Exchange4Students Milestone 2 Report

Project Design using UML Diagrams

SSW 322 Group 2

James Grant, Justin Phan, Jacob Gelman-Funk, Edmund Yuen

March 11, 2025

Introduction

Our project, Exchange4Students, aims to serve as a practical software tool for Stevens students who are seeking to exchange items on campus. After previously building a prototype and performing usability testing on other Stevens students, our group is now visualizing the scope of our project via several UML diagrams focused on our users' intended experience. Once the requirements of the Exchange4Students system were made clear between group members, we started creating Use Case diagrams to help identify main actors or processes of our system and to narrow down specific user experiences to be implemented. After use cases were created, our group then focused on creating both the activity diagram, to better model the system's functionality on a larger scale, and the class diagram, which models the structure and interfaces performed by the system. Finally, we added an optional diagram, the state diagrams, primarily modelling the listing and order processes.

Team Collaboration

- James Grant: During this project, I contributed to:
    - Creating Use Case diagrams and descriptions for UC1, UC2, and UC4 based on previously discussed requirements for Exchange4Students
    - Documenting our team's progress via this report and a presentation given on March 11, 2025.
- Justin Phan: During this project, I:
    - Created the class diagram
    - Provided a description of all of the classes in the diagram in this report
    - Prepared the presentation
- Jacob Gelman-Funk:

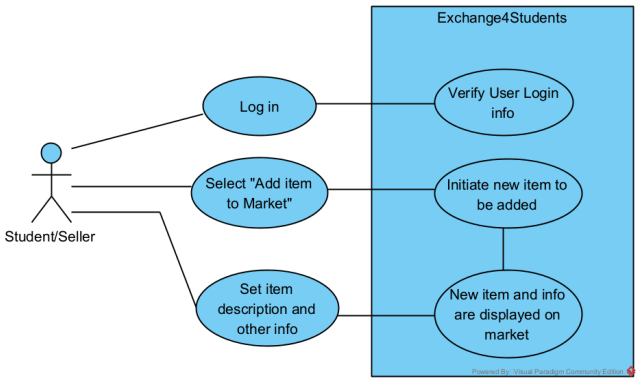
- Created Use Case diagrams for the search and filtering of the different categories of items (UC3).

- Edmund Yuen: For this milestone I contributed:
  - Activity Diagram to model the flow between buyer/seller and the system
  - State diagram modelling the item listing and order processing processes for Exchange4Students
  - Documented all changes on report and prepared for presentation

Use Case Diagrams

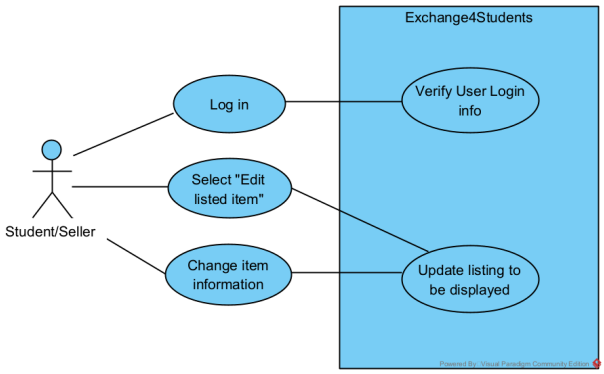
Below are 4 use cases of Exchange4Students agreed upon by our group, accompanied with various use case diagrams for specific functionality of each use case.

1. UC1 - Listing and Managing Items for Selling
   1.1. UC1.1 - Listing new item



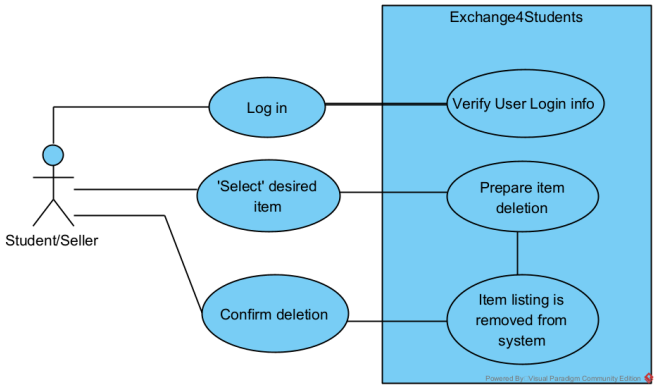
When listing a new item, the seller will first have to log into an account to be verified by our system. The system will then allow the user to add and customize an Item Listing to the Exchange4Students marketplace, with the user's account linked to the item as the seller.

