Virtual Try-On

Software Design and Requirement Specification

Session: 2021 – 2025

Submitted by:

Sehrish Saddique 2021-SE-12

Kausar Fatima 2021-SE-25

Laiba Amber Ejaz 2021-SE-37

Supervised by:

Ma'am Alina

Department of Computer Science, New Campus

University of Engineering and Technology

Lahore, Pakistan

Declaration

Contents

List of Figures

List of Tables

Chapter 1

Software Requirement Specification

The project aims to develop a virtual dressing room called Virtual Tryon, which will allow users to try on clothes virtually on a screen inside a store without physically using fitting rooms. This will be achieved through augmented reality using a Kinect sensor and 3D models of clothing [1]. The system will be developed using the Unity 3D platform, also enabling users to change their backgrounds and save their photos. Additionally, the admin interface will be developed using .NET Framework, allowing admin to manage inventory. The system will be user-friendly, ensuring data validity and integrity.

A Software Requirements Specification (SRS) is a detailed document that outlines the functional and non-functional requirements of a software project. It is often concise, explains the desired functionality and behavior of the product, and typically created in collaboration with stakeholders in the early phases of the software development life cycle.
Following are the key components of SRS:
Functional Requirements: Detailed descriptions of the software's features, capabilities, and interactions.
Non-Functional Requirements: These include qualities like performance, reliability, scalability, security, and usability that the software must possess.

1.1 Functional Requirement

Functional requirements outline the essential behaviors of a system, detailing what it must do or avoid doing in response to inputs. They typically describe if/then scenarios, encompassing calculations, data input, and business processes. In essence, functional requirements represent the product features that prioritize user needs. The functional requirements of our project include:

1.1.1 Administrative Requirements

These are the actions the admin can perform in the system. Table 1.1 presents the administrative requirements of our project.

TABLE 1.1: Administrative Requirements

ID	Name	Description	Priority
FR-01-01	Sign-Up	The admin shall be able to register.	1
FR-01-02	Log-In	The admin shall be able to login.	1
FR-01-03	Log-Out	The admin shall be able to logout.	1
FR-01-04	View-Profile	The admin shall be able to view profile.	1
FR-01-05	Edit-Profile	The admin shall be able to edit profile.	1
FR-01-06	Add-Categories	The admin shall be able to add categories.	1
FR-01-07	View-Categories	The admin shall be able to view categories.	1
FR-01-08	Delete-Categories	The admin shall be able to delete categories.	1
FR-01-09	Edit-Categories	The admin shall be able to edit categories.	1
FR-01-10	Search-Categories	The admin shall be able to search categories.	1
FR-01-11	Add-Items	The admin shall be able to add items.	1
FR-01-12	View-Items	The admin shall be able to view items.	1
FR-01-13	Delete-Items	The admin shall be able to delete items.	1

FR-01-14	Edit-Items	The admin shall be able to edit items.	1
FR-01-15	Search-Items	The admin shall be able to search items.	1

1.1.2 User requirements

These are the actions the user can perform in the system. Table 1.2 presents the user requirements of our project.

TABLE 1.2: User Requirements

ID	**Name**	**Description**	**Priority**
FR-02-01	Accept-Terms	The user shall be able to accept terms and conditions.	1
FR-02-02	View-Categories	The user shall be able to view categories.	1
FR-02-03	Select-Categories	The user shall be able to select categories.	1
FR-02-04	View-Items	The user shall be able to view items within categories.	2
FR-02-05	Select-Items	The user shall be able to select items.	2
FR-02-06	View-Item-Details	The user shall be able to view details of selected items.	2
FR-02-07	Navigation	The user shall be able to navigate through the application using hand gestures.	1
FR-02-08	Try-On-Clothing	The user shall be able to view themselves wearing 3D clothing.	2
FR-02-09	Change-Background	The user shall be able to change the background.	2
FR-02-10	Save-Picture	The user shall be able to save the picture.	2

1.1.3 System requirements

These include software and hardware specifications, as well as system responses. Table 1.3 presents the system requirements of our project.

TABLE 1.3: System Requirements

ID	Name	Description	Priority
FR-04-01	Sensor	The system shall have a Kinect v2 sensor attached.	1
FR-04-02	RAM	The system shall have at least 8GB of RAM.	1
FR-05-01	Operating-System	The system shall have Windows 10 installed.	1

1.2 Non-functional Requirement

Non-functional requirements outline how the system operates without impacting its core functionality. They define the overall quality, performance, scalability, availability, security and usability of a system. Table 1.4 presents the non-functional requirements of our project:

TABLE 1.4: Non-Functional Requirements

ID	Name	Description	Priority
NR-01-01	Usability	The system must have a clear, concise, and intuitive UI, visually appealing layout, and optimized button size according to the screen.	1
NR-01-02	Performance	The system must minimize latency times to ensure real-time performance.	3
NR-01-03	Scalability	The system must be capable of handling more categories and items.	1
NR-01-04	Availability	The system must be highly available and reliable.	2
NR-02-01	Data-Integrity	The system must maintain data integrity by implementing validation and verification processes for all data inputs and updates.	1

NR-02-02	Data-Accuracy	The system must ensure data accuracy by applying consistent standards and checks during data entry, processing, and storage.	1
NR-02-03	Security	The system must not share data with third-party apps without consent.	2
NR-03-01	Portability	The system must be desktop-based.	1

Chapter 2

Design specification

2.1 System Behavioral Design

Behavioral diagrams offer a dynamic perspective of a system, showcasing its behavior and functions over time. They outline the interactions and processes within the system, shedding light on its operational aspects.

2.1.1 Use case Diagram

Use case diagrams provide a visual overview of the actors in a system, the functions they require, and the interactions between these functions. They are usually created in the early stages of development and serve as an important communication tool among stakeholders, enabling them to understand the primary functions of the system and how different users will interact with it.
Figure 2.1 presents the use case diagram for the system, outlining the key interactions between users and the system's functionalities, including login, item selection, virtual try-on, and category and item management, thereby providing a comprehensive view of the system's use cases.

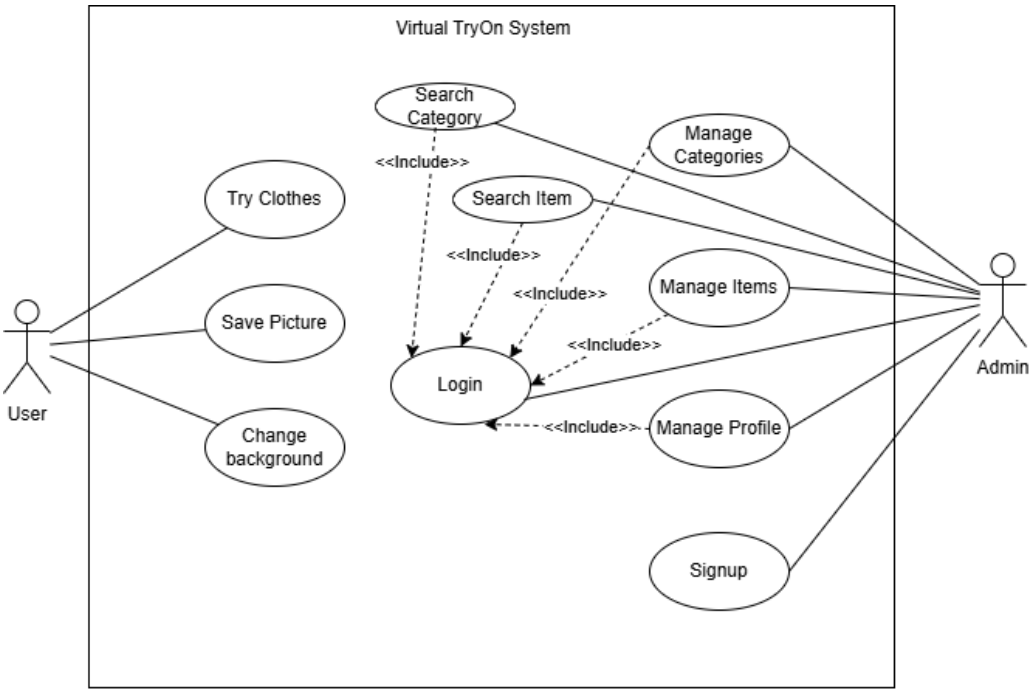

FIGURE 2.1: Use-Case diagram

2.1.1.1 Admin Signup

Table 2.1 presents the description of signup use case. This use case outlines the steps an administrator must follow to create a new account in the system.

TABLE 2.1: SignUp

Use Case	**Sign-Up**
ID	1
Actor	Admin
Pre-condition	The admin must have access to the system and the system must be functional.
Goal	The goal of this use case is to allow the admin to create an account on the system.
Trigger	The admin adds details to create an account on the system.
Post-condition	The admin has an account on the system.
Exception	If the inputs are missing, the admin will not be able to create an account.
Alternate Flow (Database)	Admin accesses the database and creates an account in the database.

Main Flow	1. The admin opens the system and clicks on the sign-up button. 2. The system presents a form to the admin to fill in their details such as name, email, password. 3. The admin fills in the required details and submits the form. 4. The system validates the details and creates an account for the admin.

2.1.1.2 Admin Login

Table 2.2 presents the description of the login use case, outlining the steps and requirements for admin to access the system.

Table 2.2: Login

Use Case	**Login**
ID	2
Actor	Admin
Pre-condition	The admin must have an account on the system and the system must be functional.
Goal	Allow the admin to log in to the system.
Trigger	The admin requests to log in to the system.
Post-condition	The admin has successfully logged in to the system.
Exception	If the admin's credentials are incorrect or the account is not already registered, the admin will not be able to log in to the system.
Alternate Flow (Incorrect-Credentials)	1. If the credentials are incorrect: 2. The system displays an error message. 3. Admin is prompted to re-enter the correct credentials.

Main Flow	1. The admin opens the system and clicks on the login button. 2. The system presents a form to the admin to fill in their credentials such as email and password. 3. The admin fills in their credentials and submits the form. 4. The system validates the credentials and grants the admin access to the system.

2.1.1.3 Admin Manage Category

Table 2.3 presents the description of the admin manage category use case, detailing the steps and actions the admin can take to oversee categories within the system.

TABLE 2.3: Manage Category

Use Case	**Manage Category**
ID	3
Actor	Admin
Pre-condition	The admin must be logged in to the system and the system must be functional.
Goal	The goal of this use case is to allow the admin to manage categories.
Trigger	The admin requests to manage categories.
Post-condition	The category has been managed (added, deleted, updated, viewed) by the system.
Exception	1. If the connection to the database fails, the categories would not be managed.
Alternate Flow (Database)	Admin accesses the database and manages the categories in the database.
Main Flow	1. The admin clicks on the category button. 2. The system presents a category view page. 3. The admin manages categories by adding, deleting, updating categories, viewing, or managing stock within categories. 4. The system validates the details and sends data to the database.

