Case study

As an initiative towards the 'Green Campus' for minimising the carbon footprint, traffic congestion, and noise pollution by vehicles, the University's administration has decided to provide a bicycling service to the campus visitors. The visitors can rent the available bike(s) on an hourly or daily basis to enhance in-campus mobility by using the CampusBike mobile application. The By using the application, campus visitors as potential bikers can register with the CampusBike system to view available bikes, e.g., location proximity < 500 meters to reserve the bike for a specific time interval, e.g., 60 minutes. To support this scenario, the university administration needs to develop a software application CampusBike that operates in a networked environment by enabling the users to register with the system, view available bikes, reserve and return a bike and make payments for the ride. The data of the visitors, i.e., bike riders must be managed securely and preserve privacy as per the GDPR regulation.

Functional Requirements	**Non-functional Requirements**
1. User Registration: Allow campus visitors to register as bike riders. 2. Bike Availability: Provide information about the available bikes. 3. Bike Reservation: Enable users to reserve a bike for a specific time interval. 4. Bike Return: Provide a mechanism for users to return the bike. 5. Payment Processing: Support payment processing for bike rides.	1. Green Campus: Contribute to minimizing carbon footprint, traffic congestion, and noise pollution by promoting bicycle usage. 2. Networked Environment: Operate in a networked environment to facilitate user interactions and data exchange. 3. GDPR Compliance: Manage user data securely and preserve privacy according to GDPR regulations. 4. User-Friendly Interface: Have an intuitive and easy-to-use interface for users to interact with. 5. Reliability: Ensure accurate bike reservations and correct payment processing. 6. Performance: Handle a large number of users and provide timely responses for bike availability and reservations. 7. Scalability: Accommodate a growing number of campus visitors and bikes in the future. 8. Security: Implement appropriate security measures to protect user data and prevent unauthorized access. 9. Availability: Ensure the system is available and accessible to users whenever needed. 10. Maintainability: Design and implement the

	system for easy maintenance and updates.

UML diagram components and their relationships based on the given requirements

The main components of the UML diagram for the CampusBike system can include:

1. User:
   - Attributes: ID, Name, Contact Information
   - Operations: Register, Login
2. Bike:
   - Attributes: ID, Availability, Location
   - Operations: Reserve, Return
3. Payment:
   - Attributes: Amount, Transaction Details
   - Operations: Process Payment
4. CampusBike System:
   - Operations: View Available Bikes, Update Bike Availability, Manage User Accounts
5. Database:
   - Store and manage user information, bike details, and payment records.

The relationships between the components can be represented as follows:

1. User-Bike Association:
   - User can reserve a bike (1 user to many bikes)
   - Bike can be associated with the user who reserved it (1 bike to 1 user)
2. User-Payment Association:
   - User can make a payment for the bike ride (1 user to many payments)
   - Payment is associated with the user who made it (1 payment to 1 user)
3. CampusBike System-User Association:

- The CampusBike system manages user accounts (1 system to many users)

4. CampusBike System-Bike Association:

- The CampusBike system manages bike availability (1 system to many bikes)

Note that the UML diagram can have additional elements such as class relationships, inheritance, and multiplicity indicators to further illustrate the system structure and behavior.