1.2. UC1.2 - Edit item listing



When editing an existing Item Listing, users can navigate to their seller list after logging in and select ‘Edit listed item’. The system will then allow the user to make changes and update the edited information on the listing displayed on the marketplace.

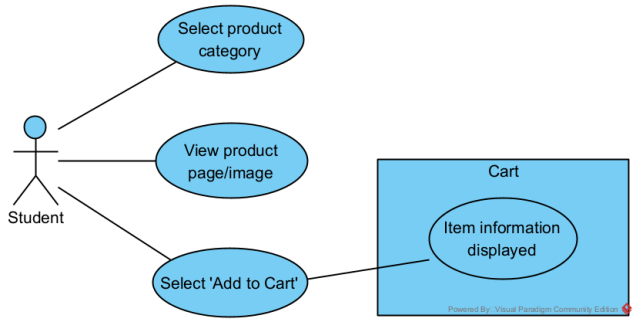
1.3. UC1.3 - Deleting a listing



When deleting a listing, users can navigate to their seller list after logging in, select the desired item to be deleted and confirm their choice. The system will then remove the listing from its marketplace and erase the listing’s data.

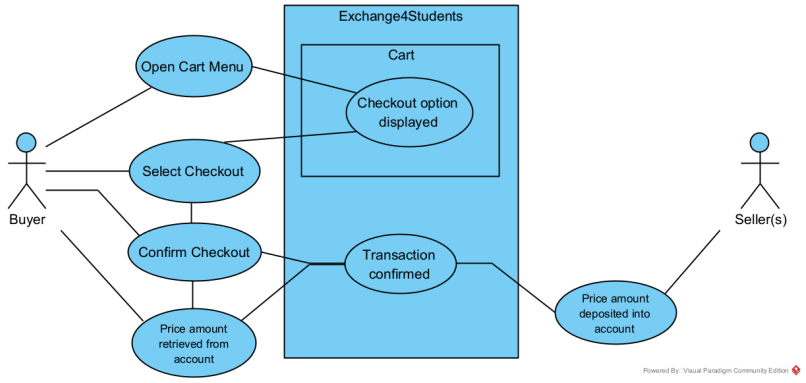
2. UC2 - Purchasing an Item

2.1. UC2.1 - Add item to cart



When adding an item to their cart, users can navigate to the desired item on the marketplace after viewing its description and displayed image. After selecting the ‘Add to cart’ option, the system will then place the item in their Cart and prepare the checkout function.

