**STUDENT ATTENDANCE MANAGEMENT SYSTEM**

Submitted by

### VINOTH KUMAR P

### (1P21CS029)

In partial fulfillment of the requirements for the award of the Degree of

### MASTER OF SCIENCE IN COMPUTER SCIENCE

from Bharathiar University, Coimbatore.

Under the Internal Supervision of

**Dr. Chitra. K M.Sc., M.Phil., Ph.D.**

**Associate Professor**

**School of Computer Studies**

****

**SCHOOL OF COMPUTER STUDIES (PG)**

**RATHNAVEL SUBRAMANIAM COLLEGE OF ARTS AND SCIENCE (AUTONOMOUS)**

**Sulur, Coimbatore – 641 402.**

**April 2023.**

**RATHNAVEL SUBRAMANIAM COLLEGE OF ARTS AND SCIENCE (AUTONOMOUS)**

**Sulur, Coimbatore – 641 402.**

**School of Computer Studies (PG)**

****

**Register Number: 1P21CS029**

Certified bona fide original record work done by **VINOTH KUMAR P**

**Guide HoD**

Submitted for the project Evaluation and Viva voce held on\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Internal Examiner External Examiner

# CERTIFICATE

**TRANSFORM TECH PRIVATE LIMITED**

This is to certify that the project work titled **“Student Attendance Management System”** using **React.js** and **Node.js** is a bonafide work of **Vinoth Kumar P (1P21CS029)**

He has completed the project work at Transform **Tech private limited** during the period from Jan 2023 to April 2023 towards the partial fulfillment of the requirement for the award of M.Sc. Computer Science prescribed by RVS College of Arts and Science, sulur.

**Authorized Signature**

# DECLARATION

I am **Vinoth Kumar P** hereby declare that the project entitled **“Student Attendance Management System”** submitted to the School of Computer Studies (PG), Rathnavel Subramaniam College of Arts and Science, in partial fulfillment of the requirements for the award of the Degree of Master of Science in Computer Science is a record of original project work done by me during the period Jan 2023 to April 2023 under the internal supervision of **Dr. Chitra. K, M.Sc., M.Phil., Ph.D., Associate Professor, Rathnavel Subramaniam College** **of Arts and Science (Autonomous)** from Bharathiar University, Coimbatore.

Signature of the Candidate

# ACKNOWLEDGEMENTS

I express my sincere thanks to our Managing Trustee **Dr. K. Senthil Ganesh MBA (USA)., MS (UK)., and Ph.D.,** for providing us with adequate faculty and laboratory resources for completing my project successfully.

I take this as a fine opportunity to express my sincere thanks to **Dr. T. Sivakumar M.Sc., M. Phil., Ph.D., Principal,** Rathnavel Subramaniam College of Arts and Science (Autonomous) for allowing me to undertake this project.

I express my sincere thanks to **Dr. P. Navaneetham M.Sc., M.Phil., Ph.D., Director (Administration), School of Computer Studies** for the help and advice throughout the project.

I express my sincere thanks to **Dr. S. Yamini M.Sc., (CC)., M. Phil., Ph.D., Director (Academic), School of Computer Studies** for her valuable guidance and prompt correspondence throughout the curriculum to complete the project.

I express my sincere thanks to **Dr. D. Maheswari, M.Sc., (CS)., M.Phil., Ph.D., Head and Research Coordinator, School of Computer Studies (PG)** for her support and advice throughout the project.

I express my gratitude to **Dr. Chitra. K, M.Sc., M.Phil., Ph.D., Associate Professor, School of Computer Studies (PG)** for his valuable guidance, support, encouragement, and motivation rendered by her throughout this project.

Finally, I express my sincere thanks to all other staff members and my dear friends, dear and near for helping me to complete this project.

