
Software Requirements Specification

for

STUDENT TRACKER NITT(STN)

Version 0.1 approved

Prepared by Team XYZ

AMEYA K, ADITYA A, TANUSH S & GAUTAM C

15th March, 2024

Table of Contents

1. Introduction	1
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions.....	1
1.4 Product Scope	2
1.5 References	2
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics.....	3
2.4 Operating Environment.....	4
2.5 Design and Implementation Constraints.....	4
2.6 User Documentation	4
2.7 Assumptions and Dependencies	4
3. System Features	5
3.1 User Registration.....	5
3.2 User Authentication	5
3.3 Attendance Marking.....	5
3.4 Attendance Record Management.....	5
3.5 Report Generation	5
3.6 User Account Management.....	5
3.7 Notifications.....	5
3.8 User Account Management.....	5
3.9 Notifications.....	6
3.10 Security and Privacy.....	6
3.11 Use-Case Diagram.....	6
3.12 Activity Diagram.....	7
4. External Interfaces and Requirements	10
4.1 User Interface	10
4.2 Software Interfaces	10
4.3 Communications Interfaces	10
5. System Requirements	11
5.1 Functional Requirements	11
6. Other Nonfunctional Requirements	12
6.1 Performance Requirement	12
6.2 Security Requirement.....	12
6.3 Software Quality Attributes.....	12
6.3.1 Usability: 12	
6.3.2 Robustness: 12	

1. Introduction

1.1 Purpose

Outlining the necessary functional and non-functional requirements for the creation of a Student Tracker NITT(STN) is the purpose of the Software Requirements Specification (SRS). The purpose of the Student Tracker NITT (STN) is to effectively monitor and handle employee or student attendance and exam result data in a workplace or educational environment, along with the registration of courses. The breadth of the STN is described in this paper, including all of its modules and features, including front-end user interface elements and back-end database systems. Moreover, it finds any external interfaces or dependencies that are necessary for the system to function flawlessly. All parties participating in the STN development lifecycle have access to the SRS as a complete point of reference, which helps to ensure that expectations about the features and capabilities of the product are communicated and aligned.

1.2 Document Conventions

- **Priority Levels:** Each requirement statement is assigned a priority level (High, Medium, or Low) to denote its importance in the overall functionality of the Student Tracker NITT (STN).
- **Formatting:** Requirement statements and sections adhere to a consistent font style and size, with key terms highlighted for emphasis using bold or italic formatting.
- **Section Headings:** Clear and distinguishable section headings are used, presented in a larger font size or bolded format to facilitate navigation within the document.
- **Language Conventions:** The document employs clear and concise language throughout, with technical terms defined upon first use to ensure unambiguous understanding by stakeholders. These conventions enhance the accessibility and usability of the SRS, aiding stakeholders in reviewing, comprehending, and contributing to the development process of the STN.

1.3 Intended Audience and Reading Suggestions

Intended audience:

- Project Stakeholders: Insight into project scope, timelines, and resource allocations.
- Users: Include Students Teachers and Admins.
- Testers: design of test cases and validation of system functionality.

Reading Suggestions:

- Project Managers: Prioritise sections outlining project scope and timelines.
- Testers: Review testing procedures and requirements validation sections.
- Users: Explore system features and user interface descriptions.

1.4 Product Scope

The Student Tracker NITT (STN) is a software solution designed to streamline the tracking and management of attendance records for students or employees within educational institutions or organizational settings and to serve as a non-pen-and-paper store for class exam results and other statistics. Its primary purpose is to digitalize the process of recording attendance and exam scores, thereby improving efficiency, accuracy, and accountability. The STN aims to benefit stakeholders by providing real-time access to attendance data, facilitating timely decision-making, and enhancing overall organizational productivity. By automating attendance and exam result management tasks, the STN aligns with corporate goals and business strategies focused on optimizing operational processes, fostering data-driven decision-making, and improving organizational effectiveness.

