In this project, students will develop a software application to address a real-world system. They will leverage object-oriented programming (OOP) concepts to model real-world entities and their interactions. Data structures will be employed to efficiently manage and organize information within the application. Unified Modeling Language (UML) diagrams will be used to visually represent the system's design and component communication.

1. **Problem Definition Document**:
    - Outline the real-world problem that the application targets.
    - Describe the functionalities that the application will provide.
    - This document serves as a blueprint for the development process, guiding the team on what needs to be implemented.
2. **UML Diagrams**:
    - **Class Diagrams**: Depict the system's classes and their relationships, including inheritance, composition, and association.
    - **Use Case Diagrams**: Illustrate interactions between actors (users, external systems) and the system's use cases.
    - **Activity Diagrams**: Detail the flow of actions within each use case, showing how the system responds to user interactions.
    - These diagrams provide a visual representation of the system's design and help ensure clarity and consistency in development.
3. **Code Implementation**:
    - Write the complete source code of the application in the C++ programming language.
    - Utilize object-oriented programming (OOP) concepts to model real-world entities as classes and define their behaviors and interactions.
    - Employ data structures to efficiently manage and organize information within the application.
    - Follow best practices for code organization, readability, and maintainability.