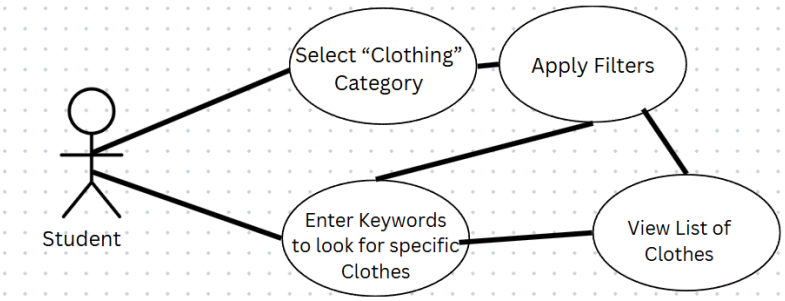
2.2. UC2.2 - Checkout items

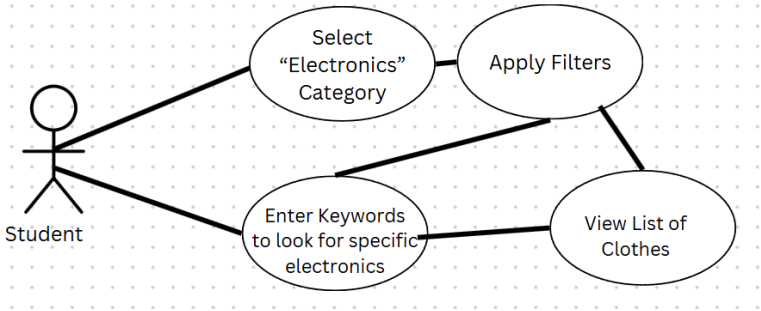


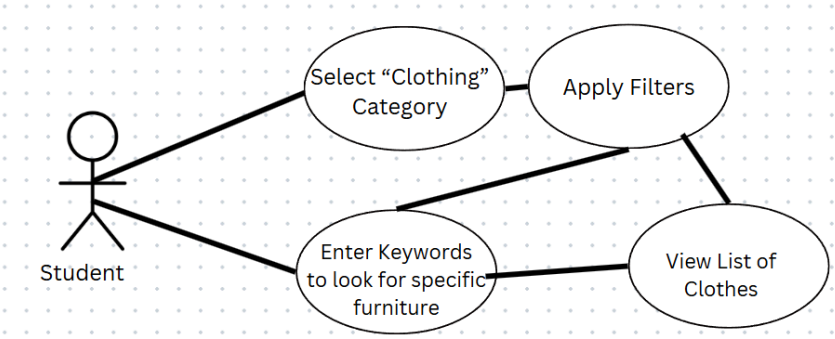
When checking out, users can navigate to their Cart Menu and confirm their checkout after the total price is displayed. The total price of items in the cart is retrieved from the Buyer’s account after the transaction is confirmed and distributed to the seller(s) of the item(s).

3. UC3 - Search for a product

3.1. UC 3.1 - Searching for items

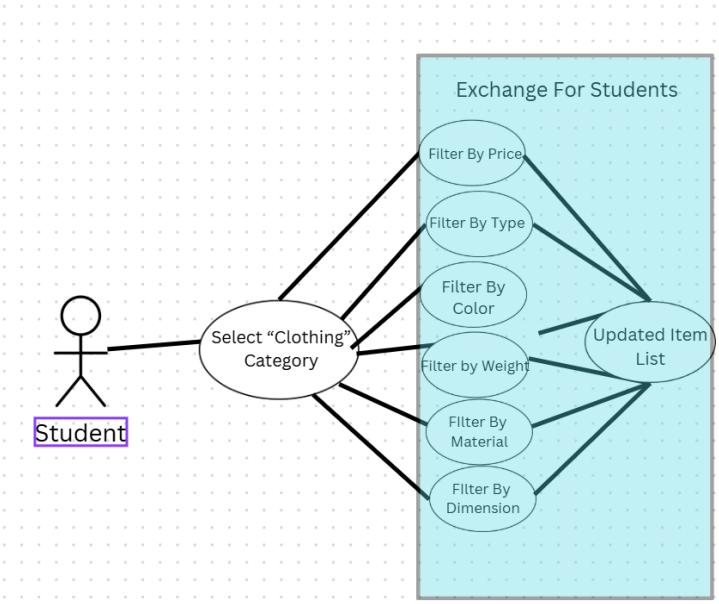


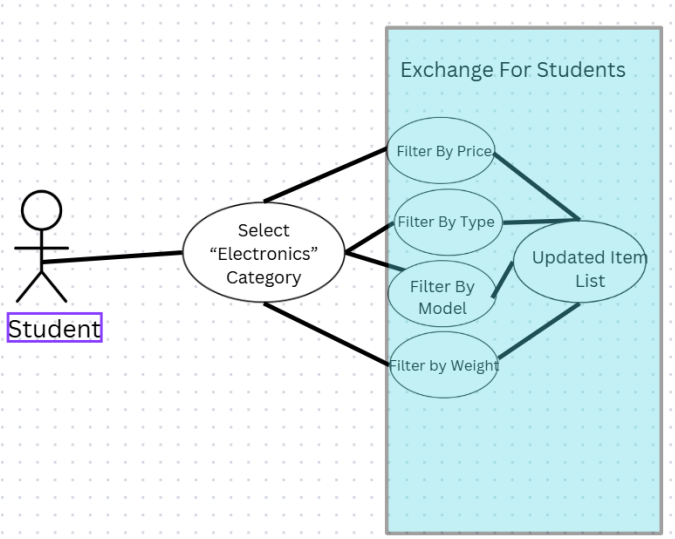


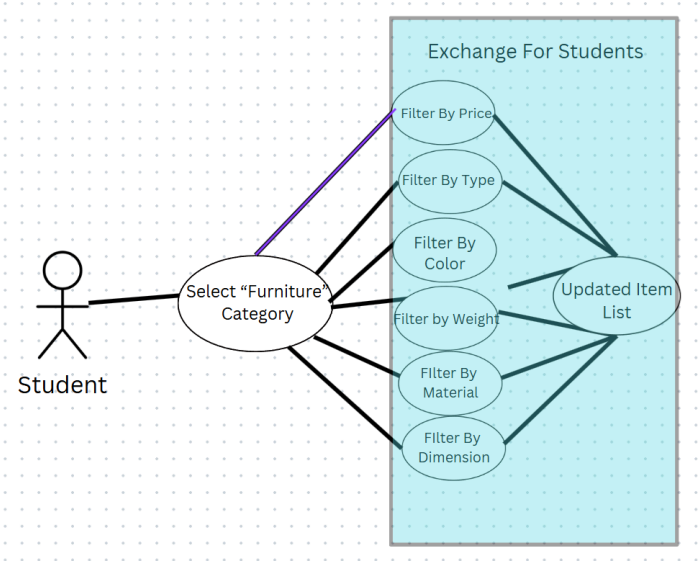


When searching for items, users can select the appropriate category, apply necessary filters, or enter keywords to narrow their search. The system will then display a filtered list of items corresponding to the user’s input.

