



Software Engineering - Coursework– Sprint 2

Team Members ID:

Evan Balson - BAL18466416
Niwhar Amin - AMI23589889
Hala Bakhtiar - BAK23592238
Mahboubia Rezaei - REZ23579670

Roehampton University

Course Number: CMP020N204S

Lecturer: Dr Mamoon Humayun

24/02/2025

Table of Contents

Technical Components & Design	3
Overview	3
Project Documentation.....	4
Application Features	4
User Stories:.....	4
Use Case Diagram	5
Sprint 3 Specifications.....	6
Wireframes	6
Activity Diagrams	7
Sequence Diagram	9
Entity Relation Diagram	10
Physical Data Model.....	11
Class Diagram.....	12
Colour Scheme & Design.....	13
Kanban Board & Project Backlog	14
Sprint 3 Tickets.....	14
Meeting Minutes.....	15
Database Implementation	16
Appendix	17
Detailed Wireframe Views	17
Landing Page:	17
Login Page:	17
My Account Page:	18
Registration Page:	18
Add Outfit as A Seller:	19
Customer View: Listing Page:.....	19
My Cart:.....	20
Checkout View:	20
Outfit Advice Page:	21
Order History View:	21



Technical Components & Design

Overview

The FitXchange platform is designed to facilitate the exchange of outfits for various social and professional occasions through a robust, database-driven web application. This overview highlights the technical infrastructure and the application of systematic diagrams that guide the development process, ensuring that FitXchange is both scalable and user-centric.

Technical Infrastructure:

- **Front-End Technologies:** The client-facing side of FitXchange is built using HTML and Pug, a high-performance template engine, which together streamline the creation of HTML markup. JavaScript is extensively used to enhance the interactivity of the web pages, providing a dynamic user experience.
- **Back-End Technologies:** Node.js serves as the core back-end runtime environment, allowing for scalable server-side scripting. Docker containers are employed to encapsulate the application's environment, ensuring consistency across development, testing, and production landscapes.
- **Database Solution:** MySQL, a robust relational database management system, is used to manage structured data efficiently. It supports complex queries and transactions, making it ideal for handling the intricate data relationships within FitXchange.

Application of Systematic Diagrams:

- **Entity-Relationship Diagram (ERD):** Critical for visualizing the structured data in MySQL, the ERD helps define and refine the relational database schema, crucial for effective data management and integrity.
- **Use Case Diagram:** This diagram provides insights into user interactions with the system, clarifying the various functionalities available to different user roles within FitXchange and guiding the user interface design.
- **Sequence Diagrams:** Essential for detailing the interactions between objects over time, sequence diagrams depict the order of operations triggered by transactions, helping developers understand and implement the flow of processes accurately.
- **Activity Diagrams:** These diagrams are used to visualize the flow of operations within specific functionalities, helping to map out the logic behind user activities and system responses, which is vital for coding the back-end services.

Project Documentation

Application Features

User Stories:

Borrowing:

1. **As a user** seeking professional or special occasion attire, I want to **browse and filter a collection of outfits**, so that I can **quickly identify pieces that match my style and my event needs**.
2. **As a user**, I want **assurance that each outfit is authentic, well-maintained, and ethically sourced**, so that I can **feel confident in my choice and the integrity of the platform**.
3. **As a user**, I want an **effortless transaction process, including eco-friendly shipping or pickup options**, so that my **experience is both convenient and in line with ethical and environmental considerations**.
4. **As a user** unfamiliar with professional or formal dress codes, I want **access to fashion tips and trend insights**, so that I can **dress appropriately and stay stylish**.

Lending:

5. **As a user** with well-maintained items that are no longer in regular use, I want to **easily create detailed and visually appealing listings**, so that I can **share my outfits with others while ensuring that they're presented accurately**.
6. **As a user**, I want to **define lending conditions such as rental duration, pricing, deposit and care instructions**, so that I can **manage risk, protect my belongings and ensure proper use**.
7. **As a lender** with fashion experience, I want to **share styling tips and outfit recommendations alongside my listings**, so that **borrowers can trust my fashion advice and feel more confident in their choices**.

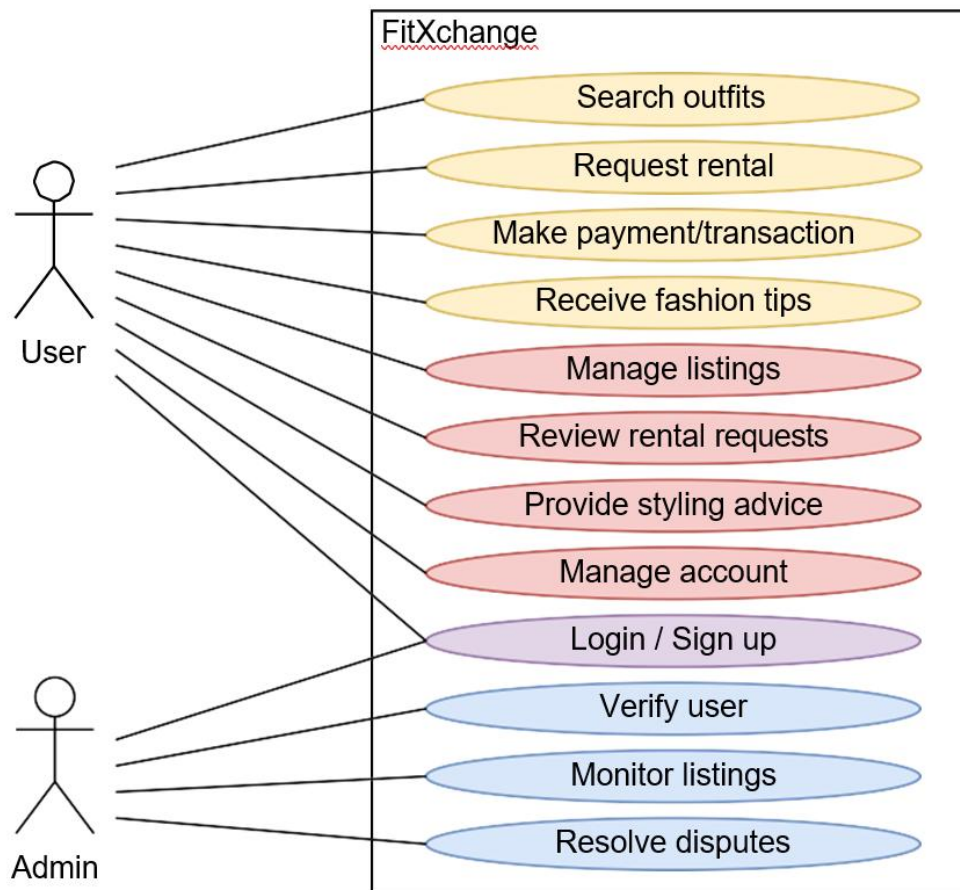
Administrating:

8. **As an administrator**, I want to **verify new users/user identities, monitor listings and oversee transactions**, so that **the platform remains safe, transparent and authentic for all users**.
9. **As an administrator**, I want to **effectively manage conflicts and ensure adherence to platform guidelines such as ethical, environmental and intellectual property standards**, so that **disputes are resolved fairly, and the platforms reputation is maintained**.

Login / Sign up (Sign up only for user):

10. **As a user** of the system, I want to **be able to create and log in to my account securely while also being notified of the terms and conditions of the service**, so that I can **make an account and safely become a functioning member of the platform**.

