CS319 - Object Oriented Software Engineering

D1 & D2 Final Report

Ahmet Kaan Sever - 22102278
Atakan Şerifoğlu - 22102313
Dila Tosun - 2212100
Konuralp Kılınç - 22101791
Perit Dinçer - 22103102
Selimhan Tokat - 22003145

1. TECH STACK

- React.js will be used to develop the front-end of the application.
- .NET Core will be used to develop the back-end of the application.
- MSSQL will be used for the database of the application, and our database will be deployed to Azure Cloud for easy development.

2. NON - FUNCTIONAL REQUIREMENTS

2.1. User Interface and Human Factors:

- The interface should be responsive and consistent across various devices and screen sizes.
- Visual design elements should align with Bilkent University's branding and colors to create a cohesive and recognizable identity.
- Compatibility testing should be conducted to identify and resolve browser-specific issues.
- Dark mode should be available as an option to enhance user experience, accessibility, and user preference.

2.2. Performance Characteristics:

- Response time for user interactions should be within acceptable limits (e.g., < 2 seconds for page loading).
- The system should be scalable to handle an increasing number of users and transactions.
- Database queries should be optimized to ensure quick retrieval and update of data, preventing performance bottlenecks.
- Ensuring that third-party integrations, such as the Iyzico payment system, do not introduce performance bottlenecks is crucial for smooth system operation.

2.3. Error Handling and Extreme Conditions:

- Appropriate error-handling systems should be in place for users to receive concise error messages.
- When mistakes or exceptions occur, the system should produce thorough logs that include error codes, timestamps, and contextual data to aid in quickly identifying and fixing problems.

2.4. System Interfacing:

- The system should seamlessly integrate with external services and APIs, such as the Iyzico payment system.
- Interactions with external systems should be reliable and secure.

2.5. System Modifications:

- The system should be designed to accommodate future modifications and updates.
- A version control system, such as Git, should be used to manage codebase versions, ensuring that modifications are tracked and documented.

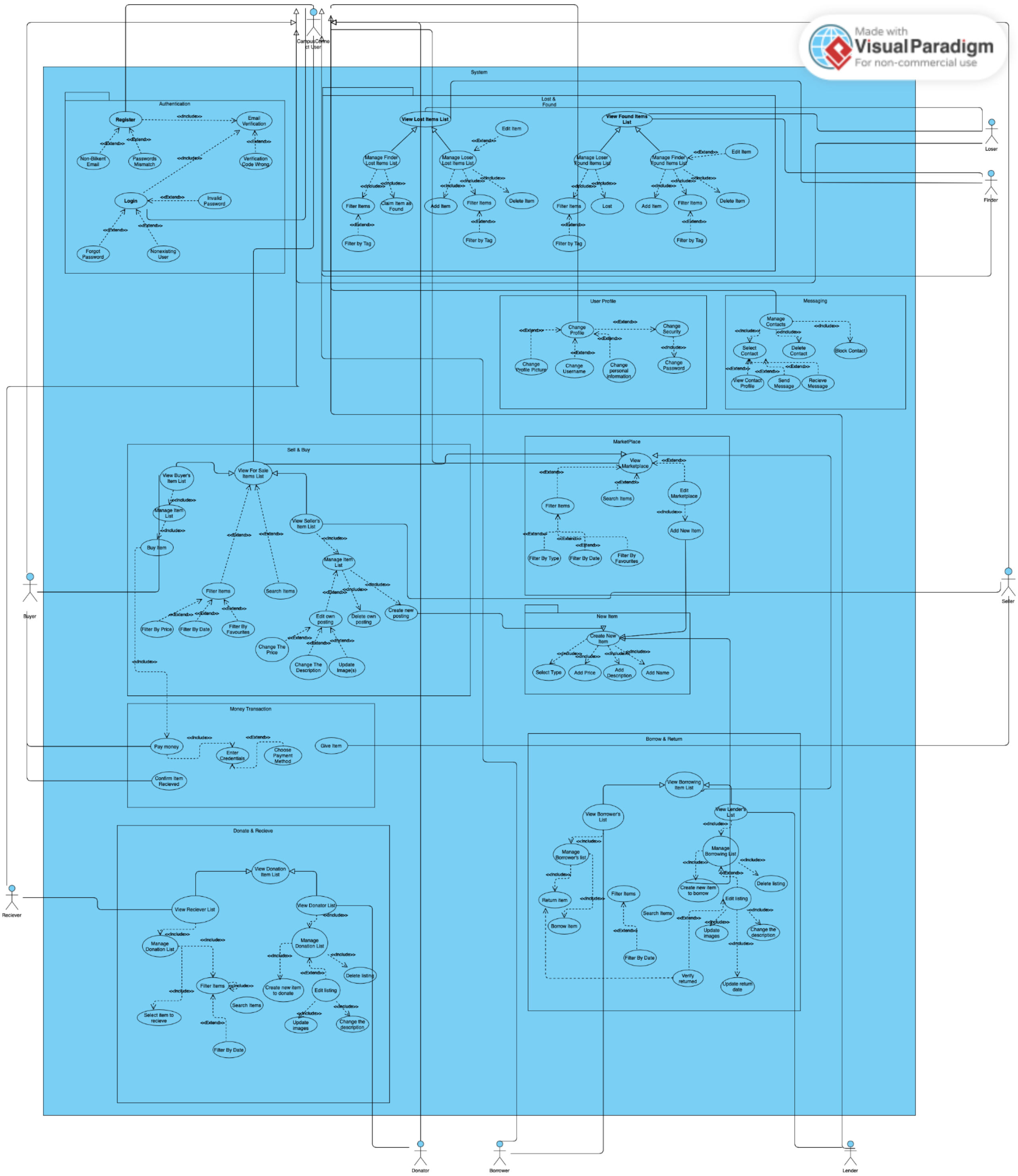
2.6. Passwords and Authentication:

- User passwords should be longer than 8 characters and must include at least 1 lowercase, 1 uppercase character, 1 number, and 1 special character.(e.g. Bilkent.1984)
- Users should have the ability to reset forgotten passwords securely.
- Password hashing and salting should be implemented for secure storage.
- Only the authorized Bilkent University community with a valid bilkent.edu.tr email can access the platform.

2.7. Legislative Requirements

- The use of Iyzico for payment processing should comply with financial services and payment processing regulations and standards.
- The system must adhere to Iyzico's API usage policies and contractual obligations.
- The system must adhere to all relevant legal and legislative requirements, including data protection and privacy laws applicable to Turkey, e.g. Turkish Data Protection Law (KVKK).
- The website must adhere to all relevant legislative requirements, such as those governing financial services and payment processing.

3. USE CASE DIAGRAM



Use Case New (1).png

Figure 1: UML Use Case Diagram of the CampusConnect Application

4. USE CASE TEXTUAL SPECIFICATIONS

4.1. Authentication Package

Context: This package is responsible for creating new users and verifying the new users to be able to use the application.

1. **Use case name:** Register
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** Clicking the “Sign Up” button on the “Log In” page.
4. **Exit Condition:** The User clicks on “Log In instead” or the User signs up successfully.
5. **Flow of events:**
   - **a)** The User decides to sign up.
   - **b)** If an invalid student ID is entered**,** “Invalid student ID.” error message is displayed.
   - **c)** Else if the two passwords given by the user do not match**,** “Passwords do not match.” error message is displayed.
   - **d)** Else if an invalid Bilkent web-mail address is given**,** “Invalid mail address.” error message is displayed.
   - **e)** Else the User signs up successfully. The User can only use the platform after verifying their email address via the verification mail sent after signing up.

1. **Use case name:** Login
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** Opening the app through a web browser.
4. **Exit Condition:** The User signs in successfully.
5. **Flow of events:**
   - **a)** The User decides to log in.
   - **b)** The User fills in the text boxes labeled “E-mail address” and “Password”
   - **c)** The User clicks the “Log In” button.
   - **d)** If the given mail address does not exist, “This user does not exist” error message is displayed.
   - **e)** If the given password is incorrect, “Incorrect Password” error message is displayed.
   - **f)** Else the User logs in successfully.

4.2. Sell & Buy Package

Context: If the User wishes to buy an item or sell an item they will be using this package to view items that are listed or create new listings. This package is responsible for listing the items that are on sale in the system.

1. **Use case name:** View For Sale Items List
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and the User clicks the buy & sell button.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button to navigate to another part of the application.
5. **Flow of events:**
   - a) The User decides to view the Items that users are selling on the platform.
   - b) The User then clicks on the “Sales” button located on the side bar.
   - c) After clicking, the User is redirected to the Main Page of All Items.

d) The User can browse the listed items.

1. **Use case name:** View Buyer's Item List
2. **Participating Actors:** Buyer User.
3. **Entry Condition:** The Buyer User is logged into their account and clicks the buy & sell button.
4. **Exit Condition:** The Buyer User decides to exit the section by clicking on another redirecting button to navigate to another part of the application or the Buyer User clicks on 'Buy Now' or 'Add to Cart' button to initiate purchase.
5. **Flow of events:**
   a) The Buyer decides to view the Items that users are selling on the platform.
   b) The system presents a list of items available for sale, including their titles and prices for users to browse.
   c) While viewing the list of items for sale, the Buyer can select a specific item they want to purchase.
      i. The system displays the item's details, including the description and price.
      ii. The Buyer clicks on a "Buy Now" or "Add to Cart" button to initiate the purchase.
      iii. The system redirects the Buyer to the payment module for transaction processing.

1. **Use case name:** View Seller's Item List
2. **Participating Actors:** Seller User.
3. **Entry Condition:** The Seller User is logged into their account and clicks the buy & sell button
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button to navigate to another part of the application or the Seller decides to add, edit, or remove an item and clicks to the appropriate button("Add Item", "Edit Item" and "Remove Item" respectively.).
5. **Flow of events:**
   a) The Seller decides to view the Items that users are selling on the platform.
   b) The system presents a list of items available for sale, including their titles and prices for users to browse.
   c) The Seller decides to create a new item to sell and clicks add a new item pane.
      i. The system presents an option to create a new item for sale.
      ii. The Seller provides details about the item, including name, description, price, and optionally, an image.
      iii. The Seller submits the form to list the item for sale.
      iv. The system processes the listing and adds the item to the Seller's list of items for sale.
   d) The Seller already has the item(s) for sale and decides to edit it/them by clicking on the item.
      i. The system allows the Seller to modify the item's details (e.g., description, price, image).

ii. The Seller submits the changes.
iii. The system updates the item's details in the Seller's list.

e) The Seller decides to remove an item he is listing.
i. The Seller selects a specific item they want to delete.
ii. The system prompts Seller for confirmation to delete the item.
iii. The Seller confirms the deletion.
iv. The system removes the item from the Seller's list.