3.2. UC 3.2 - Filtering by category



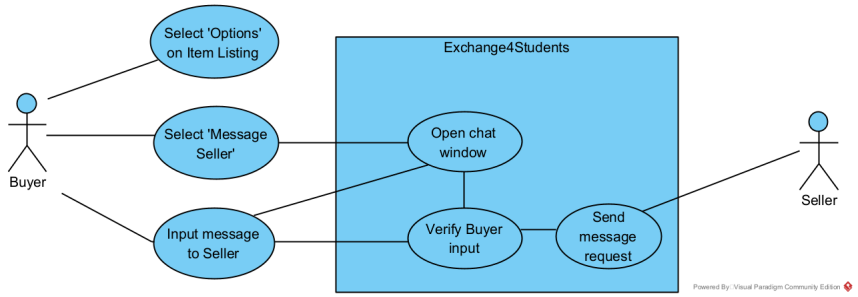




When filtering by category, users can filter displayed items by general information, such as price or weight, or category-specific information, such as furniture material or model of electronics. The system will then display an updated item list corresponding to the selected filters.

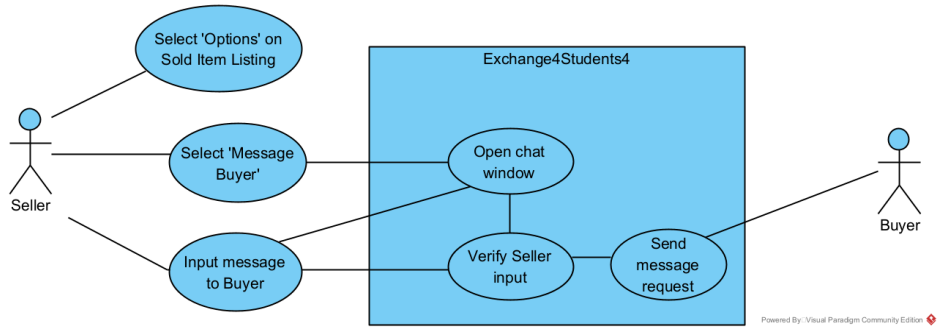
4. UC4 - Communicating with Users

4.1. UC 4.1 - Message seller



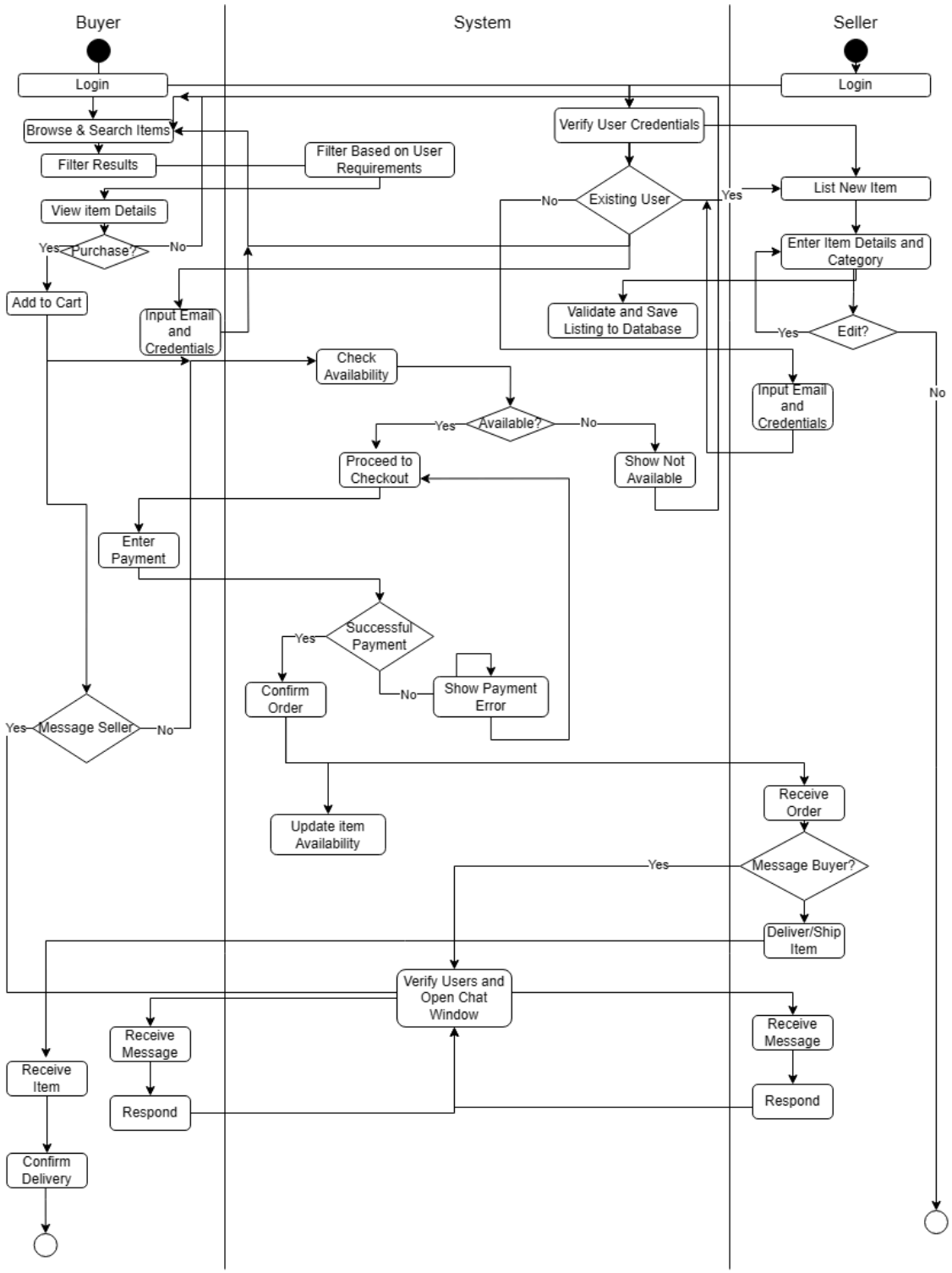
When messaging a Seller, Buyers can request to send a direct message to the Seller through the ‘options’ menu on item listing. The system will then notify the Seller of the message request.

4.2. UC 4.2 - Message buyer



When messaging a Buyer, Sellers can request to send a direct message to the Buyer through the ‘Options’ menu on item listings that have been confirmed to be sold. The system will then notify the Buyer of the message request.