Use Case Diagram



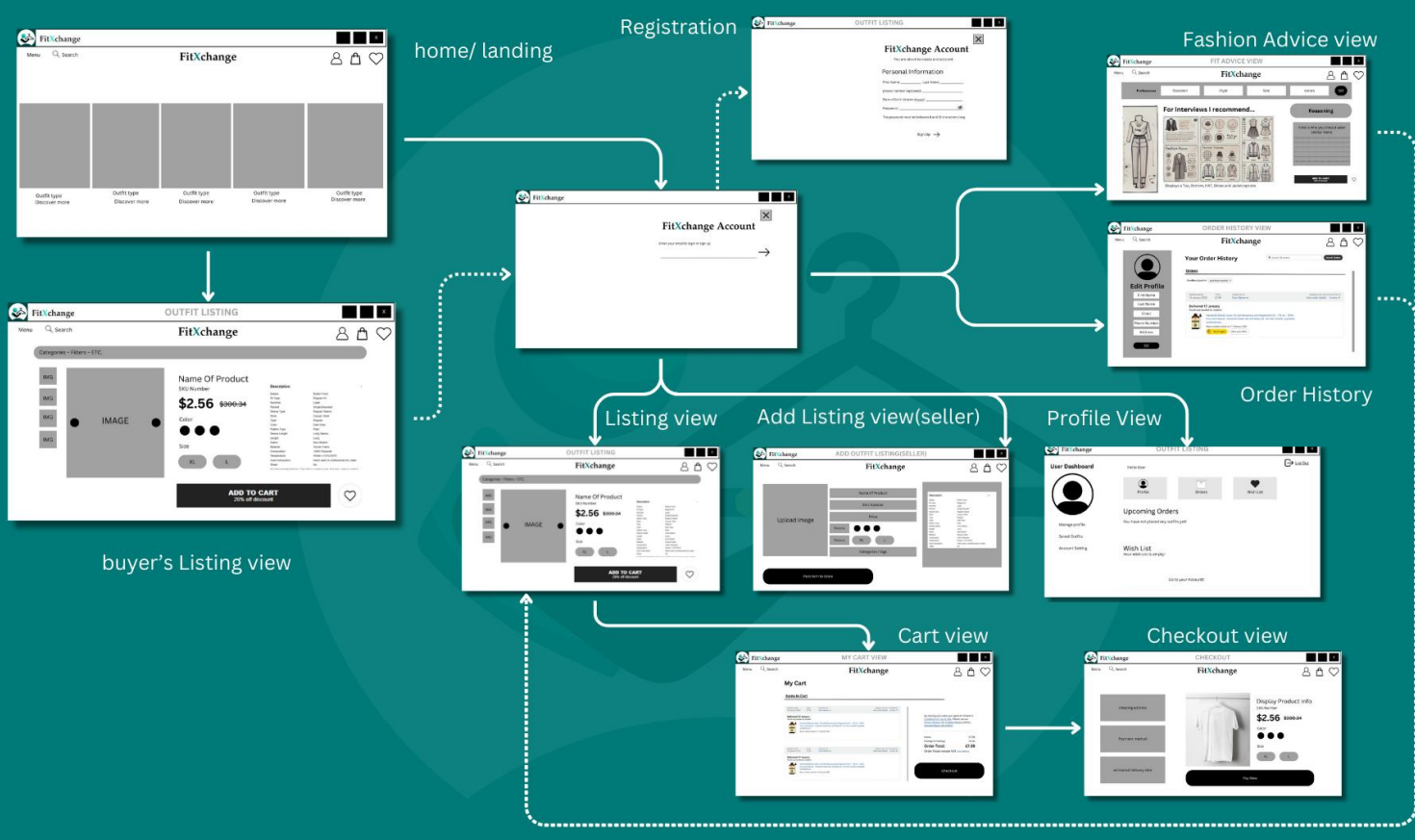
Rationale for User Role Generalization:

- Generalized User Roles:** In FitXchange, while there are theoretically three types of interactions—buying, selling, and administrative—these roles share common attributes and behaviors that allow them to be generalized into two main roles: User and Administrator.
- User Role:** This role encompasses both buyers and sellers. The decision to generalize these roles into a single 'User' actor in the use case diagram stems from the overlapping functionalities they share, such as browsing items, managing accounts, and processing transactions. A user on FitXchange can seamlessly switch between buying and selling outfits, reflecting the dual nature of their engagement with the platform. This design simplifies the system architecture and reduces complexity by eliminating redundant processes.

Sprint 3 Specifications

Wireframes

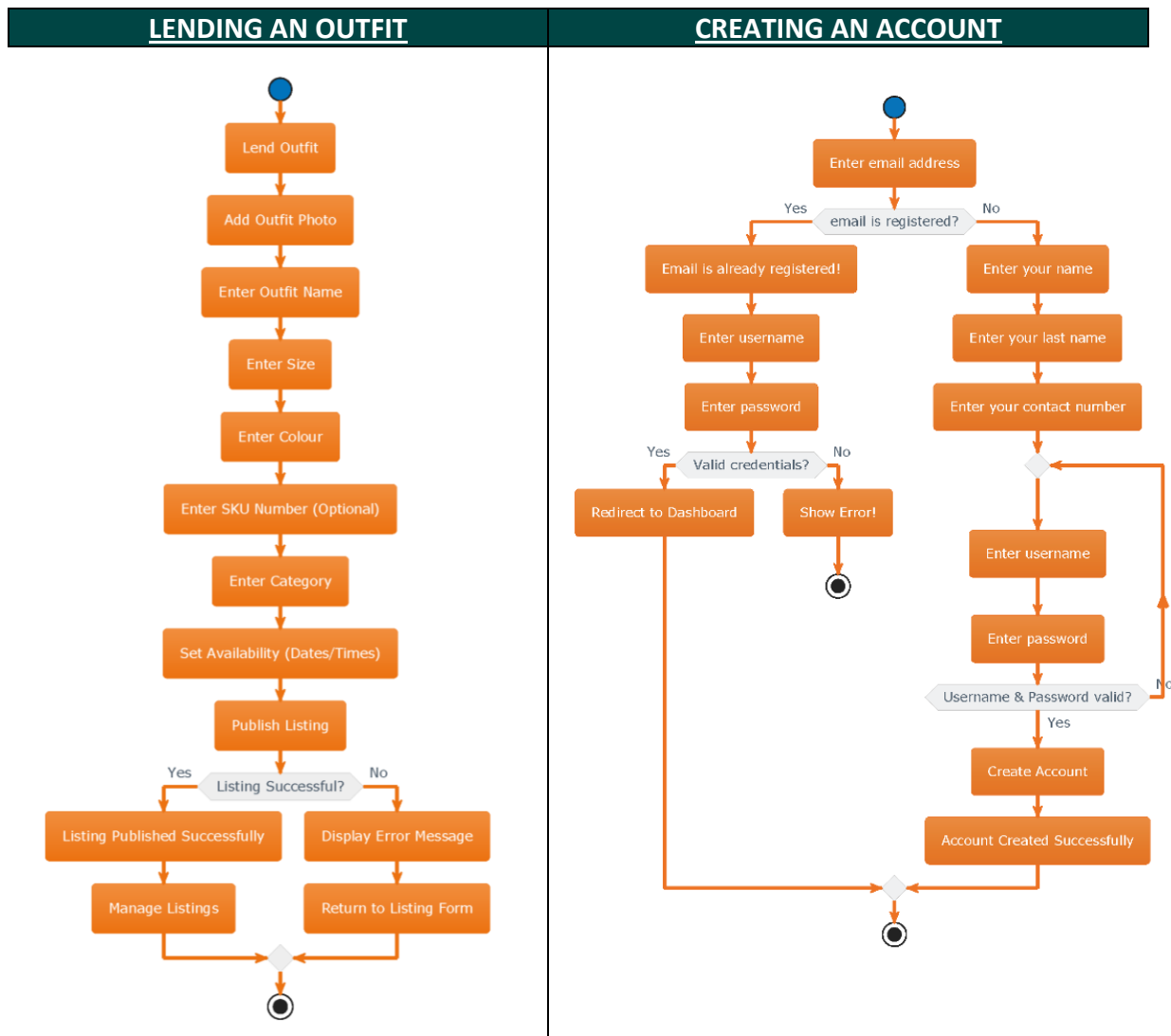
For FitXchange, wireframes were instrumental in visualizing the seamless transition between the roles of buyers and sellers, highlighting dual functionality within a single, cohesive user interface. This approach supports the platform's goal of enabling users to effortlessly switch between buying and selling outfits, enhancing user experience and platform efficiency.



Development Integration:

- **Guidance for UI Development:** The wireframes are our a guide to building the front-end code, ensuring that all functional and design requirements are met.
- **Iterative Design and Feedback:** Wireframes will be revised based on feedback from usability testing sessions and stakeholder reviews. This iterative process helps refine the interface for better user engagement and functionality.