### Vinoth Kumar P

# ABSTRACT

# “Student Attendance Management System” is a software application designed to help educational institutions manage and track student attendance efficiently and effectively. Student Attendance Management System is developed using MERN Stack. Login credentials are provided by the Admin. Here the user can log in to the application according to their user credentials. i.e., mark the student’s attendance, and a Report of the student’s attendance on a daily, monthly, and semester basis is generated.

# Student Attendance Management System deals with the maintenance of attendance details. It is for maintaining a daily basis of attendance. The staff will be provided with a separate username and password to make student attendance.

# The staff handling the particular subject is responsible to make the attendance of all students. The attendance will be calculated only if the student is present on a particular date. The student attendance report based on daily/monthly and semester wise and reports will be generated.

# Benefits of implementing a student attendance management system may include improved accuracy and timeliness of attendance records, reduced administrative burden on teachers and staff, and increased parent and student engagement.

# CONTENTS

1. Certificate
2. Declaration
3. Acknowledgements
4. Abstract

**CHAPTER 1**

1. Introduction

1.1 An Overview of Project 1

1.2 Mission of Project 1

1.3 Background Study 1

1.3.1 A Study of the existing system 2

­­

**CHAPTER 2**

2. System Analysis

2.1 A study of the proposed System 3

2.2 User requirement specification 3

2.2.1 Major Modules 4

2.2.2 Submodules 4

2.3 Software requirement specification 4

2.4 System Specification 5

2.4.1 React.js 5

2.4.2 Node.js 5

2.4.3 Hardware Configuration 5

2.4.4 Software Configuration 6

**CHAPTER 3**

3. System design and development

3.1 Fundamentals of design concepts

3.1.1 Abstraction 7

3.1.2 Refinement 7

3.1.3 Modularity 8

3.2 Design Notations 8

3.2.1 System structure chart 8

3.2.2 System flow chart 9

3.2.3 Data flow diagram / UML 10

3.3 Design Process 12

3.3.1 Software Architecture 12

3.3.2 Control Hierarchy 13

3.3.3 Structural Partitioning 13

3.3.4 Data Structure 13

3.3.5 Software Procedure 14

3.4 Table Design 14

3.5 Input design 16

3.6 Output design 16

**CHAPTER 4**

4. Testing and implementation

4.1 Testing

4.1.1 System testing

4.1.2 White box testing

4.1.3 Unit testing

4.2 System implementation

4.2.1 System Maintenance

**CHAPTER 5**

5. Conclusion

5.1 Directions for future enhancements 17

BIBLIOGRAPHY

**ANNEXURE**

ANNEXURE - A - Output design

ANNEXURE - B - Source code

**CHAPTER 1**

**INTRODUCTION**

* 1. **AN OVERVIEW OF THE PROJECT:**

**“Student Attendance Management System”** is a software developed for maintaining the attendance of the student on the daily basis the college staff, who are handling the subjects, will be responsible to mark the attendance of the students. Each staff will be given a separate username and password based on the subject they handle. An accurate report based on student attendance is generated here.

This system will also help in evaluating the attendance eligibility criteria of a student. A report of the student’s attendance on a monthly and weekly basis is generated.

* 1. **MISSION OF THE PROJECT:**

To improve the accuracy and efficiency of student attendance tracking in an educational institution. They provide teachers and administrators with real-time tendance information to help them make data-driven decisions.

To reduce the administrative burden associated with manual attendance tracking enabling educators to focus on teaching and student engagement and encourage student accountability and responsibility by making attendance records transparent and accessible.

**1.3 BACKGROUND STUDY:**

* Manual attendance tracking: describe the traditional methods used for tracking attendance in educational institutions, including their limitation and challenges.
* Benefits of attendance management systems**:** identify the benefit of using an attendance management system for educational institutions, including increased accuracy, efficiency, and accountability.
* Challenge of implementing an Attendance management system: Address the challenges that may arise during the implementation of an attendance management system, such as infrastructure requirements, training needs, and resistance to change.
* Literature review: review existing literature on attendance management systems, including research studies, case studies, and best practices.