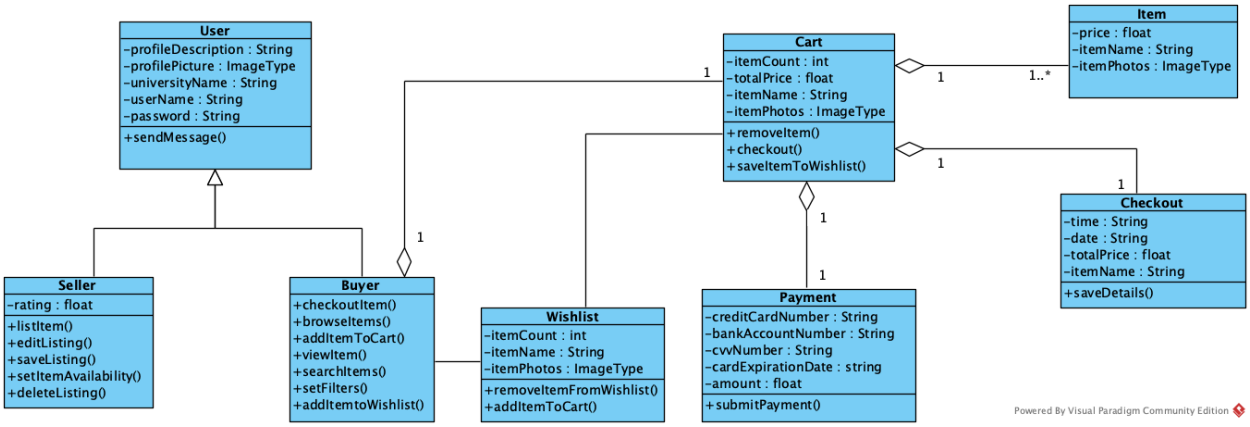
Activity Diagram



This activity diagram is a structured visualization of the key processes in the Exchange4Students marketplace, dividing responsibilities between buyers, the system, and sellers through well-defined swim lanes. It begins with buyers browsing and filtering search results, then viewing item details before deciding whether to purchase. The purchase process involves checking item availability, adding items to the cart, and proceeding through checkout, where the system validates payment. If payment is successful, the order is confirmed, and item availability is updated. The seller then fulfills the order by shipping the item, and the buyer confirms delivery, completing the transaction. These are all based on our use cases each implemented into the activity diagram.

Additionally, the diagram integrates user communication, allowing buyers to message sellers and vice versa. This feature is facilitated by a system verification step before opening a chat window, ensuring secure interactions. The seller-side activities also include item listing and editing, with a verification step before saving new listings. This activity diagram does not go into direct specifics of how each process works, rather the generalized procedures. For example, the chat feature where the Users are verified to ensure proper connection and communication. It is delved into direct detail in the use cases. We did not add the message buyer option on the seller lane as that is typically a process that occurs after a Buyer selects an item and chooses to message the buyer so that originates from the Buyer lane.

Class Diagram

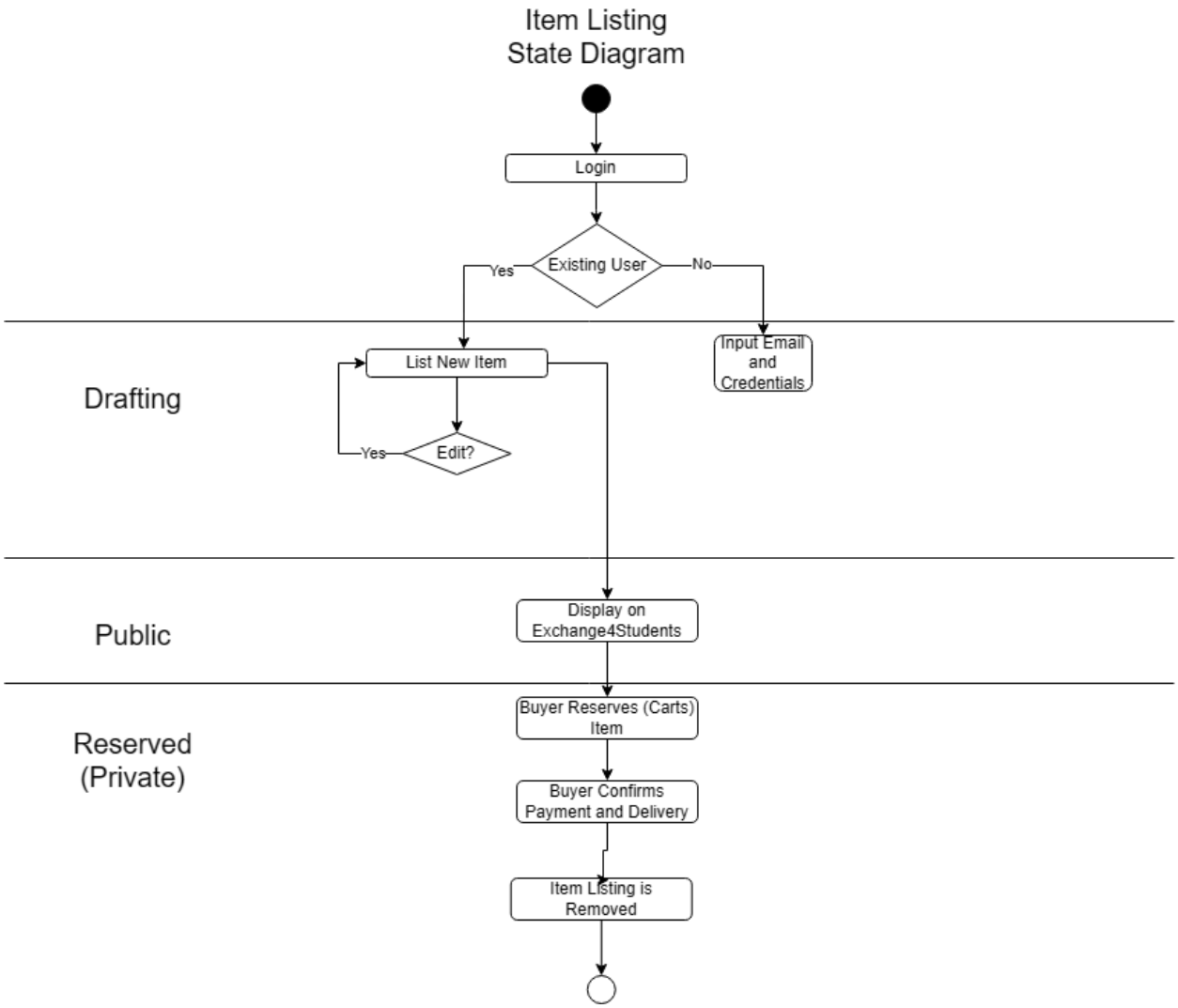


The User class represents the general attributes of Exchange4Students users and the ability to message each other. Since Exchange4Students is geared towards university students, it's important for users to have their university on their profile so the location is more convenient for them to meet to either buy and sell items. The sellers should have ratings to show previous reviews of previous transactions so buyers can know how trustworthy they are. And the Buyer and Seller classes have their respective actions they can do. The Wishlist class represents items that buyers wish to buy in the future. It shows the number of items in the wishlist, the name of each item, and photos of the item(s). Buyers have the ability to add an item from the wishlist into their cart and to remove an item from their wishlist. Each buyer will have exactly one cart, hence the one to one cardinality indicated in the aggregation relationship. The cart has several attributes representing the number of items in the cart, the total price of all the items added together, the name of each item, and photos of each item. From the cart, buyers can remove items, save the items to their wishlist if they change their mind and wish to purchase the item later, and the ability to checkout if they are ready to purchase their items. Each cart can have one or more than one item which is shown by the one to many cardinality in their aggregation relationship. The attributes of an item include the price listed by the seller, the name of the item, and photos

uploaded by the seller. The Checkout class is part of the cart class. It has the time and date of the order, the total price paid, and the name(s) of the items purchased. Every cart will have exactly one checkout. The Payment class has the attributes of credit card information, bank account number for bank transfers, and an attribute to represent the amount that needs to be paid. And from the payment page, buyers can choose to finally submit their payment.

State Diagram

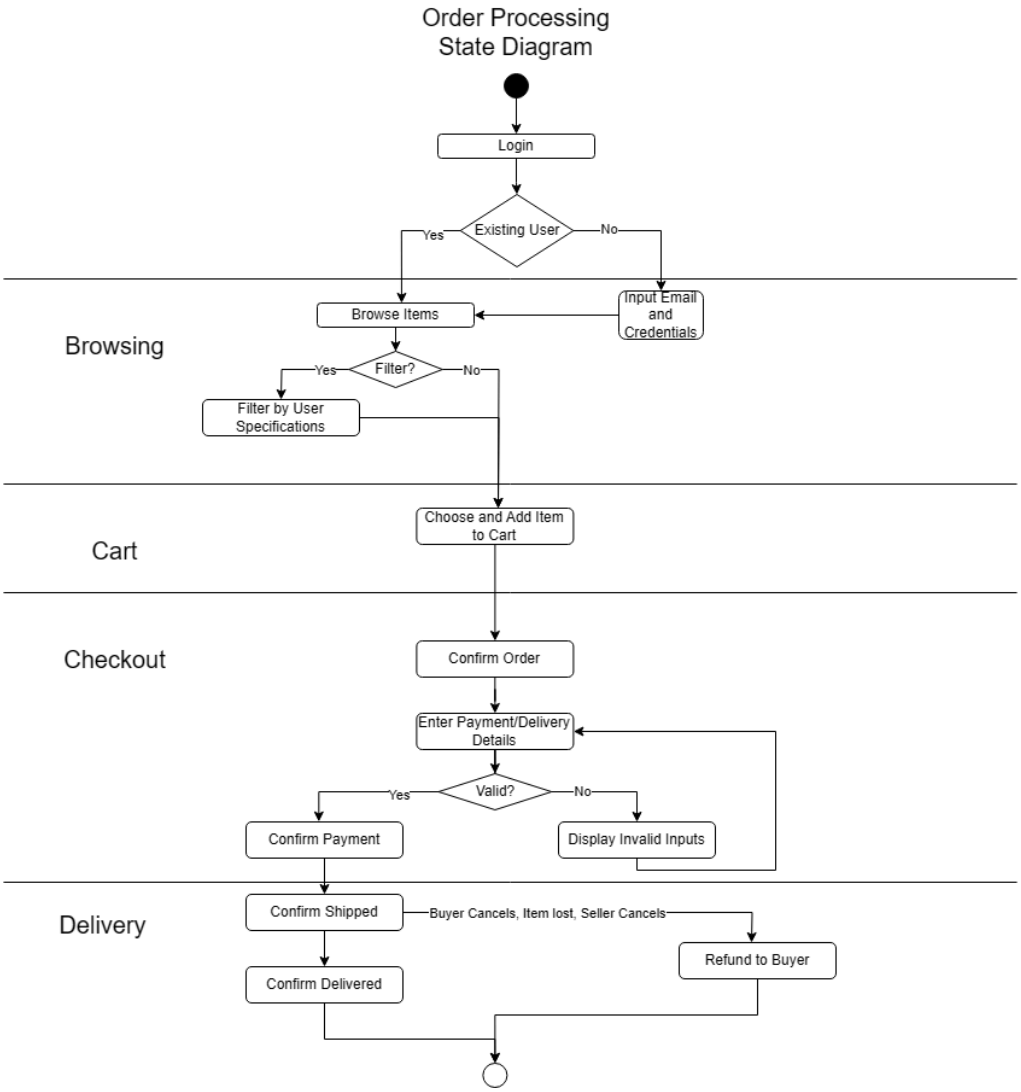
SD 1: Item Listing



The Item Listing state diagram models the lifecycle of an item that a seller posts on the Exchange4Students marketplace.

1. Draft – A seller begins creating a new item listing. At this stage, the listing is incomplete and not visible to buyers. Once the seller submits the listing, the system verifies the details, ensuring that all required fields are complete and images are uploaded correctly.
2. Public – After approval, the item is published and available for buyers to view and purchase. The seller may modify the listing while it is active. Updates are saved, and the listing remains available.
3. Reserved – When a buyer starts a purchase, the system marks the listing as sold. It is no longer available for new buyers but remains in the seller's records. Once payment and delivery is confirmed, the system permanently deletes the listing, and it is no longer recoverable.

SD 2: Order Processing



The Order Process state diagram represents how an order moves through different stages once a buyer initiates a purchase.

1. Browsing – The buyer navigates through the marketplace, searching for items by category, keywords, or filters. At this stage, no order exists yet.
2. Cart – The buyer adds items to their cart but has not yet proceeded with the purchase. They can continue browsing, remove items, or proceed to checkout.

3. Checkout – The buyer initiates the checkout process, where they confirm item selection, provide payment details, and finalize the order. If payment is successful, the order transitions to Delivery. If payment fails, the system notifies the buyer, and the order remains in Checkout until payment is retried or canceled.
4. Delivery – The seller ships the item, and the buyer receives tracking details. Once the item is delivered and confirmed by the buyer, the order is marked as completed. The seller or buyer can still cancel the order before delivery. Should the item be lost or destroyed in transit, the buyer receives a refund.

Lessons Learned

A few things we learned were that you have to be articulate and think very deeply and specifically about a project in order for it to be logical and easy to understand. For example, we had to think of specific ways users would use our project in order to create our Use Case diagrams.

