental model uses style and development phase.
3. **Testing:** In the incremental model, the testing phase checks the performance of each existing function as well as additional functionality. In the testing phase, the various methods are used to test the behavior of each task.
4. **Implementation:** Implementation phase enables the coding phase of the development system. It involves the final coding that design in the designing and development phase and tests the functionality in the testing phase.
5. **Maintenance:** In this phase, all the bugs are fixed and performance also maintained.

## **4.2 Gantt Chart**

A Gantt chart is a popular project management tool used to visualize the schedule of tasks or activities in a project over time. It provides a graphical representation of project timelines, showing the start and end dates of each task, as well as their dependencies and durations. Gantt charts are widely used for planning, scheduling, and monitoring project progress.



**Figure 2: Gantt Chart**

The given chart shows the task to be performed on the vertical axis and the time specified

to perform the task in the horizontal axis. This chart is produced to show the management

of School. Gantt Chart helps us to provide a clear view of task durations and dependencies, helps set and meet project deadlines, simplifies project planning and monitoring etc. Further it enables adjusting schedules when changes occur real-time assessment of project progress.

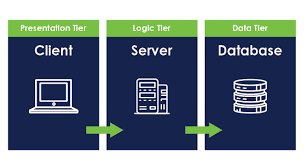
# **Chapter 5: System Design**

## **5.1 System Architecture**

The proposed system is the three-tier architecture which is a client-server architecture in

which the functional process logic, data access, computer data storage and user interface

are developed and maintained as independent modules on separate platforms.



**Figure 3: Three Tier Architecture**

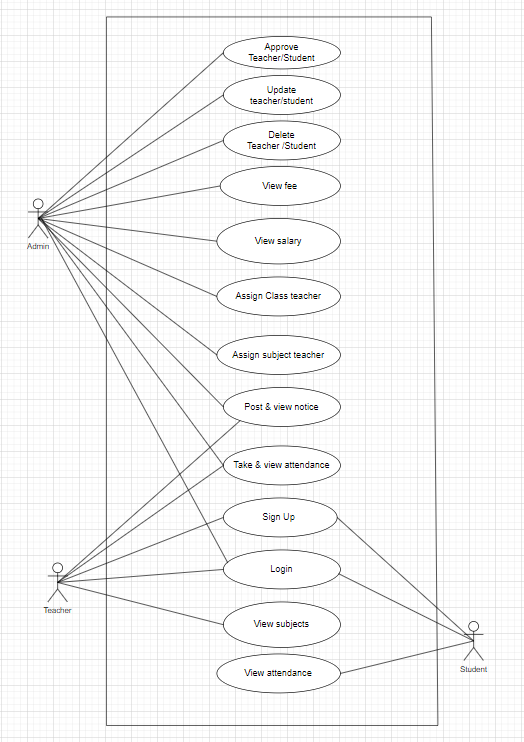
When designing the architecture for a School Management System, it's essential to consider scalability, reliability, security, and ease of maintenance. Here's an example of a common architectural design for a School Management System:

1. **Client-Server Architecture:** The system can adopt a client-server architecture where the client-side represents the user interfaces accessed by administrators, teachers and students and the server-side handles the processing and storage of data.
2. **Presentation Layer:** This layer focuses on user interfaces and interactions. This layer can be developed using, HTML, CSS and Bootstrap.
3. **Application Layer:** The application layer contains the core business logic and processes of the School Management System.It handles functionalities such as user authentication, data validation, workflow management, and integration with external systems.This layer can be built using programming language Django Python.
4. **Database Layer:** The database layer manages the storage and retrieval of data.It includes the database system where user information, attendance records and other relevant data are stored.Commonly used database management systems include SQLite.
5. **Security and Data Protection:** Implement strong security measures at various levels, including authentication mechanisms, role-based access control, and data encryption to ensure the privacy and integrity of student and staff data.

