Software Requirements and Design Document

for

Hunar Bazaar

Prepared by Muhammad Ahmed Mufti, Muhammad Irtaza Khan, Muhammad Usman

MIUsys Technologies

Monday, November 25, 2024

Table of Contents

1.   Introduction

1.1   Purpose

The purpose of HunarBazaar is to create a seamless platform for connecting local technicians—such as plumbers, mechanics, and carpenters—with customers across Pakistan. By providing real-time, location-based solutions, the platform addresses the challenges of finding reliable service providers, especially during emergencies. The initial version of the product focuses on enhancing access to verified professionals in major cities like Karachi, Lahore, and Islamabad, ensuring convenience for users and growth opportunities for technicians.

1.2   Product Scope

HunarBazaar *is a mobile and web application tailored for the Pakistani market, aiming to revolutionize how customers access and technicians provide on-demand services. The platform's purpose is to bridge the gap between customers and reliable service providers by offering an easy-to-use, location-aware solution. HunarBazaar addresses urgent service needs with features like verified profiles, instant availability checks, and emergency service requests.*

The platform aligns with broader business strategies to digitize underserved sectors and empower local professionals. By leveraging the increasing penetration of smartphones and digital services in Pakistan, HunarBazaar aims to enhance convenience for users while creating new opportunities for technicians to grow their businesses digitally.

Key Benefits:

- *Faster and more reliable access to verified technicians.*
- *Expansion of service providers' market reach via digital tools.*
- *Improved customer trust through verified profiles and reviews.*
- *Location-based matching for quicker and more efficient service delivery.*

1.3   Title

HunarBazaar: Empowering Technicians, Simplifying Lives
A location-based platform connecting Pakistani technicians to customers in need, ensuring quick, reliable, and accessible solutions for daily and emergency service requirements.

1.4   Objectives

Real-time Connectivity: Create an efficient, location-based system that instantly connects customers with nearby, verified technicians.

Business Empowerment: Enable local technicians to expand their customer base and showcase their skills through a digital marketplace.

Customer Convenience: Offer a seamless platform to address urgent service needs in emergencies or routine situations.

Trust and Reliability: Build a network of verified service providers to ensure high-quality and dependable services.

Social Impact: Address the lack of digital representation for local technical workers, improving their livelihood and contributing to economic growth.

1.5   Problem Statement

In Pakistan, finding reliable service providers during emergencies—such as car breakdowns, plumbing issues, or electrical failures—is a daunting challenge. Traditional solutions, like word-of-mouth referrals or outdated online directories, are often ineffective, leading to wasted time, frustration, and, in some cases, higher costs.

Existing platforms like OLX and Facebook Marketplace are not designed for immediate, location-based service needs, focusing instead on product listings. This creates a gap for customers who require quick and trustworthy solutions to urgent technical problems. For example, a traveler stranded on the motorway between Islamabad and Lahore often struggles to locate an available mechanic.

HunarBazaar aims to fill this gap by providing a dedicated platform that connects users with nearby, verified technicians in real-time. Its feasibility is bolstered by the growing reliance on smartphones and digital platforms in Pakistan, coupled with a lack of direct competitors focused on service-based interactions. This initiative has the potential to transform access to technical services, making life easier for millions while empowering technicians to thrive in the digital era.

2.   Overall Description

2.1   Product Perspective

HunarBazaar *is a self-contained platform developed entirely in-house, featuring a JavaFX-based frontend, an SQL Server backend, and a built-in chat system for seamless communication between users and technicians. The platform is designed as a standalone solution without integration with external systems or APIs, ensuring complete control over functionality, data flow, and user interactions.*

This approach enables the team to tailor every aspect of the system to the specific needs of Pakistani customers and technicians. The absence of dependencies on third-party systems simplifies maintenance and enhances reliability, especially in environments where external services may be inconsistent or unavailable.

System Overview:

1. ***Frontend (JavaFX):*** *Provides an intuitive interface for users to register, search for services, request assistance, and interact with technicians.*

2. **Backend (SQL Server):** *Manages all data storage, including user profiles, service requests, technician details, and chat logs.*
3. **Built-In Chat System:** *Allows real-time communication between customers and technicians directly within the platform, eliminating the need for external messaging solutions.*

This architecture ensures a cohesive, secure, and efficient user experience, tailored to the target audience's requirements.

2.2   Product Functions

*The major functions of **HunarBazaar** include:*

- **User Registration and Profile Management:**
    - *Allows customers to create and manage profiles.*
    - *Enables technicians to create verified profiles showcasing their expertise, location, and service areas.*
- **Service Requests:**
    - *Customers can search for and request services based on their location and type of work.*
    - *Offers filters like urgency, proximity, and service ratings.*
- **Real-Time Matching:**
    - *Matches customers with nearby, available technicians using geolocation services.*
    - *Provides estimated time of arrival (ETA) and service costs (if applicable).*
- **Verification System:**
    - *Ensures technicians are verified through reviews, ratings, and optional identity verification.*
- **Emergency Assistance:**
    - *Enables customers to request urgent services (e.g., car breakdowns, water pipe bursts).*
- **Notification System:**
    - *Sends real-time updates about service requests, technician arrival, or completion status.*
- **Review and Rating System:**
    - *Allows customers to rate and review technicians after the service.*
    - *Helps build trust and transparency in the platform.*
- **Technician Dashboard:**
    - *Lets technicians track their service requests, earnings, and customer feedback.*

2.3   List of Use Cases

Use Cases:

- Create Account (Buyer and Seller)

- Exchange Information (Using External Communication Gateway)

- Cancel Order

- Submit Order

- Place Order

- Authenticate Credentials

- Customer Support

- Edit Profile (Buyer and Seller)

- Report Issue

- Manage Gigs

- Make Payment

- Mark Order as Complete

- Resolve Disputes

- Login (Buyer and Seller)

- Give Feedback (Buyer and Seller)

- Withdraw Payment

- Set Availability

- Search Gigs

- Manage Users

2.4   Extended Use Cases

BUYER AND SELLER COMMON ONES

1.Use Case: Create Account (Buyer and Seller)

Scope: Hunar Bazaar application

Level: User goal

Primary Actor: Buyer/Seller

Stakeholders and Interests:

- **Buyer**: Wants to easily create an account to purchase services.

- **Seller**: Wants to create an account to offer services.

- **Hunar Bazaar**: Wants to ensure smooth, secure account creation for both buyers and sellers.

- **Government/Regulators**: Requires proper identification and verification to comply with legal regulations.

Preconditions:

- The user has access to the Hunar Bazaar platform.
- The user does not already have an account linked to their email or phone number.

Success Guarantee (Postconditions):

- The account is created successfully, and the user (either buyer or seller) is logged into the platform.

- Verification email or SMS is sent for account confirmation.

- System prepares the appropriate dashboard depending on whether the user is a buyer or seller.

Main Success Scenario (Basic Flow):

1. Selects "Create Account" on the Hunar Bazaar platform.	
	2. Displays a registration form.
3. Fills in required details (name, email, password, phone number) and selects role (buyer or seller).	

	4. Validates the information and checks for existing accounts.
	5. Sends a verification email or SMS for confirmation.
6. Verifies the account through the provided link or code.	
	7. Confirms verification, creates the account, and logs the user in.

	8. Directs the user to the appropriate dashboard:
	
	    8a. Buyers: Directed to the buyer's homepage.
	
	    8b. Sellers: Directed to complete their profile and set up services.

Extensions (Alternate Flows):

- **4a. Invalid email or phone number**:

  1. System detects an invalid email or phone number format.

  2. User is prompted to enter valid details.

- **5a. Existing account**:

  1. System detects that an account with the same email or phone number already exists.

  2. User is informed and given the option to log in or reset their password.

- **6a. Verification failure**:

  1. User does not receive or fails to complete the verification.

  2. System prompts user to resend verification or enter a different contact method.

2. Use Case: Login (Buyer and Seller)

Scope: Hunar Bazaar application

Level: User goal

Primary Actor: Buyer/Seller

Stakeholders and Interests:

- **Buyer**: Wants to log in quickly to purchase services.

- **Seller**: Wants to log in easily to manage their services and interact with buyers.

- **Technician**: Wants secure and efficient logins to ensure smooth user experience.

Preconditions:

- The user (buyer or seller) has an existing account.