2.1.1.4 Admin Manage Item

Table 2.4 presents the description of the admin manage item use case, detailing the steps and actions the admin can take to oversee items within the system.

TABLE 2.4: Manage Item

Use Case	**Manage Item**
ID	4
Actor	Admin
Pre-condition	The admin must be logged in to the system and the system must be functional.
Goal	The goal of this use case is to allow the admin to manage items.
Trigger	The admin requests to manage items.
Post-condition	The item has been managed (added, deleted, updated, viewed) by the system.
Exception	1. If the connection to the database fails, the items would not be managed.
Alternate Flow (Database)	Admin accesses the database and manages the items in the database.
Main Flow	1. The admin clicks on the item button. 2. The system presents item view page. 3. The admin manages items by adding, deleting, updating items, viewing, or managing stock within items. 4. The system validates the details and sends data to the database.

2.1.1.5 Admin Search Category

Table 2.5 presents the description of the admin search category use case, detailing the steps and actions the admin can take to search categories within the system.

TABLE 2.5: Search Category

Use Case	**Search Category**
ID	5
Actor	Admin

Pre-condition	The admin must be logged in to the system and the system must be functional.
Goal	The goal of this use case is to allow the admin to search categories.
Trigger	The admin requests to search categories.
Post-condition	The category has been retrieved by the system.
Exception	1. If the connection to the database fails, the categories would not be retrieved.
Alternate Flow (Database)	Admin accesses the database and searches the categories in the database.
Main Flow	1. Admin logs into the system. 2. Admin navigates to the Search Categories section. 4. Admin enters the desired category. 5. Admin initiates the search. 6. System retrieves and displays a list of categories matching the desired search. 7. Admin reviews the search results.

2.1.1.6 Admin Search Item

Table 2.6 presents the description of the admin search item use case, detailing the steps and actions the admin can take to search items within the system.

Table 2.6: Search Item

Use Case	**Search Item**
ID	6
Actor	Admin
Pre-condition	The admin must be logged in to the system and the system must be functional.
Goal	The goal of this use case is to allow the admin to search items.
Trigger	The admin requests to search items.
Post-condition	The item has been retrieved by the system.
Exception	1. If the connection to the database fails, the items would not be retrieved.

Alternate Flow (Database)	Admin accesses the database and searches the items in the database.
Main Flow	1. Admin logs into the system. 2. Admin navigates to the Search items section. 3. Admin enters the desired item. 4. System retrieves and displays a list of items matching the desired search. 5. Admin reviews the search results.

2.1.1.7 Admin Manage Profile

Table 2.7 presents the description of the admin manage profile use case, outlining the actions the admin can perform to update and maintain profile within the system.

TABLE 2.7: Manage Profile

Use Case	**Manage Profile**
ID	7
Actor	Admin
Pre-condition	The admin must be logged in to the system and the system must be functional.
Goal	The goal of this use case is to allow the admin to manage their profile.
Trigger	The admin requests to manage their profile.
Post-condition	The system displays the profile to the admin.
Exception	If the connection to the database fails, the details would not be retrieved.
Alternate Flow (Database)	Admin accesses the database and manages their profile in the database.
Main Flow	1. The admin clicks on the profile button. 2. The system presents profile details to the admin. 3. The admin edit profile. 4. The system validates the details and sends to the database.

2.1.1.8 Admin Logout

Table 2.8 presents the description of the admin logout use case, detailing the steps the admin must take to securely log out of the system.

Table 2.8: Logout

Use Case	**Logout**
ID	8
Actor	Admin
Pre-condition	The admin must be logged in to the system.
Goal	The goal of this use case is to allow the admin to log out of the system.
Trigger	The admin requests to log out of the system.
Post-condition	The admin has been logged out of the system.
Exception	Error in logging out.
Alternate Flow	N/A
Main Flow	1. The admin clicks on the logout button. 2. The system logs the admin out of the system.

2.1.1.9 Try Clothes

Table 2.9 presents the description of the visualize dress use case, detailing how users can view and interact with clothing items in the virtual environment.

Table 2.9: Try Clothes

Use Case	**Try Clothes**
ID	9
Actor	User
Pre-condition	The user must have accepted the terms and conditions, have access to the virtual try room, have selected a category, and have selected an item.
Goal	Allow the user to select a dress item to try on in the virtual tryroom.
Trigger	The user wants to try on a specific dress item from the selected category.

Post-condition	The user has successfully selected a dress item and tried it on in the virtual tryroom.
Exception	If the user goes out of the sensor range, he will not be able to virtually try a dress item.
Main Flow	1. The user has accepted the terms and conditions and has access to the virtual tryroom. 2. The user selects a category from the list of available categories. 3. The system retrieves and displays a list of dress items within the selected category. 4. The user views the list of items within the selected category. 5. The user selects a dress item from the list. 6. The system confirms the user's selection and prepares the dress item for virtual try-on. 7. The user successfully views himself in virtual dress.

2.1.1.10 User Save Image

Table 2.10 presents the description of the user save image use case, detailing the process for users to store images in the system.

TABLE 2.10: Save Image

Use Case	**Save Image**
ID	10
Actor	User
Pre-condition	The user must have accepted the terms and conditions, have access to the virtual tryroom, and have tried on an item.
Goal	Allow the user to save an image of themselves with the item tried-on.
Trigger	The user wants to save a photo of themselves with the tried-on item.
Post-condition	The user has successfully saved the photo.
Exception	N/A

Main Flow	1. The user has accepted the terms and conditions and has access to the virtual tryroom. 2. The user selects an item from the list of available categories. 3. The user tries on the selected item in the virtual tryroom. 4. The user decides to save an image of themselves with the item tried on. 5. The user selects the save image button.

2.1.1.11 User Change Background

Table 2.11 presents the description of the user change background use case, detailing the process for users to change background.

TABLE 2.11: Change Background

Use Case	**Change Background**
ID	10
Actor	User
Pre-condition	The user must have accepted the terms and conditions, have access to the virtual tryroom, and have selected a category.
Goal	Allow the user to change the background while trying on clothing items in the virtual tryroom.
Trigger	The user wants to change the background environment to see how the clothing item looks in different settings.
Post-condition	The user successfully changes the backround, and the new background is displayed.
Exception	If the user goes out of the sensor range, they will not be able to change the background.

Main Flow	1. The user has accepted the terms and conditions and has access to the virtual tryroom. 2. The user selects a clothing item from the list of available categories. 3. The user decides to change the background. 4. The system displays a list of available backgrounds. 5. The user selects a background from the list. 6. The system changes the background to the selected one. 7. The user views the new background environment.

2.1.2 Activity Diagram

An activity diagram visually depicts the progression of activities, actions, and decisions in a system or process. The significance of activity diagrams lies in their ability to provide a clear and thorough representation of complex processes, which supports various facets of software development and system analysis.

2.1.2.1 Admin SignUp

Figure 2.2 presents the activity diagram for the admin signup side, illustrating the flow of actions that can be performed.

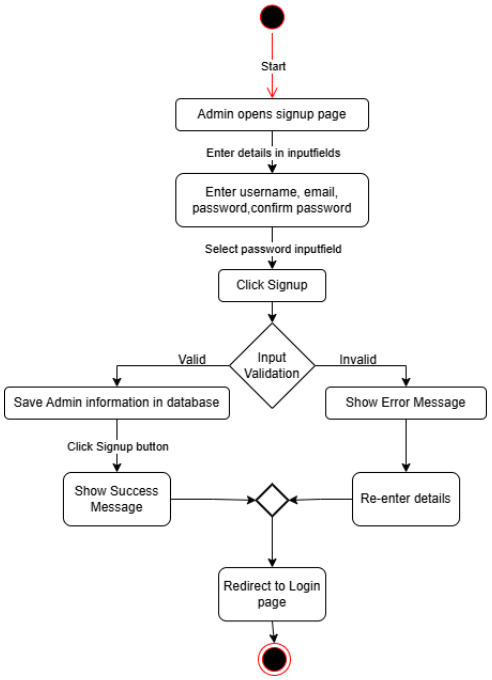

FIGURE 2.2: Activity Diagram:Admin SignUp

2.1.2.2 Admin Login

Figure 2.3 presents the activity diagram for the admin login side, illustrating the flow of actions that can be performed.

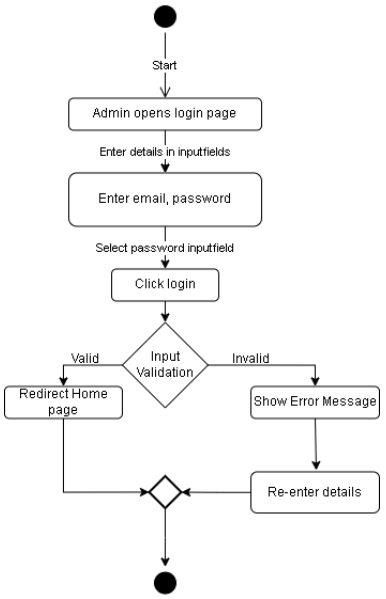

FIGURE 2.3: Activity Diagram:Admin Login

2.1.2.3 Admin View Dashboard

Figure 2.4 presents the activity diagram for the admin view dashboard side, illustrating the flow of actions that can be performed.

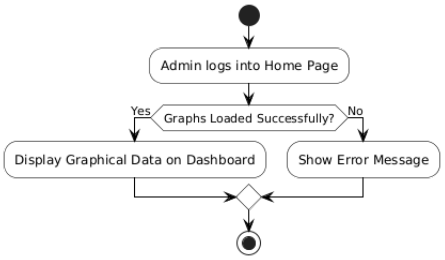

FIGURE 2.4: Activity Diagram:Admin View Dashboard

2.1.2.4 Admin CRUD Category

Figure 2.5 presents the activity diagram for the admin crud category side operations, illustrating the flow of actions that can be performed.

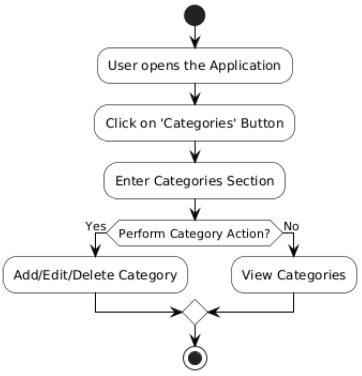

FIGURE 2.5: Activity Diagram:Admin CRUD Category

2.1.2.5 Admin Search Category

Figure 2.6 presents the activity diagram for the admin search category side operations, illustrating the flow of actions that can be performed.