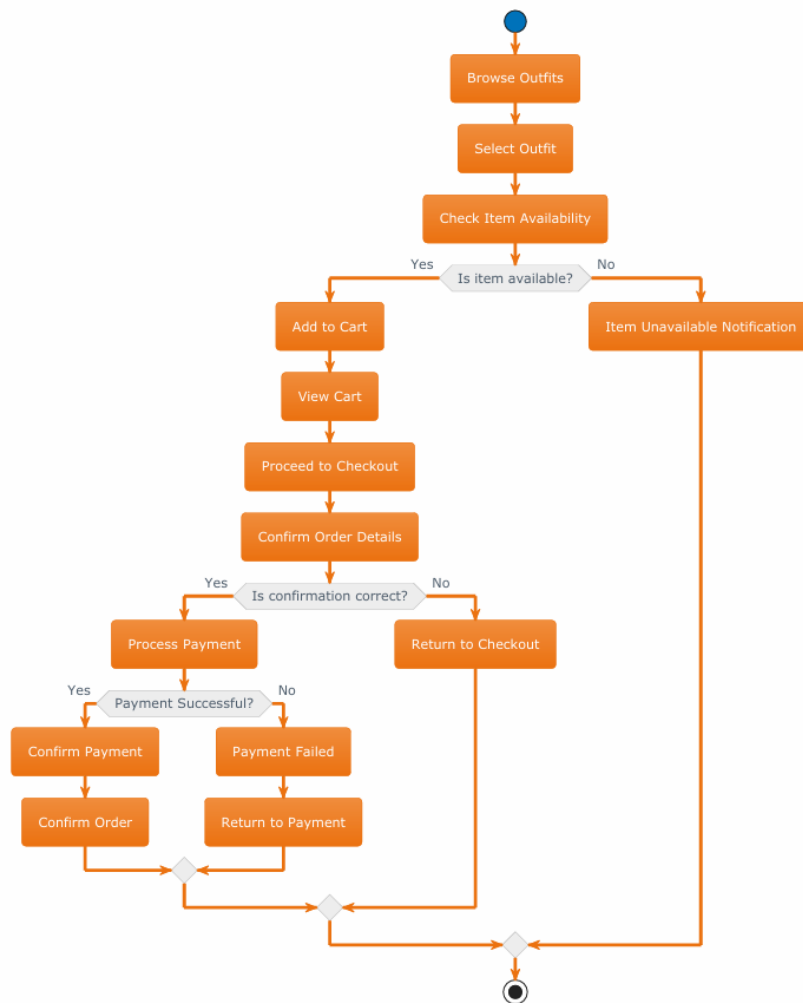
Activity Diagrams



Lending an Outfit: This diagram simplifies the process of listing an outfit for lending. It delineates each step, from user authentication to the final listing approval, pinpointing areas where steps can be consolidated or removed to enhance ease of use. This ensures that users can list outfits quickly and without hassle, increasing the likelihood of frequent user participation and interaction.

Creating an Account: The activity diagram for creating an account highlights the necessary steps a new user must take to successfully register on the platform. By visualizing this process, we can identify and eliminate any redundant steps and ensure that the account creation is as streamlined as possible, reducing barriers to entry and encouraging more user sign-ups.

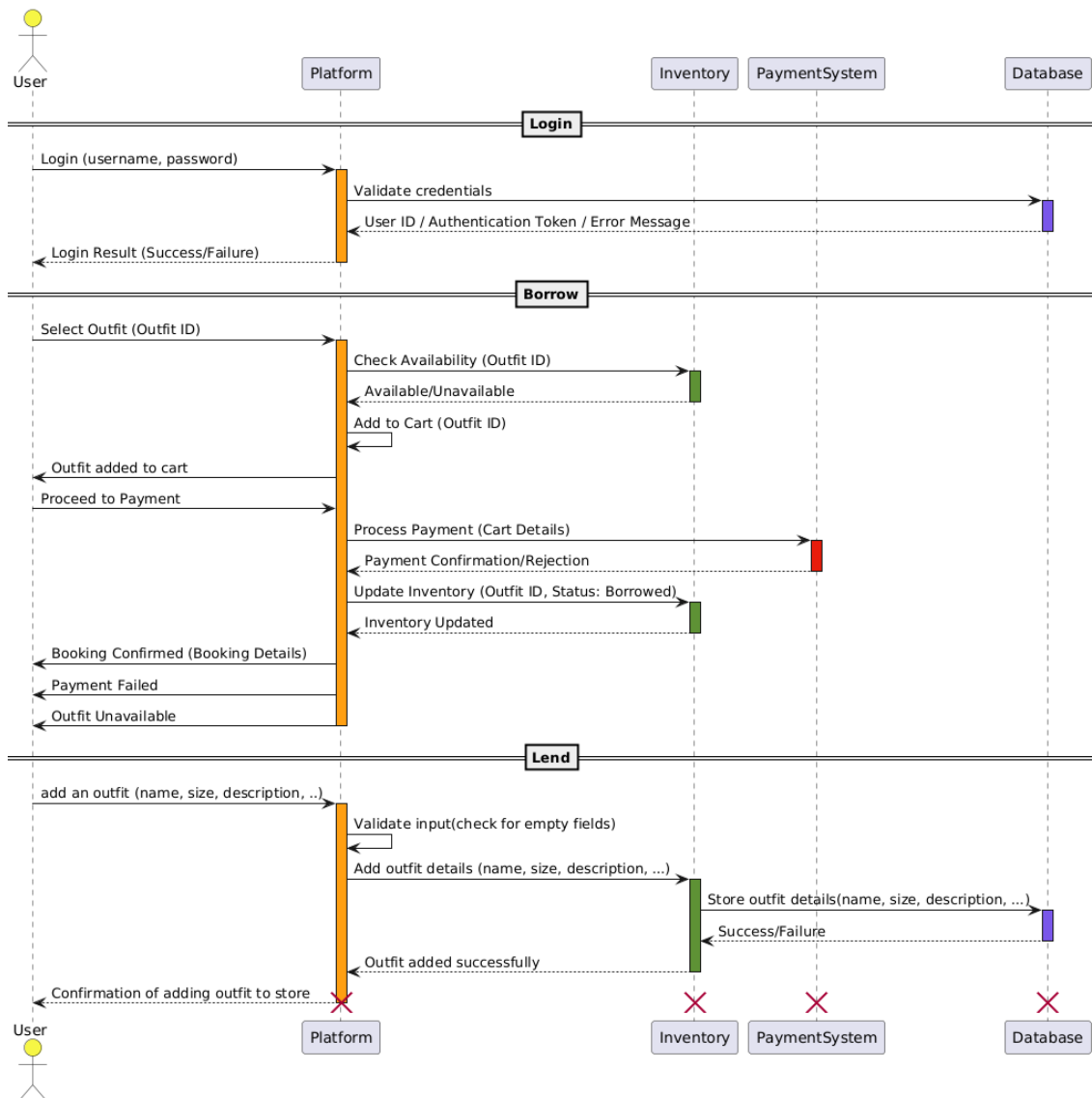
PURCHASING AN OUTFIT



Purchasing an Outfit:

The activity diagram for purchasing an outfit illustrates the decision points and support mechanisms involved in the buying process. It highlights opportunities to integrate helpful tooltips and contextual help, guiding users through selections based on size, color, and availability, and providing real-time assistance. This diagram also shows where we can offer personalized recommendations, enhancing the shopping experience by suggesting related items or accessories, thereby increasing transaction value and customer satisfaction.