Success Guarantee (Postconditions):

- The user is logged in successfully and directed to their respective dashboard.

Main Success Scenario (Basic Flow):

1. User navigates to the login page.	
	2. System prompts user to enter credentials.
3. User enter their credentials. (email/phone number and password etc)	
	4. System verifies the credentials.

	5. User is logged in and directed to the appropriate dashboard: ○ **Buyers**: Directed to the buyer's homepage. ○ **Sellers**: Directed to the seller's dashboard to manage services.

Extensions (Alternate Flows):

- **3a. Invalid credentials**:

    1. System detects incorrect credentials (email or password etc.).

    2. User is informed and prompted to retry or reset the authentication method.

---

3. Use Case: Edit Profile (Buyer and Seller)

Scope: Hunar Bazaar application

Level: User goal

Primary Actor: Buyer/Seller

Stakeholders and Interests:

- **Buyer**: Wants to update personal information like name, email, password, payment details, etc.

- **Seller**: Wants to update their profile, business information, and service-related details.

- **Admin:**: Wants to ensure that profiles are up-to-date and reflect accurate information for both buyers and sellers.

Preconditions:

- The user is logged into their Hunar Bazaar account.

Success Guarantee (Postconditions):

- The user's profile is updated successfully, and the changes are saved.
- System reflects the updated profile information for both buyers and sellers.

Main Success Scenario (Basic Flow):

1. User Navigates to the "My Profile" section.	
	2. System Display current Information
3. User Click on Edit Button	
4. User Edits the desired fields	
5. User Submits the changes.	
	6. System provides information to admin for authentication
	7. System updates the information
	8. System display updated information redirect to My Profile

Extensions (Alternate Flows):

- **6a. Invalid input**:

  1.   System return invalid input after authentication.

2. User is prompted to correct the invalid input.

- **7a. System failure**:

    1. The system fails to save changes due to a technical issue.

    2. System informs the user and asks them to retry after some time.

---

4. Use Case: Exchange Information (Using External Communication Gateway)

Scope: Hunar Bazaar application

Level: User goal

Primary Actor: Buyer/Seller

Stakeholders and Interests:

- **Buyer**: Wants to communicate with sellers regarding the services being purchased, ask questions, or share details related to the project.

- **Seller**: Wants to communicate with buyers to clarify service requirements, provide updates, or finalize project details.

- **Hunar Bazaar**: Wants to ensure secure and seamless communication between buyers and sellers, preventing any breach of platform rules (e.g., sharing personal contact info).

- **External Communication Gateway**: Facilitates real-time messaging, file sharing, and other exchanges between buyers and sellers.

Preconditions:

- Both buyer and seller have active Hunar Bazaar accounts and are logged in.
- A service request or ongoing transaction exists between the buyer and seller.

Success Guarantee (Postconditions):

- The buyer and seller successfully exchange information using the external communication gateway (e.g., messages, files).
- All messages and shared data are stored and logged by the system.

Main Success Scenario (Basic Flow):

1. Buyer selects the "Message Seller" option on the order or service page.	
	2. Opens the communication window integrated with an external communication gateway.
3. Buyer types and sends a message to the seller.	
	4. Routes the message through the external communication gateway and ensures it reaches the seller.
	5. Notifies the seller about the incoming message.
6. Seller receives the message and replies.	
	7. Sends the seller's reply to the buyer via the same communication channel.
8. Both parties can continue exchanging messages or attachments (e.g., project files, images).	
	9. Logs all messages and file exchanges securely.

Extensions (Alternate Flows):

- **4a. External gateway unavailable**:

    1. System detects that the external communication gateway is unavailable.

    2. User is informed, and communication is postponed until the system restores the connection.

- **6a. File size exceeds limit**:

    1. Buyer or seller attempts to share a file that exceeds the allowed size.

    2. System prompts the user to compress the file or share a smaller file.

---

5. Use Case: Give Feedback (Buyer and Seller)

Scope: Hunar Bazaar application

Level: User goal

Primary Actor: Buyer/Seller

Stakeholders and Interests:

- **Buyer**: Wants to provide feedback on a completed service, rating the seller based on the quality of the service provided.

- **Seller**: Wants to provide feedback on the buyer, rating their communication and experience working with them.

- **Technician**: Wants to ensure feedback is collected to help maintain quality and reputation for both buyers and sellers. Also, uses feedback for service improvement and dispute resolution.

Preconditions:

- A transaction between the buyer and seller has been completed (e.g., service delivered, payment made).
- Both buyer and seller are logged into their accounts.

Success Guarantee (Postconditions):

- Feedback is successfully submitted and stored.
- The feedback/rating is visible on the profiles of the respective buyer or seller.

Main Success Scenario (Basic Flow):

	1. After completing the service, buyer and seller are prompted to provide feedback.
2. Both parties submit their feedback (rating (e.g., 1-5 stars) and comments about the service etc).	
	3. System stores the feedback and updates the respective profiles.
	4. Feedback is visible to future buyers or sellers as part of the user's profile.

Extensions (Alternate Flows):

- **2a. Buyer/Seller skips feedback**:

    1. The buyer/seller decides not to leave feedback at the time of prompt.

    2. The system provides a reminder to leave feedback within a certain time window (e.g., 7 days).

- **4a. Inappropriate feedback detected**:

    1. System detects inappropriate or offensive language in the feedback.

    2. System prompts the user to revise the feedback or flags it for review by an administrator.

6.Use Case: Report Issue

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Buyer or Seller

Stakeholders and Interests:

- **Buyer**: Wants to report an issue with an order, service quality, or platform functionality.

- **Seller**: Wants to report issues related to the order, or platform functionality.

- **Technical Support Team:** Make measures to handle the issues created.

Preconditions:

- User (buyer or seller) is logged in to their account.
- A valid issue related to platform functionality exists.

Success Guarantee (Postconditions):

- The issue is reported and logged.
- The system acknowledges the issue and assigns it for review or resolution.

Main Success Scenario:

1.　　User navigate to "Report Issue"	
	2.　　System generate options regarding reporting issue
3.　　User selects option in which he is having issue	

4. User fills details of issue	
5. User Submit the report	
	6. System confirms the user and forward to relevant person

Extensions (Alternate Flows):

- **3a. No relevant order or feature**:

  ○   System informs the user that no relevant order or feature is available for reporting. The user can then select a generic issue category.

- **5a. Missing or incomplete details**:

  ○   System prompts the user to complete the missing details before submitting.

---

7. Use Case: Cancel Order

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Buyer or Seller

Stakeholders and Interests:

- **Buyer**: Wants to cancel an order due to a change in requirements or dissatisfaction with service quality.

- **Seller**: Wants to cancel an order if the buyer is unresponsive, or the project cannot be completed for valid reasons.

- **Hunar Bazaar**: Ensures that cancellations are processed smoothly, and disputes are minimized.

Preconditions:

- Buyer or seller is logged in to their account.
- A valid order exists, and it is within the cancellation window based on platform policies.

Success Guarantee (Postconditions):

- The order is successfully canceled.
- Refunds (if applicable) are processed, and both parties are notified.

Main Success Scenario:

1. User navigates to the "My Orders" section.	
	2. Displays the list of orders.
3. User selects the order they wish to cancel.	
	4. Displays the selected order details.
5. User clicks on the "Cancel Order" button and provides a reason for cancellation.	
	6. Confirms the cancellation request and displays the refund policy (if applicable).
7. User confirms the cancellation.	
	8. Cancels the order and updates the order status to "Canceled."
	9. Notifies both buyer and seller of the cancellation.

User navigates to the "My Orders" section.

1. User selects the order they wish to cancel.
2. User clicks on the "Cancel Order" button and provides a reason for cancellation.
3. System confirms the cancellation request and displays the refund policy (if applicable).
4. User confirms the cancellation.
5. System cancels the order and updates the order status to "Canceled."
6. Both buyer and seller are notified of the cancellation.

Extensions (Alternate Flows):

- **3a. Order cannot be canceled**:

    ○ If the order is past the cancellation window or already completed, the system informs the user that the order cannot be canceled.

- **4a. Disagreement on cancellation**:

    ○ If one party disputes the cancellation (e.g., seller disagrees with buyer's cancellation request), the system escalates the case for admin intervention.