1.5 References

The following references have been used in the preparation of this Software Requirements Specification (SRS) document:

- IEEE Standard for Software Requirements Specifications (IEEE Standard 830-1998)

2. Overall Description

2.1 Product Perspective

The Student Tracker NITT (STN) described in this SRS is a new, self-contained product aimed at automating attendance tracking and management processes. It is not part of an existing product family or a replacement for any existing systems. The STN operates independently within an educational institution or organizational setting, facilitating attendance recording and analysis. It does not integrate with larger systems but may have interfaces with external databases or user authentication services.

2.2 Product Functions

- **User Authentication:** Allow users to log in securely using their credentials to access the system.
- **Mark Attendance:** Enable users to mark attendance for students or employees, recording date and time stamps.
- **View Attendance Records:** Provide users with the ability to view and analyze attendance records for individuals or groups.
- **Mark/View Exam Results:** Provides users (teachers or instructors) with the ability to mark and view exam marks/results for students/employees.
- **Manage User Accounts:** Allow administrators to create, modify, or deactivate user accounts with appropriate permissions.

- **Set Attendance Policies:** Enable administrators to define and configure attendance policies, such as minimum attendance requirements or late arrival thresholds.
- **Customize User Interface:** Provide customization options for the user interface to tailor the system to the needs of different users or organizations.
- **Integration with External Systems:** Support integration with external systems, such as Student Information Systems (SIS) or Human Resource Management Systems (HRMS), to synchronize data and streamline processes.
- **Notifications:** Send notifications to users for upcoming events, such as classes or meetings, to remind them to mark attendance.
- **Data Security:** Implement measures to ensure the security and integrity of attendance data, including encryption and access controls.
- **System Administration:** Provide tools and functionalities for system administrators to monitor and manage the overall system, including backups, updates, and performance optimization.

These functions collectively form the core capabilities of the Student Tracker NITT (STN), enabling efficient and effective management of attendance records within educational institutions or organizational settings. Details regarding the implementation and requirements for each function will be provided in Section 3 of this document.

2.3 User Classes and Characteristics

The platform caters to three primary user classes:

- **Teacher:**
 - *Frequency of Use:* Teachers are expected to use the system regularly to mark attendance for their classes.
 - *Security/Privilege Levels:* Teachers typically have permission to mark attendance, update test scores, view attendance records for their classes, and generate reports.
 - *Experience:* Teachers are experienced professionals with knowledge of classroom management and attendance tracking.
- **Student:**
 - *Frequency of Use:* Students may access the system as and when to view their attendance records.
 - *Security/Privilege Levels:* Students have limited privileges, usually restricted to viewing their own attendance records and test scores.
 - *Experience:* Students are familiar with using technology for educational purposes and expect user-friendly interfaces for accessing attendance information.
- **Admin:**
 - *Monitoring:* Only Admin can make classrooms and may access the system as and when to view and edit any attendance records.
 - *Experience:* Admins prefer easy-to-use websites and a good functioning and interactive user interface.

2.4 Operating Environment

The Student Tracker NITT (STN) will operate in a standard computing environment, compatible with common hardware platforms such as desktop computers, laptops, tablets, and smartphones. It will support multiple operating systems, including Windows, macOS, and Linux, as well as mobile operating systems like iOS and Android. The STN should seamlessly coexist with web browsers such as Chrome, Firefox, Safari, and Edge.

2.5 Design and Implementation Constraints

- **Corporate Policies:** The system must adhere to corporate policies regarding data privacy, security, and usage.
- **Hardware Limitations:** The software should be designed to operate efficiently within specified timing and memory constraints on various hardware platforms.
- **Interface Compatibility:** Interfaces with external applications or systems must be compatible and adhere to specified protocols.
- **Security Considerations:** The system must incorporate robust security measures to protect sensitive data and prevent unauthorized access or breaches.

2.6 User Documentation

The user documentation for the Student Tracker NITT (STN) will include:

- **User Manuals:** Detailed guides on how to use various features and functionalities of the system.
- **Online Help:** Interactive help resources are accessible within the system interface.
- **Tutorials:** step-by-step tutorials to assist users in getting started with the STN.

Documentation will be delivered in digital formats, adhering to standard documentation delivery standards and formats.