## **5.2 System Diagram**

### **5.2.1 Use Case Diagram**

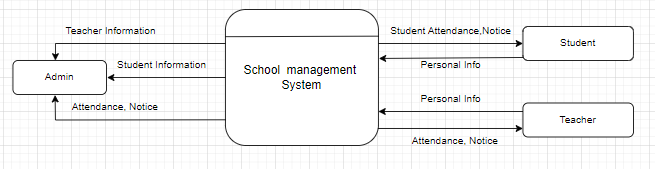
A use case diagram is a type of behavioral diagram in Unified Modeling Language (UML) that depicts the interactions between users and a system to achieve specific goals. It provides a high-level view of the system's functionalities from a user's perspective. The diagram consists of users represented by stick figures and use cases represented by ovals. Arrows between users and use cases show the interactions. Use case diagrams help to understand the requirements of a system, identify user needs, and serve as a blueprint for designing and communicating the system's functionality.



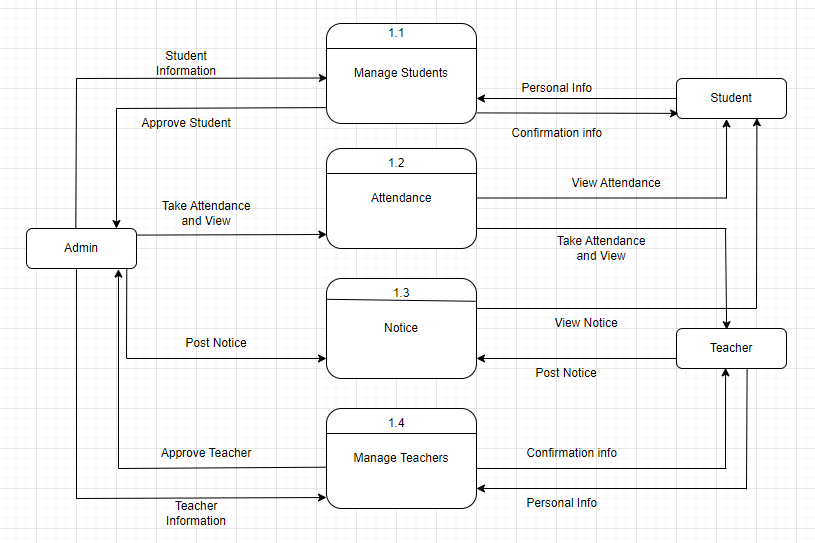
**Figure 4: Use Case Diagram**

### **5.2.2 Data Flow Diagram**

A Data Flow Diagram (DFD) is a visual representation that illustrates the flow of data within a system, depicting processes, data flows, data stores, and external entities. It abstracts the system's complexities, presenting a clear view of how data moves and is processed. DFDs are used during system analysis to understand data requirements and during system design to design the system's architecture.



