Analysis

- **Identify Stakeholders:**
  1. **Administrators:** Responsible for system configuration, user management, and overall system administration.
  2. **Users:** Individuals who will be reserving vehicles, managing bookings, and accessing vehicle-related information.
  3. **IT Staff:** Involved in system deployment, maintenance, and technical support.
  4. **Management:** Decision-makers who provide guidance and oversee the project.

- **Gather Requirements:**
  1. Conduct interviews and workshops with stakeholders to elicit detailed requirements.
  2. Document the requirements in a structured manner, including functional, UX, security, and integration aspects.

- **Define System Objectives:**
  1. Develop a secure and efficient vehicle management system that optimizes fleet utilization and improves operational processes.
  2. Enhance user experience by providing intuitive interfaces and streamlined workflows. (if I found a good template, I will use it and handle it to make it be suitable for our project)
  3. Ensure data privacy and security through proper authentication, access control, and secure communication, for this part I will implement:
     1- Authentication, 2- Permissions, 3- Secure Communication, 4- Data Encryption, 5- Input Validation

- **System Scope Definition:**

  1- Authentication, 2- Vehicle tracking, 3- Vehicle management, 4- Reservation system, 5- Maintenance and Repairs, 6- Fuel management, 7- Driver management, 6- Reporting and Analytics, 7- Notifications and Alerts, 9- Multiple Languages

- **System used Designs:**
  1. **System Architecture:** I'll use *UML use case diagrams* to represent the interactions between system actors (users, administrators) and the system. Show the main use cases, actors, and their relationships, and I'll use *UML sequence diagrams* to show the sequence of actions and system responses, including method calls, parameters, and return values.
  2. **Data Modeling:** I'll use *ERD (Entity Relationship Diagram)* to show the entities (classes), their attributes, relationships, and multiplicity.
  3. **User Interface Design:** I'll *ReactJS framework* for admin dashboard and *React Native framework* for user application.
  4. **Error Handling and Exception Management:** Design the error handling and exception management mechanisms for the system. Error logging, error messages, and recovery procedures. *This will be handled in the server side when coding the backend with NodeJS.*

- **System used Technologies:**
  1. **Express.js:** is a web application framework for Node.js. It provides a set of features and tools that simplify the process of building web applications and APIs. *I will use it to develop the backend side.*
  2. **ReactJS:** is an open-source JavaScript library for building user interfaces (UIs) in web applications. *I will use it to build the admin dashboard site for our project.*
  3. **React Native:** is a framework used to build mobile applications using JavaScript and React. It enables the creation of native mobile apps for iOS and Android using a single codebase. *I will use it to build the user application for our project.*
  4. **MongoDB:** is a popular open-source, NoSQL (non-relational) database management system. It is designed to handle large amounts of data and provide high scalability and performance for modern applications. *I will use it as the database for our project.*
  5. **Cloudinary:** is a cloud-based media management platform that provides a comprehensive set of tools and services for managing, optimizing, and delivering media assets (such as images, videos, and audio files) in web and mobile applications. *I will use it to store the media for the project.*

6. **Twilio:** is a cloud communications platform that provides developers with APIs for building and integrating voice, video, messaging, and other communication capabilities into their applications. *I will use it to send a verification token to the user.*

7. **JWT (JSON Web Token):** is an open standard (RFC 7519) that defines a compact and self-contained way of securely transmitting information between parties as a JSON object. *I'll use it to encrypt important information like passwords.*

8. **Firebase:** is a comprehensive mobile and web development platform provided by Google. It offers a suite of tools and services that help developers build and deploy applications quickly, without the need for managing infrastructure or backend services. *I'll use a Cloud Messaging feature for sending push notifications and I'll use Analytics feature to provides insights into user behavior and app usage.*

9. **Google Cloud Platform (GCP):** a wide range of cloud services and products to help businesses and developers build, deploy, and scale their applications and infrastructure. *I'll use google map tool to access the google map for vehicle tracking and I'll use translate tool to translate some text.*