2.7 Assumptions and Dependencies

Assumptions:

- *Availability of Necessary Resources:* It is assumed that the necessary resources, including hardware, software, and personnel, will be available throughout the project's lifecycle.
- *Compliance with Regulations:* Assumption that the system will comply with relevant regulations and standards related to data privacy, security, and accessibility.
- *Stability of Third-Party Components:* Assumption that third-party components or libraries used in the development process will remain stable and supported.

Dependencies:

- *Integration with External Systems:* The project depends on successful integration with external systems, such as Student Information Systems (SIS) or Human Resource Management Systems (HRMS), to synchronize data and facilitate seamless operations.

- *Availability of APIs*: The project relies on the availability and functionality of APIs (Application Programming Interfaces) provided by external systems for data exchange and interoperability.

3. System Features

3.1. User Registration:

- Description: Allows administrators to create new student and teacher accounts by providing necessary information such as a unique institution ID and personal details.
- Functionality: The system verifies user-provided information, generates unique user identifiers, and stores user accounts securely in the database.

3.2. User Authentication:

- Description: Authenticates users' identities to grant access to the system.
- Functionality: Users provide their credentials (password) for authentication. The system verifies the credentials against stored records and grants access upon successful verification.

3.3. Attendance Marking:

- Description: It enables users (e.g., teachers or administrators) to mark attendance for students or employees.
- Functionality: Users select the class or group for which attendance is to be marked, specify the date, and mark attendance individually or in bulk for each member of the class or group.

3.4. Attendance Record Management:

- Description: Allows users to view, edit, and manage attendance records.
- Functionality: Users can search for attendance records based on various parameters (e.g., date range, student/employee name), view detailed attendance reports, and make edits or corrections as necessary.

3.5. Test Scores Marking:

- Description: It enables users (e.g., teachers or administrators) to mark exam results/scores for students or employees.
- Functionality: Users select the class or group for which the score is to be marked, specify the exam, and mark scores individually or in bulk for each member of the class or group.

3.6. Test Scores Management:

- Description: It enables users to view exam results/scores.
- Functionality: Students/tutors can view test scores and other statistics such as class best, and average scores.

3.7. Report Generation:

- Description: Facilitates the generation of attendance reports for analysis and reporting purposes.
- Functionality: Users can generate attendance reports based on specified parameters such as date range, class or group, and attendance status (e.g., present, absent, late). Reports can be customized and exported in various formats (e.g., PDF, CSV).

3.8. User Account Management:

- Description: Allows administrators to manage user accounts, permissions, and roles.
- Functionality: Administrators can create new user accounts, modify existing accounts, assign or revoke permissions, and define user roles (e.g., teacher, student, administrator).

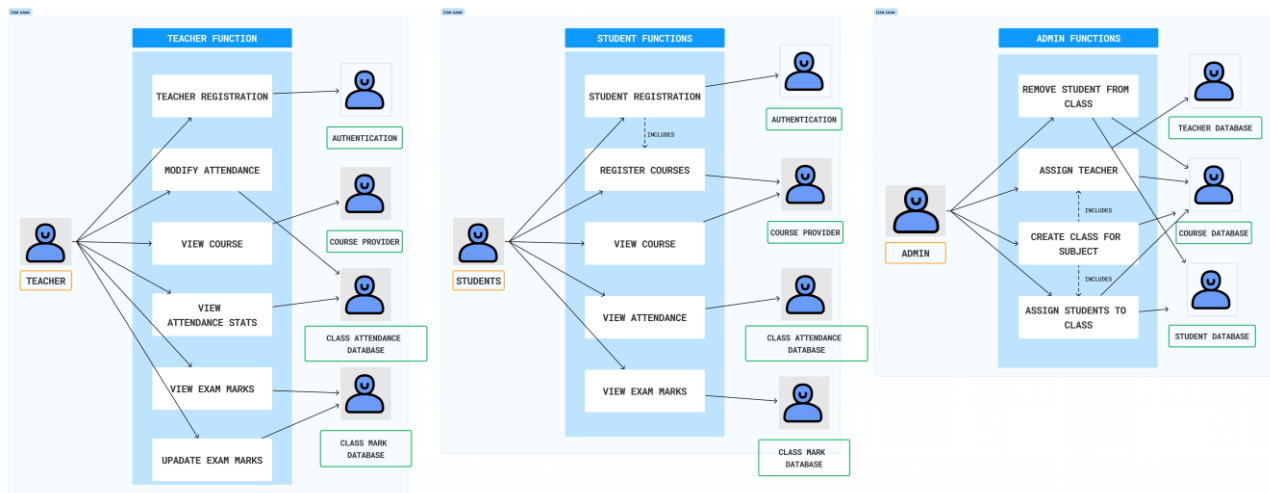
3.9. Notifications:

- Description: Sends notifications to users for events such as upcoming classes, meetings, or reminders to mark attendance.
- Functionality: The system generates and sends notifications via email or in-app notifications based on predefined triggers or user-defined preferences.

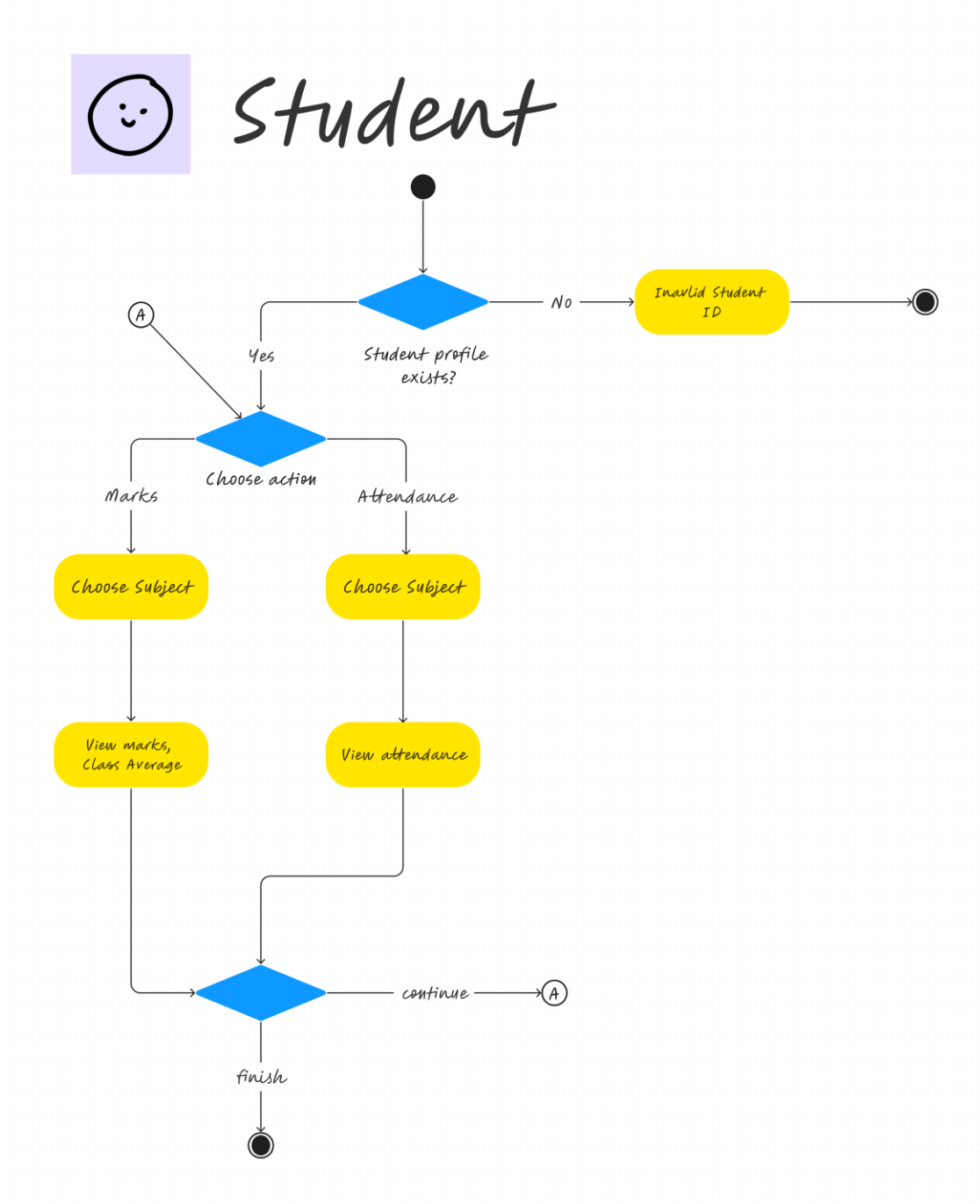
3.10. Security and Privacy:

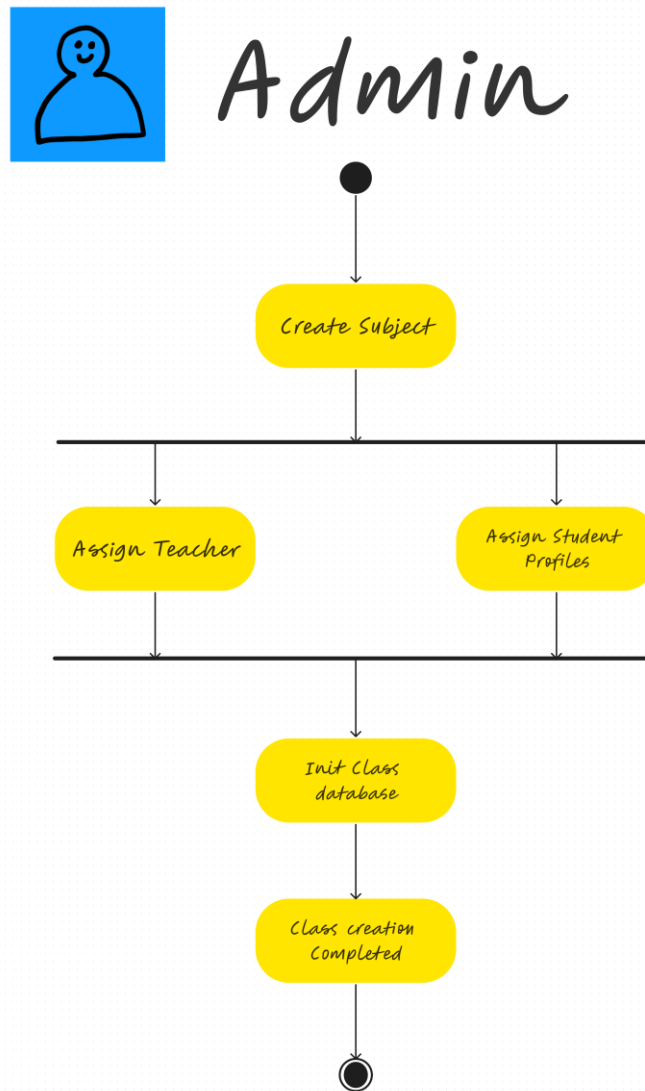
- Description: Ensures the security and privacy of attendance data and user information.
- Functionality: The system implements encryption for data transmission and storage, enforces access controls and user permissions, conducts regular security audits, and complies with data privacy regulations (e.g., GDPR).