ONLY SELLER

8. Use Case: Withdraw Payment

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Seller

Stakeholders and Interests:

- **Seller**: Wants to withdraw earned payments from completed orders efficiently.
- **Technician**: Aims to ensure secure and timely payment processing while maintaining user trust.

Preconditions:

- Seller is logged in to their account.

- Seller has a minimum withdrawal amount reached and valid payment methods set up.

Success Guarantee (Postconditions):

- Payment is successfully withdrawn and transferred to the seller's external payment account.
- Seller is notified of the transaction details.

Main Success Scenario:

1. Seller navigates to the "Withdraw Payment" section.	
	2. System displays balance details and prompts users to enter information to complete transactions.
3. Seller reviews available balance and selects the amount to withdraw.	
4. Seller selects the external payment gateway (e.g., PayPal, bank transfer).	
	5. System verifies the selected payment method and available balance.
6. Seller confirms the withdrawal request.	
	7. System processes the withdrawal through the external payment gateway.
	8. System notifies the seller of the successful transaction and expected processing time.

Extensions (Alternate Flows):

- **3a. Payment method not set up**:

      ○ System prompts the seller to set up a payment method before proceeding.

- **5a. Withdrawal amount exceeds balance**:

      ○ System informs the seller of insufficient funds and requests a valid amount.

9.Use Case: Manage Gigs

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Seller

Stakeholders and Interests:

- **Seller**: Wants to create, edit, or delete gigs to showcase services and attract buyers.
- **Hunar Bazaar**: Aims to maintain a diverse range of services and ensure that listings comply with platform standards.

Preconditions:

- Seller is logged in to their account.

Success Guarantee (Postconditions):

- Gig information is successfully created, updated, or deleted in the system.
- Buyers can view the updated gig listings.

Main Success Scenario:

1. Seller navigate to Manage Gigs	
	2. System Displays all the gigs of seller
3. Seller select new gig or edit gig or delete gig	

4. In case of new gig, user enter details of new gig	
5. Seller submits information of the gig	
	6. System takes information and forwards relevant information to accept the gig.
	7. System publish the gig

Extensions (Alternate Flows):

- ● **4a. Selecting other options**

  ○ **4ai. Edit Gig**

    ■ User will select the gig he wants to edit.

    ■ System will show current details of the gig

    ■ User will edit information

    ■ System will check and make changes to gig

  ○ **4aii. Delete Gig**

    ■ User will select the gig he wants to delete.

    ■ System will delete the gig.

- ● **6a. Incorrect information error**:

  ○ System informs the seller of incorrect information and asks him to enter again.

10. Use Case: Submit Order

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Seller

Stakeholders and Interests:

- **Seller**: Wants to submit orders for services or products requested by buyers.

- **Buyer**: Expects timely and accurate fulfillment of their orders.

- **Hunar Bazaar**: Aims to facilitate smooth transactions and ensure that orders are processed efficiently.

Preconditions:

- Seller is logged in to their account.
- Seller has an active gig that the buyer has selected.

Success Guarantee (Postconditions):

- The order is successfully submitted and recorded in the system.
- Buyer is notified of the order confirmation.

Main Success Scenario:

1. Seller receives a request from a buyer for a service.	
	2. Notifies the seller of the service request.
3. Seller reviews the buyer's requirements.	
	4. Prepares for order submission.

5. Seller navigates to the "Submit Order" section and enters order details.	
	6. Records the order and updates the status.
7. Seller confirms the submission.	
	8. Notifies the buyer of the order status update.

Extensions (Alternate Flows):

- **4a. Missing information from buyer**:

    ○ System prompts the seller to request additional details from the buyer.

- **5a. Order submission fails**:

    ○ System informs the seller of the error and requests a retry.

---

11. Use Case: Set Availability

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Seller

Stakeholders and Interests:

- **Seller**: Wants to manage their availability to ensure they can fulfill orders without overcommitting.
- **Technician**: Aims to maintain accurate availability information to enhance user experience and service reliability.

Preconditions:

● Seller is logged in to their account.

Success Guarantee (Postconditions):

● Seller's availability status is updated in the system.
● Buyers can see the seller's current availability when browsing gigs.

Main Success Scenario:

1. Seller navigates to the "Set Availability" section.	
	2. System prompts user to enter availability details.
3. Seller selects the days and hours they are available for work.	
4. Seller confirms the changes.	
	5. System updates the availability status and reflects it on the seller's gig listings.

Extensions (Alternate Flows):

● **2a. Invalid availability settings**:

  ○ System alerts the seller about overlapping schedules or invalid time ranges.

● **3a. System error during update**:

  ○ System informs the seller of the error and prompts them to retry.

ONLY BUYER

12. Use Case: Make Payment

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Buyer

Stakeholders and Interests:

- **Buyer**: Wants to securely complete payment for a selected gig or service.

- **Seller**: Expects prompt payment for services rendered.

- **Bank:** Bank will validate the payment information.

Preconditions:

- Buyer is logged in to their account.

- Buyer has selected a gig and placed the order.

- Buyer has a valid payment method linked to their account.

Success Guarantee (Postconditions):

- Payment is successfully processed and recorded.
- The buyer receives confirmation of the transaction.

Main Success Scenario:

1. Buyer place the order and navigate to checkout	
	2. System Displays the total amount to be paid
3. Buyer give his payment method	

	4. System take buyer to external gateway
5. Buyer provide information and pay the bill	
	6. Gateway validates the payment and returns the receipt
	7. System prompts "Payment received" and updates the status of order.

Extensions (Alternate Flows):

- **4a. Payment gateway is unavailable**:

  ○   System informs the buyer of the issue and suggests trying again later.

- **5a. Payment fails**:

  ○   In case the buyer enters incorrect information, the gateway displays "Incorrect information" and asks the buyer to enter again.

---

13. Use Case: Place Order

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Buyer

Stakeholders and Interests:

- **Buyer**: Wants to place an order for a service easily and accurately.

- **Seller**: Expects to receive clear and complete orders for timely fulfillment.

- **Hunar Bazaar**: Aims to streamline the order process to enhance user experience and efficiency.

Preconditions:

- Buyer is logged in to their account.
- Buyer has selected a gig they want to order.

Success Guarantee (Postconditions):

- Order is successfully created and recorded in the system.
- Both buyer and seller are notified of the new order.

Main Success Scenario:

1. Buyer navigates to the gig listing page.	
	2. Displays the list of gigs.
3. Buyer selects a gig and reviews the details (description, pricing, delivery time).	
	4. Shows detailed gig information.
5. Buyer clicks on the "Order Now" button.	
	6. Prompts the buyer to enter specific order requirements.
7. Buyer submits the order.	
	8. Confirms the order and notifies the seller.

1. Buyer navigates to the gig listing page.
2. Buyer selects a gig and reviews the details (description, pricing, delivery time).
3. Buyer clicks on the "Order Now" button.
4. System prompts the buyer to enter any specific requirements or details related to the order.
5. Buyer submits the order.
6. System confirms the order and notifies the seller.

Extensions (Alternate Flows):

- **3a. Gig is unavailable**:

  ○ System informs the buyer that the selected gig is no longer available and suggests alternatives.

- **4a. Missing information**:

  ○ System prompts the buyer to fill in required fields before submission.

---

14. Use Case: Search Gigs

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Buyer

Stakeholders and Interests:

- **Buyer**: Wants to efficiently search for relevant gigs or services that meet their needs.

- **Seller**: Expects to be discoverable by potential buyers through effective search functionality.

- **Technician**: Aims to provide a robust search feature to enhance user engagement and service accessibility.

Preconditions:

- Buyer is logged in to their account or browsing as a guest.

Success Guarantee (Postconditions):

- Search results are successfully generated and displayed based on the buyer's input.

● Buyer can view and select gigs from the search results.

Main Success Scenario:

1. Buyer navigates to the "Search Gigs" section.	
2. Buyer enters search keywords (e.g., service type, keywords) and applies any filters (category, price range).	
	3. System processes the search query and retrieves matching gigs.
	4. System displays the search results to the buyer.
5. Buyer reviews the results and selects a gig for more details.	

Extensions (Alternate Flows):

● **3a. No results found**:

○ System informs the buyer that no gigs match the search criteria and suggests modifying the search.

● **4a. Too many results**:

○ System prompts the buyer to apply more filters to refine the search.