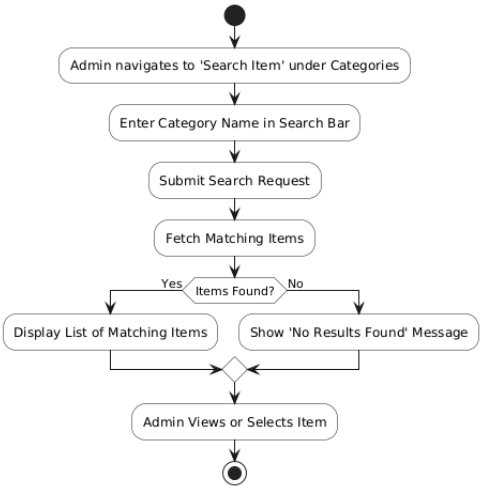

FIGURE 2.6: Activity Diagram:Admin Search Category

2.1.2.6 Admin CRUD Item

Figure 2.7 presents the activity diagram for the admin crud item side operations, illustrating the flow of actions that can be performed.

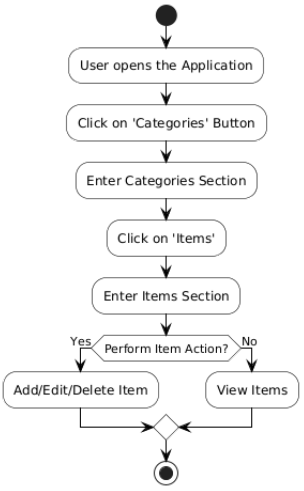

FIGURE 2.7: Activity Diagram:Admin CRUD Item

2.1.2.7 Admin Search Item

Figure 2.8 presents the activity diagram for the admin search item side operations, illustrating the flow of actions that can be performed.

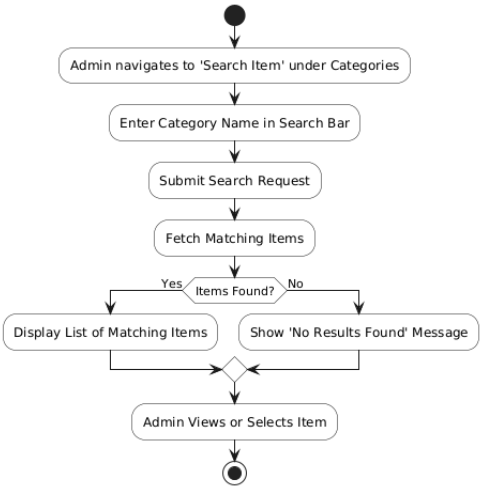

FIGURE 2.8: Activity Diagram:Admin CRUD Item

2.1.2.8 Admin Edit Profile

Figure 2.9 presents the activity diagram for the admin edit profile side operations, illustrating the flow of actions that can be performed.

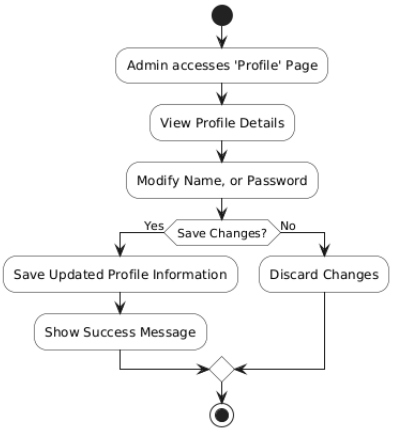

FIGURE 2.9: Activity Diagram:Admin Edit Profile

2.1.2.9 User Activity

Figure 2.10 presents the activity diagram for the user dashboard, illustrating the flow of actions such as accepting terms, selecting categories and items, thereby providing a clear overview of the various administrative tasks that can be performed.

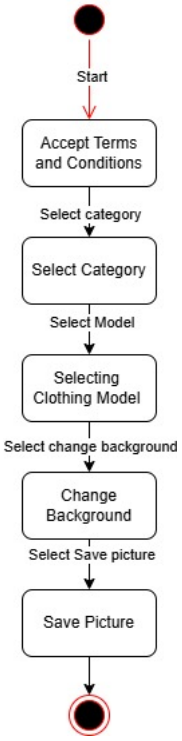

FIGURE 2.10: Activity Diagram:User

2.1.3 State Diagram

A state diagram visually depicts various states that a system or object can assume, the events that cause transitions between these states, and the actions linked to those transitions. The significance of state diagrams lies in their ability to model and illustrate the dynamic behavior and life cycle of objects or systems.

2.1.3.1 Admin State

Figure 2.11 presents the state diagram for the admin operations, depicting the various states the system transitions through as a result of actions performed by the admin, thereby illustrating the system's dynamic behavior during different actions.

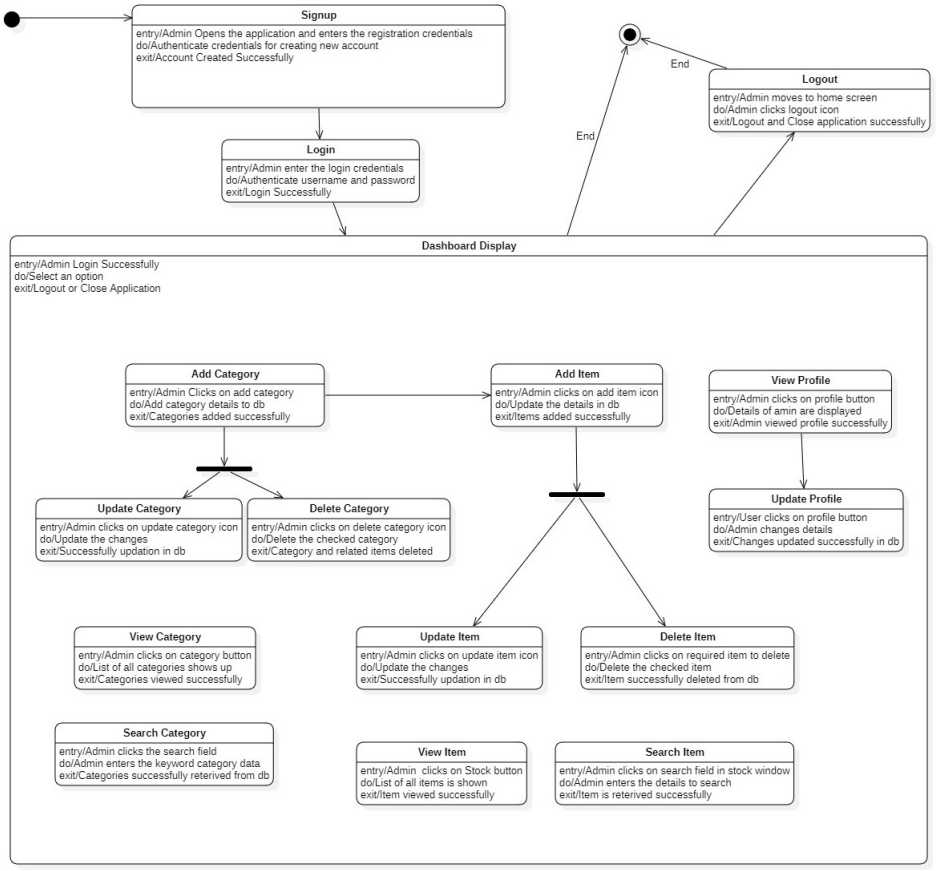

FIGURE 2.11: State Diagram:Admin

2.1.3.2 User State

Figure 2.12 presents the state diagram for the user operations, depicting the various states the system transitions through as a result of actions performed by the admin, thereby illustrating the system's dynamic behavior during different actions.

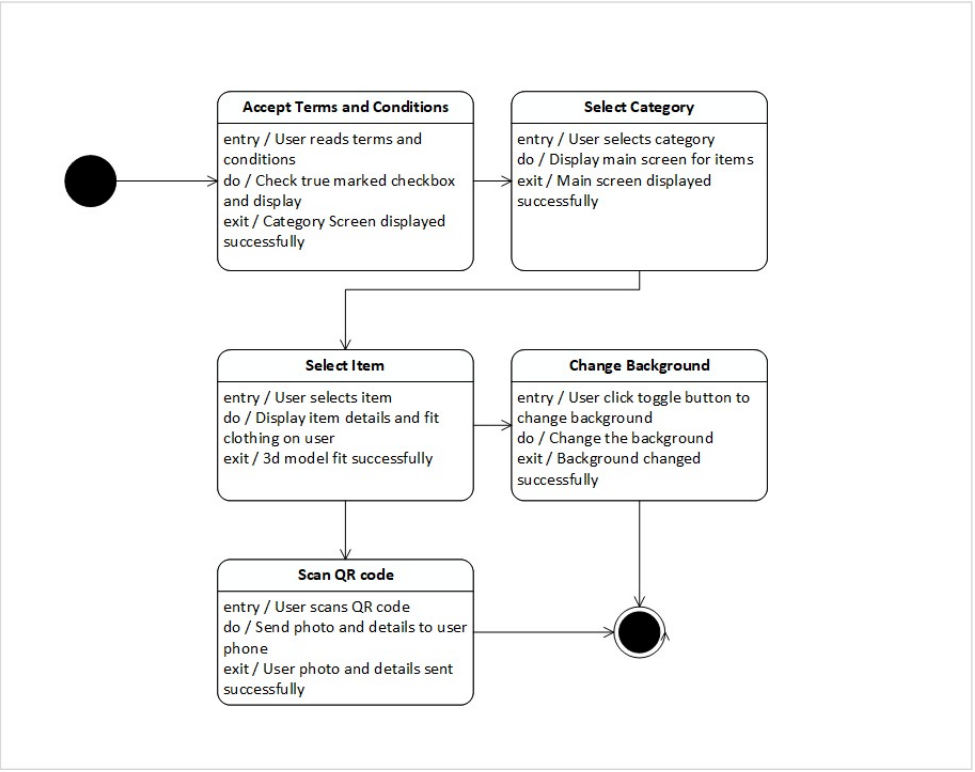

FIGURE 2.12: State Diagram:User

2.1.4 Sequence Diagram

A Sequence Diagram visually depicts how objects interact with each other in a specific scenario by depicting the sequence of interactions and processes. The significance of Sequence Diagrams lies in their ability to provide a clear and structured view of the order of interactions, with processes represented vertically and interactions shown as arrows.

2.1.4.1 Admin SignUp

Figure 2.13 presents the sequence diagram for the signup operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the signup functionalities.