**1.3.1 A STUDY OF THE EXISTING PROJECT:**

The existing system is to analyze the current state of attendance management systems in use, their features, and their advantages and disadvantages. Here the attendance will be carried out in the handwritten registers. It will be a tedious job to maintain the record for the user. The human effort is more here.

The retrieval of the information is not as easy as the records are maintained in the handwritten registers. This application requires correct feed-on input into the respective field. Suppose the wrong inputs are entered. the application resists working, so the user finds it difficult to use.

**CHAPTER 2**

**SYSTEM ANALYSIS**

**2.1 A STUDY OF THE PROPOSED SYSTEM:**

To overcome the drawbacks of the existing system, the proposed system has evolved. This project aims to reduce the paperwork and save time to generate accurate results from the student’s attendance. The system provides the best user interface. efficient reports can be generated by using this proposed system.

* It is trouble-free to use.
* It is a relatively fast approach to entering attendance.
* Is highly reliable, approximate result from a user.
* Best user interface
* Efficient reports

**2.2** **USER REQUIREMENT SPECIFICATION**

**2.2.1 Major Modules**

There are several types of end users. They are divided into Admin, Staffs. Each of these classes has its features.

* **ADMIN:** who can view and edit the details of any Staff and Students, and can add and edit departments, courses, classes, and reports.
* **STAFF:** who can view Student details, and add and update the attendance of students. And they can view the reports.

**2.2.2 Sub-Modules**

The features that are available to ADMIN are:

An Admin can log in to the system and perform any of the available operations:

* + - Add a student
    - View & edit the student details
    - Delete student record
    - Add the staff
    - View & edit the staff details.
    - Delete staff records.
    - Add & edit department
    - Can view the report of all the students of a particular class.

The feature that is available to STAFF are:

Staff can log in to the system and perform any of the available operations:

* + - View the personal details (Name, Dob, email, phone number)
    - Edit the details (Name, Dob, email, phone number)
    - View the department that belongs to and the classes they teach
    - Generate class report
    - Mark the attendance of a particular student
    - Update the attendance
    - See the monthly report of a class.

**2.3 SOFTWARE REQUIREMENT SPECIFICATION:**

* **OPERATING SYSTEM:** Most of the Software and the applications are designed for the Windows operating system. It is very easy to learn many things from the Windows operating system. The Windows operating system supports all the games and media components. Anything can be installed and used in the Windows operating system.
* **FRONT-END:** Design the front end of our project using React.js. It’s used for building interactive user interfaces and web applications quickly and efficiently with significantly less code than JavaScript
* **BACK-END:** Design a back end of our project using Node.js. It is widely used for the backend of applications, like using Express.js to build the backend of the classic web application. Also, It is used for server-side programming and non-blocking, Event-driven servers like typical websites and backend API services

.

* **DATABASE**: Using MongoDB can provide many benefits to a software development team. Its flexible schema makes it easy to evolve and easily store data. for programming to work with. MongoDB is also built to scale up quickly and support all the main features of modern databases such as transaction

**2.4 SYSTEM SPECIFICATION:**

**2.4.1 REACT.JS:**

React.js is a JavaScript library built and maintained by Facebook. React is an efficient, declarative, and flexible open-source JavaScript library for building simple, fast, and scalable frontends of web applications.react.js was designed to provide high performance in mind.

The core of the framework offers a virtual DOM program and server-side rendering. Which makes complex apps run extremely fast.

One of the main benefits of using React.js is the potential to reuse components. It saves time for developers as they don’t have to write various codes for the same feature. if any changes are made in any particular part it will not affect other parts of the application

**2.4.2 NODE.JS:**

Node.js is a server-side platform based on the JavaScript engine in Google Chrome. This is a cross-platform runtime environment for developing server-side and networking applications that are open source. Node.js is sometimes misunderstood by developers as a backend framework that is exclusively used to construct servers, node.js can be used on the front end as well as the back end. The event-driven, non-blocking nature of Node.js framework.

