Mobi-e-rides

Business Analysis and Modeling for an eBike Rental Startup

Rodrigo Garza García | IBM Business Analyst coursework | November 2026

Mobi-e-rides is a fictional startup renting e-bikes to residents and tourists. I used this case to connect requirements gathering, prioritization and three basic models. The proposed service lets customers check availability, reserve and pay, then collect and return a bike with employee assistance.

This is an academic design exercise. Interviews, workshops, observations and surveys are proposed activities; no research results or operating outcomes are claimed.

Step 1 Stakeholder identification

Stakeholder	Main interest
Customers	Easy booking, clear prices and available bikes.
Business owners	Profitability and customer satisfaction.
Employees	Clear handover and return procedures.
Local government	Safety and applicable operating rules.
Investors	Business viability and growth.

Step 2 Interview questions

Group	Questions to explore
Customers	How do you usually travel across the city? What would make renting and returning an e-bike convenient?
Business owners	What would define success in the first six months? Which features matter most at launch?
Employees	What could slow down handover and returns? What information would help you manage bike availability?

Step 3 MoSCoW prioritization

Must have: booking, payment processing and bike availability tracking.
Should have: a dedicated mobile app.
Could have: a loyalty program and tourist attraction partnerships.
Won’t have at launch: GPS tracking of individual bikes.

A basic website covers the core rental journey first. I deferred the loyalty program until the service has repeat customers.

Step 4 Core requirements traceability

This matrix links the three Must have requirements to the models below. They are proposed requirements, not implemented features.

ID	Requirement	Stakeholder	Model connection
REQ-01	Easy booking	Customers	Reserve bike; Customer and Rental records.
REQ-02	Payment processing	Owners and customers	Pay rental; payment_status in Rental.
REQ-03	Bike availability tracking	Employees and customers	View and update availability; status in Bike.

Step 5 Proposed requirements workshop

Participants: customers, an owner and rental employees. A short session would review the rental journey, discuss the most important needs and agree on launch priorities. The output would be a revised requirements list and any open questions.

Discussion prompts: Is availability clear before payment? What details do employees need at handover? What information confirms that a bike has been returned?

Step 6 Proposed observation

Observe pickup and return at a comparable service, such as Ecobici, to understand the steps customers follow. Record waiting time, instructions and common questions. Compare relevant steps with a staffed rental service before proposing improvements. This observation has not been conducted.

Step 7 Draft customer survey

1. What would you mainly rent an e-bike for? Commuting / tourism / leisure / other.
2. How often would you consider renting one? Daily / weekly / monthly / never / unsure.
3. Which matters most? Price / booking convenience / safety / location / availability.
4. How comfortable are you booking and paying online? Rate from 1 to 5.
5. What concern might stop you from renting an e-bike? Open response.

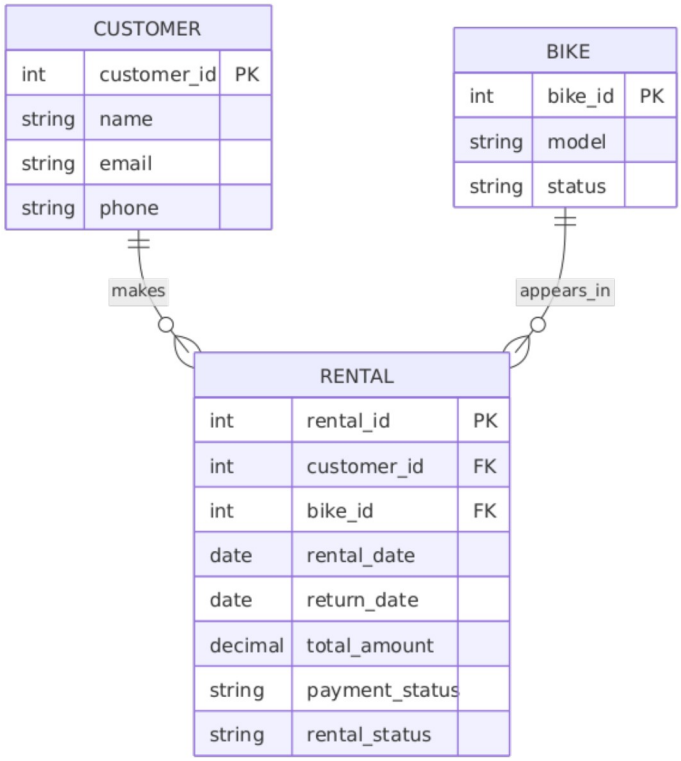
These questions would help review customer priorities. No survey responses have been collected.

Step 8 Optional prototype plan

A simple prototype would contain three screens: available bikes, booking and payment, and rental confirmation with return instructions. These screens are proposed for a later usability check; they have not been built or tested.

Step 9 Entity Relationship Diagram

The model contains three entities: Customer, Bike and Rental. Rental connects a customer to a bike and records the rental period, amount, payment status and rental status.



One customer can have zero or many rentals. One bike can appear in zero or many rentals over time. Each rental belongs to exactly one customer and one bike.