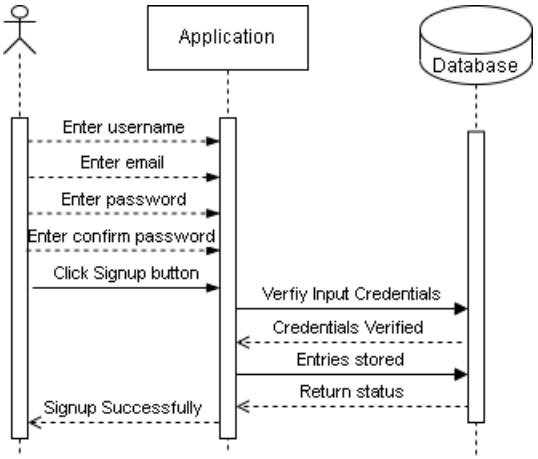

FIGURE 2.13: Sequence Diagram:Admin SignUp

2.1.4.2 Admin Login

Figure 2.14 presents the sequence diagram for the login operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the login functionalities.

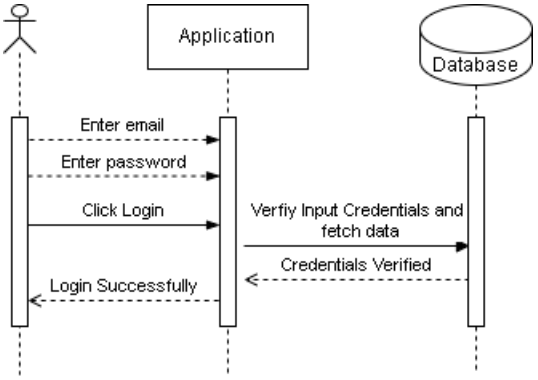

FIGURE 2.14: Sequence Diagram:Admin Login

2.1.4.3 Admin View Dashboard

Figure 2.15 presents the sequence diagram for navigating to dashboard operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the viewing dashboard functionalities.

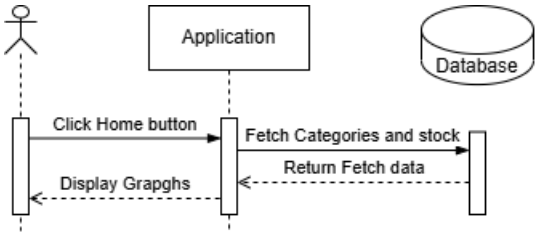

FIGURE 2.15: Sequence Diagram:Admin View Dashboard

2.1.4.4 Admin Add Category

Figure 2.16 presents the sequence diagram for adding category operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the add category functionalities.

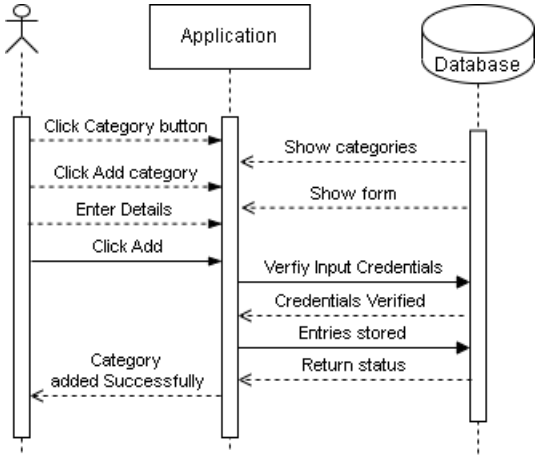

FIGURE 2.16: Sequence Diagram:Admin Add Category

2.1.4.5 Admin Edit Category

Figure 2.17 presents the sequence diagram for editing category operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the edit category functionalities.

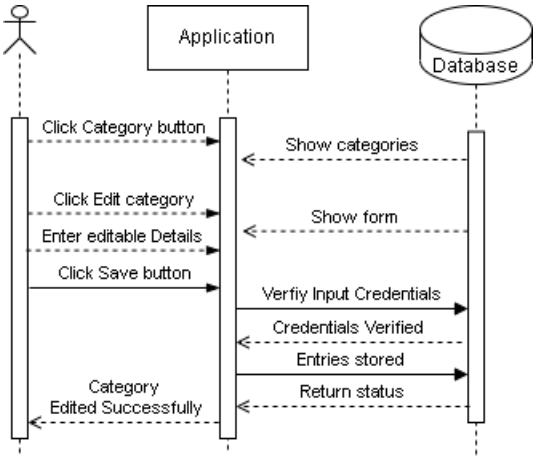

FIGURE 2.17: Sequence Diagram:Admin Edit Category

2.1.4.6 Admin Delete Category

Figure 2.18 presents the sequence diagram for deleting category operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the delete category functionalities.

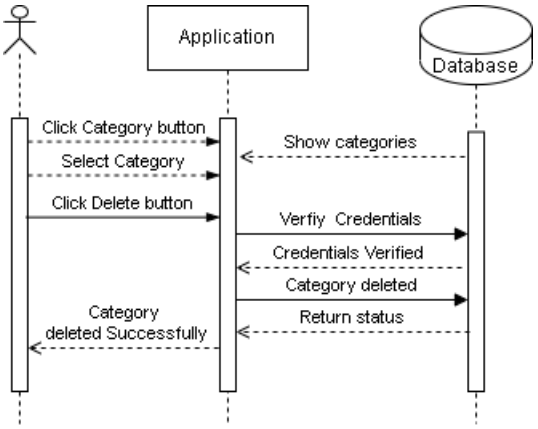

FIGURE 2.18: Sequence Diagram:Admin Delete Category

2.1.4.7 Admin Search Category

Figure 2.19 presents the sequence diagram for searching category operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the search category functionalities.

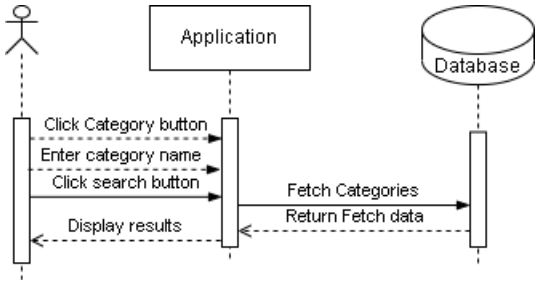

FIGURE 2.19: Sequence Diagram:Admin Search Category

2.1.4.8 Admin Add Item

Figure 2.20 presents the sequence diagram for adding item operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the add item functionalities.

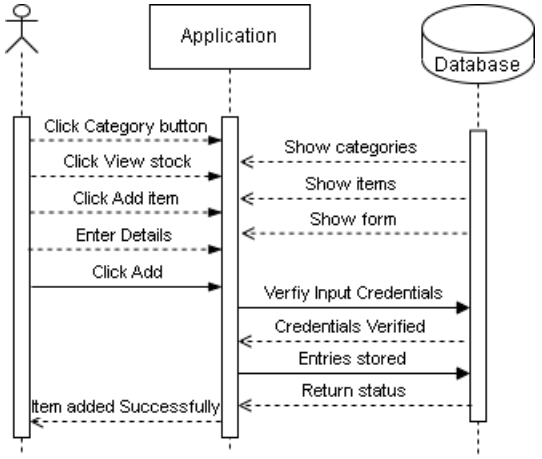

FIGURE 2.20: Sequence Diagram:Admin Add Item

2.1.4.9 Admin Edit Item

Figure 2.21 presents the sequence diagram for editing item operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the edit item functionalities.

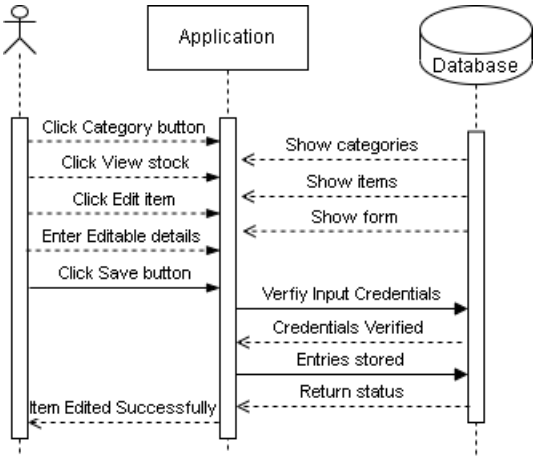

FIGURE 2.21: Sequence Diagram:Admin Edit Item

2.1.4.10 Admin Delete Item

Figure 2.22 presents the sequence diagram for deleting item operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the delete item functionalities.

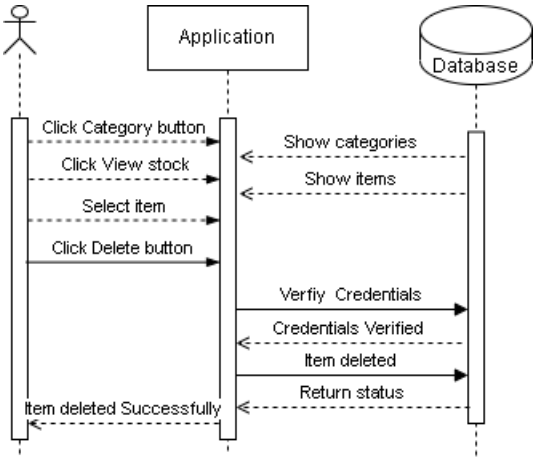

FIGURE 2.22: Sequence Diagram:Admin Delete Item

2.1.4.11 Admin Search Item

Figure 2.23 presents the sequence diagram for searching item operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the seaching item functionalities.

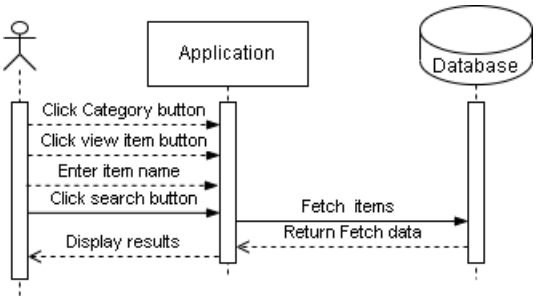

FIGURE 2.23: Sequence Diagram:Admin Search Item

2.1.4.12 Admin View Profile

Figure 2.24 presents the sequence diagram for viewing profile operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the viewing profile functionalities.

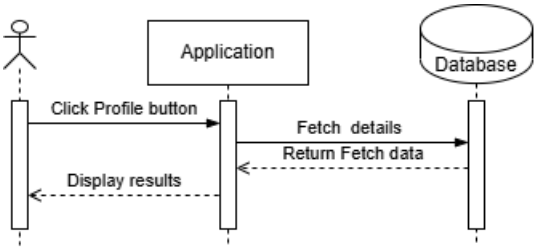

FIGURE 2.24: Sequence Diagram:Admin View Profile

2.1.4.13 Admin Edit Profile

Figure 2.25 presents the sequence diagram for editing profile operations, illustrating the interactions between the admin, the interface, and the back-end systems as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the editing profile functionalities.

