Design Activity - Smart vehicle booking system

==========

For this activity each student will create a high-level design of a software system. The design will be presented as a single UML diagram showing the overall class structure of the system. The diagram should contain class names, responsibilities (don't worry about including every single minor behavior), and the relationships among the classes (inheritance, association, aggregation, and possibly cardinality and role names). A separate section of the document should give brief (1-2 sentence max.) descriptions of the responsibilities for each class. The total document size, including diagrams, must not exceed 2 pages.

Each individual is responsible for submitting his or her own design sketch by **11:00 pm on 14th January**. Please have a copy of the design sketch to class for your reference in the next class, i.e., on **Monday, 16th January**.

Students who make a good faith effort to develop a design can expect to receive 10 points, even if the design itself is suboptimal. The goal is to start students on the path to reflecting about design at a higher level than individual classes and methods.

==========

PROBLEM DESCRIPTION:

Book smart vehicle for in-campus transportation
We're looking forward to designing a software system to book smart bikes for in-campus transportation at IIIT-H campus. The design must accommodate users being able to book smart bikes via a mobile app and make payment. The user must be able to complete onboarding, scan QR code on the bikes parked in the parking lot to start the trip, end the trip and make payment either manually or through auto-deduct facility.
For this prototype, you need to accommodate various kinds of users which must be obvious from the design, its documentation, and your presentation.

Smart vehicle:
The smart vehicles(bike, bicycle or moped) can be used by users(staff,student,teacher) by registering on the application and making payment on the same. The vehicle can be docked in docking stations provided on campus and users will be charged according to a defined scheme.
The vehicle can be used both inside and outside campus. The payment can be made using the software app.

User Onboarding
The software product must allow the user to get onboarded. Once the user opens the app, they should be able to:

1. Create an account - *You can decide on the login mechanism to be implemented*

2. Upload id - *You can decide on what ids should be uploaded for various kinds of users.*
3. Add money to Wallet - *Decide on how the wallet functionality will work with respect to the payment mechanisms, boundations like minimum balance etc.*

Bike rates:
The payment is such that for the first 2 kilometers, a base rate of 12 rupees is taken. After that, it is at 1 rupees per 100 meters.

Charges:
The user books a vehicle using the application. Keep track of the current vehicle, money due, user details, etc. If a bike is not returned to the docking station within 8 hours and the bike is not renewed, a fine of 50 rupees is deducted every day.

Booking a smart bike
The bikes will be parked in the designated parking lots in the campus. Users can make a booking by scanning the QR code present in the parking lot. The same QR code will be used while starting and ending a trip. The QR code will contain all details to individual bikes and when the user scans it, the connection has to be established between the logged-in user and the smart bike. There should be an authentication mechanism to confirm the booking.

Starting and ending a trip
To move the smart bike from the parking lot, the trip must be started. Also, trips can only be started and ended at the designated parking lots of the campus. Trips can be started and ended by using the bike's QR code.

1. For starting the trip- Scan the QR code -> bike's details get listed in the app -> Start the trip
2. For ending the trip- Park the bike in parking lot -> End trip on app screen -> Scan QR code

You may add more details/functionalities pertaining to this. List all the assumptions you make.

Making payment
The user should be able to make payment for the rides they've taken.
Payment can be made via in-app wallet. If there is sufficient money in the wallet, the amount can either be auto deducted (if so enabled by the user before) or else, can be done manually via the app. Else, if the wallet doesn't have sufficient money, money has to be added to the wallet via other payment options before proceeding for making the payment. Money can be added by existing UPI apps.
Other options can include deducting from the salary of the user(in case of staff or professors) or adding to the fees(of students).
You may add details for adding money to the wallet as per your understanding of payment systems. Your design and/or presentation must make the added details obvious.

NOTE: Also list all assumptions made in the document.

Submission Guidelines:

1. The submission portal will remain open till **14th Jan. ,i.e., Saturday till 11PM.**
2. Make sure your **name** and **roll number** is printed on top of the submitted sheet.
3. The submission format should be **Activity1_RollNum.pdf.** Please follow the format strictly otherwise a penalty may be incurred.