---

15. Use Case: Mark Order as Complete

Scope: Hunar Bazaar application
Level: User goal

Primary Actor: Buyer

Stakeholders and Interests:

- **Buyer**: Wants to mark the order as complete once satisfied with the service.
- **Seller**: Expects confirmation of order completion to receive payment and feedback.

Preconditions:

- Buyer is logged in to their account.
- The order must be in a status indicating that it is complete or delivered.

Success Guarantee (Postconditions):

- The order is marked as complete in the system.
- Seller is notified, and the payment process (if applicable) is initiated.

Main Success Scenario:

1. Buyer navigates to "Orders".	
2. Buyer marks the order completed.	
	3. System confirm completion of order and ask for feedback of seller
4. Buyer provide review/ feedback and set completed	
	5. System notifies seller and start processing payment

Extensions (Alternate Flows):

- **4a. Feedback submission fails**:
  - System informs the buyer of the error and allows them to retry or skip feedback.

ONLY ADMIN

16. Use Case: Authenticate Credentials

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Admin

Stakeholders and Interests:

- **Admin**: Wants to access the system securely to manage platform operations.
- **Hunar Bazaar**: Ensures only authorized admins have access to sensitive data and functions.

Preconditions:

- Admin must have valid login credentials.

Success Guarantee (Postconditions):

- Admin is successfully logged in and can access the administrative dashboard.

Main Success Scenario:

1. Admin navigates to the login page.	
	2. Displays the login form.
3. Admin enters their username and password.	

	4. Validates the credentials.
5. Admin is logged in.	
	6. Directs the admin to the admin dashboard.

Extensions (Alternate Flows):

- **3a. Invalid credentials**:

  ○ System prompts the admin to re-enter correct credentials or reset the password.

- **3b. Admin account locked**:

  ○ After multiple failed attempts, system locks the account, and admin is notified of the lockout procedure.

---

17. Use Case: Manage Users

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Admin

Stakeholders and Interests:

- **Admin**: Wants to manage user accounts, including suspending, deleting, or updating profiles.
- **System Moderators:** Ensures that user activity complies with platform policies.

Preconditions:

- Admin is authenticated and logged in.

Success Guarantee (Postconditions):

- User accounts are successfully managed, including actions such as suspension, deletion, and updates.

Main Success Scenario:

1. Admin navigates to the user management section.	
	2. System prompts admin to select user type
3. Admin selects type and searches for a specific user or views a list of all users.	
4. Admin selects a user account to manage.	
	5. System offers admin to take actions. (suspend, delete, edit etc)
6. Admin selects action (e.g., suspends, deletes, or edits the account).	
	7. System confirms the action and updates the database.
	8. User is notified of any changes via email/SMS.

Extensions (Alternate Flows):

- **3a. User not found**:
    - System displays a "User not found" error.
- **4a. Action fails**:
    - System reports an error, and admin retries or escalates the issue to tech support.

---

18. Use Case: Resolve Disputes

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Admin

Stakeholders and Interests:

- **Buyer**: Wants their dispute addressed fairly and quickly.

- **Seller**: Wants their dispute addressed fairly and quickly.

- **Admin**: Ensures that disputes are resolved efficiently and fairly to maintain trust on the platform.

Preconditions:

- Admin is authenticated and logged in.
- A dispute has been raised between a buyer and a seller.

Success Guarantee (Postconditions):

- Dispute is resolved, and both parties are informed of the decision.

Main Success Scenario:

1. Admin navigates to the disputes section.	
	2. System displays a list of disputes.
3. Admin views the details of specific disputes.	
4. Admin view detail and performance of seller and buyer.	
	5. System prompts the detail of dispute

	6. System gives access to the admin in the message box through an external gateway.
7. Admin contacts both seller and buyer and makes a decision.	
	8. System set dispute as completed and removed from the list.

Extensions (Alternate Flows):

- **7a. Admin decision:**

  ○   Admin will either give a penalty to the seller or block the buyer or make any other decision.

- **7b. Decision contested**:

  ○   Buyer or seller can appeal the decision, which escalates the case to a higher level.

- **8a. Couldn't set to completed:**

  ○   In case buyer or seller appeal the decision system will not set dispute as completed.

---

19. Use Case: Customer Support

Scope: Hunar Bazaar application
 Level: User goal

Primary Actor: Admin, Seller, Buyer

Stakeholders and Interests:

- **Users (Buyers and Sellers)**: Want their issues or questions addressed promptly.
- **Hunar Bazaar**: Aims to maintain high user satisfaction by resolving customer issues efficiently.

Preconditions:

- Admin is authenticated and logged in.
- User has submitted a support ticket or request.

Success Guarantee (Postconditions):

- User issue or query is resolved, and the customer is notified of the resolution.

Main Success Scenario:

1. Users will navigate to the customer support dashboard.	
	2. System will display common issues to user
3. User will choose the issue he is facing.	
	4. System will assign user ticket number
5. User will enter details of the problem he is facing.	
	6. System will take the user's details to the admin.
7. Admin will contact the user to get further detail.	
8. Admin will suggest customer solutions.	

	9. System will mark completed against this ticket number.

Extensions (Alternate Flows):

- **3a. Issue not Available:**

  ○ In case an issue to be solved is not present, the suggested user will write his details in the text box.

- **8a. Not Solved:**

  ○ User will again contact admin that solution didn't work

  ○ System will not move any further until issue is solved

2.5 Use Case Diagram
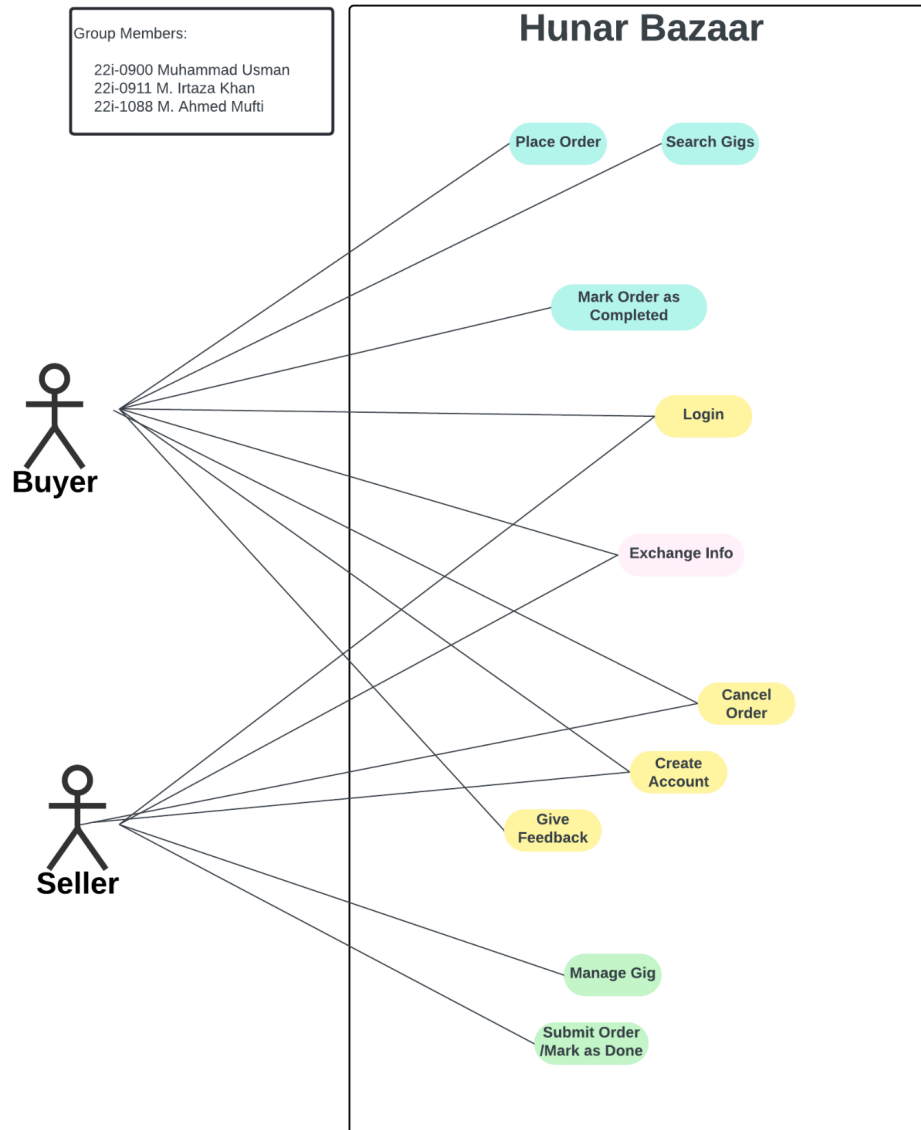
3.   Other Nonfunctional Requirements

3.1   Performance Requirements

4.   ***Response Time:*** *The platform must respond to user queries (e.g., searching for a technician or initiating a chat) within 2 seconds under normal load conditions.*
5.   ***Scalability:*** *The system should handle up to 1,000 concurrent users during peak times without performance degradation.*
6.   ***Data Processing:*** *Service request matching and technician availability checks should be processed in under 1 second.*
7.   ***Resource Usage:*** *Memory usage on the client side (JavaFX) should not exceed 500 MB, ensuring compatibility with mid-range hardware.*
8.   ***Database Performance:*** *SQL Server queries must execute within 500 ms for standard operations like fetching user profiles, chat history, and service details.*