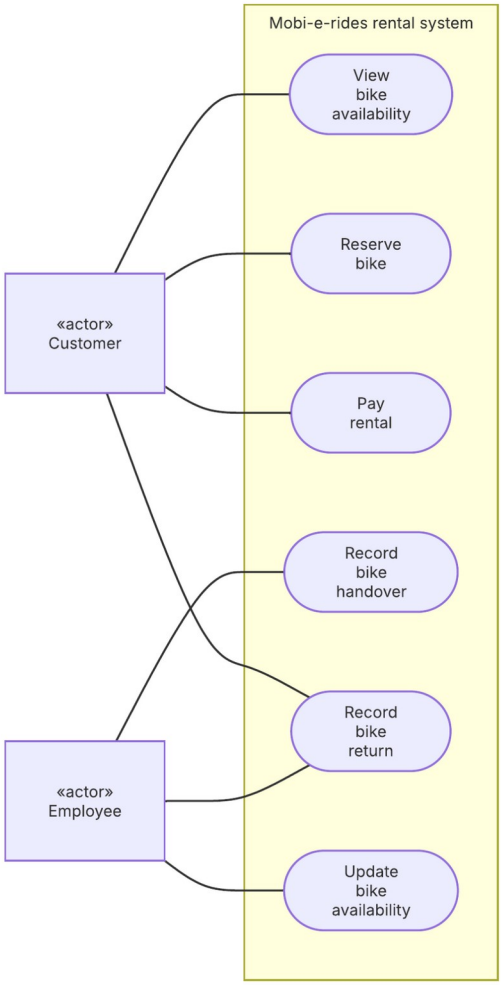
PK identifies a record. FK connects it to another entity. The crow’s foot symbol with a circle means zero or many. rental_date and return_date define the agreed rental period; rental_status records whether the rental is pending, confirmed, active or closed.

Basic rule: a bike cannot have overlapping confirmed or active rentals. A rental is confirmed only after payment is approved. Payment details stay in Rental for this introductory model.

View the three editable diagrams in Lucidchart

Step 10 Use Case Diagram

The system has two actors: Customer and Employee. Customers view availability, reserve and pay. Employees record handovers and returns and update availability. The customer also participates in the return; the employee records receipt in the system.



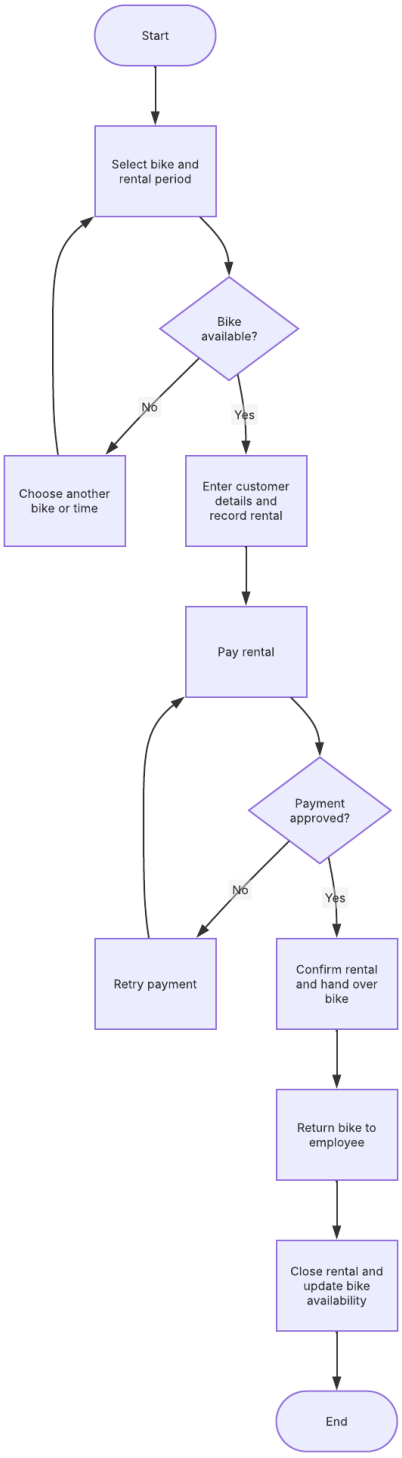
The system boundary shows which functions belong to Mobi-e-rides. Actor associations show participation; they do not describe the order of the process.

REQ-01 is represented by Reserve bike, REQ-02 by Pay rental, and REQ-03 by View bike availability and Update bike availability. Handover and return connect these functions to the physical service.

For simplicity, payment processing is represented as a customer-facing function. A payment provider would be considered when developing a more detailed system model.

Step 11 Rental Process Flowchart

The flow covers selection, payment, handover and return. Its two alternative paths handle unavailable bikes and rejected payments. It is a basic flowchart, not a BPMN diagram.



Step 12 Basic systems analysis

The service needs a booking website, customer and bike records, payment processing and an employee view of availability. Basic safeguards should prevent overlapping rentals and limit access to customer information.

Two risks guide the design: outdated availability can lead to conflicting bookings, and a failed payment must not confirm a rental. The availability and payment decisions in the flow address these risks.

Step 13 Reflection and validation

This exercise helped me understand how requirements connect to data, actors and process steps. The main challenge was keeping the three models consistent. Before developing the service, I would review the proposed requirements with customers and employees and test the booking journey with a simple prototype.

Academic reference: IBM, Business Analysis and Modeling for an eBike Rental Startup, exemplar solution. This case adapts the exercise using my stakeholder questions, priorities and rental models.