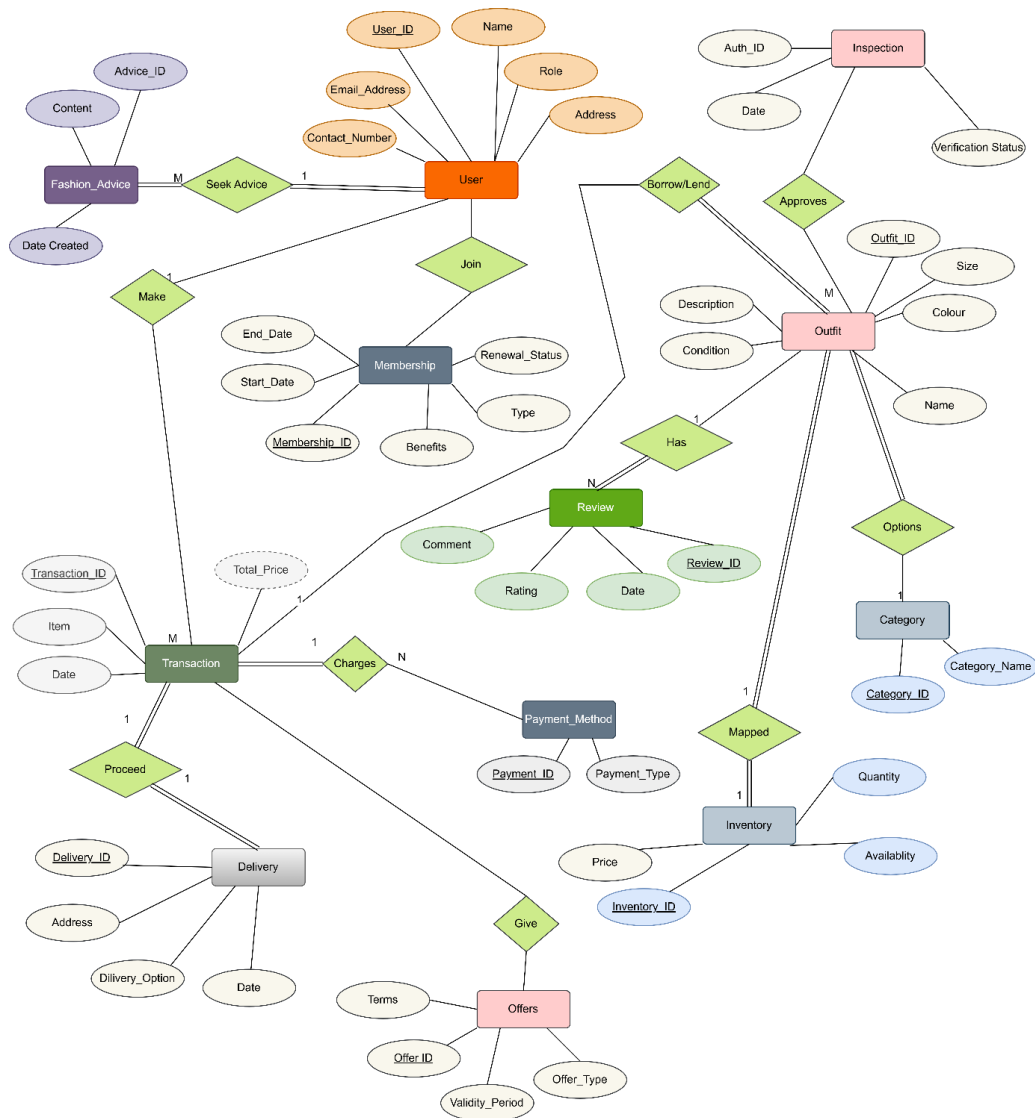
Sequence Diagram



This diagram effectively visualizes how FitXchange processes unfold over time, detailing the communication between objects and the sequence of events that lead to a completed transaction or action. They are instrumental in identifying and approaching dependencies between system components, ensuring that each step in a user interaction is executed efficiently.

Given the nature of our service, where users often rent clothes under time constraints, such as for upcoming events or meetings, it is critical that our system processes requests quickly and reliably. Sequence diagrams help us ensure that our platform meets these user expectations by allowing us to streamline operations and minimize response times. This focus on efficiency is not just a technical requirement but a fundamental aspect of user satisfaction, making the sequence diagram a key tool in our development process to enhance overall user experience.

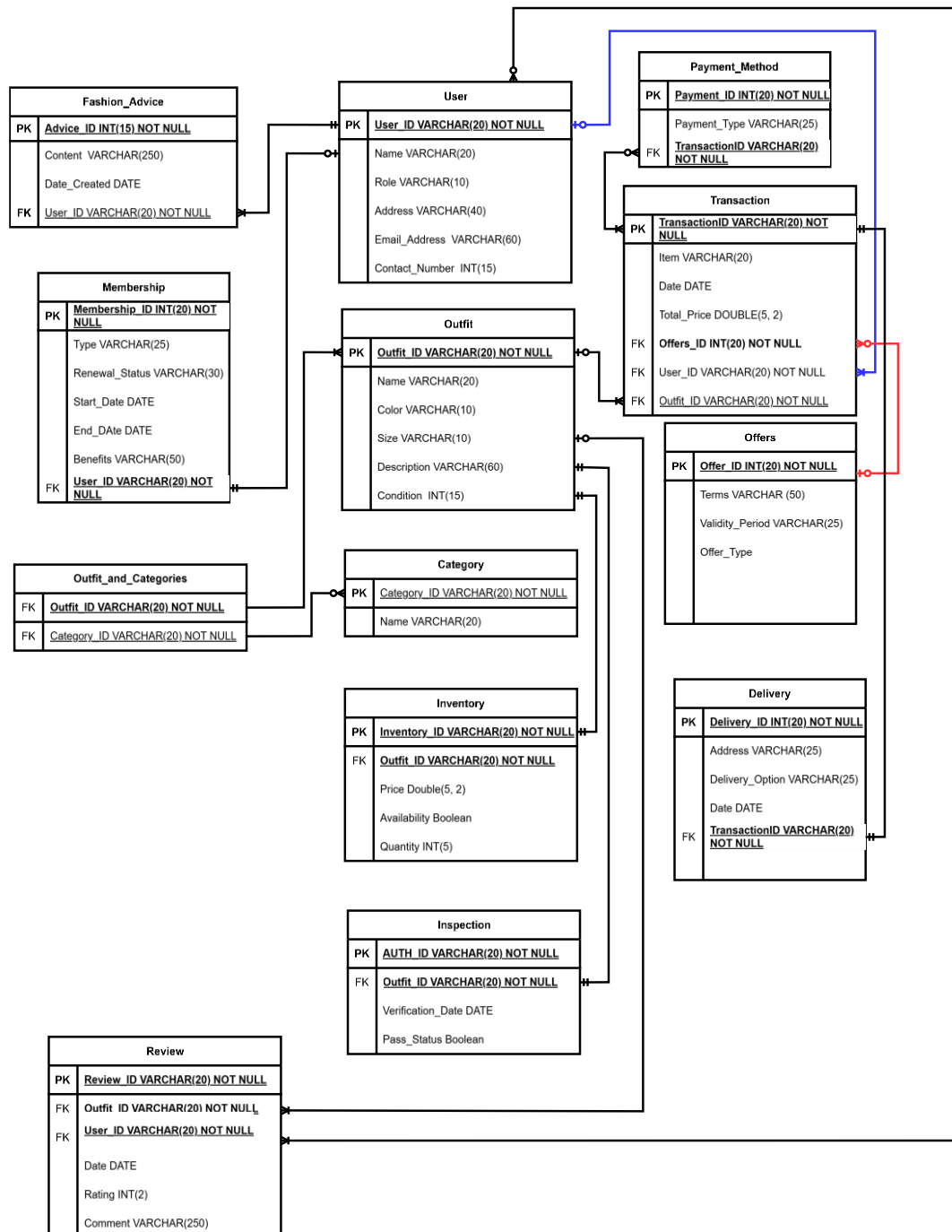
Entity Relation Diagram



The Entity-Relationship Diagram (ERD) for FitXchange outlines the database schema, ensuring that data is organized and accessed efficiently for optimal performance.

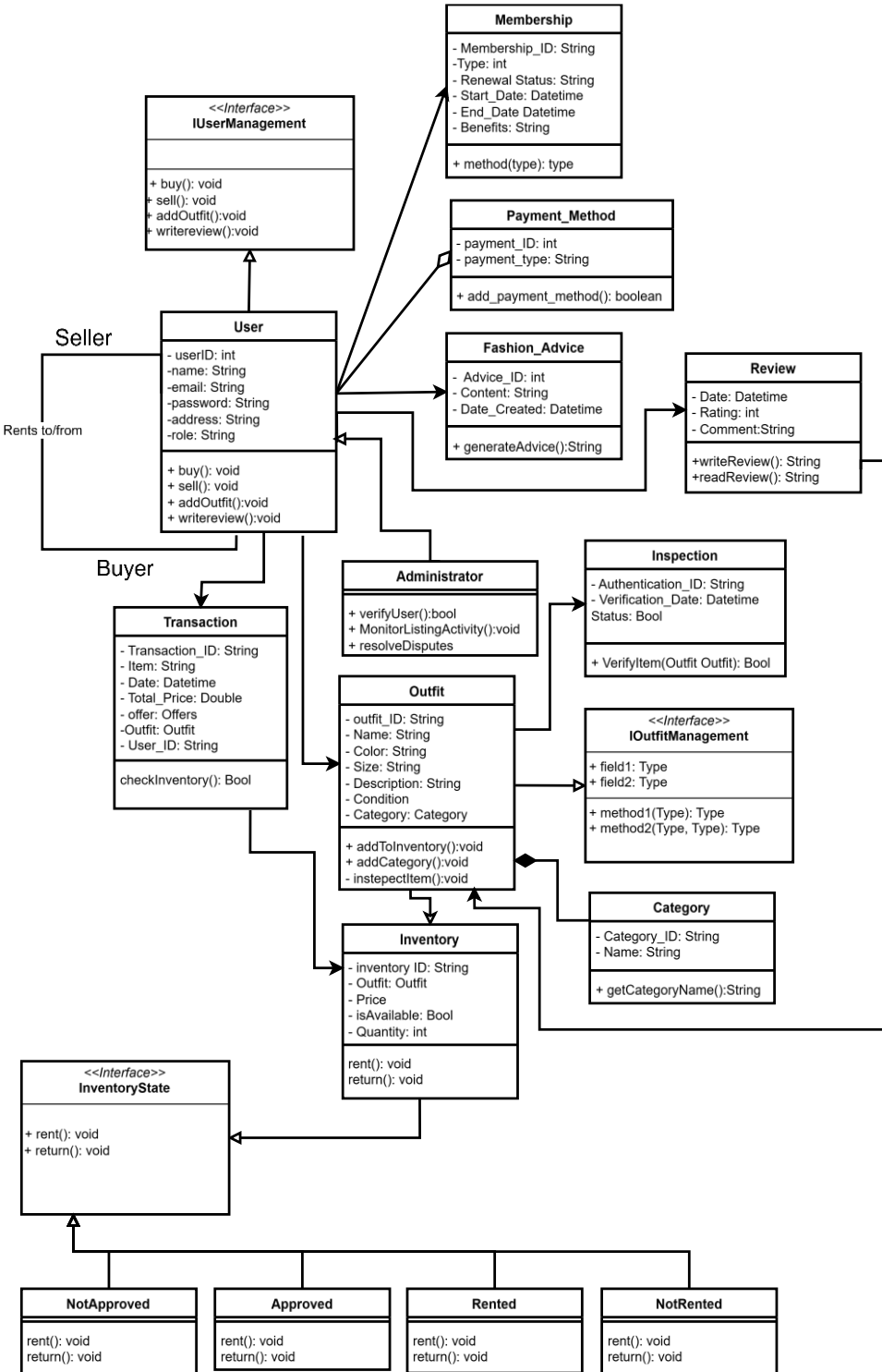
We anticipate that users may frequently engage in time-sensitive rental transactions for events or meetings. Therefore, it is imperative that our database structure facilitates quick and reliable data retrieval. The ERD plays a pivotal role in optimizing these processes by clearly defining entity relationships, which helps in streamlining database operations and enhancing query performance. This focus on efficient data management is not merely a technicality but a core aspect of user satisfaction, making the ERD a vital tool in our development arsenal to ensure a superior user experience.