8.1   Safety Requirements

Data Integrity: *All data, including technician profiles, customer requests, and chat messages, must be stored securely to prevent corruption. Regular backups should be maintained to recover from data loss.*
Error Handling: *The system must handle unexpected crashes gracefully, displaying clear error messages and preserving user progress wherever possible.*
Prevent Misuse: *Safeguards should be implemented to prevent unauthorized access to sensitive functions (e.g., technician verification or profile editing).*

8.2   Security Requirements

9.   ***Authentication:*** *All users must authenticate using unique credentials (username and password). Passwords must be stored using hashing algorithms to prevent unauthorized access.*
10.   ***Access Control:*** *Role-based access must be enforced, ensuring that only authorized users can access certain functionalities (e.g., technicians managing their profiles or admins handling verifications).*
11.   ***Data Encryption:*** *Communication between the frontend and backend must be encrypted to prevent data breaches, especially for sensitive information such as contact details and chat messages.*
12.   ***Audit Logging:*** *The system should log critical events (e.g., user registration, service requests, and profile updates) to identify and prevent potential security breaches.*

12.1   Software Quality Attributes

Reliability: *The platform should operate without failures for 99.9% of the time over a one-month period.*
Usability: *The user interface must be intuitive, requiring no more than 5 minutes of learning for first-time users.*
Maintainability: *The system codebase should be modular and well-documented to facilitate future updates and debugging.*

Scalability: *The platform must support increased users and data loads as the user base grows, with provisions for horizontal scaling.*
Robustness: *The system should handle incorrect inputs or unexpected user actions without crashing.*
Testability: *Automated tests should cover 90% of the system's code to ensure quick detection and resolution of bugs.*

12.2   Business Rules

Technician Verification: *Only verified technicians are allowed to offer services on the platform. Verification involves an admin reviewing technician details and documentation.*
Service Matching: *Customers can only request services from technicians within a 10 km radius for non-urgent needs and up to 50 km for emergencies.*
Role Restrictions:

- *Customers can initiate service requests and provide reviews.*
- *Technicians can update their profiles, accept/reject requests, and view ratings.*
- *Admins manage user verifications, resolve disputes, and monitor system activity.*

12.3   Operating Environment

Client Hardware: *The platform should operate smoothly on mid-range computers and laptops with at least 4 GB RAM and a dual-core processor.*
Operating Systems: *The JavaFX frontend must support Windows (10 and later) and Linux-based systems.*
Backend Environment: *The SQL Server database will run on a server with a minimum configuration of 8 GB RAM, quad-core CPU, and 100 GB SSD storage.*
Network Requirements: *A stable internet connection with a minimum bandwidth of 2 Mbps is recommended for seamless user experience.*

12.4   User Interfaces

Design Standards: *The user interface should adhere to a clean and minimalistic design, ensuring ease of navigation.*
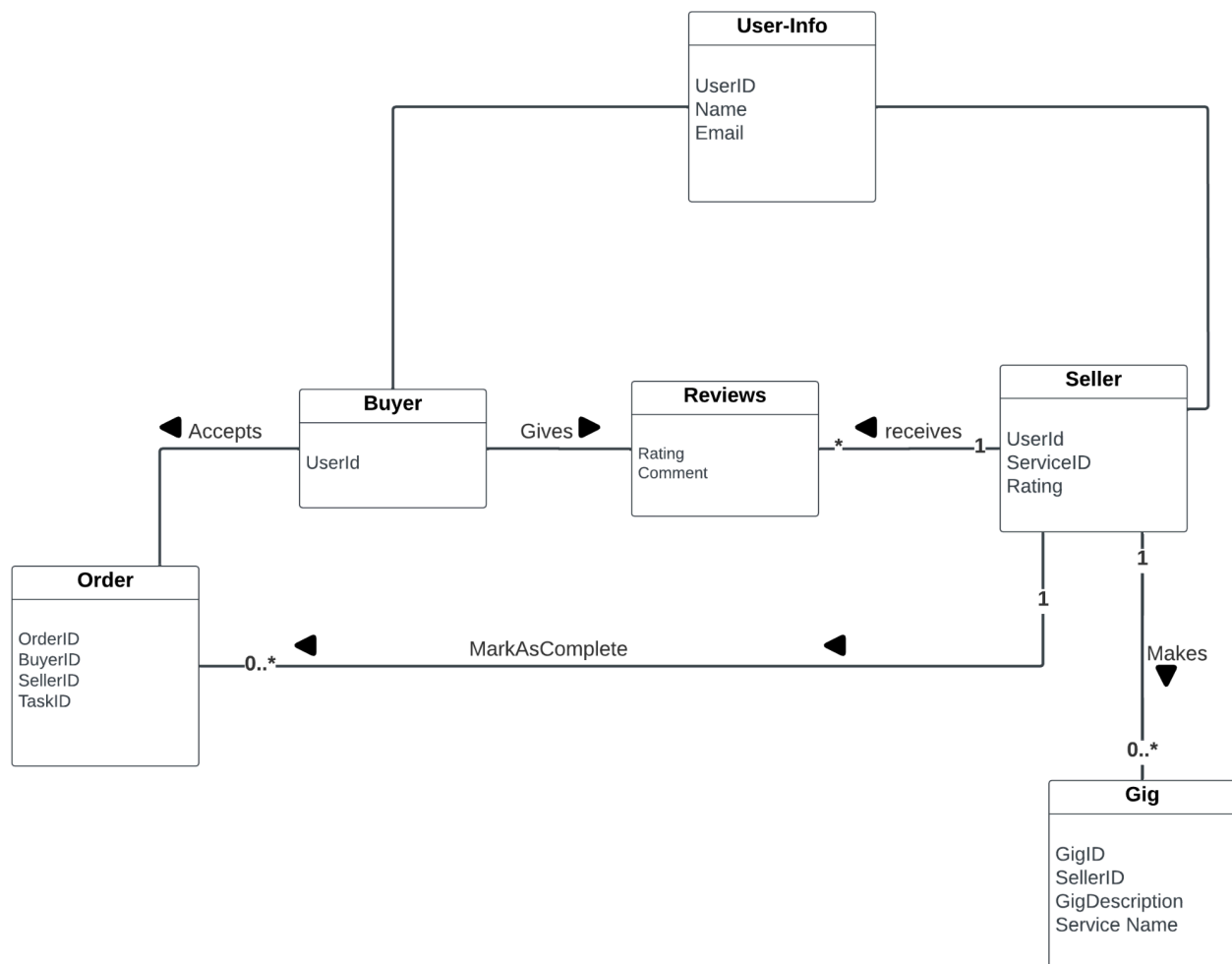**Key Components:**

- **Homepage:** *Includes a search bar for service needs and quick links to categories like plumbers, electricians, etc.*
- **Technician Profiles:** *Display verified badges, service ratings, and reviews.*
- **Chat Window:** *A simple interface for real-time text communication with a typing indicator and message timestamps.*
- **Service Request Screen:** *Allows users to specify details of the service they require, including location and urgency.*
- **Admin Dashboard:** *Provides tools for user verification, dispute resolution, and performance monitoring.*