3.11. Use-Case Diagram



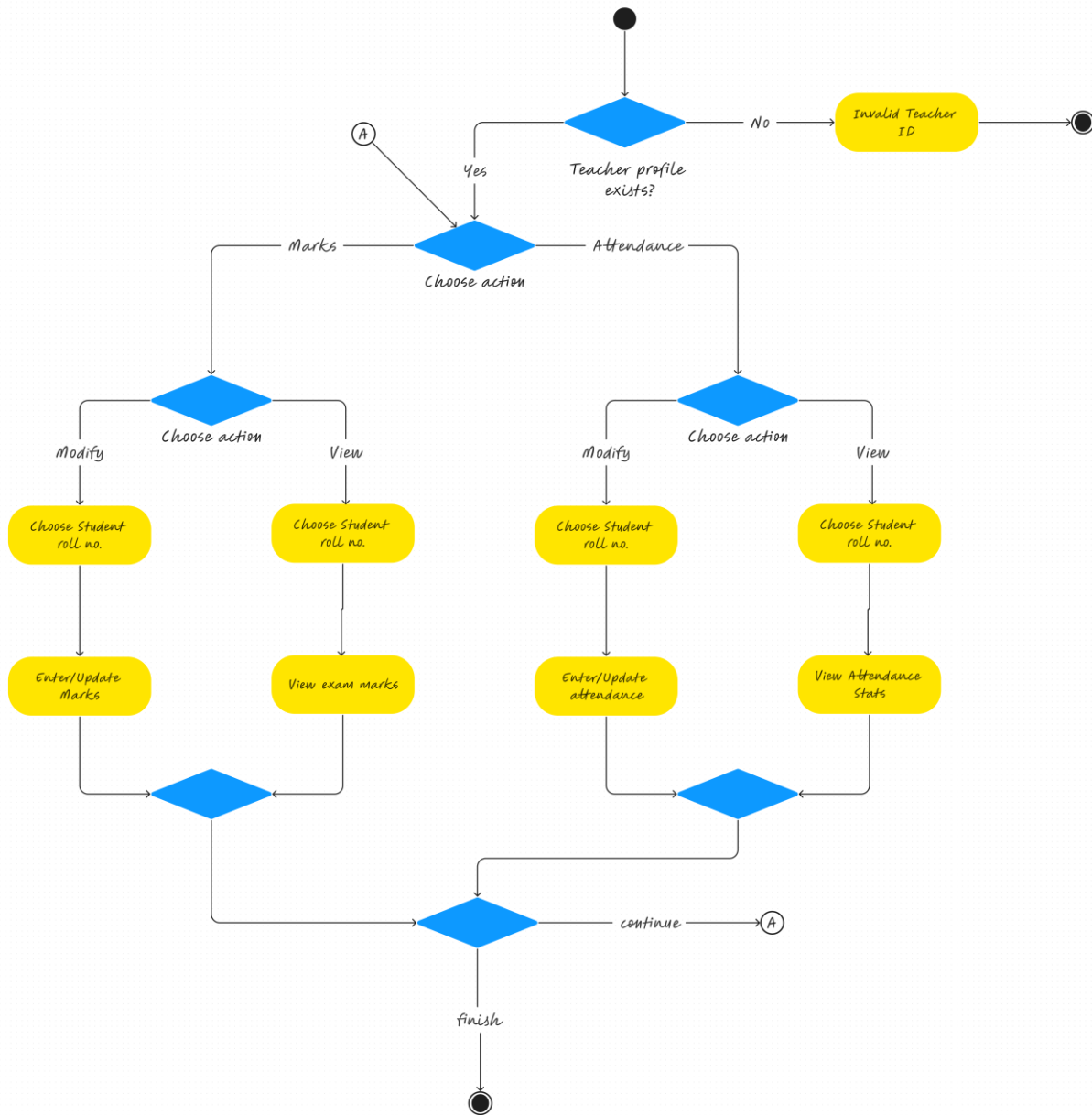
3.12. Activity Diagrams







Teacher



4. External Interfaces and Requirements

4.1 User Interface

The user interfaces of the course management software are designed to be user-friendly and accessible across devices. The following user interfaces are included:

- **Login Interface:** Allow users to securely log in to the system.
- **Course Registration Interface:** Allow students to register/deregister appropriate courses.
- **Attendance Marking Interface:** Provide a user-friendly interface for the teachers/tutors for marking attendance.
- **Attendance Record Viewing Interface:** Enable users to view and analyze monthly attendance records in a detailed manner.
- **Marks Record Viewing Interface:** Enable users to view and analyze exam results statistics.
- **Report Generation Interface:** Facilitate the generation of attendance reports based on specified parameters.
- **User Account Management Interface:** Allow administrators to manage user accounts and permissions.

4.2 Software Interfaces

The software's architecture revolves around a combination of web, application, and database servers, along with custom-built services for authentication, attendance tracking, and course registration. The data flow involves user interactions triggering requests to the server, which processes the requests, interacts with the database, and sends back responses to the client interface. Additionally, API documentation ensures clear communication protocols between the various software components.

4.3 Communications Interfaces

The Student Tracker NITT (STN) requires various communication functions, including:

- **Email:** The system may send email notifications to users for events such as upcoming classes or meetings, as well as reminders to mark attendance. Email communication may utilize SMTP (Simple Mail Transfer Protocol) for message delivery.
- **Electronic Forms:** Users interact with electronic forms within the STN interface for tasks such as marking attendance, generating reports, or managing user accounts. Message formatting for electronic forms adheres to standard web form protocols.
- **Data Transfer Rates:** Data transfer rates depend on factors such as network bandwidth and server performance. The STN aims to optimize data transfer rates to ensure efficient communication between client and server components.

- **Synchronisation Mechanisms:** The STN may implement synchronization mechanisms to ensure data consistency across distributed components or devices. Techniques such as database replication or synchronization protocols may be used to synchronize data between servers or client devices.

5. System Requirements

In organizing the system requirements, we will categorize them based on the system features outlined previously. Each system requirement will be aligned with the corresponding feature it supports. This approach ensures clarity and coherence in understanding the functional needs of the product. By structuring the requirements according to system features, we can effectively prioritize and address each feature's specific functionalities and capabilities. This organization also facilitates traceability, allowing stakeholders to easily identify the requirements associated with each feature and track their implementation throughout the development process. Additionally, it enables efficient communication and collaboration among development teams, stakeholders, and users by providing a clear framework for discussing and validating the system's functional requirements.

5.1 Functional Requirements

Functional requirements define the specific capabilities and functionalities that the system must possess to enable users to carry out the services provided. Each requirement is uniquely identified with a sequence number or tag to facilitate organization and reference. These requirements detail the actions, tasks, or behaviors that the system should support, encompassing features such as user authentication, attendance marking, report generation, and user account management. By clearly delineating these functional requirements, the development team can ensure that the system delivers the necessary features and capabilities to meet the users' needs effectively. Additionally, these requirements serve as the basis for system design, development, testing, and validation processes throughout the software development lifecycle.

6. Other Nonfunctional Requirements

6.1 Performance Requirements

In addition to the functional requirements, the Student Tracker NITT (STN) must meet certain nonfunctional requirements to ensure optimal performance. Performance requirements dictate that the system must respond to user interactions within 1 second for 90% of requests, ensuring prompt user feedback. Scalability is crucial, with the STN expected to handle a concurrent user load of at least 1000 users without significant performance degradation. Resource utilization should be optimized to ensure efficient CPU, memory, and network bandwidth usage.

6.2 Security Requirements

Security is paramount for the Student Tracker NITT (STN) to safeguard sensitive attendance data. Security requirements include robust user authentication mechanisms, such as username and password authentication, to prevent unauthorized access. Access controls must be implemented to ensure that users can only view or modify data relevant to their roles. Encryption techniques should be employed to secure data transmission over networks and protect data at rest. Additionally, the STN should have mechanisms in place for auditing and logging user activities to detect and respond to any security breaches promptly. Regular security audits and updates are essential to maintaining the integrity and confidentiality of attendance data.

6.3 Software Quality Attributes

6.3.1 Usability

Usability is crucial for the Student Tracker NITT (STN) to ensure ease of use and a positive user experience. The system should feature an intuitive user interface with clear navigation and a minimal learning curve. User interactions, such as marking attendance or generating reports, should be straightforward and require minimal effort. Additionally, the STN should provide adequate user feedback, such as confirmation messages or error alerts, to guide users through tasks effectively. Usability testing and user feedback should inform iterative improvements to enhance the overall usability of the system.

6.3.2 Robustness

Robustness is essential for the STN to maintain functionality and performance under various conditions and stressors. The system should be resilient to unexpected inputs, errors, or adverse conditions, ensuring that it continues to operate reliably. Error handling mechanisms should be in place to gracefully handle exceptions and prevent system crashes or data corruption. The STN should undergo rigorous testing, including stress testing and fault tolerance testing, to validate its robustness and identify and address any vulnerabilities or weaknesses. Additionally, regular monitoring and maintenance are necessary to ensure the ongoing robustness and stability of the system.