Physical Data Model





FitXchange





Colour Scheme & Design

LOGO



LOGO



Brand Scheme



INDIVIDUAL FONTS



Aa

Palantino Bold

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890

Aa

Calibri (MS)

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
1234567890



Kanban Board & Project Backlog

Software Engineering Project							
<div> <div>Tasks</div> <div>Kanban Board</div> <div>Current iteration</div> <div>Next iteration</div> <div>Prioritized backlog</div> <div>In review</div> <div>My items</div> <div>New view</div> </div>							
<div>Filter by keyword or by field</div> <div>Discard Save</div>							
Title	Assignees	Status	Start Date	End Date	Priority		
25 Set up development environment using MySQL, Express, and PUG #26	BAK23592238, Eva...	Backlog					
26 Implement initial requirements, ensure database connectivity #27	BAK23592238, Eva...	Backlog					
27 User List Page #28	M12rezaei	Backlog					
28 User Profile Page #29	BAK23592238	Backlog					
29 Listing Page #30	Evan-Balson	Backlog					
30 Detail Page #31	ninoa2000	Backlog					
31 Tags/Categories #32	BAK23592238, Eva...	Backlog					
32 Implement Data for USER in MySQL #33	BAK23592238	Backlog					
33 Implement Data for Outfit in MySQL #34	M12rezaei	Backlog					
34 Implement Data for CATEGORY in MySQL #35	M12rezaei	Backlog					
35 Implement Data for INVENTORY in MySQL #36	Evan-Balson	Backlog					
36 Implement Data for INSPECTION in MySQL #37	Evan-Balson	Backlog					
37 Implement Data for PAYMENT_METHOD in MySQL #38	ninoa2000	Backlog					

Backlog 37
This item hasn't been started
SE-Spring-Project #58 Implement NotApproved Class
SE-Spring-Project #59 Implement Approved Class
SE-Spring-Project #60 Implement Rented Class
SE-Spring-Project #61 Implement NotRented Class
SE-Spring-Project #62 Implement Inventory Class
+ Add item

Done (Sprint2) 8
SE-Spring-Project #20 Wireframes
SE-Spring-Project #25 Features for the application defined as user stories
SE-Spring-Project #22 Sequence Diagram
SE-Spring-Project #23 ... ERD Diagram
SE-Spring-Project #19 Use Case Diagram

Sprint 3 Tickets

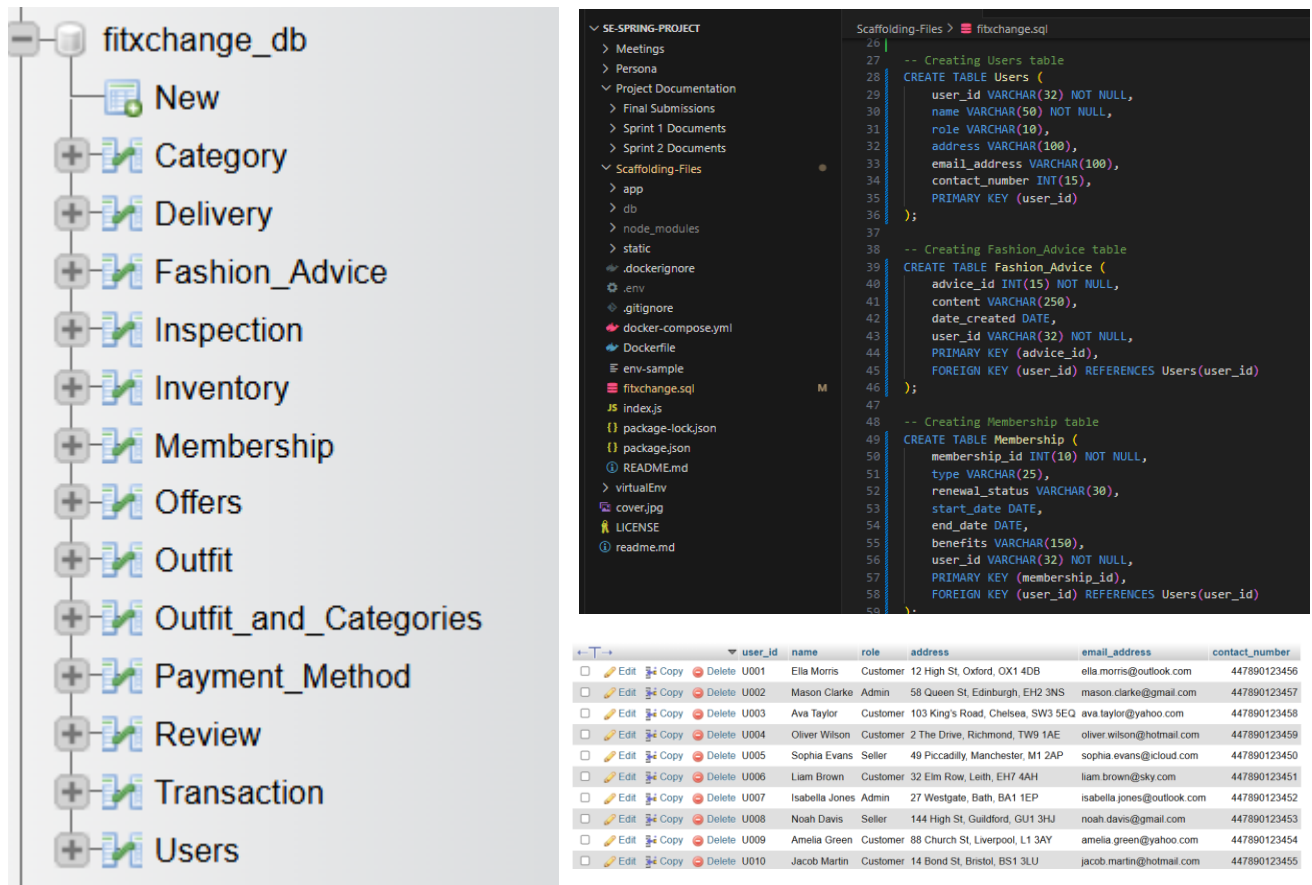
Leveraging insights gained from earlier phases such as wireframes and entity-relationship diagrams, the team has effectively identified critical items for the project backlog. These include implementing the database structure, determining the web pages that need to be developed, assigning classes for backend functionality, and formulating the JavaScript queries necessary for database interactions.

This proactive organization on the Kanban Board ensures that each aspect of the development is clearly visualized and strategically aligned for efficient execution, facilitating a smooth transition into the upcoming sprint with well-defined priorities and tasks.