Error Messages: Must be informative and user-friendly, avoiding technical jargon (e.g., "Network error. Please try again later.").
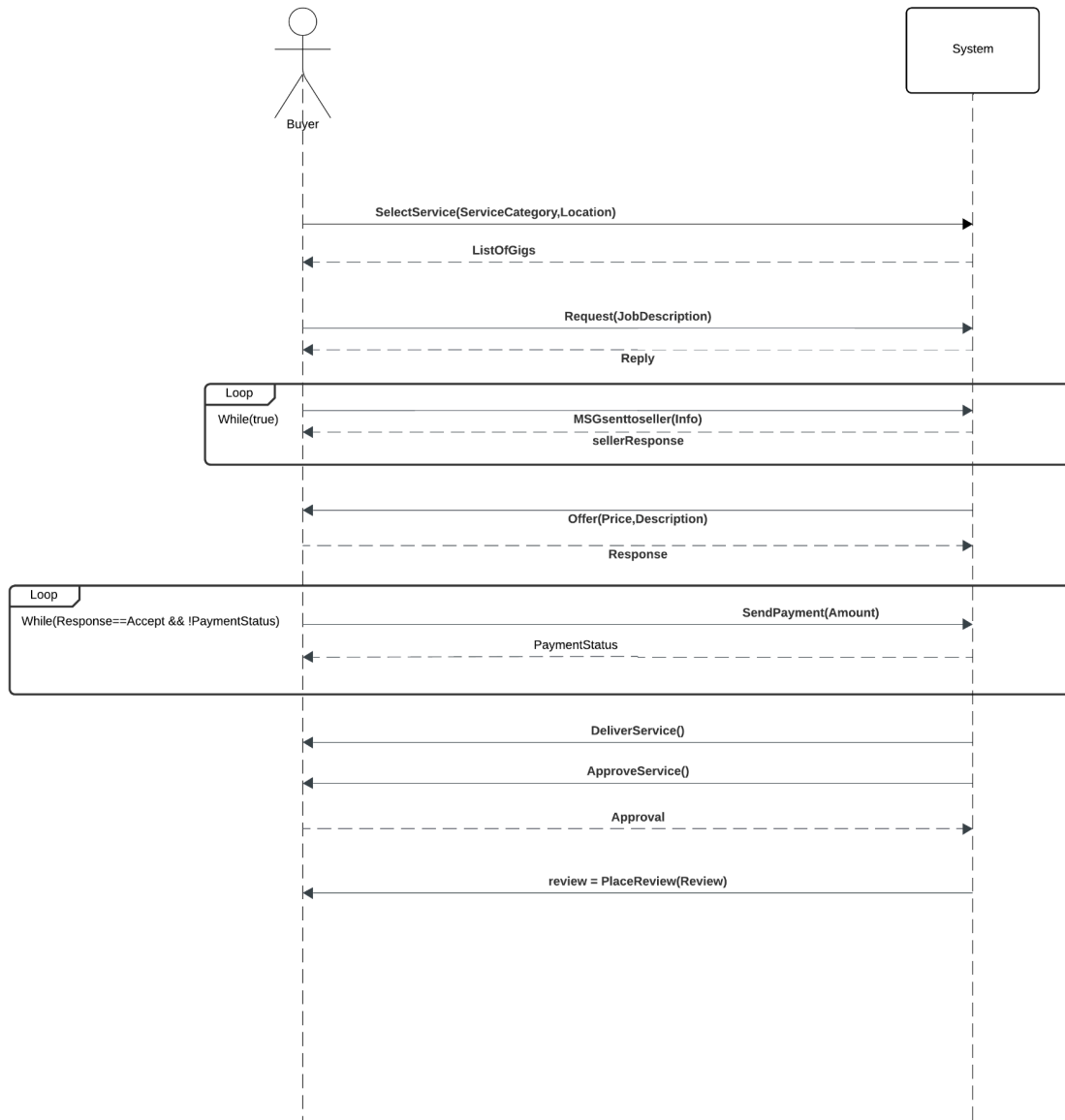*Standard Features: Include a "Help" button on all screens, consistent navigation menus, and keyboard shortcuts for quick access to main functions.*
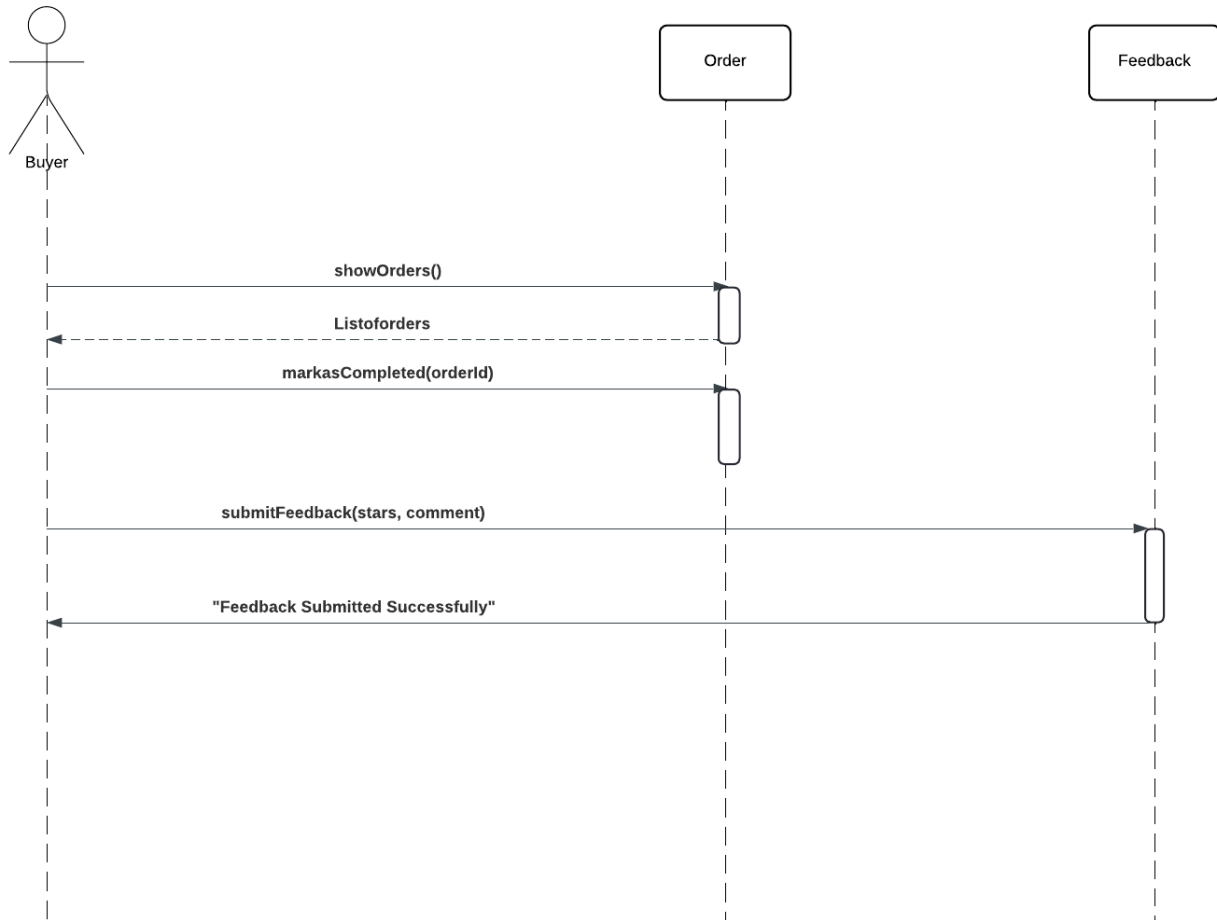
13. Domain Model
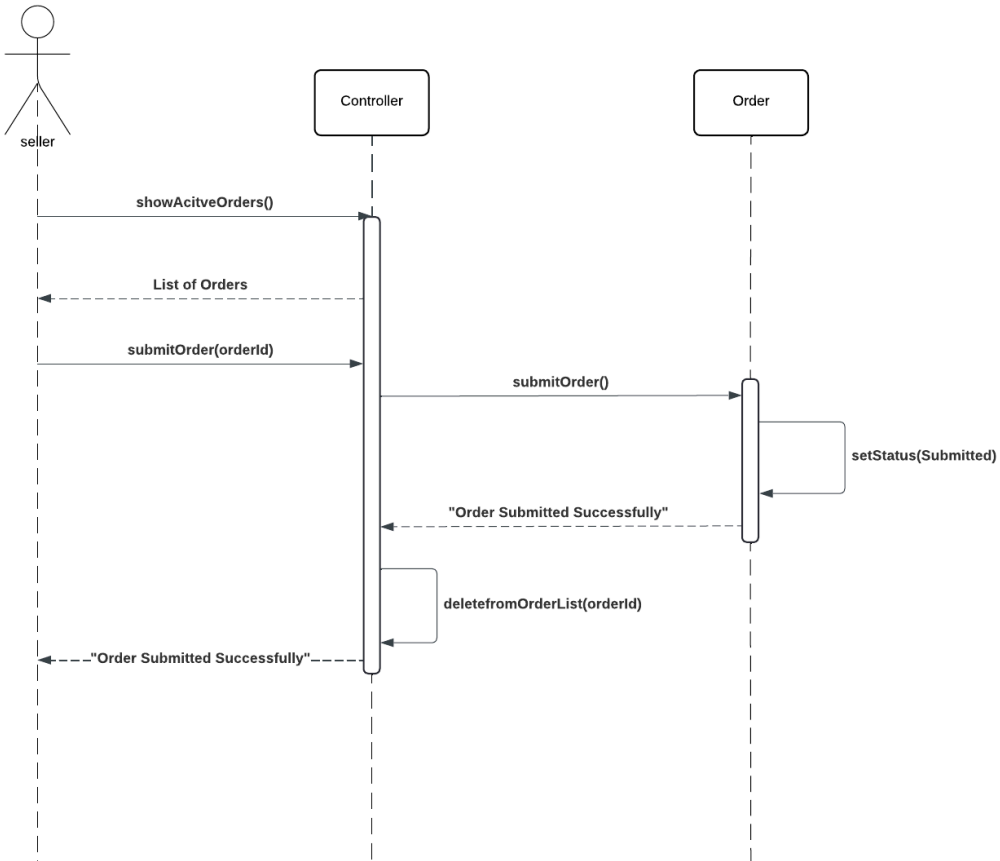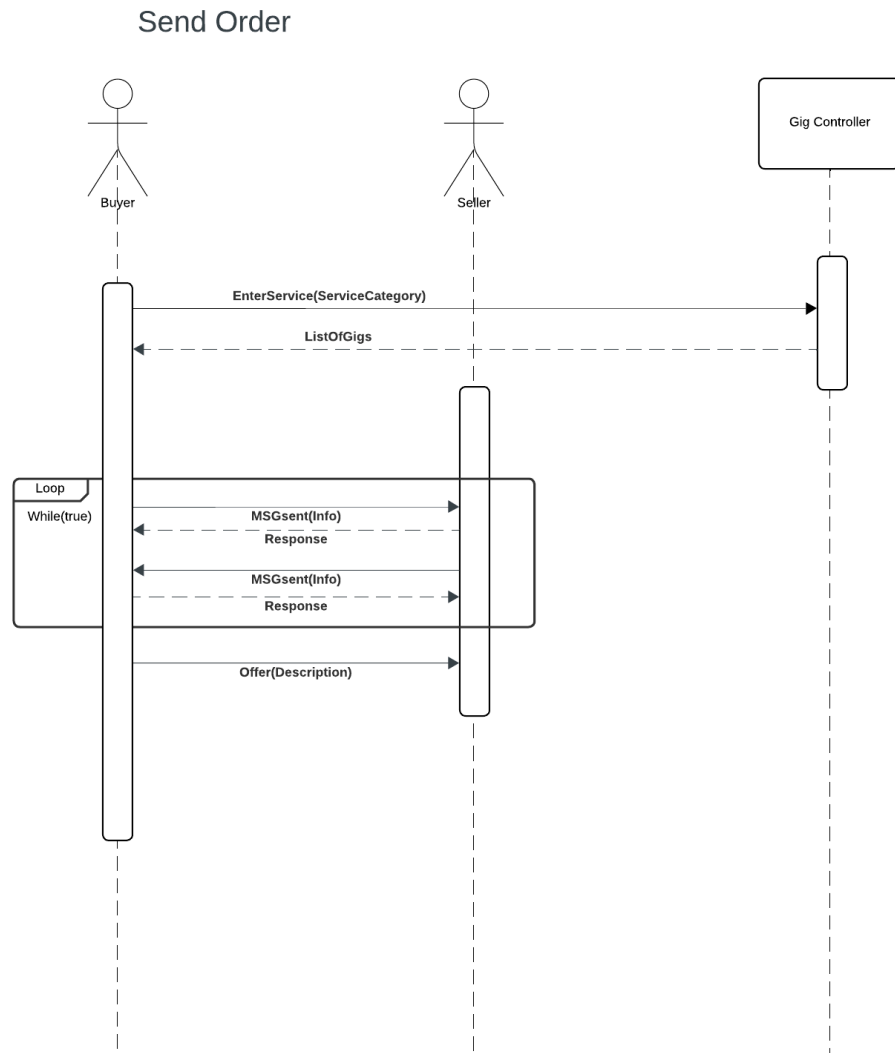
14. System Sequence Diagram