4.3. Lost and Found Package

Context: This use case comes into play when a registered user, referred to as the "User", intends to browse items listed as found or wishes to create an item typed lost. The user has the ability to view all the items typed lost and found with the options for filtering by type or date.

1. **Use case name:** View Lost Items List
2. **Participating Actors:** Loser User, Finder User.
3. **Entry Condition:** The User is logged into their account and clicks the "Lost Items" button.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button to navigate to another part of the application or the Loser User clicks on an item to view more details.
5. **Flow of events:**.
   a) The system presents a search interface with various filters such as location, date, and item type.
   b) The User provides relevant details and initiates the search.
   c) The system processes and presents users a decision between the search request or creating a listing for the lost item.
   d) Users can view the listing matching the definition of the lost item.
   e) The User reviews the list and clicks on an item to view more details.

1. **Use case name:** Manage Finder Lost Items List
2. **Participating Actors:** Finder User.
3. **Entry Condition:** The Finder User is logged into their account and navigates and clicks on the " Lost and Found" button.
4. **Exit Condition:** The Finder User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or the Finder User cancels the creation of a new listing for the lost items or presses the 'Claim as Found' button.
5. **Flow of events:**
   a) The Finder User can filter items by their tag to see whether the item they have found has been posted as lost.
   b) If the Finder User does see the lost item listing, they can press the Claim as Found button to inform the Loser that their item has been found.
   c) This will then put the two users in communication to arrange a pickup.

1. **Use case name:** Manage Loser Lost Items List
2. **Participating Actors:** Loser User.
3. **Entry Condition:** The Loser User is logged into their account (Must already have a list for a lost item to edit).
4. **Exit Condition:** The Loser User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or the Loser User creates a listing for the lost item successfully.
5. **Flow of events:**
   a) The Loser User clicks on the “I’ve lost something” button.
   b) The system processes and presents users a decision between the search request or creating a listing for the lost item.
   c) The Loser User can create a listing for the lost item with the provided information by using the “Create a listing” button.

1. **Use case name:** View Found Items List
2. **Participating Actors:** Loser User, Finder User.
3. **Entry Condition:** The User is logged into their account and clicks the found items button.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button to navigate to another part of the application.
5. **Flow of events:**
   a) The system presents a search interface with various filters such as location, date, and item type.
   b) The User provides relevant details and initiates the search.
   c) The system processes and presents users a decision between the search request or creating a listing for the found item.
   d) The Users can view the listing matching the definition of the found item.
   e) The User reviews the list and clicks on an item to view more details.

1. **Use case name:** Manage Finder Found Items List
2. **Participating Actors:** Finder User.
3. **Entry Condition:** The User is logged into their account.
4. **Exit Condition:** The Finder User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or the User successfully submits a found item.
5. **Flow of events:**
   a) The Finder User can filter by tag to see if the item they have found has already been posted.
   b) If not, the Finder User clicks on the "I've found something" button.
   c) The system presents a form for the User to enter details about the lost item, including a description, location, date, and contact information.
   d) The Finder User provides the necessary information and submits the form.
   e) The system processes the submission and confirms receipt of the lost item report.

f) The Finder User can also edit or delete the listing they have created.

1. **Use case name:** Manage Loser Found Items List
2. **Participating Actors:** Loser User.
3. **Entry Condition:** The User is logged into their account.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or presses the “Claim as Lost” button.
5. **Flow of events:**
   a) The Loser User can filter items by their tag to see whether the item they have lost has been posted as found.
   b) If the Loser User does see the found item listing, they can press the Claim as Lost button to inform the Finder that their item has been lost.
   c) This will then put the two users in communication to arrange a pickup.

4.4. User Profile

Context: Users will use this package to edit their profile. They can use this to update their profile picture, description, mail address and their password.

1. **Use case name:** Change profile
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and clicks the edit profile button.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or the User updates their profile information successfully.
5. **Flow of events:**
   a) The User navigates to the profile settings or account management section of the application.
   b) The system displays the current profile information, including the profile picture, email address, personal details, and an option to change the password.
   c) The User selects the specific information they want to update (e.g., profile picture, email address, personal details, or password).
   d) If the User chooses to update the profile picture, they are prompted to upload a new image.
   e) If the User chooses to update the email address, they enter the new email and confirm it (the new mail address still has to be a Bilkent Web-mail address with bilkent.edu.tr extension.).
   f) If the User chooses to update personal information, they edit the relevant fields (e.g., name, bio, etc.).
   g) If the User chooses to update the password, they enter the current password, followed by the new password, and confirm it.
   h) The User reviews the changes and clicks on the "Save" or "Update" button.

i) The system processes the updates and confirms the successful modification of the profile information.

4.5. Messaging

Context: This use case is used when a registered user wants to interact with other users regarding items for sale, found items, or borrowed items. The user has the ability to view other users' profiles, send and receive messages, and potentially block other users.

1. **Use case name:** Manage Contacts
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and opens the messaging panel.
4. **Exit Condition:** The User decides to exit the panel by clicking on another redirecting button or navigates to another part of the application.
5. **Flow of events:**
   - **a)** The User decides to read or send messages.
   - **b)** They click on the "Messages" button on the left sidebar.
   - **c)** They browse throughout their message list.

1. **Use case name:** View Contact Profile
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and selects the contact from the Contacts list.
4. **Exit Condition:**The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application.
5. **Flow of events:**
   - **a)** The User navigates to the "Messaging" section of the application using the sidebar.
   - **b)** The system presents a list of available contacts or recent conversations.
   - **c)** The User selects a specific contact or searches for a user by name or username.
   - **d)** The system displays the profile of the selected user, including their username, profile picture, and any relevant information they have chosen to share.

1. **Use case name:** Send Message
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and presses send message button after selecting contact.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application.
5. **Flow of events:**
   - **a)** The User navigates to the "Messaging" section of the application.
   - **b)** The system presents a list of available contacts or recent conversations.
   - **c)** The User selects a specific contact or searches for a user by name or username.
   - **d)** The system displays the conversation interface, allowing the User to compose and send a message.

e) The User types out the message and clicks on the "Send" button.
f) The system delivers the message to the intended recipient.

1. **Use case name:** Receive Message
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application.
5. **Flow of events:**
   - **a)** The User receives a notification or navigates to the "Messaging" section of the application.
   - **b)** The system displays a list of conversations, indicating unread messages.
   - **c)** The User selects a specific conversation to view the message.
   - **d)** The system displays the message, including the Sender's username and profile picture.

1. **Use case name:** Block Contact
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and selects the User to block.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or the User confirms the decision to block or unblock another User.
5. **Flow of events:**
   - **a)** The User navigates to the profile of the User they wish to block.
   - **b)** The system provides an option to block the User.
   - **c)** The User confirms the decision to block the user.
   - **d)** The system updates the User's blocked contacts list.
   - **e)** If the User decides to unblock a contact:
     - **i.** The User navigates to their list of blocked contacts.
     - **ii.** The system provides an option to unblock a User.
     - **iii.** The User confirms the decision to unblock the User.
     - **iv.** The system updates the User's blocked contacts list.

4.6. Money Transaction Package

Context: This use case is activated when a registered user, known as the "Buyer User", intends to purchase an item. The Buyer User will make the payment through the system and provide their payment credentials. They have the option to cancel the order before making the payment. After the payment is completed, the system will prompt the Buyer User to confirm if they received the item. Upon confirmation, the funds will be transferred to the Seller User's bank account.

1. **Use case name:** Pay Money
2. **Participating Actors:** Buyer User.

3. **Entry Condition:** The Buyer must select an item they wish to purchase and submit a buy request through the system.
4. **Exit Condition:** The Buyer decides to exit the section by clicking on another redirecting button or navigates to another part of the application or the Buyer enters the payment details and the payment is completed.
5. **Flow of events:**
   - **a)** The Buyer User selects the desired item for purchase.
   - **b)** The system displays the item details, including the price and a "Checkout" button.
   - **c)** The Buyer User clicks on the "Checkout" button.
   - **d)** The system prompts the Buyer to enter their payment credentials (e.g., credit card details or use saved payment methods).
   - **e)** Before entering payment details in step 4, the Buyer decides to cancel the order.
   - **f)** The system confirms the cancellation and returns the Buyer to the item listing.
   - **g)** After payment is completed, the system prompts the Buyer to confirm if they received the item.

1. **Use case name:** Confirm Item Received
2. **Participating Actors:** Buyer User**.**
3. **Entry Condition:** The User presses the Confirm Received button on the Buy-Sell page
4. **Exit Condition:** The Buyer decides to exit the section by clicking on another redirecting button or navigates to another part of the application or they press the button to confirm the item is received successfully, or the Buyer confirms that they have not received the item.
5. **Flow of events:**
   - **a)** After payment is completed, the system prompts the Buyer to confirm if they received the item.
   - **b)** The Buyer indicates that they have received the item by selecting an appropriate option.
   - **c)** Upon confirmation of item receipt, the system processes the payment.
   - **d)** The funds are transferred to the Seller's designated bank account.
   - **e)** If the Buyer confirms that they have not received the item the User will be redirected to technical support to solve the issue.

1. **Use case name:** Give Item
2. **Participating Actors:** Seller User.
3. **Entry Condition:** The order for an item must be completed.
4. **Exit Condition:** The Seller decides to exit the section by clicking on another redirecting button or navigates to another part of the application or confirms whether they have given the item or not.
5. **Flow of events:**
   - **a)** After getting a confirmation with the system the Buyer and the Seller can message with each other to decide on a meeting time/place.
   - **b)** After giving the item to the buyer, the system prompts the Seller to confirm if they gave the item.

c) The Seller indicates that they have given the item by selecting an appropriate option.
d) Upon confirmation from both the Seller and the Buyer, the system processes the payment.
e) The funds are transferred to the Seller's designated bank account.

4.7. Donate & Receive Package

Context: This use case is activated when a user decides to donate an item. The Donator User lists the item as free and a Receiver Users can view and browse the listed donations. The Receiver User can select an item and upon selection, they can request the item. The Donator User has the ability to list an item, edit their listed items, or remove their donations from the system.