Express.js is used as a backend in certain popular stacks, such as MERN (a Node.js framework) between the front and the backend multiple components can be reused.

A significant amount of developer time can be saved by reducing context switching between several languages.

**2.4.3 HARDWARE CONFIGURATION:**

* **RAM (Minimum):** 4GB RAM
* **ROM (Minimum):** 128 GB ROM
* **PROCESSOR :** Intel i3 10325P 4.70GHz

**2.4.4 SOFTWARE CONFIGURATION:**

* **Operating System (OS):** Windows
* **Front-End Development**: React.js v18.2.0
* **Back-End Development:** Node.js v18.12.1
* **Database:** MongoDB version 3.0.4

**CHAPTER 3**

**SYSTEM DESIGN AND DEVELOPMENT**

* 1. **FUNDAMENTALS OF DESIGN CONCEPTS:**

The system should be designed with the end users. The user interface and experience should be easy to understand and navigate, the system should be designed to accommodate future growth and expansion such as increased student enrollment or additional features. the system should be designed with appropriate security measures to ensure the privacy and confidentiality of student attendance data.

The system should be designed to collect and analyze attendance data to identify patterns and trends that can help teachers and administrators make informed decisions The system should be designed to be always reliable and available to users, with minimal downtime or disruption. The system should be designed to be easily supported and maintained by college support personnel.

The system should be designed to provide adequate training and documentation for users to ensure they can effectively use the system.

**3.1.1 ABSTRACTION:**

The project should be designed with modules or components that can be easily combined or separated to achieve specific functionality. This allows for easier maintenance, testing, and scalability**.**

The project should be designed with abstract classes or interfaces that define a common behavior or contract for different components. This allows for easier code reuse and maintenance.

**3.1.2 REFINEMENT:**

Gathering feedback from end-users (teachers, students, administrators) to refine the system's features and usability. Optimizing the system to ensure fast load times and minimal downtime or disruptions.

Ensuring that the system can handle an increasing number of users and data over time. Ensuring that the system is secure and follows best practices to protect student privacy and data. Providing options for customization to meet the unique needs of different classrooms.

Providing support and maintenance to ensure the system remains reliable and functional. Providing analytics and reports to help teachers and administrators identify patterns and trends in student attendance data.

**3.1.3 MODULARITY:**

Modularity refers to the ability to break down a complex project into smaller, more manageable pieces, or modules. In a MERN stack project, modularity means organizing the code into separate, reusable components that can be easily maintained and modified. This allows developers to work on different parts of the project independently, without affecting the functionality of other modules.

A MERN stack project can also enable developers to reuse code across different parts of the application or even in other projects. By encapsulating specific functionalities into modules, they can be reused without having to rewrite the code from scratch. This can save development time and reduce the likelihood of introducing errors in the code.

**3.2 DESIGN NOTATION:**

Design notation is used when planning and should be able to communicate the purpose of a program without the need for formal code.