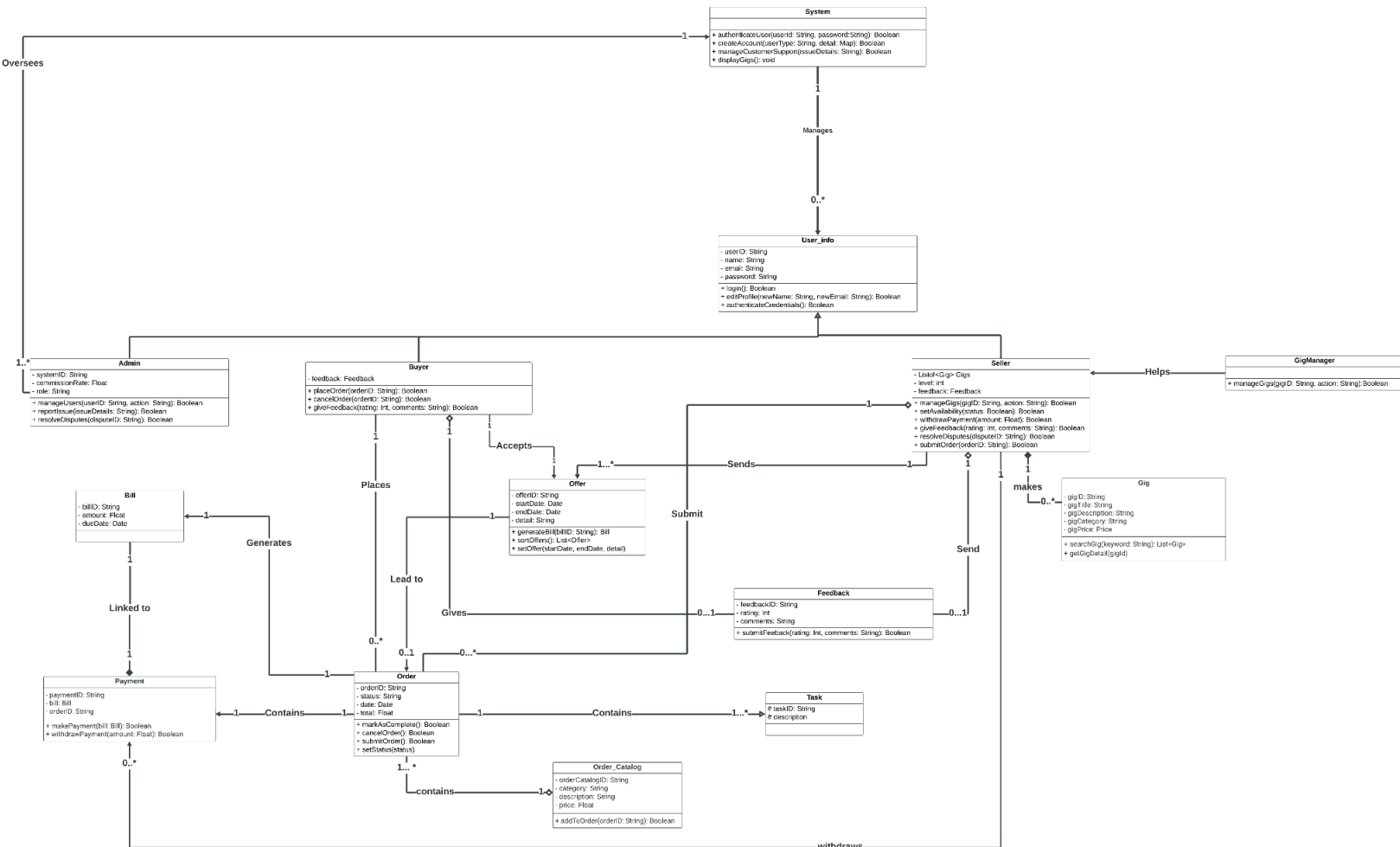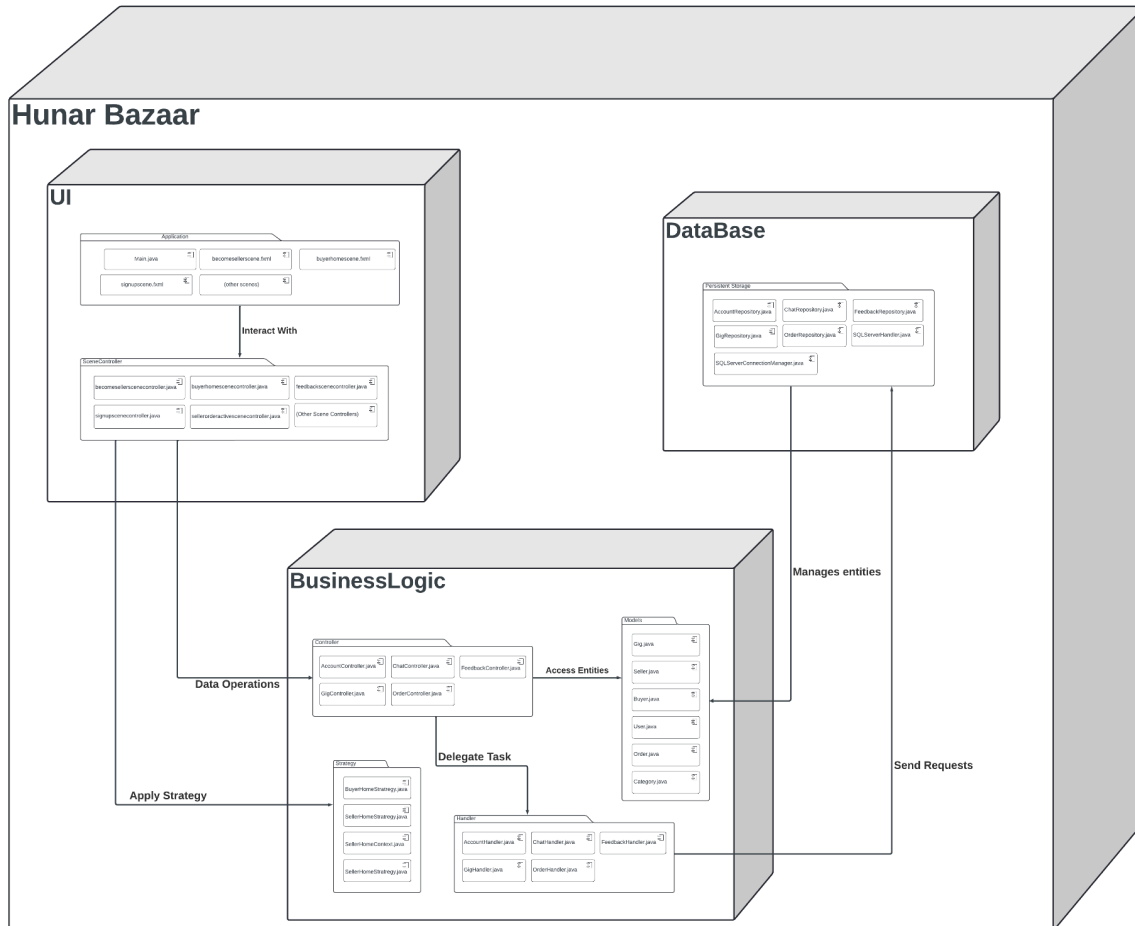For Buyer,