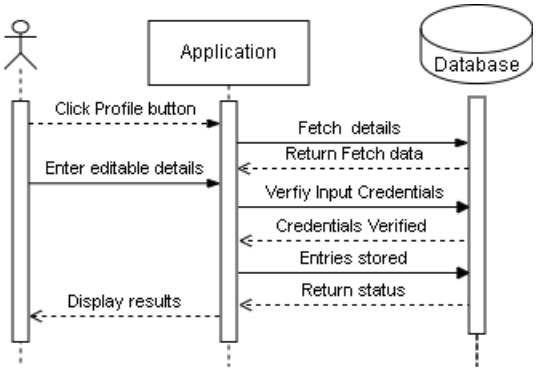

FIGURE 2.25: Sequence Diagram:Admin Edit Profile

2.1.4.14 User Sequence

Figure 2.26 presents the sequence diagram for the operations, illustrating the interactions between the user, and the user interface as they execute various tasks, thereby providing a clear view of the chronological flow of actions and responses within the admin functionalities.

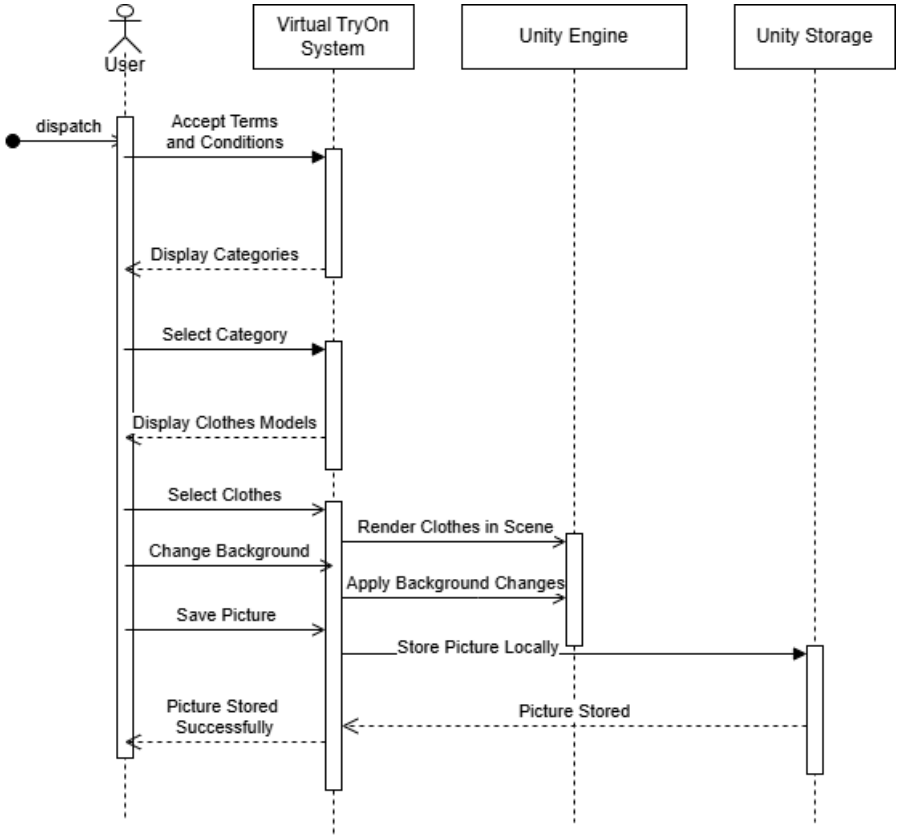

FIGURE 2.26: Sequence Diagram:User

2.1.5 Collaboration Diagram

A collaboration diagram visually depicts the dynamic interactions and relationships between objects in a system, emphasizing their roles, responsibilities, and exchanged messages. The significance of class diagrams lies in the ability of providing a clear understanding of collaboration and communication patterns within an object-oriented system.

2.1.5.1 Admin Signup

Figure 2.27 illustrates the collaboration diagram for the admin signup process, detailing the interactions between the admin user, the signup interface, and the registration system, thus showcasing the communication flow during the signup procedure.

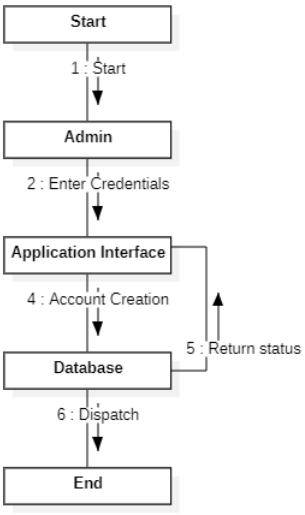

FIGURE 2.27: Collaboration Diagram:Admin Signup

2.1.5.2 Admin Login

Figure 2.28 presents the collaboration diagram for the admin login process, highlighting the interactions between the admin user, the login interface, and the authentication system, thereby demonstrating the flow of communication during the login procedure.

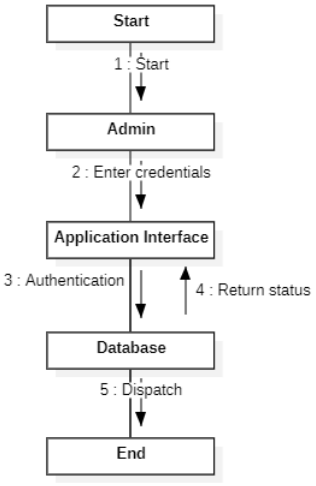

FIGURE 2.28: Collaboration Diagram:Admin Login

2.1.5.3 Admin Edit Profile

Figure 2.29 presents the collaboration diagram for the admin edit profile process, highlighting the interactions between the admin user, the edit profile interface, and the data management system, thereby illustrating how the user's profile information is modified and updated.

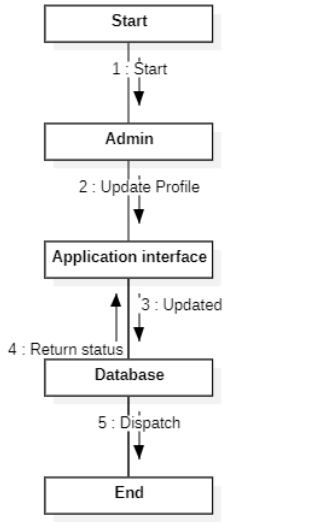

FIGURE 2.29: Collaboration Diagram:Admin Edit Profile

2.1.5.4 Admin CRUD on Category

Figure 2.30 presents the collaboration diagram for the admin CRUD operations on categories, highlighting the interactions between the admin user, the category

management interface, and the database system, thus demonstrating how category data is created, accessed, modified, and deleted.

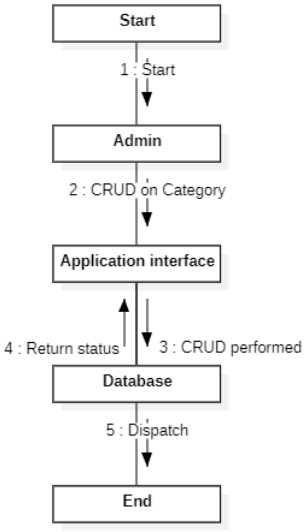

FIGURE 2.30: Collaboration Diagram:Admin Category CRUD

2.1.5.5 Admin Search Category

Figure 2.31 presents the collaboration diagram for the admin search operation on categories, highlighting the interactions between the admin user, the category management interface, and the database system, thus demonstrating how relevant category data is retrieved.

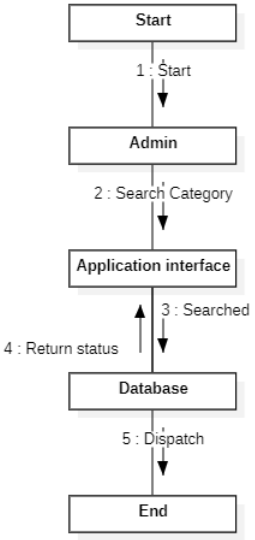

FIGURE 2.31: Collaboration Diagram:Admin Search Category

2.1.5.6 Admin CRUD on Item

Figure 2.32 presents the collaboration diagram for the admin CRUD operations on items, highlighting the interactions between the admin user, the item management interface, and the database system, thus demonstrating how item data is created, accessed, modified, and deleted.

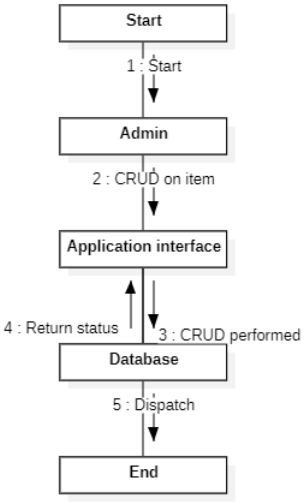

FIGURE 2.32: Collaboration Diagram:Admin Item CRUD

2.1.5.7 Admin Search Item

Figure 2.33 presents the collaboration diagram for the admin search operation on items, highlighting the interactions between the admin user, the item management interface, and the database system, thus demonstrating how relevant item data is retrieved.

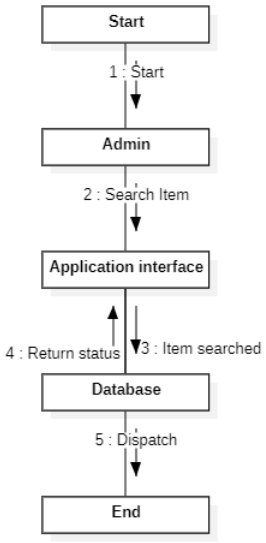

FIGURE 2.33: Collaboration Diagram:Admin Search item

2.1.5.8 Kinect Detect User

Figure 2.34 presents the collaboration diagram for the Kinect user detection process, highlighting the interactions between the Kinect device, the user interface, and the detection algorithms, thus illustrating how user presence and movements are tracked and processed.

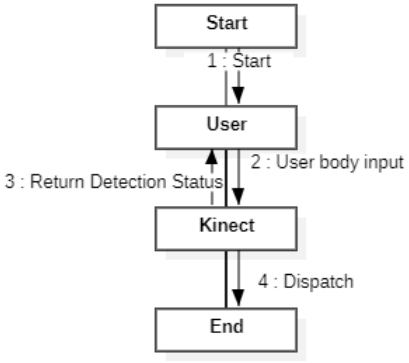

FIGURE 2.34: Collaboration Diagram:Detect User

2.1.5.9 User Accept Terms and Conditions

Figure 2.35 presents the collaboration diagram for the terms and conditions acceptance process, highlighting the interactions between the user, the terms and conditions interface, and the agreement confirmation system, thereby illustrating how the user's acceptance is recorded and processed.

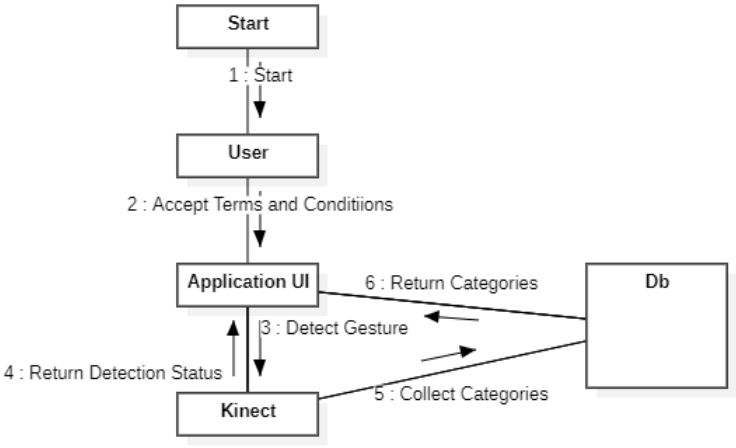

FIGURE 2.35: Collaboration Diagram:Accept Terms

2.1.5.10 User Select Category

Figure 2.36 presents the collaboration diagram for the category selection process, highlighting the interactions between the user, the category selection interface, and the system that retrieves and displays available respective items, thus illustrating how the user selects a category for further actions.

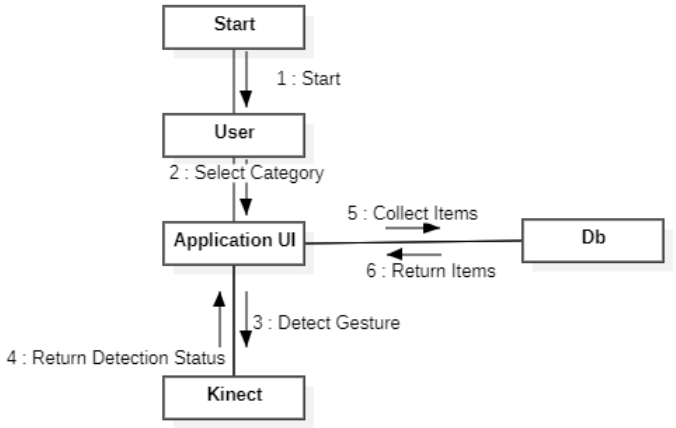

FIGURE 2.36: Collaboration Diagram:User Select Category

2.1.5.11 User Select Item

Figure 2.37 presents the collaboration diagram for the item selection process, highlighting the interactions between the user, the item selection interface, and the system that retrieves and displays available respective item details, thus illustrating how the user selects a category for further actions.

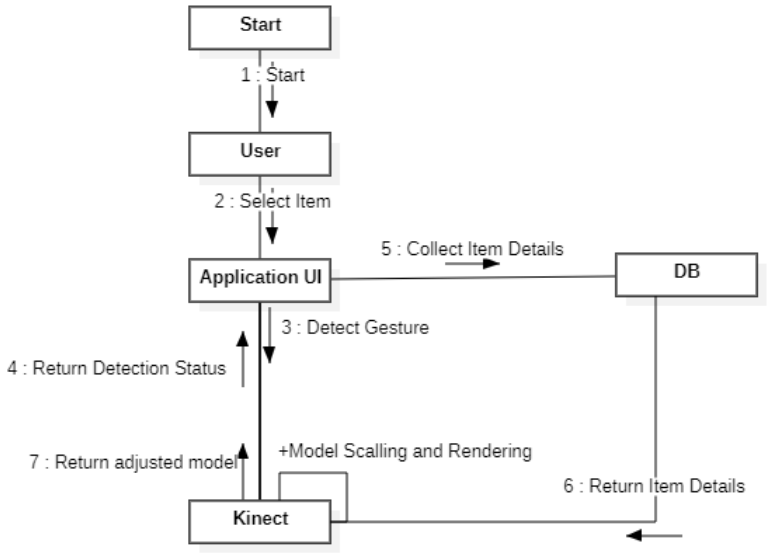

FIGURE 2.37: Collaboration Diagram:User Select Item

2.1.5.12 User Change Background

Figure 2.38 presents the collaboration diagram for the background change process, highlighting the interactions between the user, and the background selection interface, thereby illustrating how the user selects and applies a new background to the display.

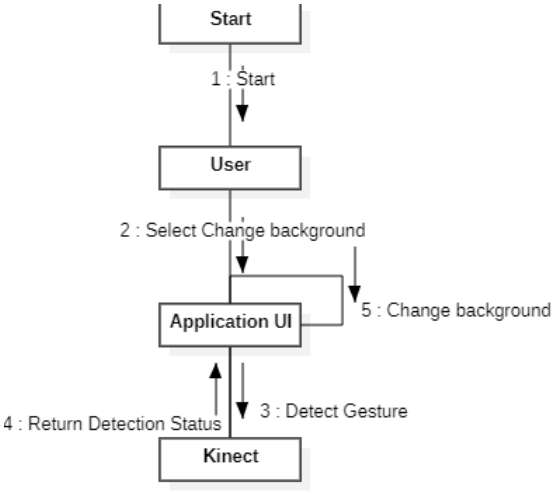

FIGURE 2.38: Collaboration Diagram:User Change Background

2.2 System Structure Design

Structural diagrams depict the static aspects or structure of a system, providing a detailed outline of the system's architecture and its components. These diagrams are essential for documenting and understanding the software architecture, as they define the components and their relationships without focusing on the dynamic behavior.

2.2.1 Class Diagram

A class diagram visually depicts the static structure of a system including classes, their attributes, methods, relationships, and interactions. The significance of class diagrams lies in their ability to provide a clear and comprehensive view of the object-oriented design and architecture of a system. Figure 2.39 presents the class diagram for our project, showcasing the system's classes, along with their attributes, methods, and relationships, thus offering a depiction of the object-oriented design.

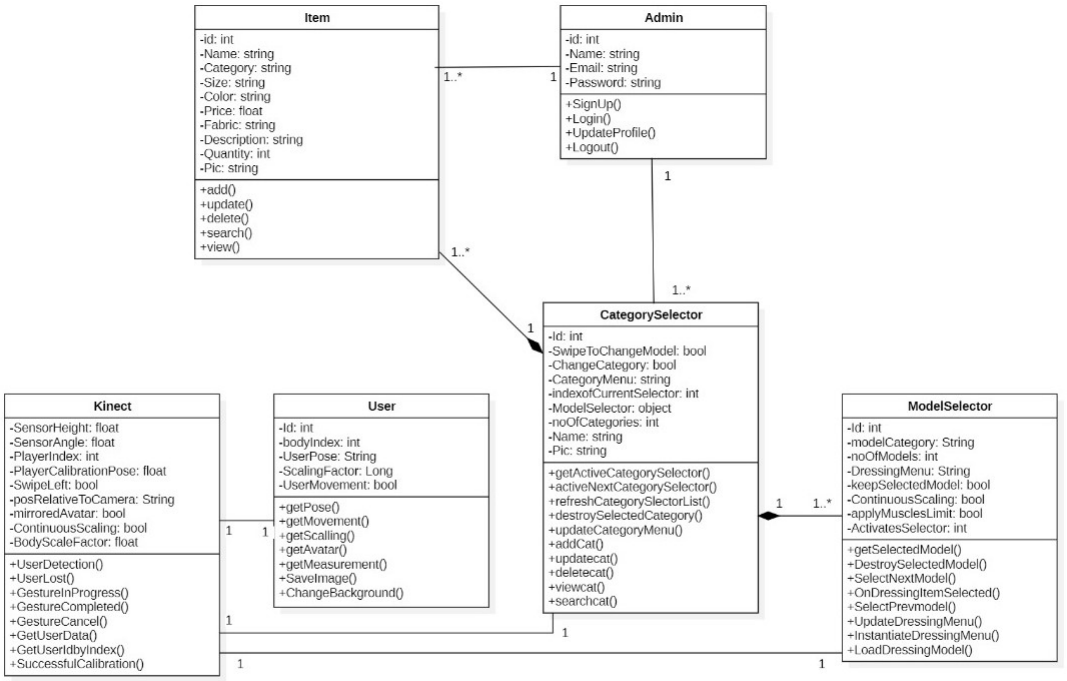

FIGURE 2.39: Class Diagram

2.2.2 Component Diagram

A component diagram visually depicts structural and organizational relationship of components of a software system along with their dependencies and relationships. The significance of component diagrams lies in their ability to provide a clear and structured view of how different components contribute to the overall system architecture. Figure 2.40 presents the component diagram for our project, highlighting the system's components, their interfaces, and relationships, thus presenting a view of its modular structure.

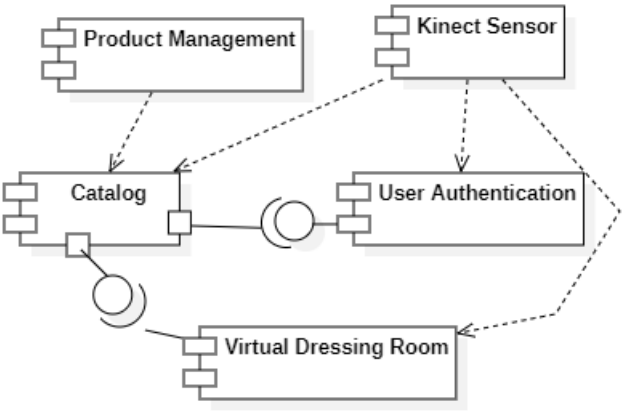

FIGURE 2.40: Component Diagram

2.2.3 Deployment Diagram

A Deployment Diagram visually depicts the hardware and the software in that hardware. The significance of Deployment Diagrams lies in their ability to provide a clear and structured view of the system's hardware and software infrastructure. Figure 2.41 presents the deployment diagram for our project, showcasing the architecture of the system, including the various hardware nodes, software components, and their interactions in the deployment environment.