Meeting Minutes

Date and Time	19/02/2025 at 19:00, 22/02/2025 at 20:00
Project Name	FitXchange
Meeting Goal	<ul style="list-style-type: none"> • Discuss the purpose of each diagram based on user stories • Share feedback on provided diagrams • Modify and prepare new updates
Facilitator	Hala Bakhtiar
Note taker	Mahboub Rezaei
Attendees	<ul style="list-style-type: none"> • 19/02/2025: Hala Bakhtiar, Evan Balson, Mahbouba Rezaei • 22/02/2025: Hala Bakhtiar, Mahbouba Rezaei, Evan Balson, Niwhar Amin
Roundtable Updates (each group member to contribute)	
Discussion points	<ul style="list-style-type: none"> • Purpose of each diagram based on user stories. • Review and provide feedback on existing diagrams • Identify areas for improvement and modifications • Assign updates and next steps
Actions (list tasks and assign a group member)	<ul style="list-style-type: none"> • Design Wireframes -> Evan, Mahbouba • Features of application -> Evan • Class Diagram -> Evan • ER Diagram -> Evan • Use Case Diagram -> Hala • User Stories -> Hala • Activity Diagram -> Mahbouba • Sequence Diagram -> Niwhar

Database Implementation



The screenshot displays the database implementation for the FitXchange project. On the left, a file explorer shows the project structure, including a 'fitxchange_db' database and various application files. The central pane shows the 'fitxchange.sql' file, which contains SQL scripts for creating three tables: Users, Fashion_Advice, and Membership. The right pane shows a table view of the 'Users' table, listing 10 users with their details.

```

-- Creating Users table
CREATE TABLE Users (
  user_id VARCHAR(32) NOT NULL,
  name VARCHAR(50) NOT NULL,
  role VARCHAR(10),
  address VARCHAR(100),
  email_address VARCHAR(100),
  contact_number INT(15),
  PRIMARY KEY (user_id)
);

-- Creating Fashion_Advice table
CREATE TABLE Fashion_Advice (
  advice_id INT(15) NOT NULL,
  content VARCHAR(250),
  date_created DATE,
  user_id VARCHAR(32) NOT NULL,
  PRIMARY KEY (advice_id),
  FOREIGN KEY (user_id) REFERENCES Users(user_id)
);

-- Creating Membership table
CREATE TABLE Membership (
  membership_id INT(10) NOT NULL,
  type VARCHAR(25),
  renewal_status VARCHAR(30),
  start_date DATE,
  end_date DATE,
  benefits VARCHAR(150),
  user_id VARCHAR(32) NOT NULL,
  PRIMARY KEY (membership_id),
  FOREIGN KEY (user_id) REFERENCES Users(user_id)
);

```

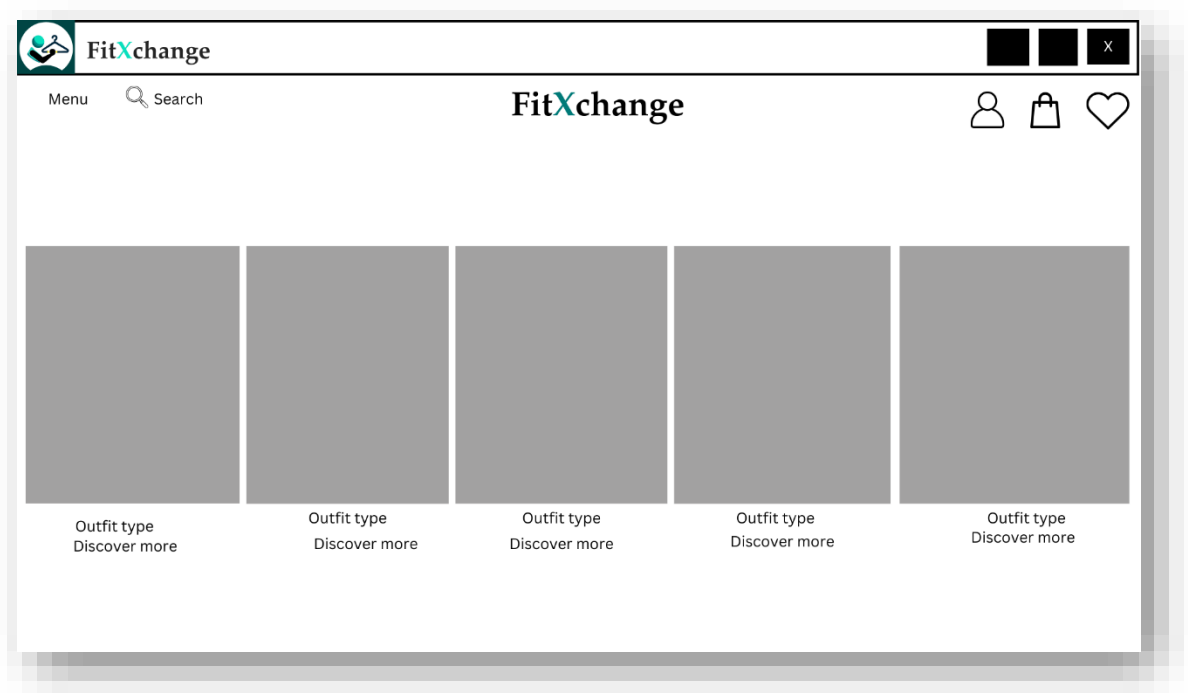
	user_id	name	role	address	email_address	contact_number
<input type="checkbox"/>	U001	Ella Morris	Customer	12 High St, Oxford, OX1 4DB	ella.morris@outlook.com	447890123456
<input type="checkbox"/>	U002	Mason Clarke	Admin	58 Queen St, Edinburgh, EH2 3NS	mason.clarke@gmail.com	447890123457
<input type="checkbox"/>	U003	Ava Taylor	Customer	103 King's Road, Chelsea, SW3 5EQ	ava.taylor@yahoo.com	447890123458
<input type="checkbox"/>	U004	Oliver Wilson	Customer	2 The Drive, Richmond, TW9 1AE	oliver.wilson@hotmail.com	447890123459
<input type="checkbox"/>	U005	Sophia Evans	Seller	49 Piccadilly, Manchester, M1 2AP	sophia.evans@icloud.com	447890123450
<input type="checkbox"/>	U006	Liam Brown	Customer	32 Elm Row, Leith, EH7 4AH	liam.brown@sky.com	447890123451
<input type="checkbox"/>	U007	Isabella Jones	Admin	27 Westgate, Bath, BA1 1EP	isabella.jones@outlook.com	447890123452
<input type="checkbox"/>	U008	Noah Davis	Seller	144 High St, Guildford, GU1 3JH	noah.davis@gmail.com	447890123453
<input type="checkbox"/>	U009	Amelia Green	Customer	88 Church St, Liverpool, L1 3AY	amelia.green@yahoo.com	447890123454
<input type="checkbox"/>	U010	Jacob Martin	Customer	14 Bond St, Bristol, BS1 3LU	jacob.martin@hotmail.com	447890123455

In preparation for Sprint 3, the FitXchange team has prioritized the early implementation of the database using MySQL, Express, and Pug within a Dockerized environment to establish a solid foundation for the development phase. This approach ensures that the core data structures are in place, allowing the team to deliver a functional prototype by the end of this sprint. Early database setup enables the focus for Sprint 4 to shift towards integrating advanced features, such as AI-based outfit advice and enhancing the user interface design. The checklist for Sprint 3 includes crucial development tasks like setting up the development environment, ensuring database connectivity, and implementing the initial user and item management features. Additionally, the sprint involves comprehensive documentation efforts, detailing user stories, database schema, task allocations, and sprint progress, ensuring that all aspects of the sprint are well-documented and transparently shared with stakeholders. This structured approach not only facilitates efficient development workflows but also aligns with our goal to enhance the platform's functionality progressively, ensuring readiness for subsequent sprints.