For the activity diagram, we needed to run through all the possible outcomes in order to showcase the flow of our project. One of the key takeaways from creating the activity diagram was the importance of structuring workflows clearly. By breaking down the buyer, seller, and system interactions into distinct swimlanes, we were able to see how different components of the system communicate and depend on one another. Complex processes were simplified: Initially, we struggled with organizing all interactions between buyers, sellers, and the system. However, by sequentially mapping out each step - such as item searching, adding to cart, checkout, and messaging - we ensured that every major action was accounted for.

While creating the diagram, we realized the necessity of handling edge cases like payment failures, item unavailability, or user verification failures. This helped us refine our

system's logic and ensure that users have a seamless experience. The activity diagram also revealed that certain actions—like messaging a seller or modifying a listing—could happen in parallel with other processes, like searching for items or completing a purchase. This insight is valuable for designing an efficient system.

The state diagrams helped us understand how objects in our system transition between different states based on user interactions. Specifically, the Order Processing state diagram demonstrated the lifecycle of an order from creation to completion. Initially, we had vague ideas about how an order moves from cart to checkout and eventually to fulfillment. Creating a state diagram forced us to clearly define each stage and transition, including alternative paths like cancellations and refunds. Through the process of defining transitions, we realized that we needed states like "Refund Processing" and "Payment Failure" to accurately capture real-world scenarios. This allowed us to refine our design and ensure that our system could handle unexpected issues. The state diagram helped us consider states that persist over time, such as "Shipped" and "Awaiting Payment," which may not have immediate transitions but are crucial for tracking an order's progress.

As for the classes we needed to specifically think about what classes we would have in our project and each class's responsibilities. Another thing is that creating diagrams allows people to easily visualize and understand what a project does because diagrams are universal and don't require much technical knowledge to understand. Creating diagrams also helps give more clarity to the developers and also simplifies the development process.