Namal University Transport Management System (NUTMS)

Software Requirements Specification

1	Altaf Hussain	NUM-BSCS-2024-11
2	Rabia Ashraf	NUM-BSCS-2024-65
3	Najia Nayab	NUM-BSCS-2024-62

Department of Computer Science
Namal University Mianwali

28th December, 2025

Table of Contents

1 Introduction

1.1 Purpose

1.1.1 Overview of the document

This Software Requirements Specification (SRS) document provides a complete and comprehensive description of the requirements for the Namal University Transport Management System (NUTMS).The primary goal is to guide the development process and ensure alignment with the expectations of all stakeholder.

1.1.2 Purpose

The purpose of this document is to:

- Define the functional and non-functional requirements of NUTMS in a clear, precise, and unambiguous manner
- Serve as a reference for the development team to ensure the successful implementation of features.
- Establish a mutual understanding between stakeholders (students, faculty, drivers, administrators) and the development team regarding system functionalities and its constraints.
- Serve as a contractual basis for system development, validation, and verification
- Provide a reference for system maintenance and future enhancements

1.1.3 Intended Audience

This SRS is intended for the following audience:

- **Software Developers :** To understand system requirements and design appropriate solutions
- **Testers :** To develop test plans, test cases, and validation criteria
- **Project Managers :** To plan resources, estimate costs, and track project progress
- **Transport Administrators :** To review and validate that the system meets operational needs
- **University Management :** To understand system scope, benefits, and investment requirements

1.2 Scope

1.2.1 System Identification

Namal University Transport Management System is a comprehensive application based platform designed to automate and modernize the university's transport operations.

1.2.2 System Features

NUTMS will provide the following major features and capabilities:

1. **GPS Tracking :** Real-time bus location tracking with live map visualization and estimated arrival times
2. **Schedule Management :** Comprehensive tools for creating, managing, and publishing bus routes and schedules
3. **Special Vehicle Booking for Faculty :** Faculty can reserve vehicles with the consent of higher authorities for their trips
4. **User Management :** Registration, authentication, and profile management for students, faculty, drivers, and administrators with role based access control
5. **Notification System :** Automated email notifications for booking confirmations, schedule changes and delays
6. **Administrative Dashboard :** Centralized interface for managing users, routes, vehicles, drivers, and generating reports
7. **Reporting and Analytics :** Generation of fuel usage reports, peak hour analysis, and transport performance

1.2.3 Applications of the Software

NUTMS addresses critical operational challenges in Namal University's current transport management system, which relies heavily on manual, paper-based processes. The system will be deployed in the following context:

Application Domain :
Beneficiaries:

- **Students :** Real-time bus tracking and timely notifications
- **Faculty :** Simplified booking process,real-time bus tracking and timely notifications
- **Drivers :** Clear route assignments and trip schedules
- **Transport Administrators :** Efficient route management, usage monitoring, and data based decision making
- **University Management :** Improved operational efficiency and enhanced student satisfaction

1.2.4 Goals and Benefits

1. **Automation :** Replace manual, paper-based processes with a fully automated digital system
2. **Efficiency:** Reduce administrative overhead and simplify transport operations

3. **User Convenience :** Provide easy to use interfaces accessible from any device with internet connectivity
4. **Administrative Supervision :** Enable comprehensive monitoring and reporting of transport operations
5. **Resource Optimization :** Support data based decisions for route planning and vehicle allocation
6. **Safety :** Enforce seat capacity limits and provide emergency notification capabilities

1.3 Definitions, Acronyms, and Abbreviations

Term/Acronym	Definition
NUTMS	Namal University Transport Management System - the software product being specified in this document
SRS	Software Requirements Specification - this document describing the requirements for system
GPS	Global Positioning System - satellite-based navigation system used for real-time bus tracking
API	Application Programming Interface - set of protocols for building and integrating application software
UI	User Interface - the means by which users interact with the system
HTTP/HTTPS	Hypertext Transfer Protocol / Secure - protocols for transmitting data over the web
WebSocket	Communication protocol providing full-duplex communication channels over a single TCP connection
SMTP	Simple Mail Transfer Protocol - standard for email transmission
SQL	Structured Query Language - standard language for managing relational databases

1.4 References

- IEEE Std 830-1984: IEEE Guide to Software Requirements Specifications
- Latex
- Project Proposal

1.5 Overview

The remaining document is organized into the following major sections to provide a complete specification of the Namal University Transport Management System:

Section 2 : General Description provides context for the requirements specified in Section 3. It describes the product perspective, product functions, user characteristics,

general constraints, and assumptions. This section helps readers understand the general factors that affect the requirements without stating specific requirements themselves.

Section 3 : Specific Requirements contains all detailed software requirements. This section provides complete specifications that developers will use to create the system. Requirements are organized into functional requirements, external interface requirements, performance requirements, design constraints, quality attributes, and other requirements. Each requirement is stated clearly, unambiguously, and in a verifiable manner.

Section 4 : Appendices includes supporting materials such as the Context Diagram, Use Case Diagram, glossary of terms, and references to meeting minutes. These appendices provide visual representations and additional information that enhance understanding of the requirements.

Section 5 : Index provides an alphabetical listing of key terms, requirements, and sections for easy reference and navigation throughout the document.

2 Genral Description

This section provides a comprehensive overview of NUTMS, its context within related systems,major functionalities, user characteristics, and the constraints under which it operates. Rather than specifying detailed requirements, this section sets the foundation for understanding those requirements by offering relevant background and descriptive elements.

2.1 Product Perspective

2.1.1 System Positioning and Independence

The Namal University Transport Management System (NUTMS) is being developed as a standalone, independent system specifically designed for Namal University's transport operations. Key positioning characteristics include:

- **New Development :** NUTMS is a new software product, not a replacement or upgrade of an existing system
- **Independant :** The system operates independently and does not require integration with existing university systems in the initial release
- **Future Integration Potential:** While independent initially, the system architecture allows for future integration with university's own software
- **Dedicated Purpose :** NUTMS focuses exclusively on transport management and does not overlap with other university administrative systems

Contextual Relationships and Interfaces NUTMS interacts with various systems and environments to provide comprehensive transport management services. The web-based application integrates with GPS tracking devices for real-time bus location monitoring and email services for automated notifications. Its interfaces allow smooth operations across :

- **External Systems :** GPS devices installed in buses for location tracking, email for notification delivery, OpenStreetMap APIs for map visualization and route display, and potential future integration with university authentication systems.
- **Internal Components :** PostgreSQL database for storing user profiles, routes, schedules, and booking records; backend API services for business logic and data processing; real-time WebSocket services for live GPS updates; and booking management system with capacity enforcement algorithms.
- **User Interfaces :** Accessible web-based dashboards for students and faculty to browse routes and book seats; administrative panels for managing transport operations, routes, and schedules; driver interfaces for viewing assignments and updating trip status; and real-time map displays for tracking bus locations.

2.1.2 Operational Environment

NUTMS will operate in the following technical and organizational environment:

Platform and Deployment Web-based application hosted on Vercel cloud platform with PostgreSQL database, accessible via HTTPS-secured connections from campus network and internet.

Client Requirements Accessible on desktop computers, laptops, tablets, and smartphones using modern web browsers.

GPS Infrastructure GPS devices installed in university buses with cellular or WiFi connectivity for data transmission via standard API protocols.

2.2 Product Functions

NUTMS is centered around the automation and management of university transport operations. This functionality is achieved through several interrelated components:

1. GPS Tracking The system integrates with GPS devices installed in university buses to display real time bus locations on interactive maps. Users can track route progress, view estimated arrival times, and monitor multiple buses simultaneously with location updates every 10 seconds.

2. Notification System The system delivers automated email notifications for booking confirmations, schedule changes, and cancellations. Users receive alerts about bus delays, route modifications, and emergency notifications. Reminder notifications are sent before scheduled trips, and users can customize their notification preferences.

3. Schedule Management Administrators can create and manage comprehensive route schedules on daily, weekly, and semester basis. The system handles bus and driver assignments, schedule modifications, holiday variations, and automatically integrates schedule changes with the booking system while notifying affected users.

4. User Management NUTMS provides comprehensive user registration and authentication system with role-based access control for Students, Faculty, Drivers, and Administrators. Users can manage their profiles, reset passwords, and maintain secure sessions throughout their interaction with the system.

5. Special vehicle Booking for faculty Faculty members can submit special trip requests with complete details including destination, date, purpose, and number of passengers. The system allowes administrators to approve or reject requests based on vehicle availability and university policies. Approved special trips are scheduled and tracked throughout their lifecycle.

6. Administrative Dashboard Administrators can monitor all bookings and route utilization in real-time through a centralized dashboard. The system provides tools for managing user accounts, driver assignments, and vehicle information. Administrators can override bookings when necessary, view system audit logs, and access comprehensive reporting tools for data-driven decision making.

7. Reporting and Analytics The system generates comprehensive usage reports on daily, weekly, monthly, and semester basis. Administrators can analyze peak hour traffic, route utilization, and booking trends. The system tracks driver performance, monitors trip completion rates, and allows creation of custom reports with various parameters. Reports can be exported in PDF with data visualization through charts and graphs.

2.3 User Characteristics

This section describes the characteristics of users who will interact with the NUTMS system, including their background, experience, and accessibility needs.

2.3.1 User Categories

NUTMS is designed to support multiple user groups with different roles and responsibilities:

- **Students:** Primary users for daily transport booking, bus tracking, and notifications.
- **Faculty :** Users for regular academic travel with access via laptops, office computers, and mobile devices.
- **Drivers:** Users who view assigned routes, passenger lists, and trip status.
- **Transport Administrators:** Users who manage routes, schedules, users, drivers, and reports.

2.3.2 Experience Levels

NUTMS is designed to accommodate users with different expertise levels:

- **Basic Users:** Students, drivers, and some faculty members with limited technical experience.
- **Intermediate Users:** Faculty and transport administrators with regular experience using web based systems.

2.3.3 Accessibility Considerations

To ensure usability for all users, NUTMS will include the following accessibility features:

- Clear and readable interface with adequate font sizes
- Full keyboard navigation support
- Compatibility with screen readers
- Responsive design for different devices
- Simple and clear language
- Helpful error messages and confirmation prompts
- Built-in help and support resources

2.4 General Constraints

2.4.1 Regulatory Policies

- **Academic Calendar:** System availability must align with university academic calendar and breaks
- **Transport Policies:** Adherence to university transport policies regarding passenger limits, route allocations, and special trip approvals
- **Safety Regulations:** Enforcement of vehicle capacity limits as per safety regulations
- **Record Retention:** Booking and trip records must be maintained for minimum of one academic year

2.4.2 Hardware Limitations

- **GPS Devices:** Dependent on availability and functionality of GPS hardware in university buses
- **Network Infrastructure :** Constrained by campus WiFi coverage and cellular network availability in the region
- **Device Compatibility :** Must function on different devices

2.4.3 Interfaces to Other Applications

- **Email System :** Integration with university SMTP server or external email service for notifications
- **GPS System:** Interface with GPS tracking devices or services used in university buses
- **Map Services :** Dependency on OpenStreetMap data availability and accuracy for Mianwali region

2.4.4 Parallel Operation

NUTMS must efficiently handle parallel operations and simultaneous user interactions. This includes:

Real-Time Interaction Management NUTMS must support real-time interactions such as GPS tracking, seat booking, and live schedule updates. The platform must be optimized to handle multiple concurrent users, especially during peak booking periods, without significant delays or performance degradation. The system must process and display location updates from multiple buses simultaneously while managing concurrent booking requests.

Data Integrity Across Sessions When multiple users interact with the platform simultaneously, NUTMS must ensure that shared data remains consistent. For instance, if multiple users attempt to book the last available seat, or multiple administrators modify routes and schedules concurrently, NUTMS must ensure that no data loss or conflicts occur.

2.4.5 Audit Functions

NUTMS must implement comprehensive audit functions to ensure transparency, accountability, and track user activities:

User Action Logging The system shall record important user actions such as seat bookings, cancellations, profile updates, and schedule changes. Each log entry shall include the user ID and the date and time of the action. The system shall also log all login attempts, including successful and failed attempts, to help monitor security and identify unauthorized access.

Audit Trail Accessibility Administrators shall have access to audit trails showing user actions and system activities, including bookings, cancellations, updates, and approvals. These records shall support dispute resolution, detect unauthorized access, and ensure accountability and data protection compliance.

2.4.6 Control Functions

- **Access Control:** Role based access control to restrict functions based on user roles
- **Capacity Enforcement:** Automatic prevention of bookings exceeding vehicle capacity
- **Session Management:** Automatic session timeout after 30 minutes of inactivity for security
- **Administrative Override:** Administrator ability to override system restrictions when necessary
- **Emergency Controls:** Ability to quickly disable booking or send emergency notifications

2.5 Higher-Order Language Requirements

The system shall be developed using the following programming languages and formats, consistent with the selected tools and technologies:

- **Frontend**: JavaScript using the React.js framework for building responsive and interactive user interfaces.
- **Backend**: JavaScript using the Node.js runtime with the Express.js framework for server-side logic and API handling.
- **Database Queries**: SQL for interacting with the PostgreSQL relational database.

- **Markup and Styling**: HTML for page structure and CSS for styling and layout design.
- **Configuration Files**: JSON format for application and environment configuration.
- **Version Control**: GitHub for source code management and collaboration.

2.5.1 Signal Handshake Protocols

- **HTTP/HTTPS:** Standard HTTP methods for client-server communication
- **WebSocket:** Two way communication protocol for real-time GPS updates using Socket.io library
- **SMTP:** Simple Mail Transfer Protocol for email notifications
- **GPS Protocol:** Compatible with standard GPS data formats (NMEA, GPX, or device-specific API)

2.5.2 Reliability Requirements

As a web-based transport management platform, NUTMS must meet high reliability standards to ensure consistent service availability:

System Availability Requirements NUTMS must remain available 99% of the time during academic semesters with minimal downtime. The system utilizes cloud infrastructure with automatic failure handling to ensure continuous service. Scheduled maintenance is performed during low-usage periods to minimize user impact.

Data Protection and Backup The system performs daily automated database backups with 30-day retention. All user profiles, bookings, routes, and schedules are regularly backed up. System recovery must be completed within 1 hour during operating hours to minimize disruption.

Graceful Degradation Critical functions like user authentication and seat booking remain operational even during component failures. If GPS tracking becomes unavailable, the system displays static schedules and last-known locations, ensuring core booking functionality continues working.

2.5.3 Criticality of the Application

- **Safety-Critical Elements :** Seat capacity enforcement prevents overcrowding and safety violations
- **Academic Impact:** Transport delays or failures can cause students to miss classes and exams
- **Reputational Impact:** System failures reflect poorly on university's technological capabilities

- **Non-Critical Periods:** System can be offline during semester breaks without major impact

2.5.4 Safety and Security Considerations

- **Data Security:** Protection of personal information (student IDs, contact details, travel patterns)
- **Authentication Security:** Secure password storage (hashing), protection against brute force attacks
- **Communication Security:** All data transmission over HTTPS with TLS encryption
- **Input Validation:** Protection against SQL injection, XSS, and other common web vulnerabilities
- **Privacy:** Users can only view their own booking history, not others'
- **Emergency Notification:** Critical system feature for safety-related communications

2.6 Assumptions and Dependencies

2.6.1 Technical Assumptions

- University has stable, high-speed internet connectivity on campus
- Modern web browsers (released within last 3 years) are available on user devices
- Cloud hosting platform (Vercel) provides reliable service with acceptable performance
- GPS devices in buses can communicate location data at regular intervals (every 10-30 seconds)
- University email system or SMTP service is available for sending notifications
- OpenStreetMap data is accurate for Mianwali region and university campus
- Database server (PostgreSQL) provides sufficient storage and processing capacity

2.6.2 Operational Dependencies

- **GPS Infrastructure:** System's real-time tracking feature depends on GPS devices being installed, powered, and operational in all buses
- **Email/SMTP Service:** Notification system depends on availability of email service provider
- **University Network:** On-campus access depends on WiFi infrastructure
- **Internet Service:** Overall system availability depends on continuous internet connectivity

- **Database Availability:** All system functions depend on database server being operational
- **Transport Schedule:** System operates based on defined transport schedules provided by administration
- **User Devices:** End-user access depends on availability of smartphones, tablets, or computers

3 Specific Requirements

3.1 Functional Requirements

3.1.1 User Registration and Authentication

FR-1.1: User Registration Description: The system shall allow students, faculty, drivers, and administrators to register by providing valid personal and institutional details.

Input: Name, university ID, email address, password, user role

Processing: System validates input data and stores user information securely in the database.

Output: Successful registration confirmation message

Dependency: None

FR-1.2: User Authentication Description: The system shall authenticate users using email and password credentials.

Input: Email, password

Processing: System verifies credentials against stored records.

Output: Access to user dashboard or error message

Dependency: FR-1.1

FR-1.3: Profile Management Description: The system shall allow users to view and update their profile information.

Input: Updated profile details

Processing: System validates and updates records.

Output: Updated profile confirmation

Dependency: FR-1.2

3.1.2 Route and Schedule Management

FR-2.1: Route Definition Description: The system shall allow administrators to define bus routes.

Input: Route name, stops, distance

Processing: Route data is stored in database.

Output: Route creation confirmation

Dependency: FR-1.2

FR-2.2: Schedule Management Description: The system shall allow administrators to create and modify bus schedules.

Input: Route ID, departure time, arrival time

Processing: Schedule is linked with route.

Output: Updated schedule information

Dependency: FR-2.1

FR-2.3: Route Assignment **Description:** The system shall assign buses and drivers to specific routes.

Input: Bus ID, driver ID, route ID

Processing: Assignment stored and activated.

Output: Assignment confirmation

Priority: Medium

Dependency: FR-2.1, FR-2.2

3.1.3 Real-Time Bus Tracking

FR-3.1: GPS Integration **Description:** The system shall integrate with GPS devices installed in buses to receive real-time location data.

Input: GPS coordinates (latitude, longitude), timestamp, bus ID

Processing: The system continuously receives and stores location updates from buses (every 10–30 seconds).

Output: Updated live location data stored in database

Dependency: Hardware availability (GPS devices in buses)

FR-3.2: Live Location Display **Description:** The system shall display the real-time location of buses on an interactive map for authorized users.

Input: Bus ID or route ID selected by user

Processing: Retrieve latest GPS coordinates and render on map using OpenStreetMap/Google Maps API.

Output: Interactive map showing current bus position, route path, and estimated progress

Dependency: FR-3.1

FR-3.3: Route Progress Tracking **Description:** The system shall calculate and display the progress of the bus along its assigned route in real time.

Input: Current GPS coordinates, route definition

Processing: Compare current location with predefined route stops and calculate percentage completed and estimated arrival times.

Output: Progress status, next stop information, and estimated time of arrival (ETA)

Dependency: FR-3.2

3.1.4 Notification System

FR-4.1: Booking Confirmation **Description:** The system shall automatically notify the user upon successful seat booking.

Input: Booking details (route, schedule, seat, user email)

Processing: Generate confirmation message and send via email.

Output: Email sent to registered email address

Dependency: FR-4.1

FR-4.2: Delay Notifications **Description:** The system shall notify affected users in case of bus delay.

Input: Delay information (bus ID, delay duration, affected schedules)

Processing: Identify booked users for the delayed trip and send alert.

Output: Email notification with delay details and updated ETA
Dependency: FR-3.1

FR-4.3: Schedule Change Alerts **Description:** The system shall inform users of any changes in routes or schedules.
Input: Updated schedule/route details
Processing: Identify affected bookings and send notification to registered users.
Output: Email with details of change and new schedule.
Dependency: FR-2.2

FR-4.4: Emergency Notifications **Description:** The system shall allow administrators to send instant emergency alerts to affected or all users.
Input: Emergency message, target users/routes (optional)
Processing: Send urgent alert via email to selected/all registered users.
Output: Immediate email delivery.
Dependency: FR-3.1 (for context), Admin authentication

3.1.5 Administrative Functions

FR-5.1: Dashboard Overview **Description:** The system shall provide administrators with a centralized dashboard showing real-time overview of transport operations.
Input: Admin login
Processing: Aggregate data on active buses, current bookings, delays, and utilization.
Output: Dashboard with summary statistics, charts, and alerts
Dependency: FR-1.2

FR-5.2: User Management **Description:** Administrators shall be able to create, update, delete, and assign roles to user accounts.
Input: User details (name, ID, email, role)
Processing: Validate and store/update user records in database.
Output: Confirmation of user action (created/updated/deleted)
Dependency: FR-1.2

FR-5.3: Driver and Vehicle Management **Description:** Administrators shall manage driver and vehicle records.
Input: Driver/vehicle details (ID, name, contact, status, capacity)
Processing: Add, update, or remove records.
Output: Updated list of drivers/vehicles
Dependency: FR-1.2

FR-5.4: Report Generation **Description:** The system shall allow administrators to generate usage and performance reports.
Input: Report type, date range, filters
Processing: Query database and generate summarized data.
Output: Report in table/chart format (exportable to PDF/CSV)
Dependency: FR-1.2

3.1.6 Special Vehicle Booking for Faculty

FR-6.1: Special Trip Request **Description:** Faculty members shall be able to submit requests for special vehicle bookings.
Input: Trip details (date, destination, purpose, number of passengers, faculty ID)
Processing: Validate request and store in pending status.
Output: Request submission confirmation
Dependency: FR-1.2

FR-6.2: Request Approval Workflow **Description:** Administrators shall review, approve, or reject special trip requests.
Input: Request ID, decision (approve/reject), comments (optional)
Processing: Update request status, schedule approved trip, notify faculty.
Output: Approval/rejection notification to faculty via email
Dependency: FR-6.1

3.2 External Interface Requirements

This section describes all external interfaces of the Namal University Transport Management System (NUTMS) in accordance with IEEE Software Requirements Specification (SRS) standards. It includes user interfaces, hardware interfaces, software interfaces, and communication interfaces.

3.2.1 User Interfaces

The Namal University Transport Management System (NUTMS) shall provide a web-based, responsive, and user-friendly interface accessible through desktops, laptops, tablets, and mobile devices. The system interface shall be:

- Consistent in layout, color scheme, and navigation
- Role-based (Student, Faculty, Driver, Administrator)
- Easy to use for non-technical users

UI-1: Homepage Interface The homepage shall provide:

- Login and registration options
- System overview
- Navigation links to main system features

UI-2: Tracking Interface The tracking interface shall:

- Display real-time bus location on a map
- Show route progress and bus status

UI-3: Administrative Dashboard The administrative dashboard shall allow administrators to:

- Manage users, drivers, and vehicles
- Configure routes and schedules
- View system reports and analytics

UI-4: Driver Interface The driver interface shall allow drivers to:

- View assigned routes and schedules
- Update trip status (start, delay, end)

3.2.2 Hardware Interfaces

This section describes the hardware components that interact with the system.

HW-1: GPS Device Interface GPS devices installed in buses shall:

- Transmit real-time location data
- Continuously update bus coordinates to the system

HW-2: Mobile Device Compatibility The system shall be accessible using:

- Smartphones
- Tablets
- Laptops and desktop computers

No special hardware configuration shall be required.

HW-3: Server Hardware The system shall be hosted on cloud-based servers capable of:

- Supporting multiple concurrent users
- Handling real-time data processing

3.2.3 Software Interfaces

This section describes software systems and services that interface with NUTMS.

SW-1: Operating System Interface The system shall operate on servers running:

- Microsoft Windows
- Linux-based operating systems

SW-2: Database Management System Interface The system shall use PostgreSQL as the relational database management system to ensure:

- Secure data storage
- Transaction reliability
- Data integrity

SW-3: Web Browser Interface The system shall be compatible with modern web browsers, including:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

SW-4: Third-Party APIs The system shall integrate with:

- Map APIs (Google Maps / OpenStreetMap) for route visualization
- Email services for notifications
- SMS gateway services for alerts and emergency messages

SW-5: Web Server Interface The backend of the system shall be implemented using:

- Node.js
- Express.js

to handle API requests, authentication, and real-time communication.

3.2.4 Communication Interfaces

This section defines communication mechanisms used by the system.

COM-1: HTTP/HTTPS Protocol The system shall use HTTP/HTTPS protocols for secure web communication. HTTPS shall be enforced for sensitive data transmission.

COM-2: WebSocket Protocol WebSocket communication shall be used to support:

- Real-time bus tracking
- Live notifications

COM-3: Email Protocol SMTP shall be used to send:

- Booking confirmations
- Schedule updates
- System notifications

COM-5: Database Communication Secure communication protocols shall be used between the application server and the PostgreSQL database.

3.3 Non-Functional Requirements

3.3.1 Performance

- System shall load all pages fast, within three seconds under the network conditions.
- Booking transactions shall be completed within two seconds.
- Real-time bus location updates shall arrive within five seconds.

3.3.2 Reliability & Availability

- System must stay up 99% of the time during each session.
- System will use PostgreSQL transactions to store and protect booking and tracking data.
- Automated backups shall run daily to prevent data loss.

3.3.3 Security

- System shall provide login functionality and permissions for students, drivers, and administrators.
- Role-Based Access Control (RBAC) shall protect access to system features.
- Data shall be encrypted during transmission using HTTPS.

3.3.4 Usability

- User interface shall be simple, intuitive, and responsive for desktop and mobile devices.
- Basic features, such as tracking, require no prior training for new users.
- Error messages shall be clear and user-friendly.

3.3.5 Scalability

- System shall handle additional users and routes without redesign.
- Architecture shall support addition of new features, apps, and payment gateways.

3.3.6 Maintainability

- System shall be built using React.js (frontend) and Node.js (backend).
- Source code shall include comments and be managed via Git version control.

3.4 Design Constraints

The design and implementation of the NUTMS system are subject to the following constraints, which define the technological, operational, and regulatory boundaries within which the system must function:

- **Web-Based Architecture:** The system shall be developed strictly as a web-based application. The frontend shall be implemented using React.js to ensure modularity, responsiveness, and maintainability, while the backend shall be developed using Node.js with the Express.js framework to support scalable server-side processing and RESTful API services.
- **Database Technology:** PostgreSQL shall be used as the primary database management system. All data related to users, bookings, routes, tracking information, and system logs shall be stored and managed using PostgreSQL to ensure data consistency, integrity, and support for ACID-compliant transactions.
- **Browser Compatibility:** The system shall be compatible with modern web browsers, including Google Chrome, Mozilla Firefox, and Microsoft Edge. The design shall avoid browser-specific dependencies to ensure consistent behavior and user experience across supported platforms.
- **Network Dependency:** Continuous internet connectivity is required for core system functionalities, including real-time bus tracking, live notifications, and online seat booking. The system shall be designed to handle temporary network disruptions gracefully where possible.
- **Security and Privacy:** The system shall comply with university IT policies, data protection standards, and applicable data privacy regulations. User authentication, authorization, and secure data transmission shall be mandatory design requirements to prevent unauthorized access and data breaches.
- **Cost and Resource:** The system shall be developed using open-source and cost-effective tools wherever possible. Deployment and hosting solutions shall prioritize free or low-cost service tiers without compromising system reliability or scalability.
- **Maintainability:** The system architecture shall support ease of maintenance and future enhancements. Source code shall follow standard coding practices, include appropriate documentation, and be managed using version control systems to facilitate collaboration and long-term maintenance.
- **Scalability:** The system design shall allow for future growth in the number of users, routes, and services without requiring significant architectural redesign.

A Meeting Minutes Reference

You can access the document from the following link:
https://docs.google.com/spreadsheets/d/180hqUDHGpHp8CMqTwAZxo9n5CBqNpvm2eWgDv0x

B GitHub Repository

The complete project repository is available at:
https://github.com/Altaf-Hussain-11/SE-Project-Transport-Management-System-.git

C Context Diagram

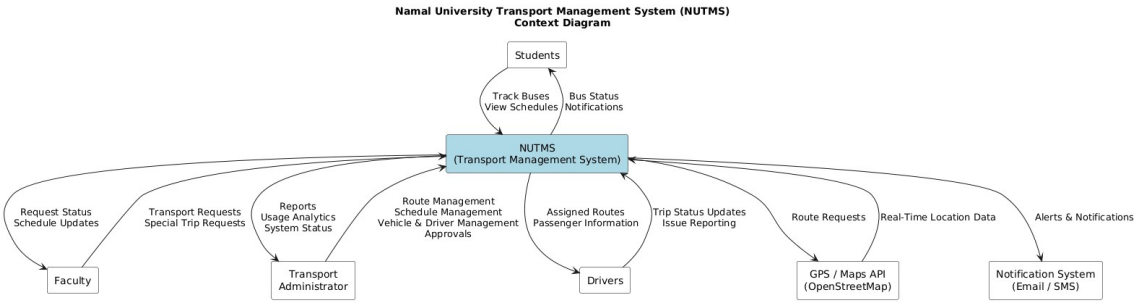

Figure 1: NUTMS Context Diagram - System interactions with external actors

D Use Case Diagram

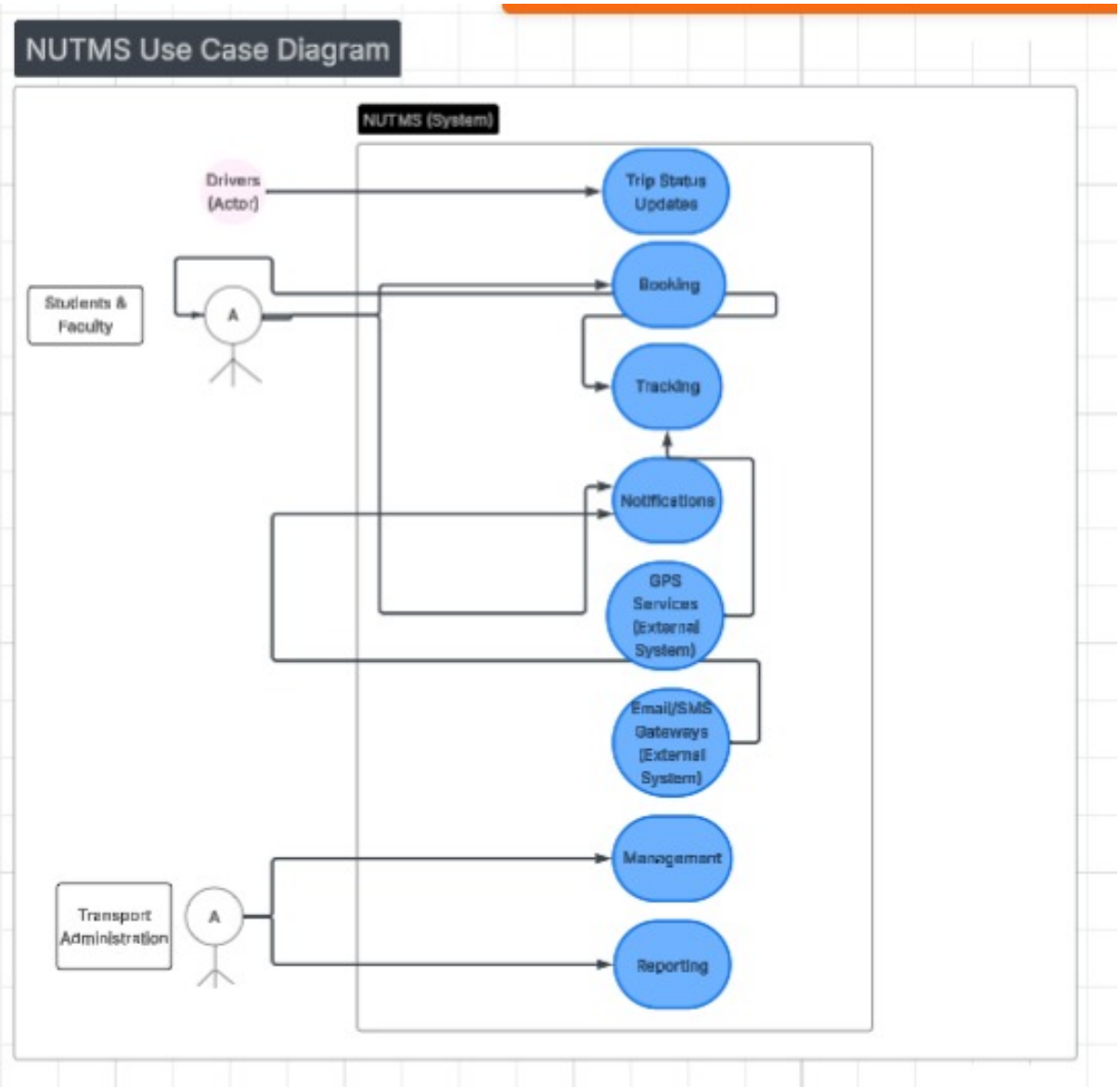

Figure 2: NUTMS Use Case Diagram - Detailed system use cases and actor interactions