1. **Use case name:** Receiver List
2. **Participating Actors:** Receiver User.
3. **Entry Condition:** The Receiver is logged into their account.
4. **Exit Condition:** The Receiver clicks on another redirecting button, or wants to go to another section of the application or the Receiver clicks on the 'Request' button to express interest successfully.
5. **Flow of events:**
   a) The Receiver navigates to the "Donations" section of the application located on the sidebar.
   b) The system displays a list of available donations, including their titles and categories.
   c) The Receiver selects a specific donation item for more details.
   d) The system displays the details of the selected item, including the description and photo (if provided by the Donator).
   e) The Receiver clicks on a "Request" button to express interest in receiving the item.
   f) The system notifies the Donator of the request.

1. **Use case name:** Donator List
2. **Participating Actors:** Donator User.
3. **Entry Condition:** The Donator is logged into their account
4. **Exit Condition**: The Donator decides to exit the 'Donate Items' section of the page by clicking to another redirecting button, or navigates to another part of the application or the Donator submits the form to list the item successfully.
5. **Flow of events:**
   a) The Donator navigates to the "Donate Items" section of the application.
   b) The system presents a form for the Donator to provide details about the item, including a title, description, and an optional photo.
   c) The donator submits the form to list the item for donation.
   d) The system processes the listing and adds the item to the donation list.

1. **Use case name:** Manage Donation List

2. **Participating Actors:** Donator User.
3. **Entry Condition:** The Donator is logged into their account
4. **Exit Condition:** The Donator decides to leave the page by clicking to another redirecting button, or navigates to another part of the application or the listing is added by the Donator successfully, or the listing is edited by the User successfully, or the listing is removed by the Donator successfully.
5. **Flow of events:**
   - **a)** The Donator can add new listings if they want by providing details about the item, including a title, description, and an optional photo.
   - **b)** The Donator can edit their listing by changing the details of the item including its title, description, or its photo.
   - **c)** The Donator can remove an item from their listings by doing so from the system.

4.8. New Item Package

Context: This package is triggered when a user, known as the "Seller, Donator, Lender, or Finder", decides to list a new item in the system. They utilize this use case to create a new item entry, selecting its type (e.g., lending, selling, donating, found or lost item). The user provides the item's name, a brief description, and optionally, an image.

1. **Use case name:** Create New Item
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User must be logged into their account.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or closes the web browser. Or the User fills the form details and submits the form successfully.
5. **Flow of events:**
   - **a)** The User navigates to the "Create New Item" section of the application.
   - **b)** The system presents a form for User to input details about the item.
   - **c)** The User selects the type of item (e.g., lending, selling, donating, found or lost item).
   - **d)** The User provides the name of the item.
   - **e)** The User enters a brief description of the item.
   - **f)** The User has the option to upload an image of the item.
   - **g)** The User submits the form to create the new item entry.
   - **h)** The system processes the information and adds the item to the system's listings.

4.9. Marketplace Package

Context: This use case is activated when a registered user, referred to as the "User", wants to browse and interact with all items available in the marketplace. The User can view all items regardless of their type. They also have the ability to filter items based on type, date, or price. Additionally, the User can use this use case to edit their existing listings, remove their listings or create new ones.

1. **Use case name:** View Marketplace

2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and clicks the buy & sell button.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or closes the web browser.
5. **Flow of events:**
   - **a)** The User decides to view Items that Users are selling on the platform..
   - **b)** Clicks on the “Sales” button placed on the sidebar.
   - **c)** If the User wants they can filter or sort the items listed
     - **i)** By price
     - **ii)** By date
   - **d)** If the User wants they can search for an item that they are looking for by name.

1. **Use case name:** Edit Marketplace
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User must be logged into their account and have an existing listing.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or closes the web browser. Or the User modifies the item and submits the changes successfully.
5. **Flow of events:**
   - **a)** The User navigates to the "My Listings" section of the application.
   - **b)** The system presents a list of the User's existing listings.
   - **c)** The User selects a specific item they want to edit.
   - **d)** The system allows the User to modify the item's details (e.g., description, condition, photo).
   - **e)** The User submits the changes.
   - **f)** The system updates the item's details in the marketplace listings.

4.10. Borrow & Return Package

Context: This use case comes into play when a registered user, referred to as the "User", intends to browse items available for borrowing. The User has the ability to view all items and apply filters to refine their search, including options for filtering by name, date.

1. **Use case name:** Borrowing Item List
2. **Participating Actors:** CampusConnect User.
3. **Entry Condition:** The User is logged into their account and is on the Dashboard page.
4. **Exit Condition:** The User decides to exit the section by clicking on another redirecting button or navigates to another part of the application or closes the web browser.
5. **Flow of events:**
   - **a)** The User navigates to the "Borrowing" section of the application.

b) The system presents a list of items available for borrowing.
c) The User can scroll through the list to view item thumbnails, titles, and basic information.
d) While browsing the items up for borrowing, users can use filters to refine their search.
e) The User can choose to filter items based on type (e.g., books, tools, electronics, etc.).
f) The User can choose to sort items by date or price (ascending or descending).

1. **Use case name:** View Lender's List
2. **Participating Actors:** Lender User.
3. **Entry Condition:** The Lender is logged into their account and is on the Dashboard page.
4. **Exit Condition:** The Lender decides to exit the section by clicking on another redirecting button or navigates to another part of the application or closes the web browser. Or the Lender views their listed items, or removes the item successfully.
5. **Flow of events:**
   - **a)** The Lender navigates to the "My Listings" section of the application.
   - **b)** The system presents a list of the Lender's currently listed items for borrowing.
   - **c)** The Lender can choose to modify the listed items' details (e.g., description, availability date, name).
   - **d)** The Lender can choose to remove an item from their listings. After confirming the deletion, the system removes the item from the database.

1. **Use case name:** View Borrower's List
2. **Participating Actors:** Borrower User.
3. **Entry Condition:** The Borrower is logged into their account and is on the Dashboard page.
4. **Exit Condition:** The Borrower decides to exit the section by clicking on another redirecting button or navigates to another part of the application or closes the web browser. Or the Borrower views their borrowed items, or returns the item successfully.
5. **Flow of events:**
   - **5.1.** Borrower navigates to the "Borrower's List" section of the application.
   - **5.2.** The system presents a list of currently borrowed items, including their titles and due dates.
   - **5.3.** The Borrower can choose to return an item they are borrowing
     - **5.3.1.** While viewing the Borrower's list of currently borrowed items, Borrower selects a specific item they want to return.
     - **5.3.2.** The system prompts Borrower for confirmation to return the item.
     - **5.3.3.** Borrower confirms the return.
     - **5.3.4.** The system updates the status of the item to indicate it has been returned.

Class Diagram

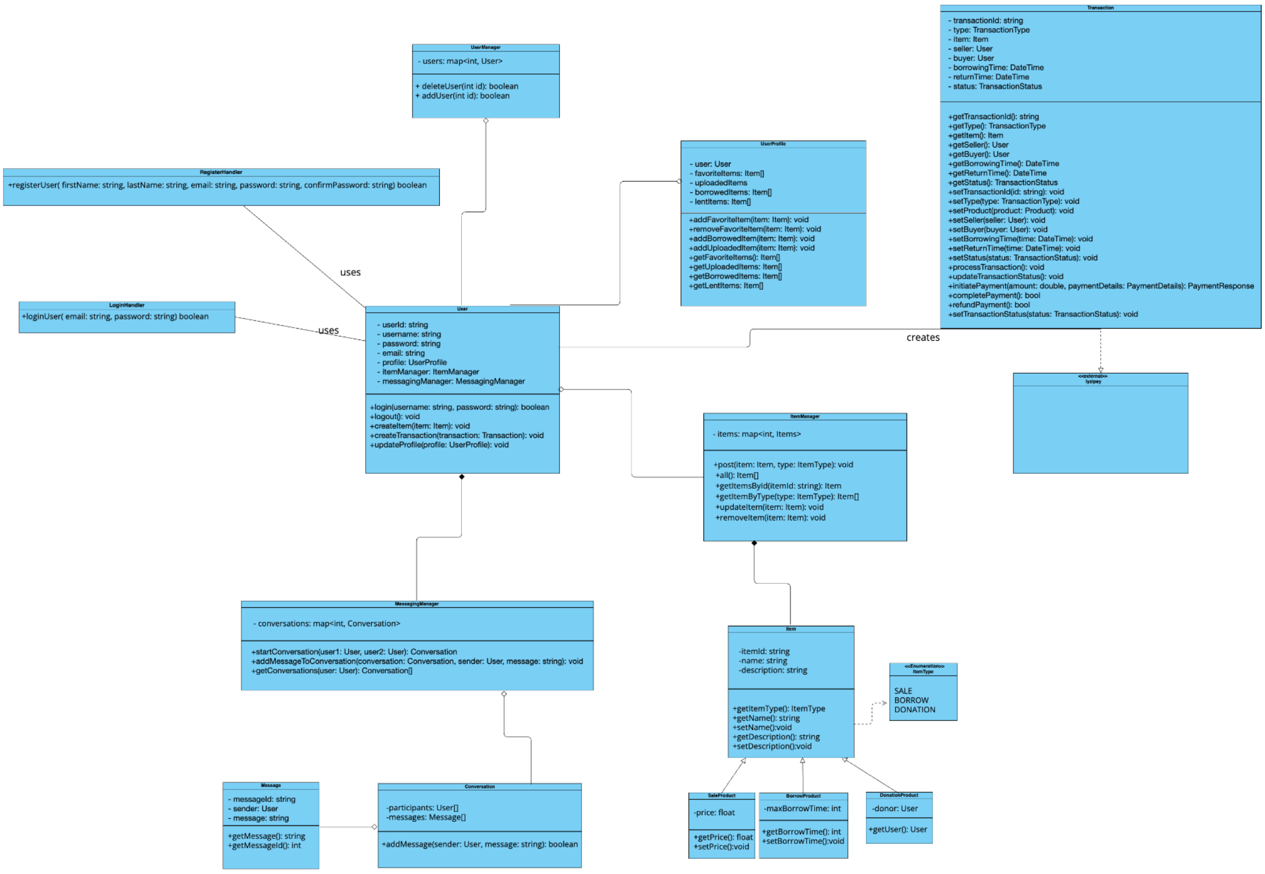


Fig. 2: Class Diagram

User: The User class represents the various users within the CampusConnect platform. It contains essential attributes such as user IDs, usernames, passwords, emails, and user profiles. Users can log in, create items, initiate transactions, and send messages. It serves as the base class for different user roles in the system.

UserManager: User manager holds user records and handles basic user manipulation methods.

Item: Item is an abstract class representing various types of items that users can post or interact with within CampusConnect. It includes shared attributes like item IDs and names. Specific item types, such as sale, borrow, and donation items, extend this class.

ItemType: ItemType is an enumeration class representing the different categories or types of items within CampusConnect, including sale, borrow, and donation. Enumerations are decided to be included just to make the code more structured. In various places it is assumed to have collections of items and the need to decide what type of items they are can be handled with enumerations.

SaleItem: SaleItem represents items available for sale. It inherits from the Item class and adds specific attributes and methods related to sale items.

BorrowItem: BorrowItem represents items available for borrowing. It extends the Item class and includes attributes for maximum borrow time and availability status for borrowing.

DonationItem: DonationItem represents items offered as donations. It extends the Item class and includes information about the donor.

ItemManager: ItemManager is responsible for handling item-related operations. Users can post items, view all items, get items by category, update and remove items, borrow and return items, buy items, and accept donations. It plays a crucial role in managing the items available within the platform.

Transaction: Transaction represents interactions between users and items. It includes information like transaction IDs, transaction types, associated items, users (sellers and buyers), borrowing and return times, and transaction statuses. Transactions can be processed, updated, initiated with payments, and finalized.

UserProfile: UserProfile contains user-specific information, including favorite items, uploaded items, borrowed items, and sales status. Users can add and manage items in their favorites, track their uploaded and borrowed items, and view their sales status.

MessagingService: MessagingService handles user conversations and messages. Users can start conversations, add messages to conversations, and retrieve their conversations. It facilitates real-time communication within the platform.

Conversation: The Conversation class represents individual communication threads between users. It contains participants and messages. Users can add participants and messages to conversations.

Message: The Message class represents individual messages within a conversation. It includes a message ID, sender, and message content. Messages are exchanged within conversations.

6. Sequence Diagrams

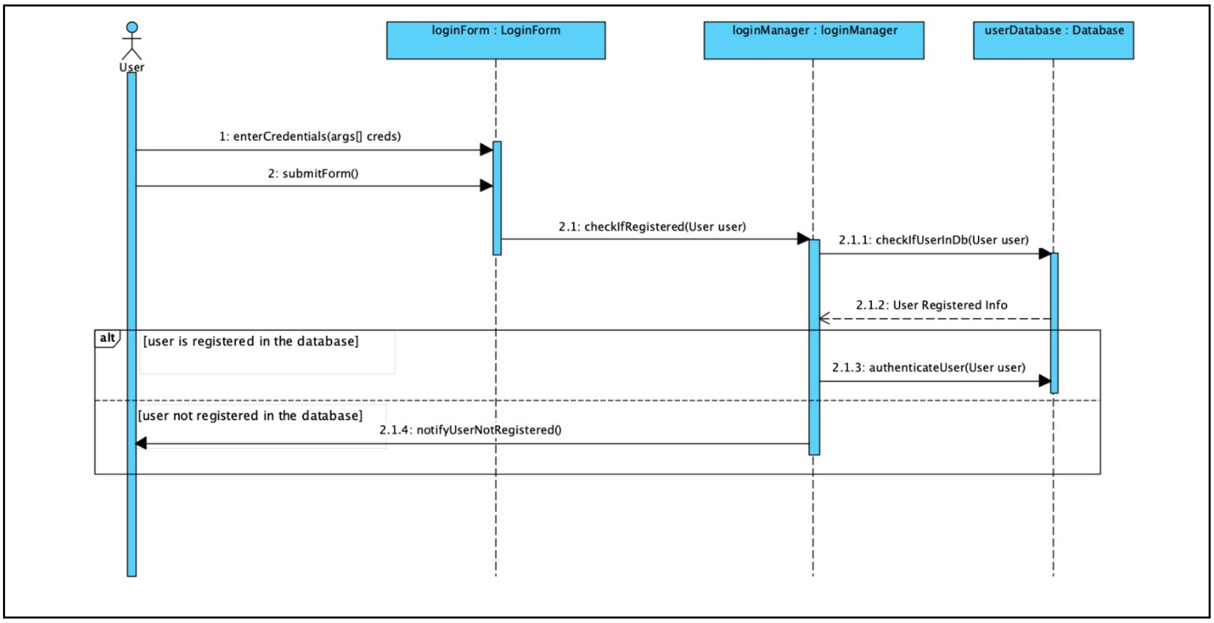


Fig. 3: Sequence Diagram for Login Functionality

A user will first enter their information into the login form and submit it. This information will then be sent to the database to be compared, to see if such a user is already registered. If the user has been registered before, the user will be authenticated and logged into the application. If no registry can be found, an error will be displayed urging the user to register first.

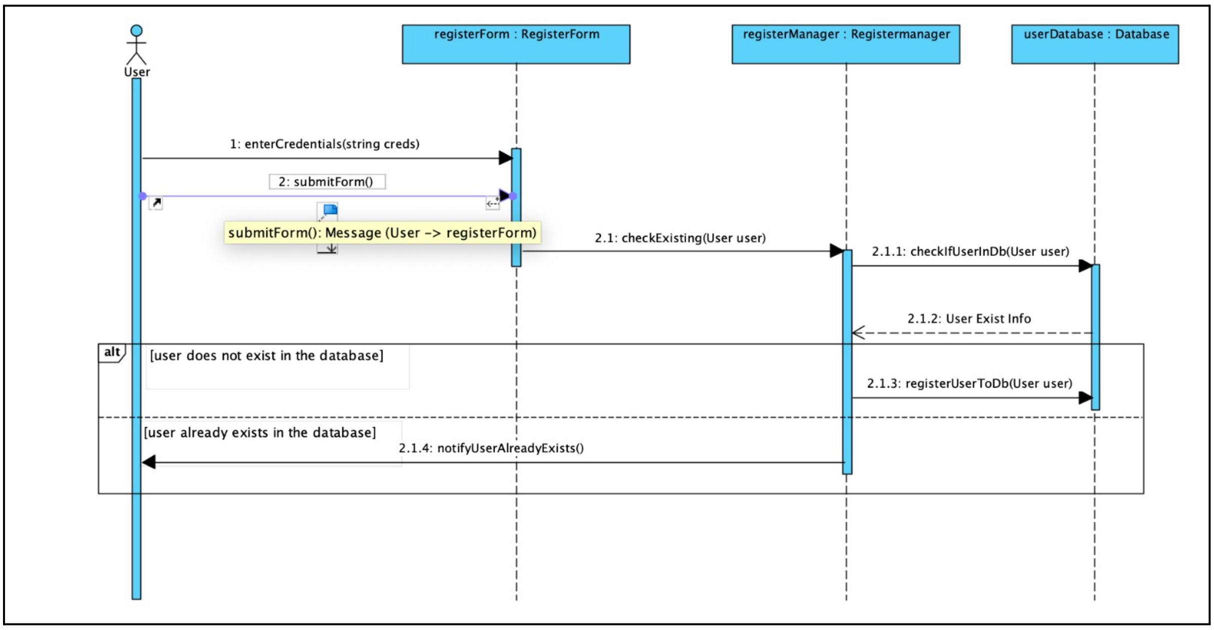


Fig. 4: Sequence Diagram for Register Functionality

A user will first enter their information into the register form and submit it. This information will then be sent to the database to be compared, to see if such a user already exists. If so, an error will be displayed saying that such a user already is registered. If not, the user and their credentials will be registered into the database.

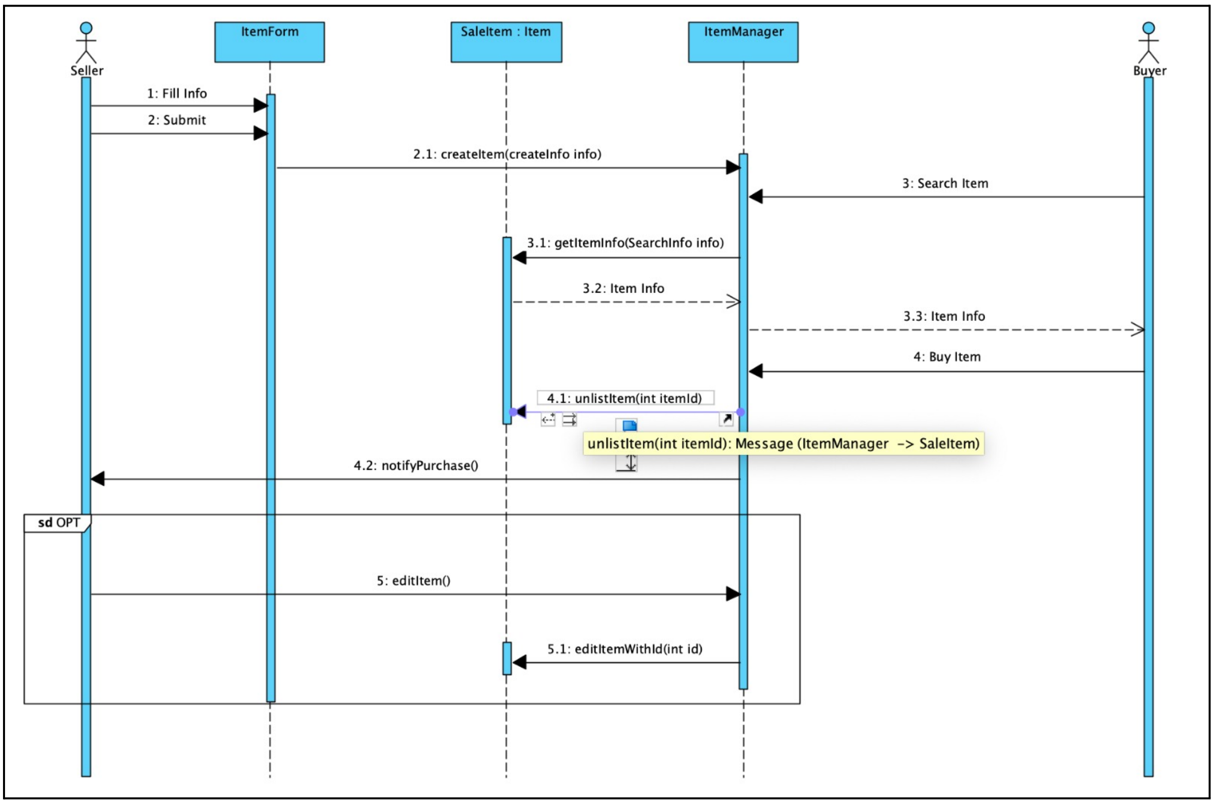


Fig. 5: Sequence Diagram for Buy-Sell Functionality

A seller can fill an item form and submit it for sale, which will create a sale listing. Buyers can search listings, and can click the buy button on any listing they wish to purchase. After a buyer presses the Buy button, the listing will be unlisted and the seller will be notified of the purchase. Additionally, sellers can also edit or delete listings if they choose to do so from their own listings.

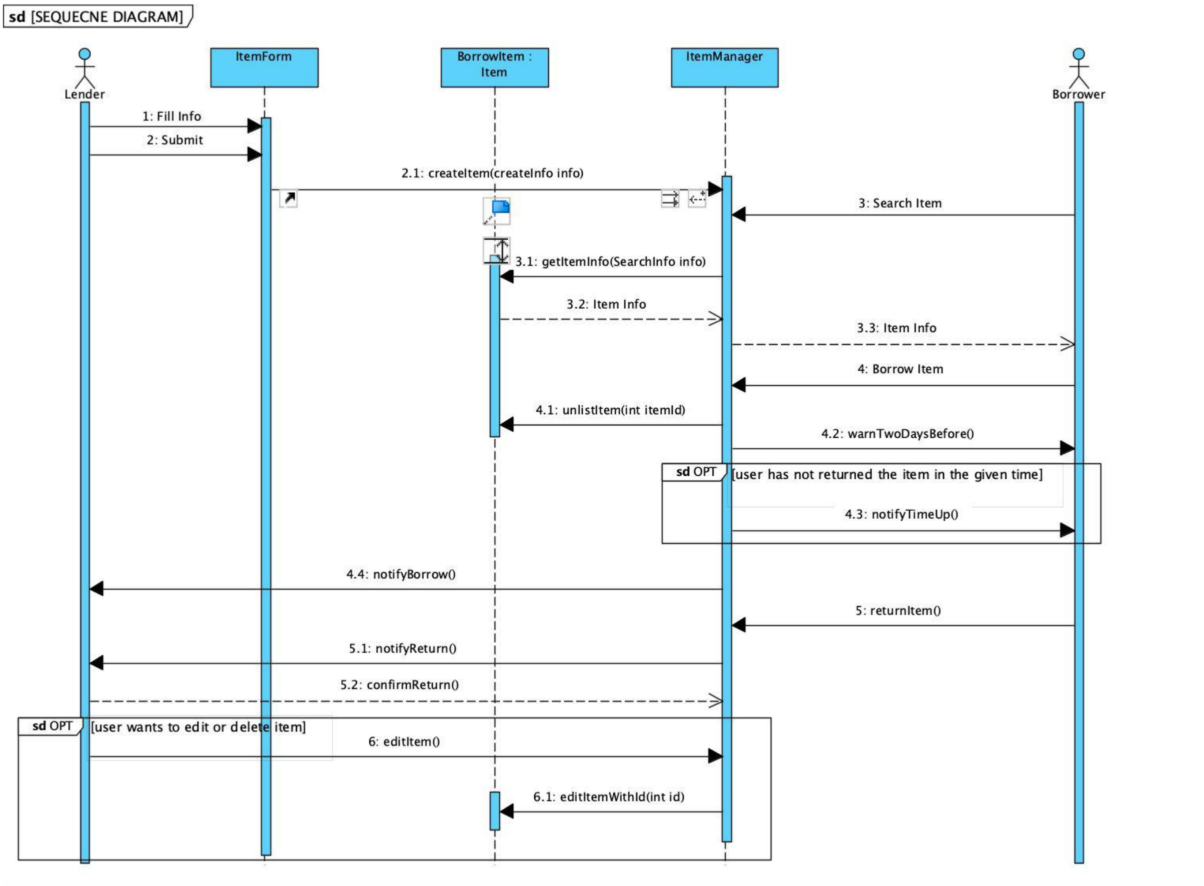


Fig. 6: Sequence Diagram for Lend-Borrow Functionality

A lender can fill an item form and submit it for borrowing , which will create a borrow listing. Borrowers can search listings, and can click the Borrow button on any listing they wish to borrow. After a borrower presses the Borrow button, the listing will be unlisted and the lender will be notified of the action. Additionally, lenders can also edit or delete listings if they choose to do so from their own listings. When the return date for the item comes and if the borrower still has not returned the item, they will be automatically notified. Upon the return of the item, the borrower can press the Return button on the listing, which will ask for a confirmation from the Lender to authenticate safe item returns.

7. Activity Diagrams

7.1. Activity Diagram for Buying and Selling an Item

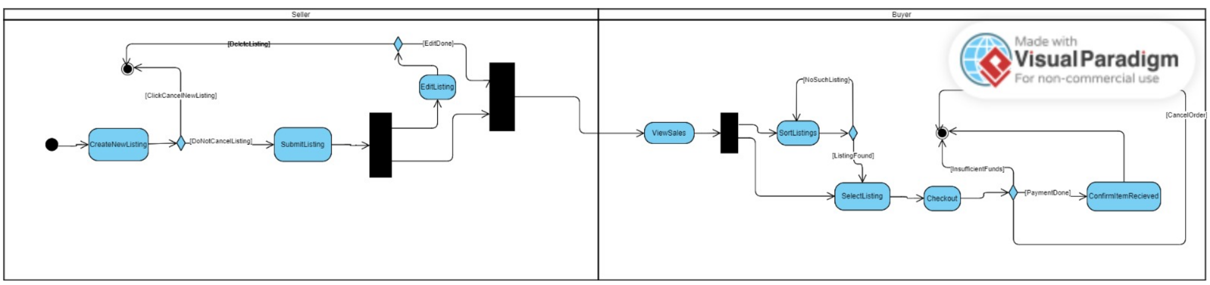


Fig. 7: Activity diagram for buying and selling function.

Seller users can create new listings(and submit it) and edit or remove their old listings(if they exist) from the "Create New Item" panel or "My Listings" panel respectively. The buyer users can view and browse the listing of other users, and filter their search. If the buyer finds something they want to purchase they can select the item they wish to purchase and after payment is received using our "Money Transaction" module mentioned above the buyer and seller will be messaging each other to set a time to give and receive the item respectively. After confirmation of the item exchange the process will be completed and the listing of the item will be removed from the "Sell & Buy" module.

7.2. Activity Diagram for Donating and Receiving an Item

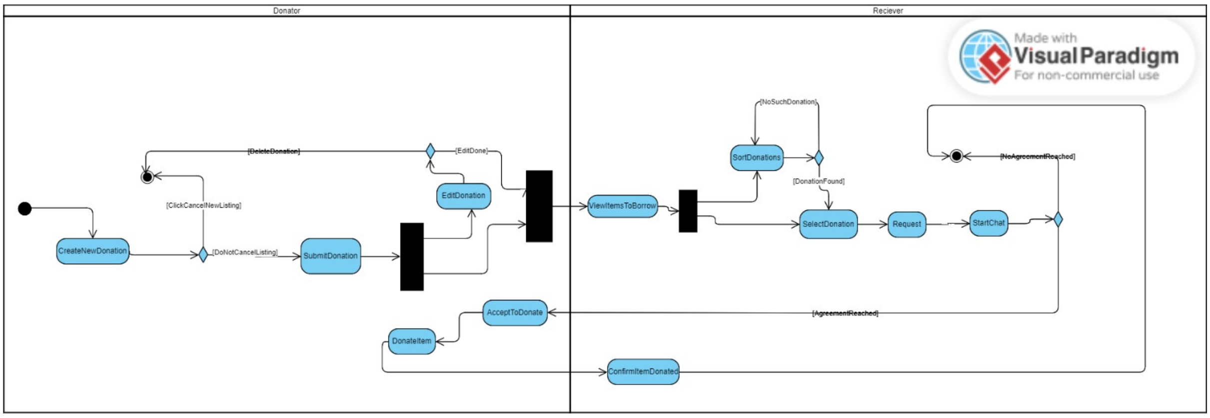


Fig. 8: Activity diagram for donation function.

Donator users (similar to seller users) can submit a new listing or edit or remove their previous listings. Receiver users can browse and filter listings of other donor users. If the receiver users wish to receive a donation they will submit a request which will be redirected to the donator. The donor has the right to accept or reject the request but the receiver user can send messages to the donator to explain themselves why they wish to get this item(e.g. financial inability). If the donator accepts the request they will use the messaging platform of the application and select a place and time to give and receive item respectively. After confirmation from the both parties this process will be completed and the listing will be removed from the "Donate & Receive" module, however if the donator rejects the request after notifying the receiver user the process will be ended without removing the listing.

7.3. Activity Diagram for Lending and Borrowing an Item

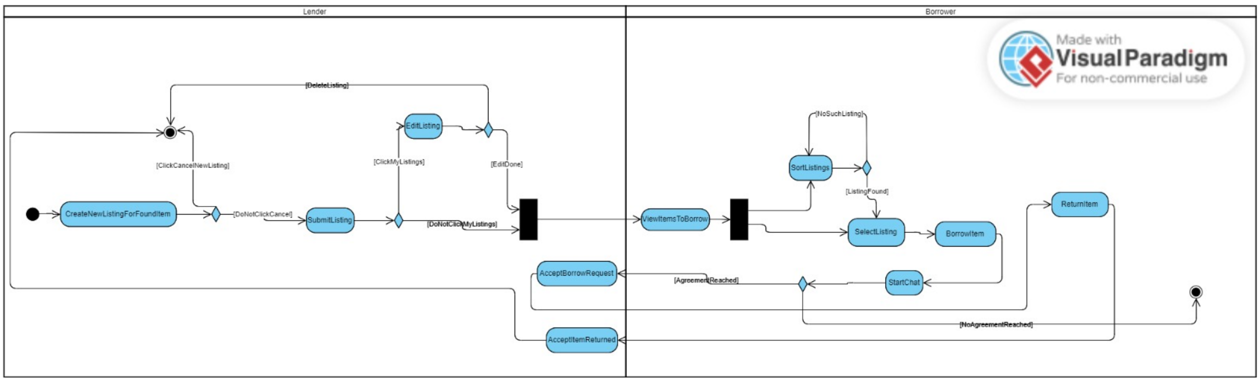


Fig. 9: Activity diagram for borrowing and lending function.

Lender users can create new listings and edit or remove their old listings. The borrower users can browse or filter listings of other users. If they wish to borrow an item they will submit a request which will be redirected to the lender user. The lender user has the right to reject or accept a request. The borrower user can message the lender to explain themselves to convince the lender. If the lender accepts the request both parties will set a time limit of how long the item will be lended and set a time and place to exchange the item. The two exchanges will be the lending of the item to the borrower and returning the item to the lender. During the exchanges both parties will be prompted to confirm the exchange. After confirming the return of the item to the lender this process will be ended. If the lender rejects the request this process will be ended after notifying the borrower user that made the request. In both cases the item will not be removed from the "Borrow & Return" module, if the lender wishes to remove an item they can remove their listings using "My Listings" panel.

7.4. Activity Diagram for Losing and Finding an Item

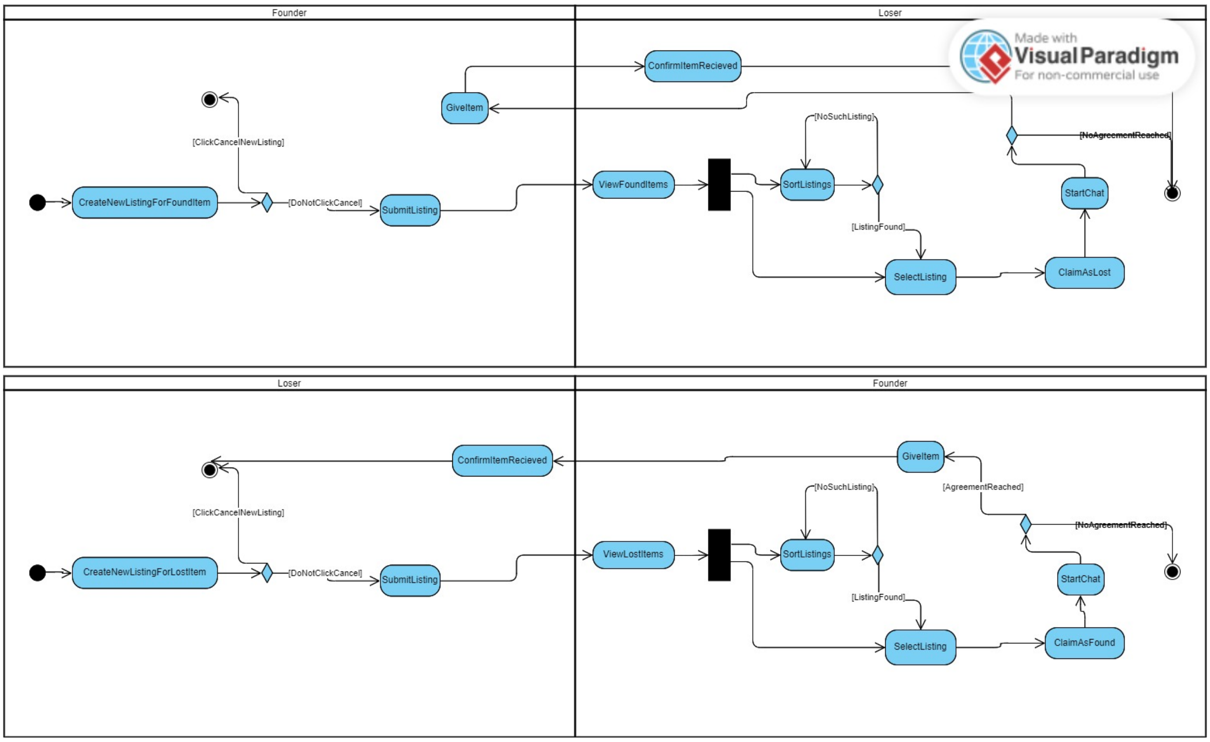


Fig. 10: Activity diagrams for lost and found function.

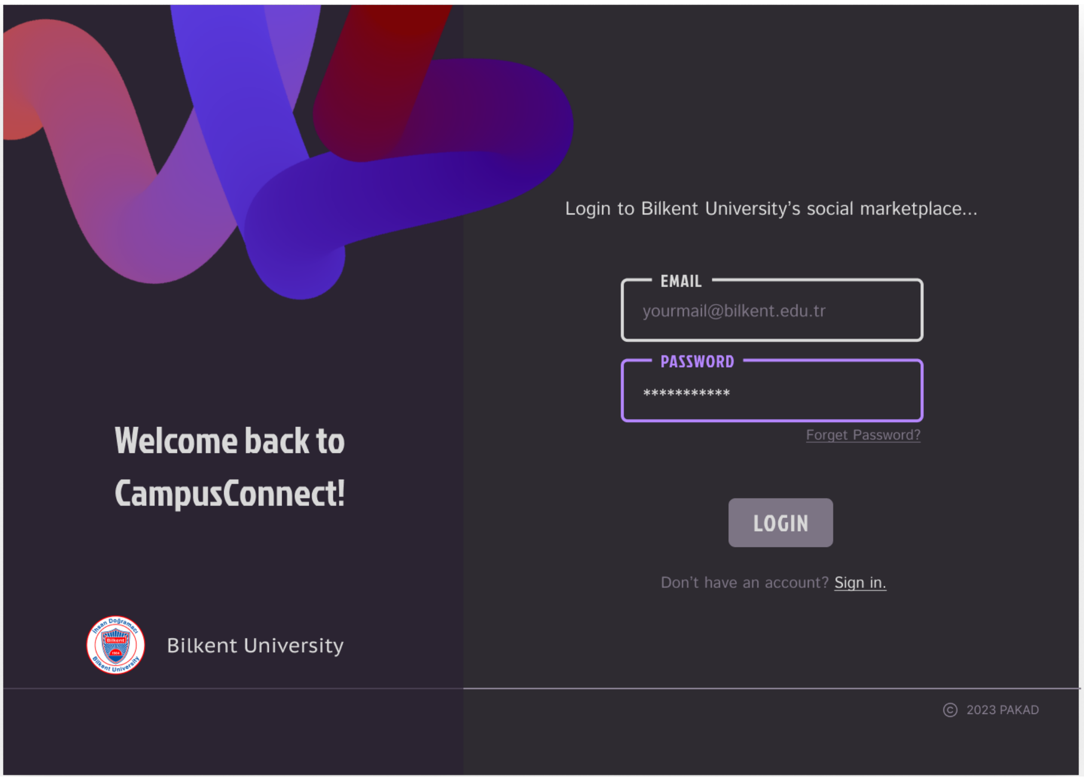
The “Lost & Found” module has two different processes.

Firstly, if a user finds an item they can submit a listing of the item that they found and the owner of the item (loser user) can view and filter the listings to find their lost item. If they find the item they can submit a request to receive the item and the loser user will be redirected to the founder user to message them and set a time and place for the exchanging of the item. During the exchange both parties will be prompted to confirm the exchange of the item and after confirmation the process will end. Listing of the item will be removed from the “Lost & Found” module.

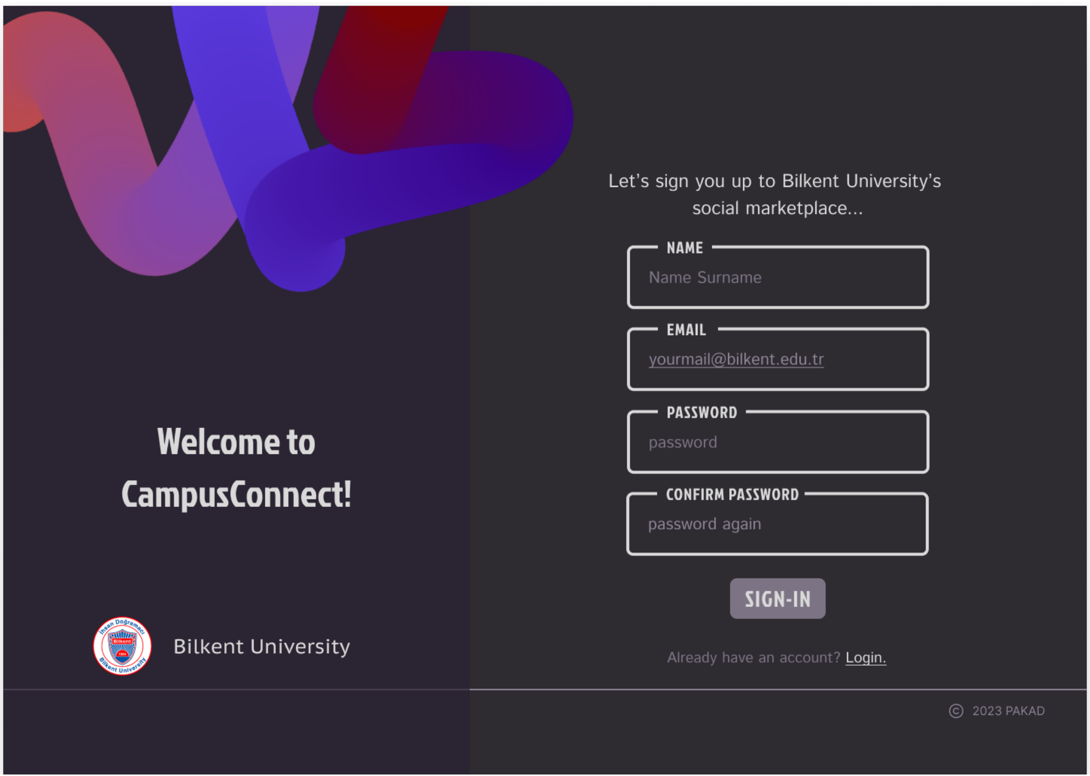
The other variant is if a user (loser user) loses an item they can submit a lost item listing. If a founder finds that item they can browse and filter the listings to find the lost item listing. After the founder user claims the listing as found both parties will be notified and prompted to use the messaging platform of the application to set a time and place for the exchange of the item. During the exchange both parties will be prompted to confirm the exchange and after the confirmation the process will end and the listing will be removed from the “Lost & Found” module.

8. Mockups

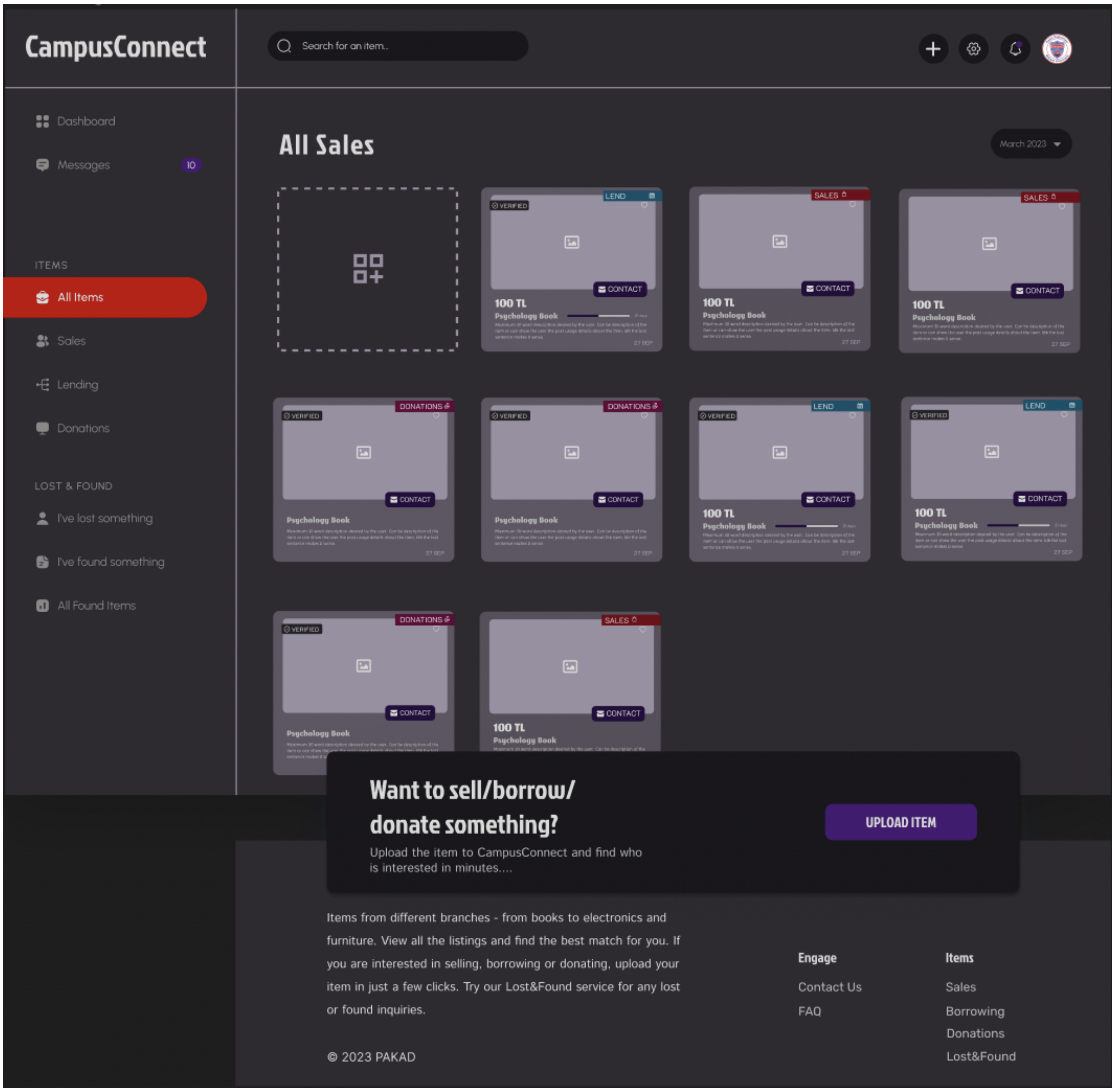
8.1. Login Page

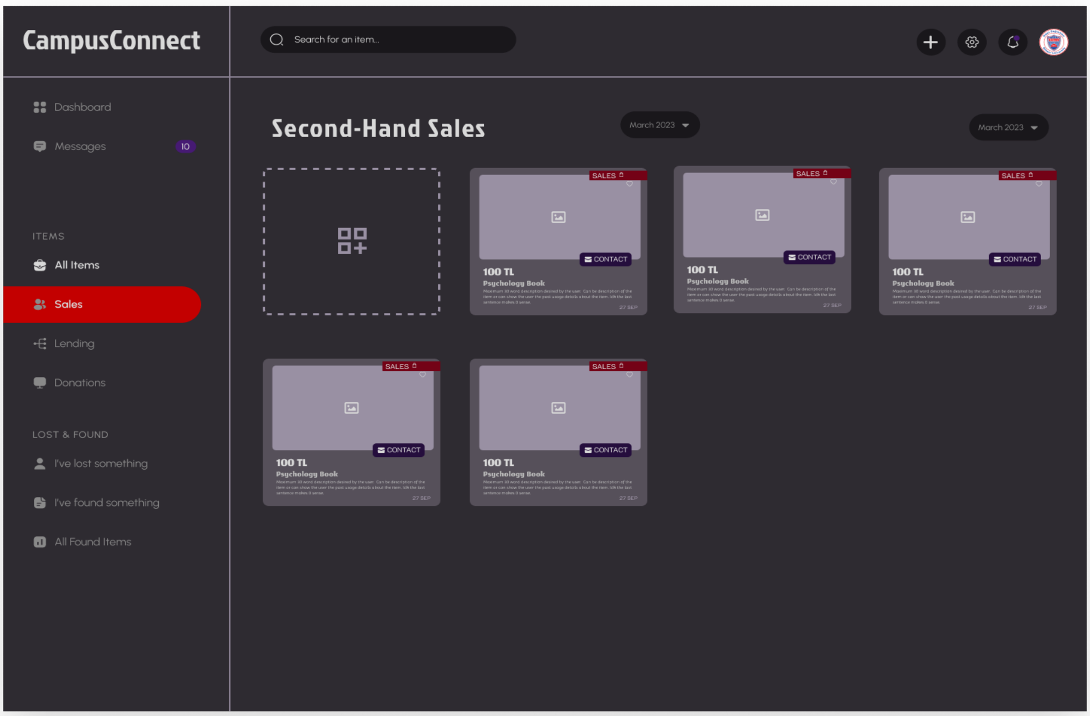


8.2. Signup Page

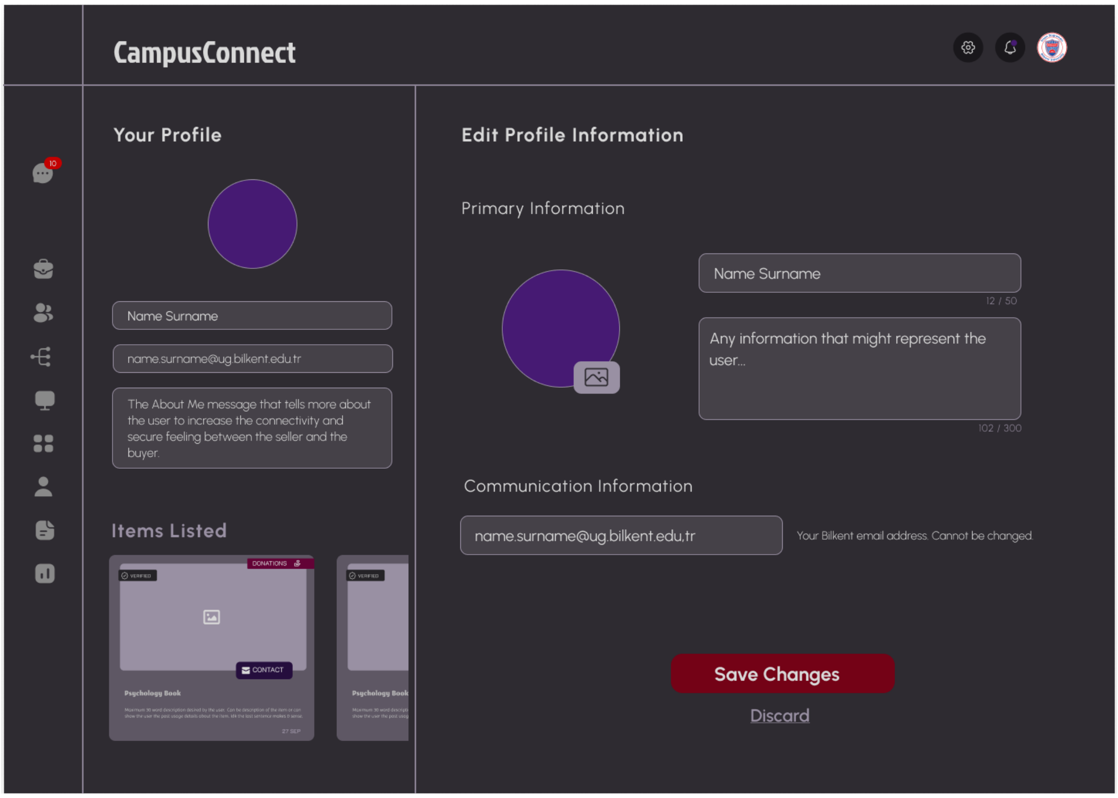


8.3. Main Page

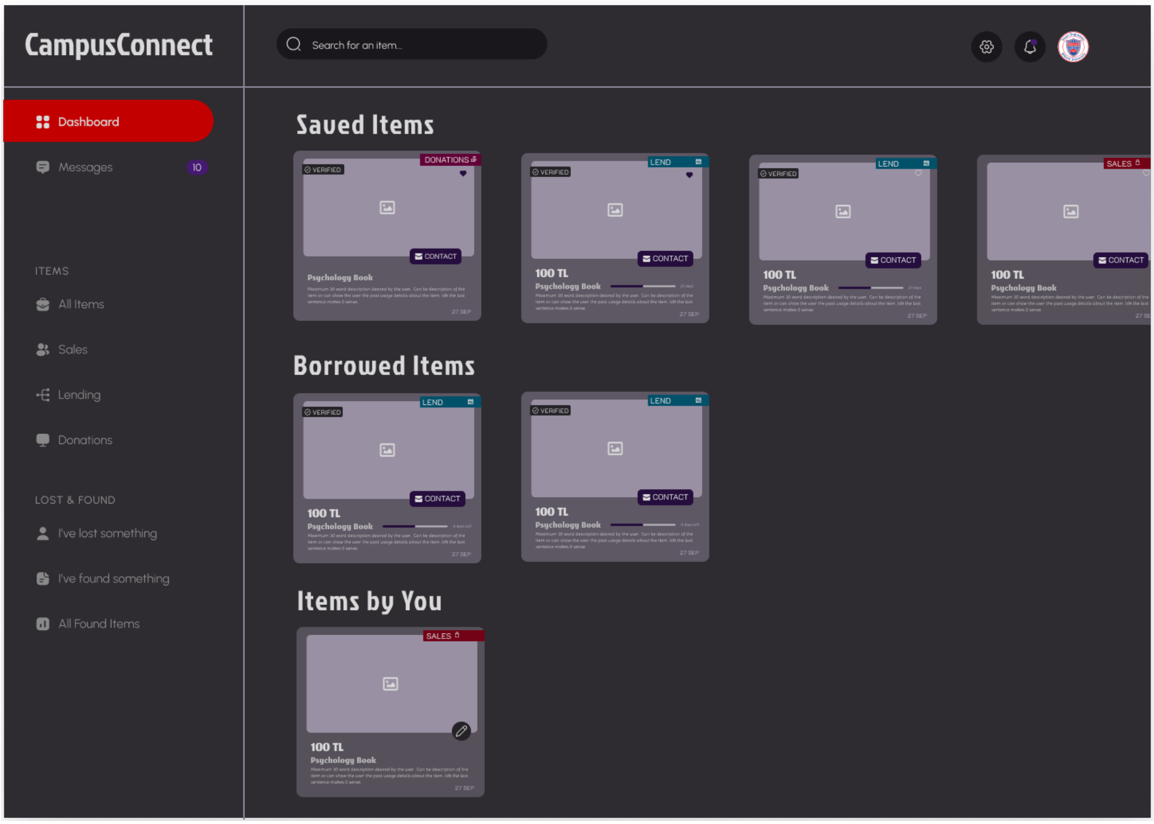

CampusConnect
Search for an item...
Dashboard
Messages
ITEMS
All Items
Sales
Lending
Donations
LOST & FOUND
I've lost something
I've found something
All Found Items
All Sales
March 2023
100 TL
Psychology Book
Want to sell/borrow/ donate something?
Upload the item to CampusConnect and find who is interested in minutes....
UPLOAD ITEM
Items from different branches - from books to electronics and furniture. View all the listings and find the best match for you. If