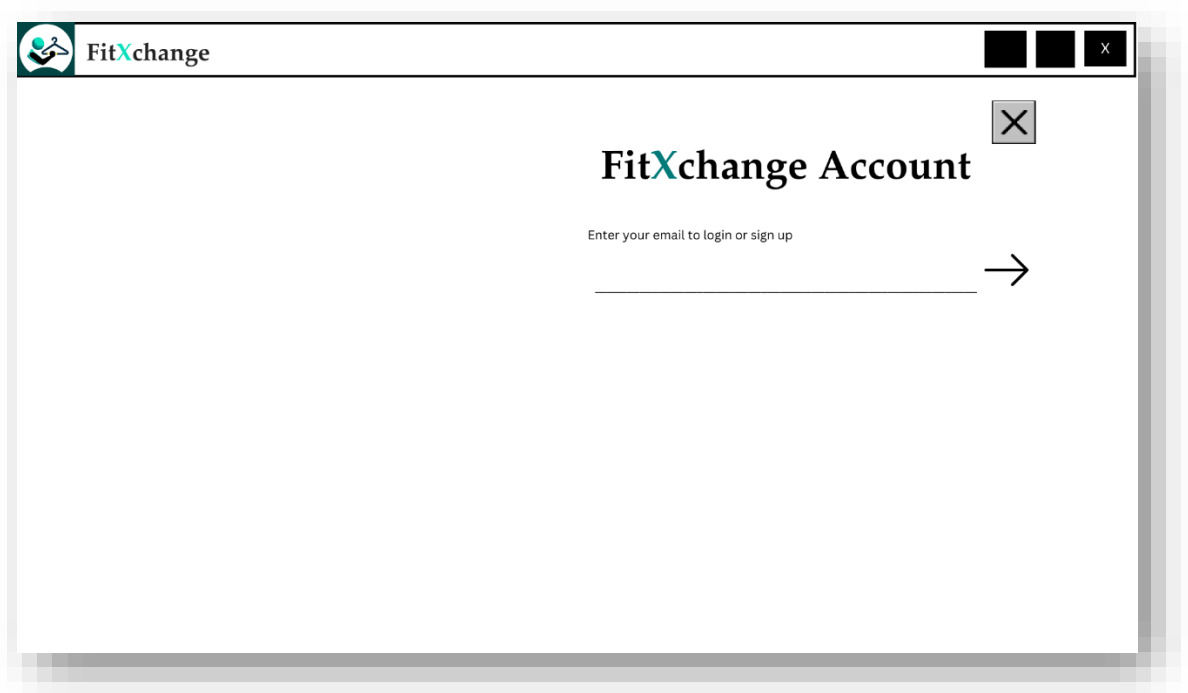
Appendix

Detailed Wireframe Views

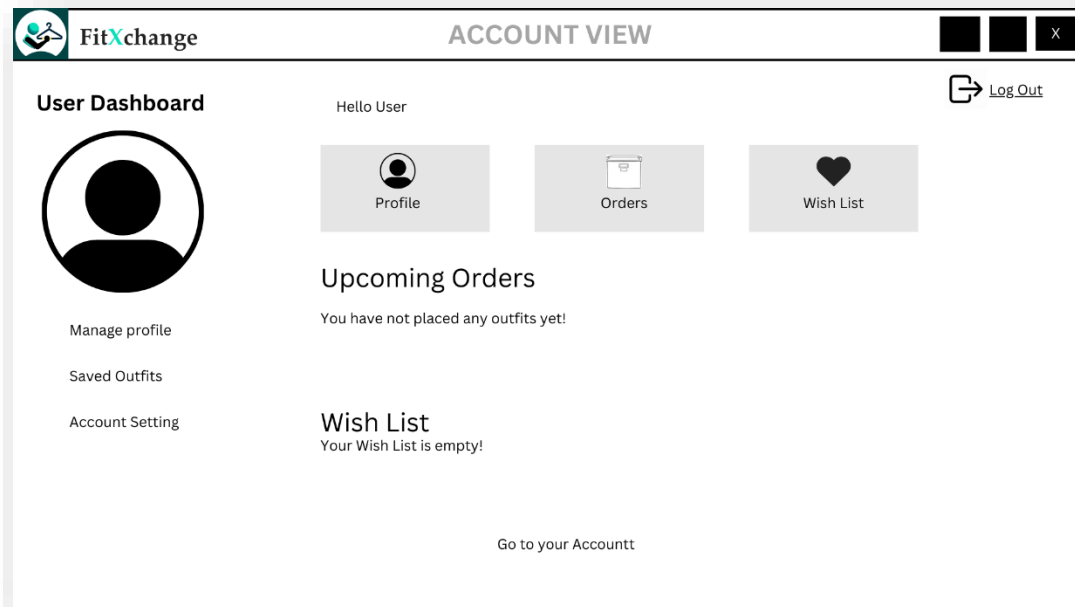
Landing Page:



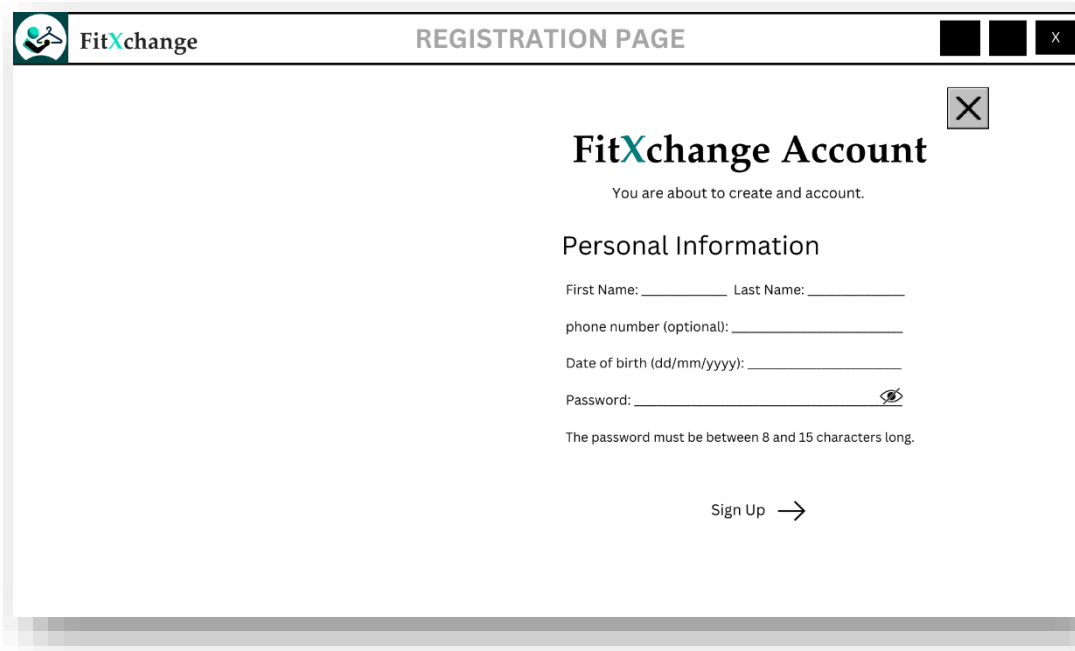
Login Page:




My Account Page:



Registration Page:



Add Outfit as A Seller:




**FitXchange**

ADD OUTFIT LISTING(SELLER)

Menu

Search

FitXchange



Upload Image

Name Of Product

SKU Number

Price

Colors

Colors


Categories / tags

Post Item to Store

Description

Details:	Button Front
Fit Type:	Regular Fit
Neckline:	Lapel
Placket:	Single Breasted
Sleeve Type:	Regular Sleeve
Style:	Casual, Work
Type:	Regular
Color:	Dark Grey
Pattern Type:	Plain
Sleeve Length:	Long Sleeve
Length:	Long
Fabric:	Non-Stretch
Material:	Woven Fabric
Composition:	100% Polyester
Temperature:	Winter (<10°C/50°F)
Care Instructions:	Hand wash or professional dry clean
Sheer:	No

Customer View: Listing Page:




**FitXchange**

OUTFIT LISTING

Menu

Search

FitXchange



Categories ~ Filters ~ ETC.