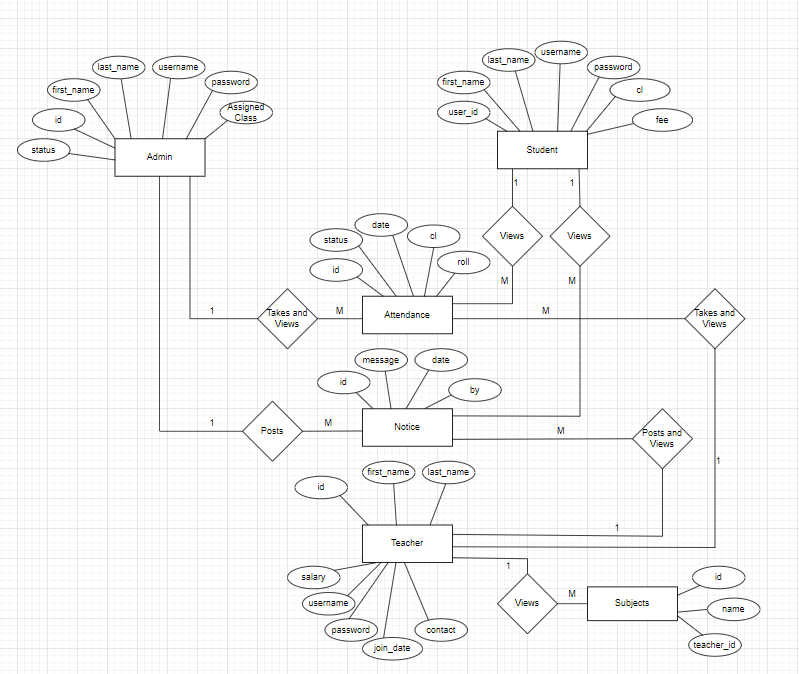
**Figure 5: Data Flow Diagram**



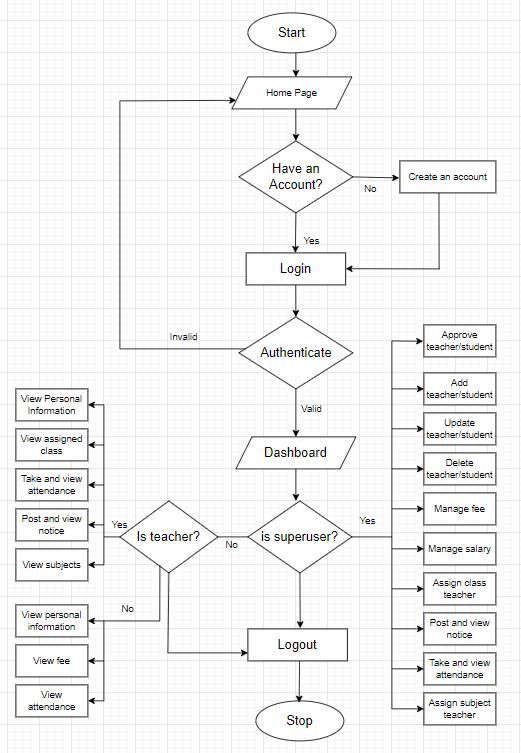
**Figure 6: Data Flow Diagram**

### **5.2.3 ER Diagram**

An Entity-Relationship (ER) diagram is a visual representation of the data model for a system or database. It uses standardized symbols to depict entities (objects or concepts), attributes (properties of entities), and relationships (associations between entities). Entities are represented by rectangles, attributes by ovals, and relationships by diamonds connecting entities. ER diagrams help in understanding the structure of data and the relationships between different entities.



**Figure 7: ER Diagram**



***Figure 8: Process Flow Diagram***

# **Chapter 6: System Development and Testing**

## **6.1 Coding tools**

### **6.1.1 Front End**

**1. HTML**

HTML (Hyper Text Markup Language) is the backbone of the internet, providing a standardized way to structure and format web content. It uses tags to define the elements of a webpage, such as headings, paragraphs, images, and links. By combining these tags, developers can create the layout and presentation of a website. As a fundamental part of web development, HTML plays a crucial role in shaping the online experience we enjoy today.

**2. CSS**

CSS (Cascading Style Sheets) is a vital technology in web development, responsible for controlling the presentation and layout of HTML elements. By defining styles such as colors, fonts, margins, and positioning, CSS enables developers to create visually appealing and consistent designs across web pages. It works by applying rules to HTML elements using selectors, determining how they should appear on the screen or other media types.

1. **Bootstrap**

Bootstrap is a popular and widely-used front-end framework that provides developers with a collection of pre-designed HTML, CSS, and JavaScript components and templates, simplifying the process of creating responsive, visually appealing, and consistent websites and web applications. With its versatility and community support, Bootstrap remains a go-to choose for web developers seeking to build modern, responsive, and visually appealing interfaces with ease.

### **6.1.2 Back End**

**Django Python**

Django is a powerful and high-level Python web framework that simplifies and expedites the process of building web applications. It follows the "Don't Repeat Yourself" (DRY) principle and emphasizes rapid development, code reusability, and modularity. Django provides a rich set of built-in features, including an ORM (Object-Relational Mapping) for database management, a template engine for creating dynamic HTML templates, authentication and authorization mechanisms, and an admin interface for easy content management.

## **6.2 Software and Hardware Requirements**

Software: Windows 11, 10, 8,7 64-bit or 32-bit, Web browser.

Hardware: Windows computer, smart phone, stable internet connection.