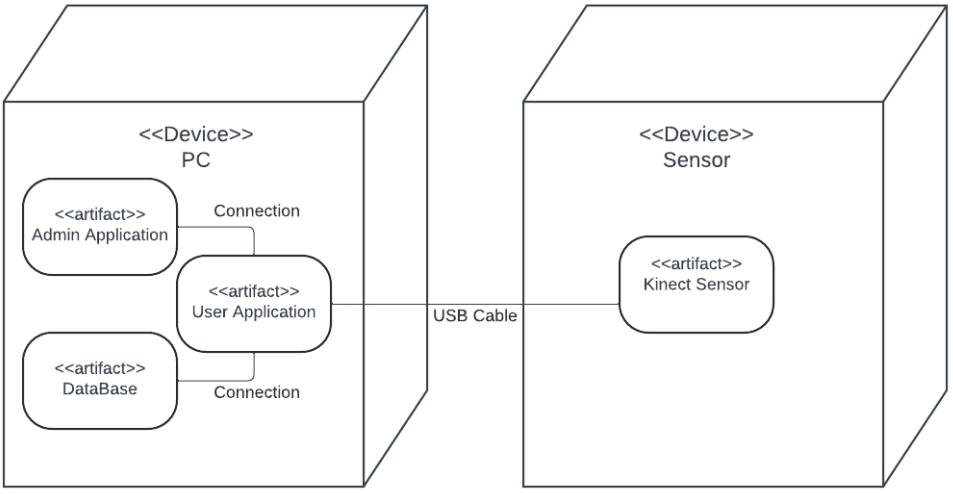

FIGURE 2.41: Deployment Diagram

2.3 User Interface Design

User interface(UI) design is essential in influencing a product's appearance, interaction, usability, actions, and overall user experience. A well-designed UI can greatly improve how users engage with a product, making it easy to navigate, efficient, and enjoyable. On the other hand, inadequate UI design can result in user frustration and a lack of engagement.

2.3.1 Wireframes

A Wireframe visually depicts the basic structure and layout of a user interface by depicting elements, layout, and functionality. The significance of Wireframes lies in their ability to provide a clear and simplified view of the user interface design of a system.

2.3.1.1 Admin Login

Figure 2.42 illustrates the login page of the admin application, where administrators can enter their credentials (username and password) to access the system after verification.

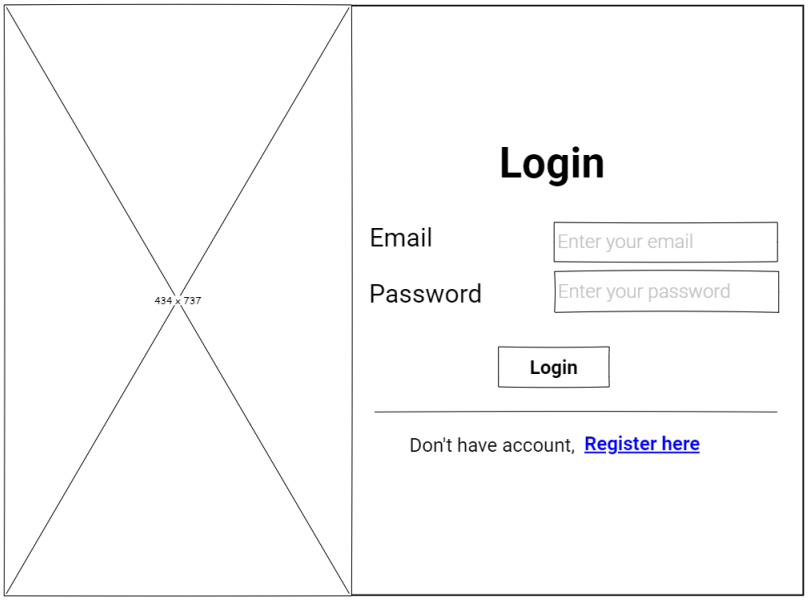

FIGURE 2.42: Admin Login

2.3.1.2 Admin Signup

Figure 2.43 illustrates the registration page of the admin application, where new administrators can create an account by entering their personal information and choosing password.

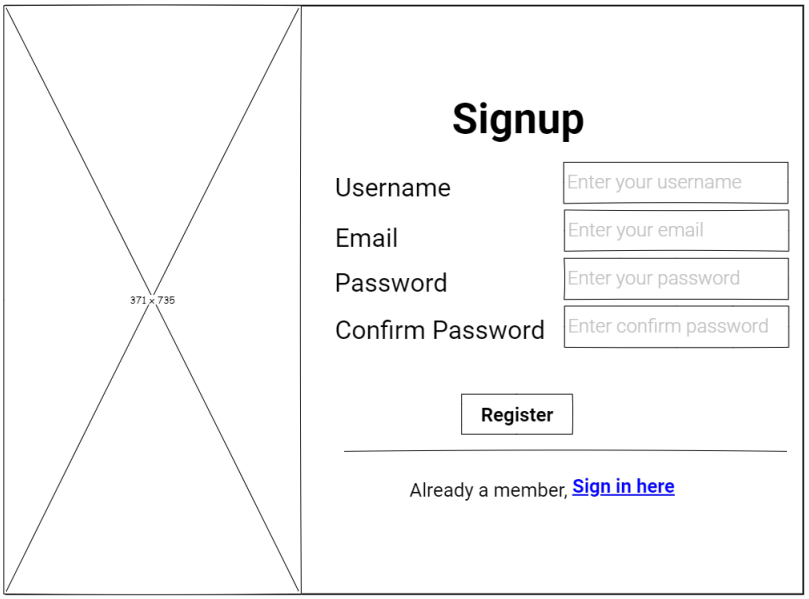

FIGURE 2.43: Admin Signup

2.3.1.3 Admin Dashboard

Figure 2.44 illustrates the home page of the admin application, providing administrators with a quick overview of the system,and navigation to different management sections.

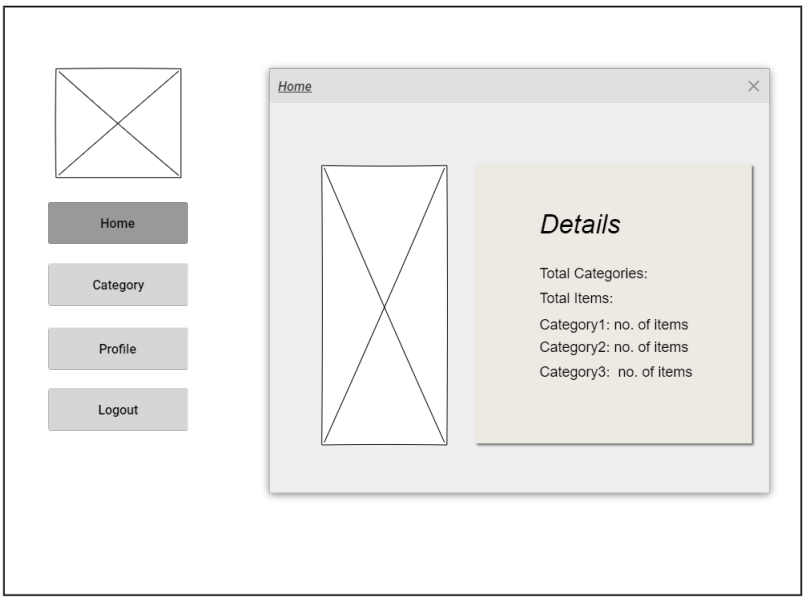

FIGURE 2.44: Admin Dashboard

2.3.1.4 Admin View Category

Figure 2.45 illustrates the view category page of the admin application, where administrators can browse and manage product categories, displaying the current list of categories available in the system.

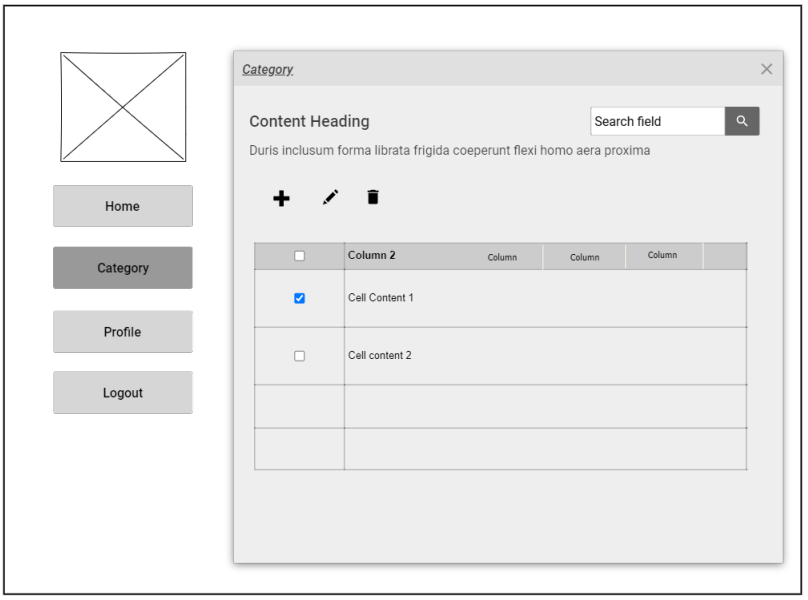

FIGURE 2.45: Admin View Category

2.3.1.5 Admin Add Category

Figure 2.46 illustrates the add category page of the admin application, which allows administrators to input and create new product categories by providing relevant details like name and image.

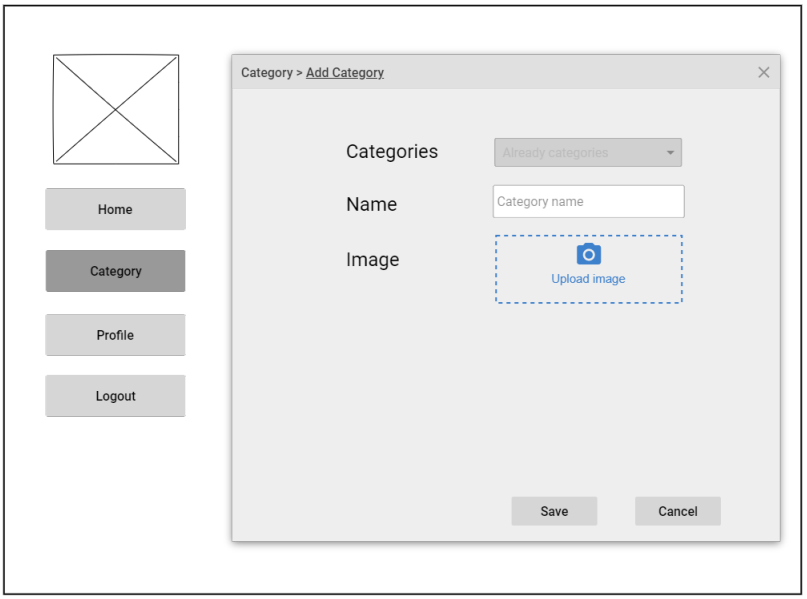

FIGURE 2.46: Admin Add Category

2.3.1.6 Admin Edit Category

Figure 2.47 illustrates the edit category page of the admin application, enabling administrators to modify existing category information, such as updating the name of a product category.