**3.2.1 SYSTEM STRUCTURE CHART:**

ADMIN

STAFF

Login

Staff

View Staff

Update Staff

Student

View student

Update student

Attendance

Report

Department

**3.2.2 SYSTEM FLOW DIAGRAM:**

Student Attendance

Management System

Login

Admin

Staff

Update

Master

Attendance

Report

Attendance

Student

Staff

Department

Daily entry

Monthly and semester

Add & Edit

Details

Attendance approval

Daily

Report

Monthly and semester

Report

**3.2.3 DATA FLOW DIAGRAM:**

**3.2.3.1 DFD LEVEL 0:**

Report

Database

User

**3.2.3.2 DFD LEVEL 1:**

Check

Database

Login

User

Authentication

UID/PWD

INVALID

**3.2.3.3 DFD LEVEL 2:**

**ADMIN:**

Report

Stored data from database

Complaint

Admin

Student

HOD

HOD

**STAFF:**

Stored Attendance

Input Attendance

Update Attendance

Attendance Entry

Update Attendance

Get All Student Attendance

report

Attendance

Admin

**3.2 DESIGN PROCESS:**

* The design process in a project refers to the steps that a development team takes to create a system that meets the needs of its users.
* Requirement gathering: In this phase, the development team works with stakeholders to identify the needs and goals of the project. This includes understanding the requirements of the system, identifying potential users, and defining the scope of the project.
* Prototyping: In this phase, the development team creates a rough prototype of the system to show stakeholders how it will work. This allows stakeholders to visualize the system and provide feedback on its design and functionality.
* Design: In this phase, the development team creates the detailed design of the system. This includes defining the data model, designing the user interface, and planning the architecture of the system.
* Implementation: In this phase, the development team begins building the system. This includes writing code for the frontend and backend components of the system, configuring the database, and setting up the development environment.
* Testing: In this phase, the development team performs various tests to ensure that the system works correctly. This includes unit testing, integration testing, and acceptance testing.
* Deployment: In this phase, the development team deploys the system to a production environment. This includes configuring servers, setting up security measures, and ensuring that the system is stable and secure.
* Maintenance: In this phase, the development team provides ongoing maintenance and support for the system. This includes monitoring performance, fixing bugs, and updating the system as necessary

**3.3.1 SOFTWARE ARCHITECTURE:**

The software architecture in a MERN stack project typically follows the Model-View-Controller (MVC) pattern. The model represents the data and business logic of the system, the view represents the user interface, and the controller handles the interactions between the model and the view.

In a project, the backend acts as the controller, while the frontend acts as the view, and the database acts as the model.

**3.3.2 CONTROL HIERARCHY:**

The control hierarchy typically starts with the UI layer, where the user interacts with the system. The UI layer contains various components that can communicate with each other to create a seamless user experience.

For example, a login form component might communicate with a dashboard component to display data once the user has logged in.

The UI layer communicates with the backend layer, which consists of the API and database. The API layer defines the endpoints that the front end can use to perform CRUD operations on the database. The database stores the data for the system and is responsible for managing the data’s consistency and integrity.

The backend layer communicates with external services, such as email services or payment gateways. For example, if the system sends email notifications to users, it would communicate with an email service to send those notifications

**3.3.3 STRUCTURAL PARTITIONING:**

Structural partitioning is a design technique that involves breaking down a complex MERN stack project into smaller, more manageable pieces or modules. This can be achieved by identifying different components of the system and separating them based on their functionality.

Each layer in a MERN stack project can be further partitioned into smaller modules or components. For example, the frontend layer may include separate components for the login form, student list, and attendance form, while the backend layer may include separate modules for authentication, data manipulation, and report generation.

**3.3.4 DATA STRUCTURE:**

The data structure in the project refers to how the data is organized, stored, and manipulated within the system. They provide a flexible and scalable architecture for building web applications, and the data structure plays a crucial role in ensuring the system works efficiently and effectively.

The data structure within the database can be organized in different ways depending on the needs of the application. For example, In my project student attendance management system, the data structure may include collections for students, teachers, classes, and attendance records. Each of these collections may have different fields and attributes, such as student ID, name, class, and attendance status.

**3.3.5 SOFTWARE PROCEDURE:**

The software development procedure in a project refers to the overall process of developing software using the MERN stack, the software development procedure in the above project also involves using agile methodologies to manage the development process.

This includes breaking down the project into smaller tasks, working in sprints, and conducting regular reviews and retrospectives to improve the development process.

Overall, the software development procedure in the project aims to ensure that the software is well-designed, meets user requirements, and is reliable and maintainable over time.