IMG

IMG

IMG

IMG

IMAGE

Name Of Product

SKU Number

\$2.56 ~~\$300.34~~

Color

Size

XL

L

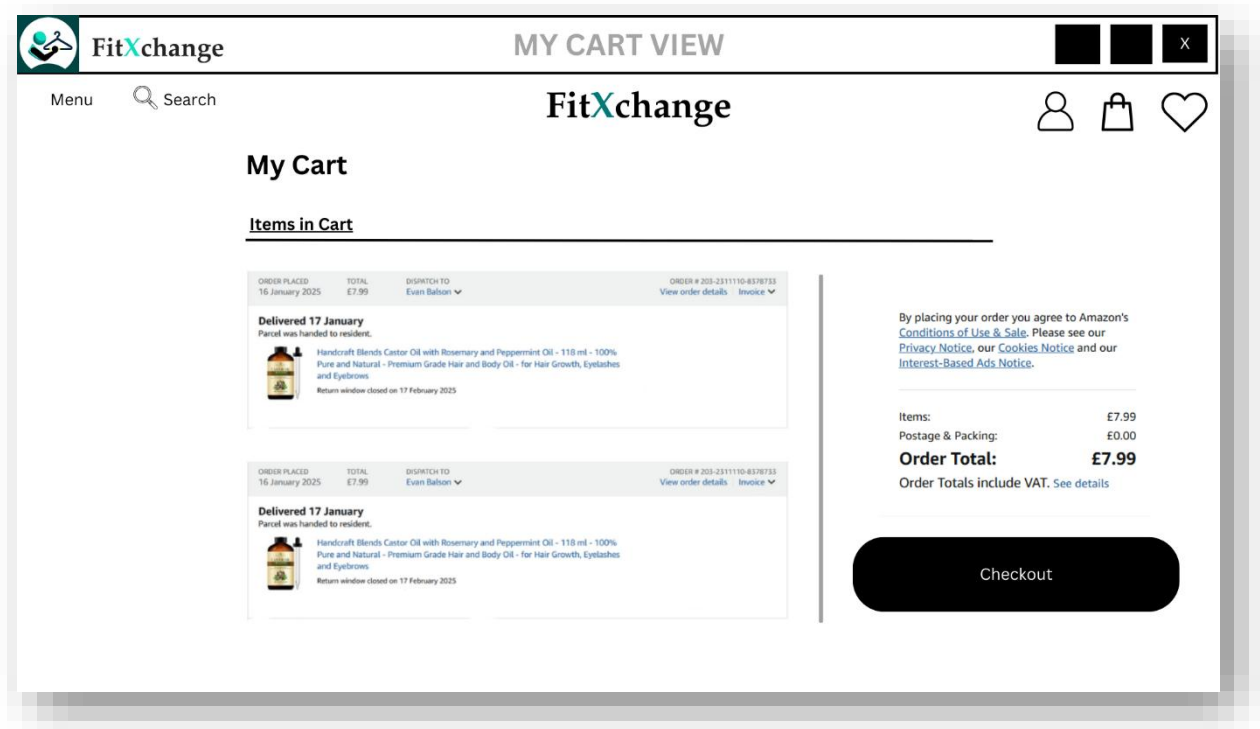
ADD TO CART

26% off discount

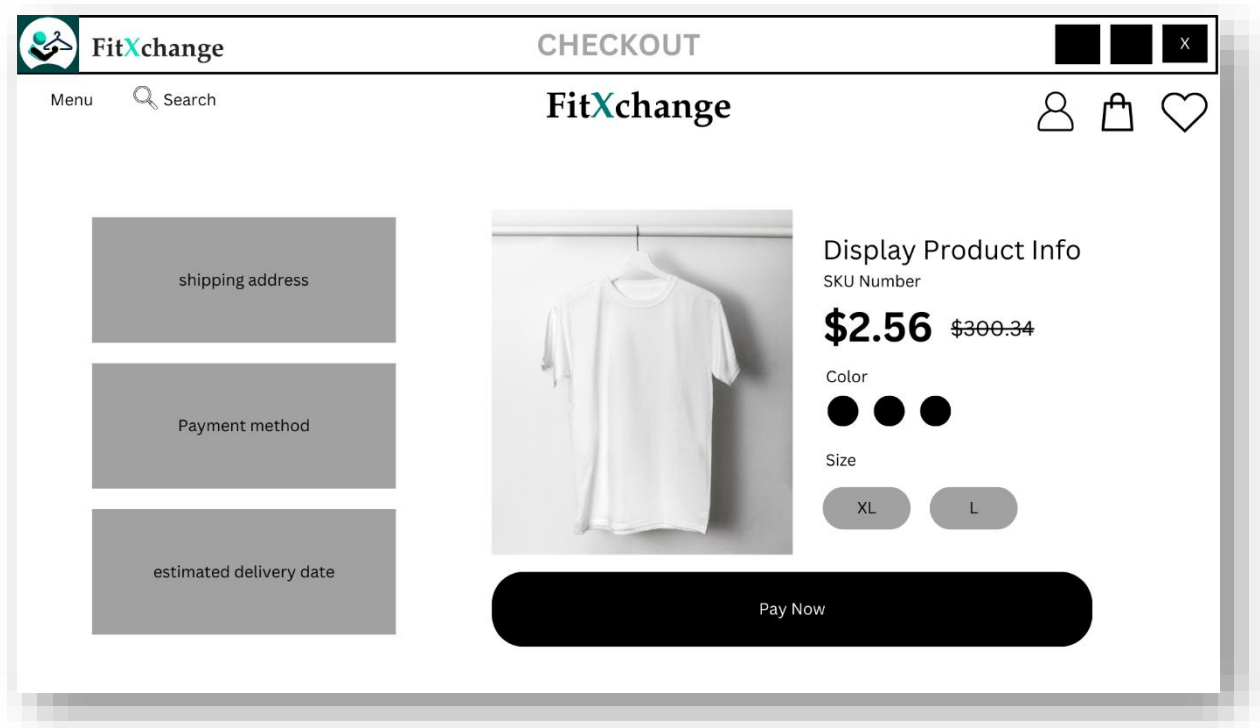
Description

Details:	Button Front
Fit Type:	Regular Fit
Neckline:	Lapel
Placket:	Single Breasted
Sleeve Type:	Regular Sleeve
Style:	Casual, Work
Type:	Regular
Color:	Dark Grey
Pattern Type:	Plain
Sleeve Length:	Long Sleeve
Length:	Long
Fabric:	Non-Stretch
Material:	Woven Fabric
Composition:	100% Polyester
Temperature:	Winter (<10°C/50°F)
Care Instructions:	Hand wash or professional dry clean
Sheer:	No

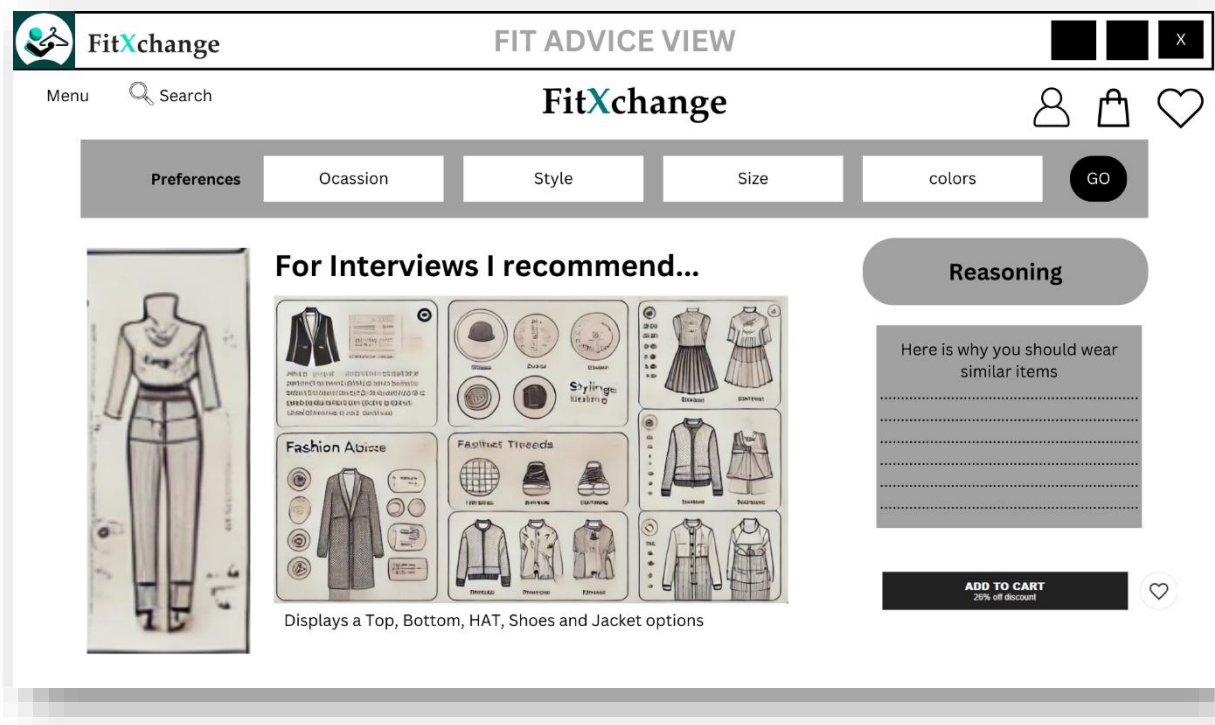
My Cart:



Checkout View:



Outfit Advice Page:



Order History View:

