**SRS - Train Tracking and Arrival Time Prediction App**

**1. Scope:**

The scope of this project is to develop a comprehensive Train Tracking and Arrival Time Prediction application that caters to the needs of various user groups, including commuters, train station personnel, and rail enthusiasts.

**2. General Description:**

**a. Target Audience:**

- Commuters and Travelers: People who rely on train transportation for their daily commute or travel.

- Train Station Personnel and Management: Station staff and managers who require tools for efficient train management.

- Rail Enthusiasts and Hobbyists: Users who have a general interest in trains and railway systems.

**b. Objectives:**

- Real-Time Train Tracking: Provide a seamless experience for users to track trains in real-time using a dynamic map interface.

- Accurate Arrival Time Predictions: Utilize historical data and real-time information to predict train arrival times at selected stations.

- Enhanced Travel Experience: Improve the overall convenience and reliability of train travel for passengers.

- Efficient Station Management: Enable train station personnel to monitor and manage train schedules effectively.

**c. Constraints:**

- Real-time Train Data: The availability of real-time train data from relevant train operators is crucial for accurate tracking and predictions.

- Data Accuracy and Reliability: Ensure that the data used for tracking and predictions is accurate and reliable.

- Privacy and Data Protection: Adhere to privacy and data protection regulations to safeguard user information.

- Accessibility: Implement accessibility features to cater to users with disabilities.

**3. Functional Requirements:**

**User Registration and Authentication:**

* Allow users to create accounts securely with email or social media authentication.
* Implement multi-factor authentication for added security.

**Train Tracking:**

* Enable users to search for trains by train number or name.
* Display real-time train locations on an interactive map.
* Allow users to set up customizable notifications for specific trains.

**Arrival Time Prediction:**

* Offer users the ability to select a train station and view predicted arrival times for upcoming trains.
* Utilize machine learning algorithms to improve the accuracy of predictions.

**Favorites and Reminders:**

* Allow users to mark favorite trains and stations for quick access.
* Implement a reminder system for train departures and arrivals, synchronized with user preferences.

**Feedback and Reporting:**

* Provide a user-friendly interface for users to submit feedback on train services.
* Allow administrators to review and respond to user feedback in a timely manner.

**Admin Panel:**

* Grant administrators the ability to manage train data, station information, and user accounts efficiently.
* Generate comprehensive reports on train performance, delays, and user engagement.

**4. Non-Functional Requirements:**

**Performance:**

* Ensure that the application loads quickly and provides real-time updates with minimal latency.
* Maintain response times within milliseconds for an optimal user experience.

**Security:**

* Encrypt and securely store user data, including personal information.
* Implement robust security measures to protect against data breaches and unauthorized access.

**Scalability:**

* Design the system to be highly scalable to accommodate a growing user base and expanding train station coverage.
* Employ load balancing and auto-scaling mechanisms to handle high concurrent user loads.

**Reliability:**

* Aim for 99.9% uptime, ensuring the application is available 24/7 with minimal downtime.
* Implement redundancy and failover mechanisms to ensure continuity of service.

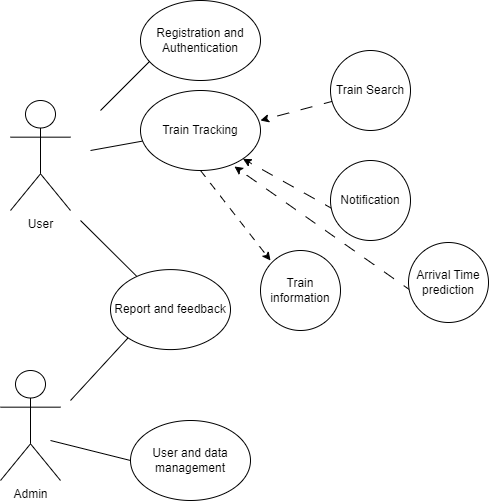
**Usability:**

* Create an intuitive and user-friendly interface that caters to users of all backgrounds and abilities.
* Support multiple languages and provide accessibility features such as screen readers and voice commands.

**Compatibility:**

* Ensure cross-device compatibility, making the application accessible on various devices, including smartphones, tablets, and desktops.
* Develop mobile app versions for both iOS and Android platforms to reach a wider audience.

**5. Use Case Models (UML Diagrams):**



**6. Appendices:**

**a. Definitions, Acronyms, Abbreviations:**

**GPS:** Global Positioning System, a satellite-based navigation system used for tracking train locations in real-time.

**API:** Application Programming Interface, a set of rules and protocols that allow different software applications to communicate with each other.

**ML:** Machine Learning, a subset of artificial intelligence that enables the application to improve arrival time predictions over time through data analysis.

**UX:** User Experience, the overall satisfaction and usability of the application from a user's perspective.

**UI:** User Interface, the graphical layout and design elements of the application that users interact with.

**MFA:** Multi-Factor Authentication, a security measure that requires users to provide multiple forms of verification to access their accounts.

**GDPR:** General Data Protection Regulation, a European Union regulation for data protection and privacy of individuals.

**Load Balancing:** The process of distributing network traffic across multiple servers to ensure optimal performance and availability.

**Redundancy:** The duplication of critical system components to provide backup in case of system failure.

**Failover:** The automatic switching to a backup system in the event of a primary system failure.

**Uptime:** The amount of time that a system or service is operational and available to users.

**Accessibility:** Ensuring that the application is usable by individuals with disabilities, including those who use screen readers or have mobility impairments.

**Cross-Device Compatibility:** The ability of the application to function consistently and effectively on different types of devices and platforms, such as smartphones, tablets, and desktop computers.

**iOS:** Apple's mobile operating system used on iPhones and iPads.

**Android:** Google's mobile operating system used on a wide range of smartphones and tablets.

**UML:** Unified Modeling Language, a standard notation for modeling software systems through diagrams and visual representations.

**b. References:**

**Data Sources:**

* [National Rail Enquiries](https://www.nationalrail.co.uk/): A source of real-time train data for the UK.

**Regulatory and Privacy Compliance:**

* [General Data Protection Regulation (GDPR)](https://eur-lex.europa.eu/eli/reg/2016/679/oj): The official EU regulation on data protection and privacy.

**Security Best Practices:**

* [OWASP (Open Web Application Security Project)](https://owasp.org/): An organization focused on improving software security.

**Load Balancing and Scalability:**

* [Amazon Elastic Load Balancing](https://aws.amazon.com/elasticloadbalancing/): A cloud-based load balancing service.
* [Kubernetes](https://kubernetes.io/): An open-source container orchestration platform for scaling applications.

**Development and Accessibility:**

* [Apple Developer Documentation](https://developer.apple.com/documentation/): Resources for iOS app development.
* [Android Developers](https://developer.android.com/): Documentation and guides for Android app development.
* [Web Accessibility Initiative (WAI)](https://www.w3.org/WAI/): Resources for web accessibility standards and guidelines.

**UML Diagrams:**

* Draw.io.