* 1. **TABLE DESIGN:**

**LOGIN TABLE:**

The login records are created with the username and password. There are two types of users. they are Admin and Staff.

|  |  |  |
| --- | --- | --- |
| FIELDS | DATATYPES | DESCRIPTION |
| Table Name | String | Stored the number of tables from the login |

**ADMIN TABLE:**

The admin records are created based on username and password. The admin table is the primary of the System.

|  |  |  |
| --- | --- | --- |
| FIELDS | DATATYPE | DESCRIPTION |
| Username | String | Valid Admin Username |
| Password | String of Characters | Valid Admin Password |

**STAFF TABLE:**

The Staff records are created based on valid email, and password where it also needs the Staff name, email, DOB, Qualification, Department, Phone number, and Password. Type of the user to which they belong and can be created only by admin.

|  |  |  |
| --- | --- | --- |
| FIELDS | DATATYPE | DESCRIPTION |
| Name | String | Staff Name |
| Email | String | Staff Email Id |
| DOB | Integer | Date of birth |
| Qualification | String | Staff qualification |
| Department | String | Particular department |
| Phone Number | Integer | Staff phone number |
| Password | String of Characters | Valid Staff password |

**STUDENT TABLE:**

To create a student table where it also needs a name, register number, dob, department, phone number, batch section, and password. Type of the user to which they belong and can be created only by admin.

|  |  |  |
| --- | --- | --- |
| FIELD | DATATYPE | DESCRIPTION |
| Name | String | Student Name |
| Register ID | String of characters | Student register number |
| DOB | Integer | Date of birth |
| Department | String | Student Department |
| Phone Number | Integer | Student Phone Number |
| Batch | Integer | Student Batch |
| Section | String | Student Section |
| Password | String of characters | Valid Student Password |

**DEPARTMENT TABLE:**

To create a department table where it also needs department name and department short name.

|  |  |  |
| --- | --- | --- |
| FIELDS | DATATYPE | DESCRIPTION |
| Department | String | Department name |

**ATTENDANCE TABLE:**

To create an attendance table for the department where it also needs Department, Section, Batch, and Date.

|  |  |  |
| --- | --- | --- |
| FIELDS | DATATYPE | DESCRIPTION |
| Date | Date | Day-by-day attendance |
| Batch | Integer | Student batch |
| Section | String | Student batch |
| Department | String | Department specialization |

**3.5 INPUT DESIGN:**

**LOGIN PAGE:**

Users must enter their valid credentials in the login form to access the modules in case of invalid credentials, you will see an alert message.

**ADMIN PAGE:**

Through this page, the admin can view the Staff or students that are created. The admin will create users by selecting their corresponding Department and selecting the Role for the Staff or Student with their name, email, Department, and phone number.

**STAFF PAGE**

Through the page, Staff can mark the attendance of the student for a particular date. The staff will be entering the Department, Section, Batch, and Date.

**ATTENDANCE PAGE:**

Through this page, Staff can Mark the Attendance they selecting the corresponding Department, Batch, Date, and Section and Mark the Attendance to the Student for a Particular Date.

**DEPARTMENT PAGE:**

Through this page, the admin can view the departments that are created. The Admin will create departments by entering their Department Name and Department Short Name

**3.6 OUTPUT DESIGN:**

Output design of this application **“Student Attendance Management System”** generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application

**CHAPTER 5**

**CONCLUSION**

The “**Student Attendance Management System”** is an essential tool for educational institutions to track and manage student attendance. It helps to ensure that students attend classes regularly, meet academic requirements, and make progress toward graduation. Automating attendance tracking and reporting reduces administrative burden and improves efficiency.

The MERN stack provides a flexible and scalable platform for building web-based applications, including student attendance management systems. The modular design approach and the use of industry-standard tools and technologies enable developers to create robust and maintainable applications.

**5.1 DIRECTIONS FOR FUTURE ENHANCEMENTS**

The project has a very vast scope in the future. The project can be implemented on the Internet in the future. The project can be updated in near future as and when the requirement for the same arises, as it is very flexible in terms of expansion.

The following is the future scope of the project.

* Discontinue students to eliminate potential attendance.
* Bar code Reader based attendance system.
* Individual Attendance system with a photo using Student login.

**REFERENCES:**