you are interested in selling, borrowing or donating, upload your item in just a few clicks. Try our Lost&Found service for any lost or found inquiries.
© 2023 PAKAD
Engage
Contact Us
FAQ
Items
Sales
Borrowing
Donations
Lost&Found


CampusConnect
Search for an item...
Dashboard
Messages
ITEMS
All Items
Sales
Lending
Donations
LOST & FOUND
I've lost something
I've found something
All Found Items
Second-Hand Sales
March 2023
100 TL
Psychology Book

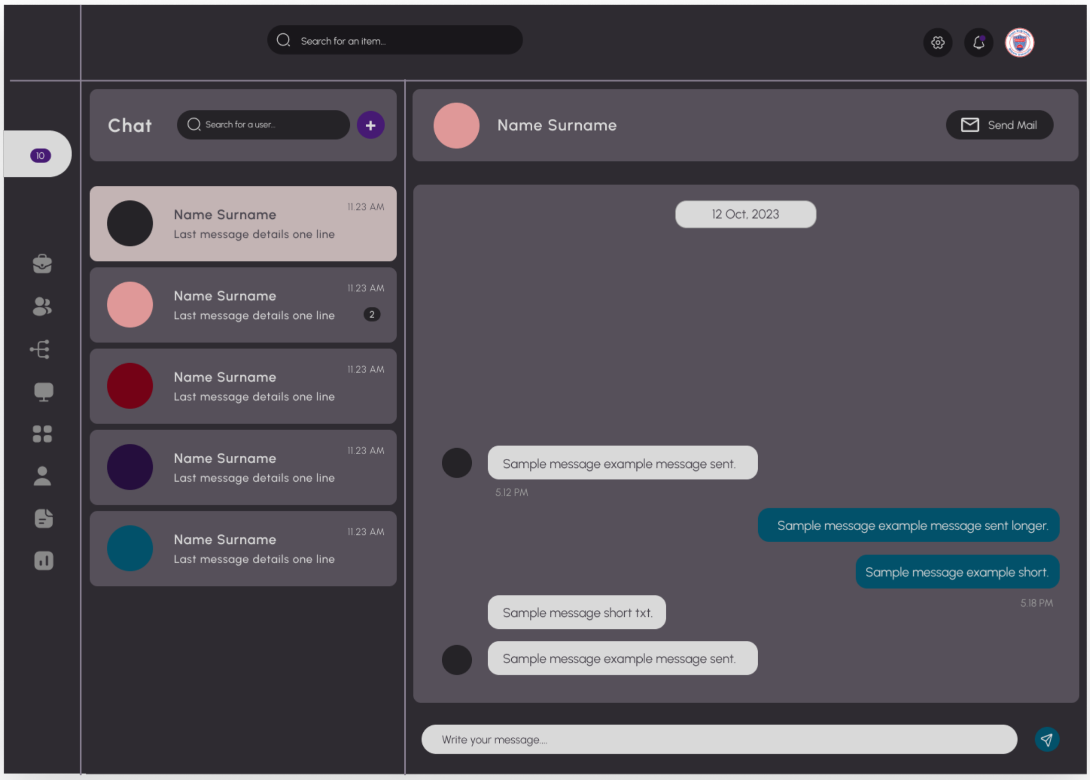
8.4. Profile Page



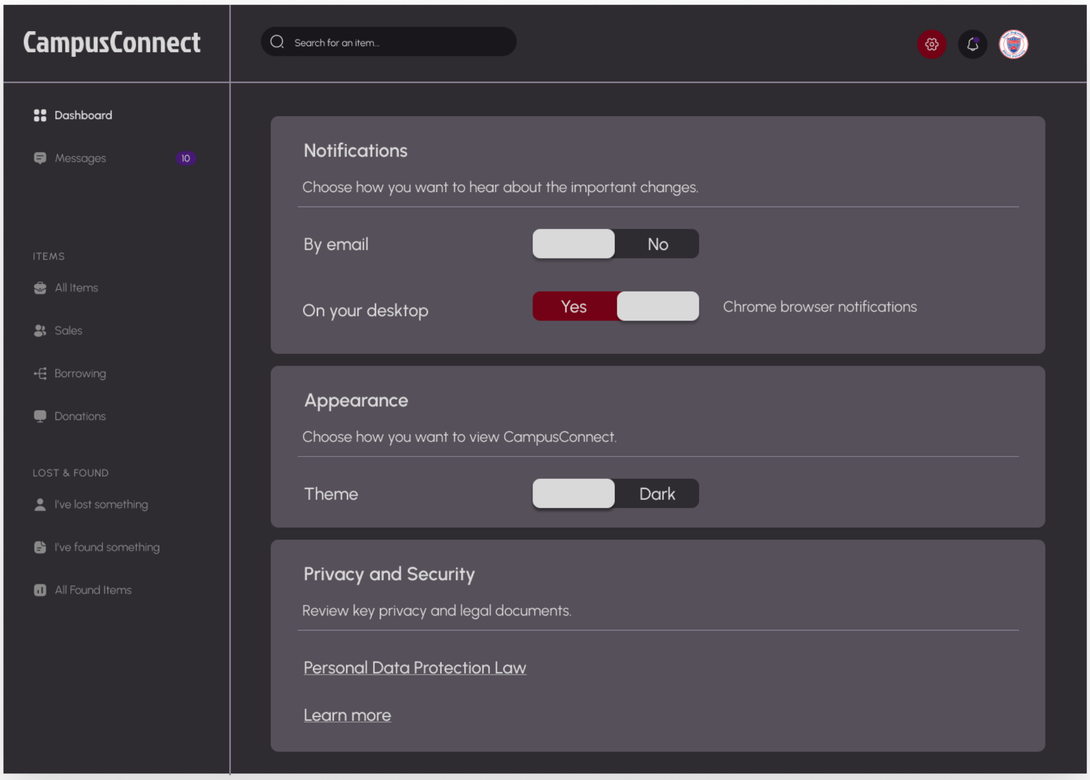
8.5. Dashboard Page



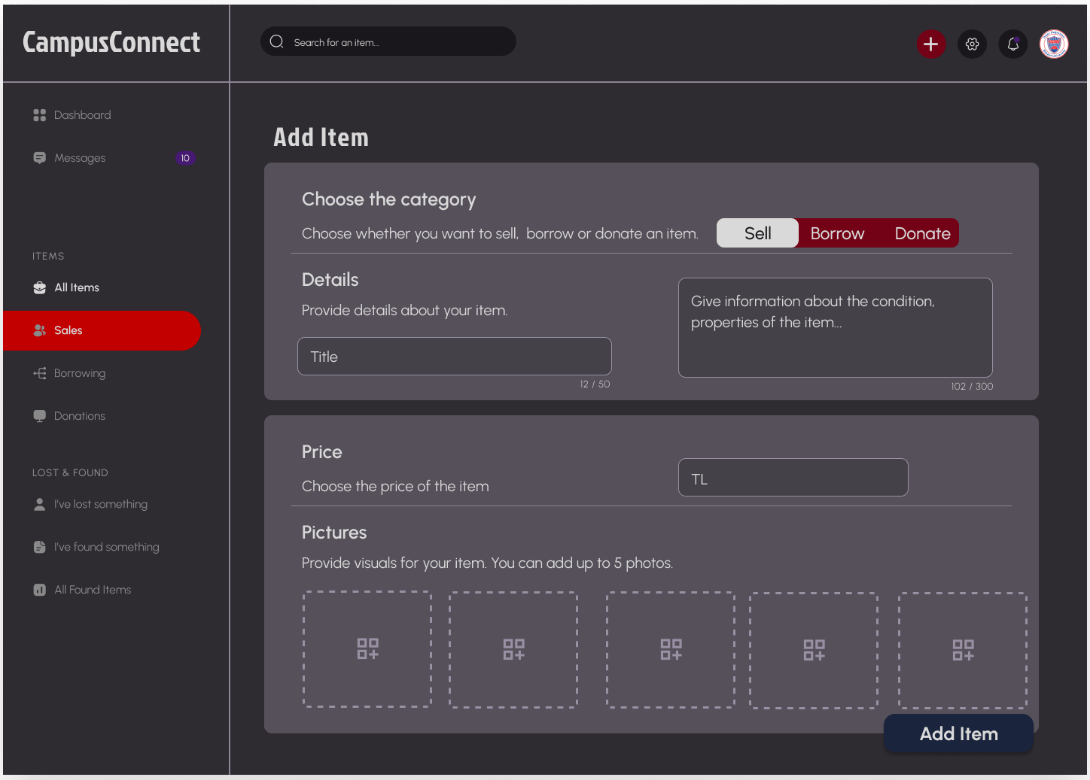
8.6. Messaging Page



8.7. Settings Page



8.8. Add Item



8.9. Lost&Found