## **6.3 Testing**

**1. Unit Testing:** In this testing method, individual functions and methods to ensure they work correctly in isolation is performed.

**2. Integration Testing:** In this testing, the interaction between different modules of the system to check if they collaborate effectively are tested and verified that data flows smoothly between components.

**3. Functional Testing**: Here it is verified that each function and feature of the school management system meets its intended requirements and works as expected.

**4. User Interface (UI) Testing:** In this testing, it is ensured that the user interface is user-friendly and visually appealing.

**5.Performance Testing:** In this testing, the system's performance under different loads and scenarios to identify and resolve performance are evaluated.

**6. Regression Testing:** After making changes or updates, re-run tested to ensure new features don't break existing functionality.

# **Chapter 7: Lesson learnt/Outcome**

## **7.1 Expected Outcome**

In this School Management System, the admin can use the system to manage various aspects of the school, such as student admissions, staff recruitment, and overall school performance. They can access important data and reports, making it easier to monitor the school's progress and identify areas for improvement. The system enables seamless communication with students, teachers and students. Administrators can send out announcements, important notices, and updates to the entire school community, ensuring everyone stays informed and connected. By automating administrative tasks like generating, managing attendance, and handling student records, the system frees up valuable time for administrators to focus on strategic planning and other important aspects of school management. Teachers can keep track of students' attendance, and Students can view it.

## **7.2 Learnt Lessons**

While building the School Management System, I learned some important things. I learned that it's really important to understand what the school needs before I start making the website and how to make the website easy for people to use, so they don't get confused. Another important lesson is that we have to keep the student and teacher information safe and secure. I also learned how to test the website to make sure it works well and how to write down all the things I did so I can make it even better in the future.

# **Chapter 8: Conclusion and Future Recommendation**

## **8.1 Conclusion**

Building a School Management System using Django offers a user-friendly website that simplifies administrative tasks and boosts efficiency in schools. The system empowers teachers to manage student records, attendance, and academic performance effectively. Moreover, administrators benefit from streamlined school operations, including admissions, staff management, and data analysis. The system's comprehensive reports provide valuable insights, aiding in data-driven decision-making and strategic planning. Additionally, the platform's security features safeguard sensitive information, ensuring data privacy and accountability.

Overall, the School Management System using Django creates a cohesive educational environment, optimizing communication, and fostering a conducive learning experience for students while easing the workload of educators and administrators alike.

## **8.2 Future recommendation**

Here is what can be added in future on this system to increase its usability, user experience and portability of the system. There is a lot to be done hence this application can be considered as a starting point for something big to come. It will need more time and resources for all these to be done but it is still very realistic and possible to achieve.

* Artificial Intelligence (AI) Integration
* Online Payment and Fee Management
* Virtual Classroom and E-Learning
* Smart Timetabling
* Parent-Teacher Communication Enhancements
* Data Visualization and Dashboards
* Integration with Learning Management Systems (LMS)
* Continuous System Updates and User Training

# **References**

[1] Kumar, S. (2022, January 9). *School Management System in Python Django* . Retrieved from CodeAstro: https://codeastro.com/school-management-system-in-python-django/

MKS075. (2022, August 2022). Incremental Model.

Verma, Y. (2022, January 8). *Django School Management System Project*. Retrieved from Projectworlds: https://projectworlds.in/django-school-management-system-project/

Citation

[1]Verma, Yugesh. “Django School Management System Project.” *Projectworlds*, 8 Jan. 2022, projectworlds.in/django-school-management-system-project/.

[2]Adminastro. “School Management System in Python Django” *CodeAstro*, 9 Jan. 2022, codeastro.com/school-management-system-in-python-django/.

[3]pandian, Vetrivel. “Student Management System Django” *VetBosSel*, 18 June 2023, www.vetbossel.in/student-management-system-django/.

[4]Ijraset. “IJRASET Journal for Research in Applied Science and Engineering Technology.” *Student Record Management System Using Django*, 1 May 2023, www.ijraset.com/research-paper/student-record-management-system-using-django.

# **Appendix**