15. Sequence Diagram

Mark as Completed

Submit Order

Send Order

Buyer

Seller

Gig Controller

EnterService(ServiceCategory)

ListOfGigs

Loop

While(true)

MSGsent(Info)

Response

MSGsent(Info)

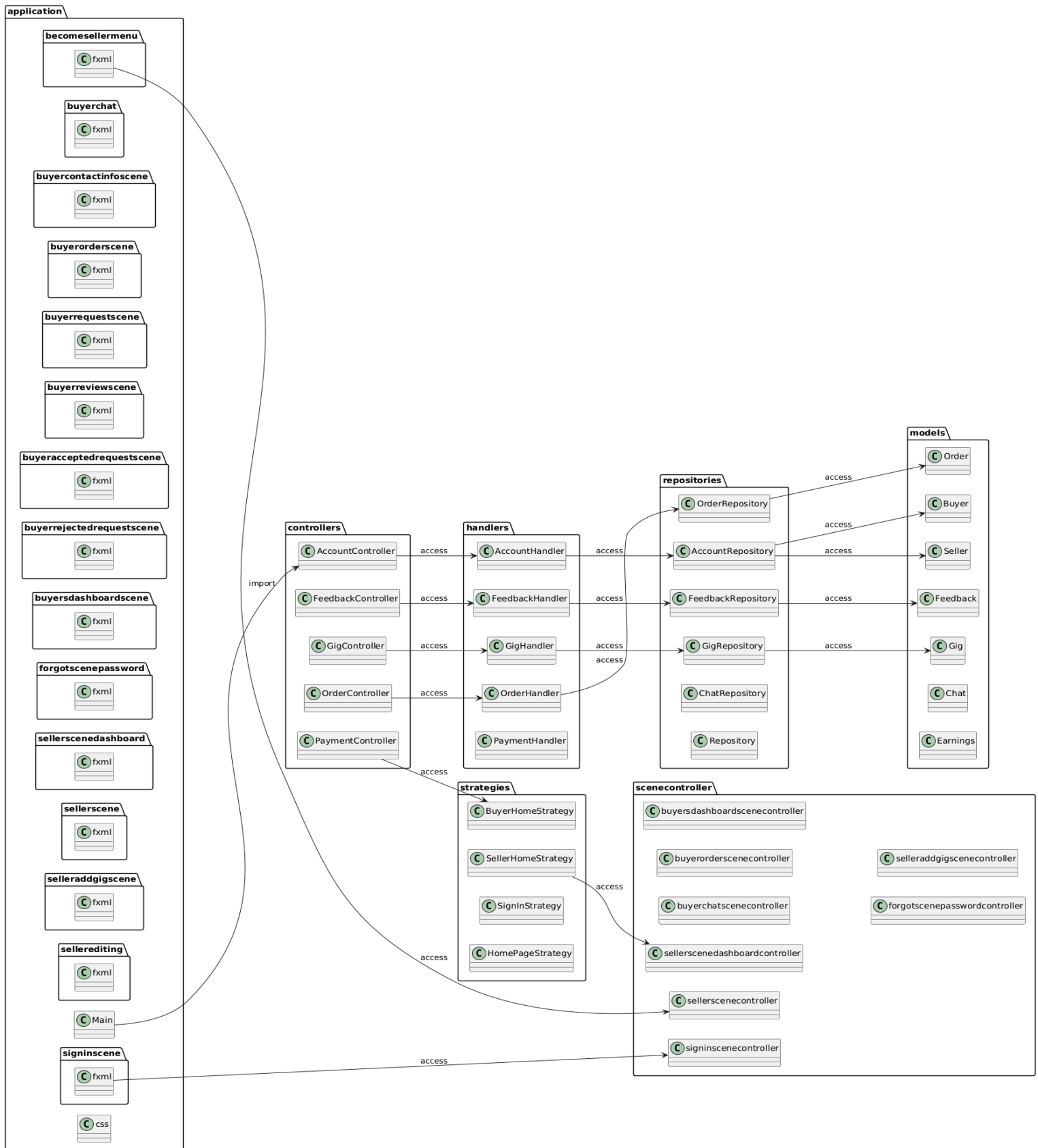**Response**

Offer(Description)

16.   Class Diagram

17.   Component Diagram

18. Package Diagram

19.   Deployment Diagram