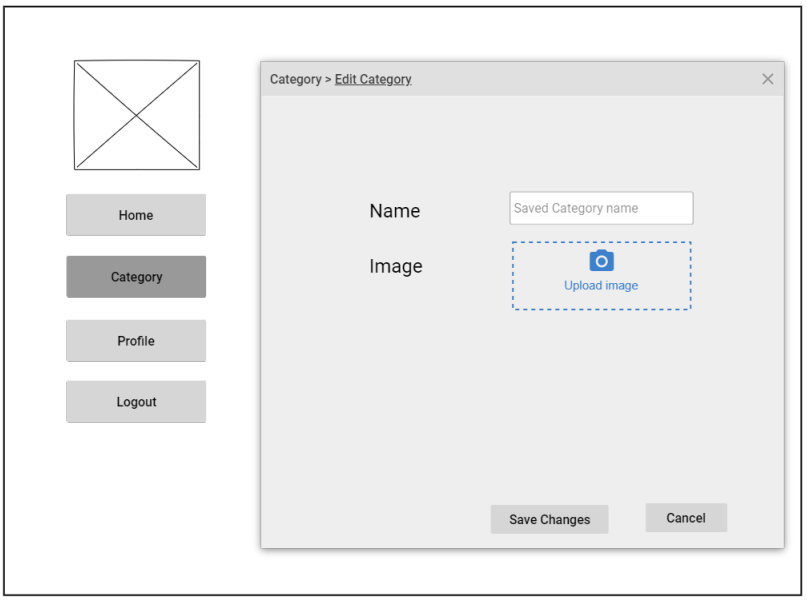

FIGURE 2.47: Admin Edit Category

2.3.1.7 Admin View Stock

Figure 2.48 illustrates the view stock page of the admin application, where administrators can view current stock levels for products, including details such as product names, quantity, and price.

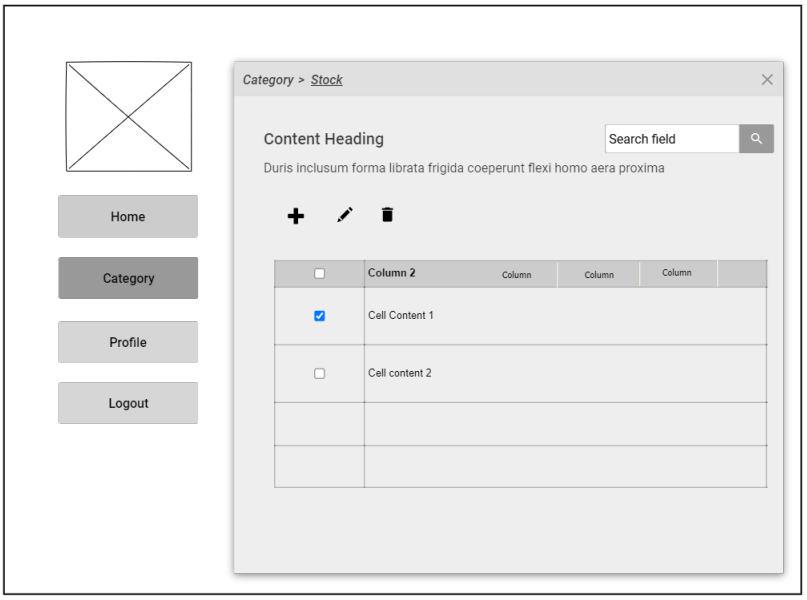

FIGURE 2.48: Admin View Stock

2.3.1.8 Admin Add Stock

Figure 2.49 illustrates the Add Stock page of the admin application, where administrators can add new stock items by entering the relevant details such as product name, quantity, and price.

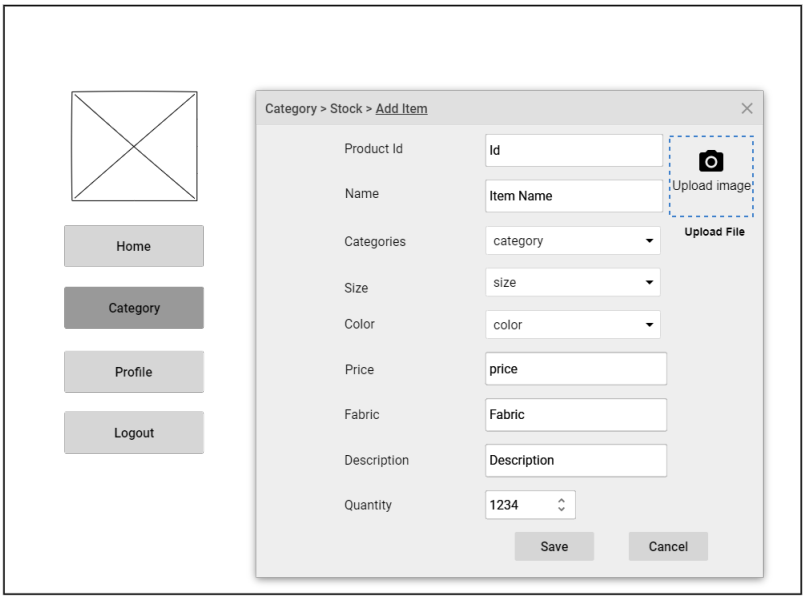

FIGURE 2.49: Admin Add Stock

2.3.1.9 Admin Edit Stock

Figure 2.50 shows the Edit Stock page of the admin application, which allows administrators to update existing stock information, including adjusting product details like price.

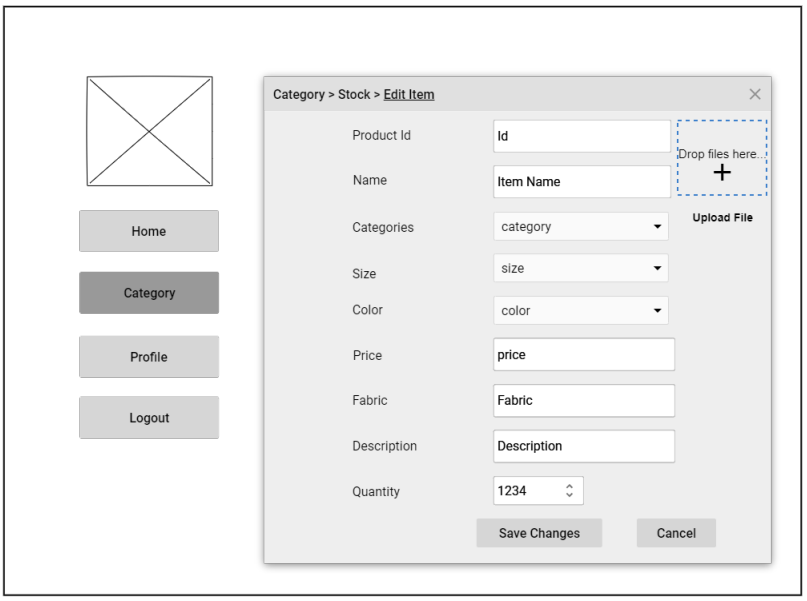

FIGURE 2.50: Admin Edit Stock

2.3.1.10 Admin View Profile

Figure 2.51 demonstrates the View Profile page for the admin, displaying personal details and options for updating account information.

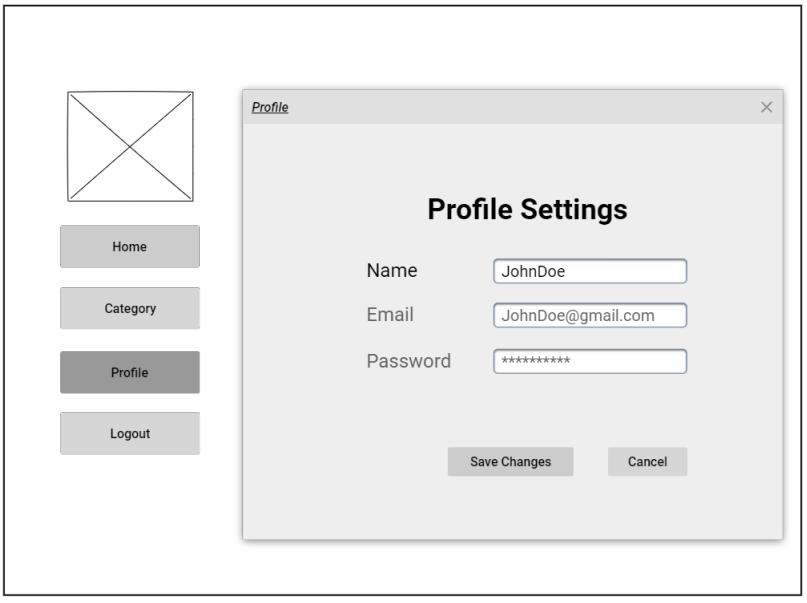

FIGURE 2.51: Admin View Profile

2.3.1.11 WelcomeScreen

Figure 2.52 illustrates the Welcome Screen of the application, where users can view and accept the terms and conditions before proceeding to the main functionalities,

ensuring that users understand their rights and responsibilities when using the app.

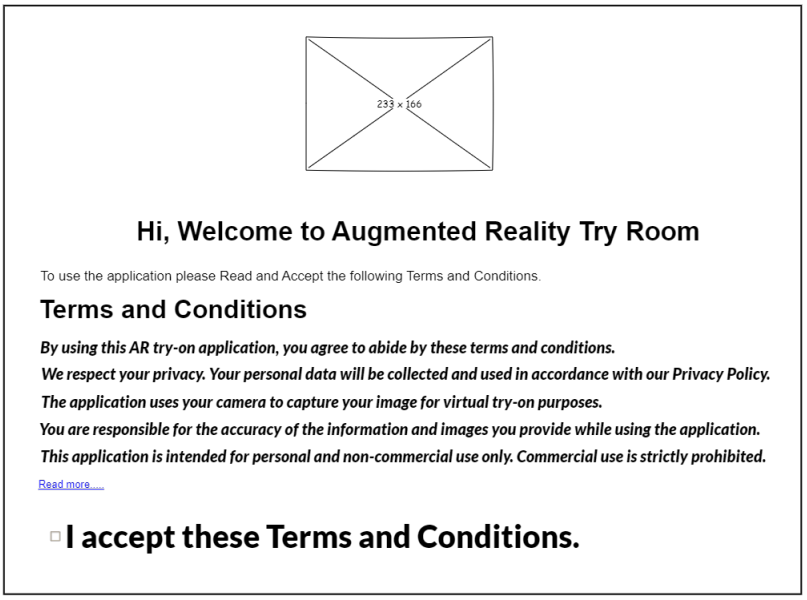

FIGURE 2.52: Terms and Conditions

2.3.1.12 HomeScreen

Figure 2.53 illustrates the Home Screen of the application, where users can view and select different categories of items serving as the central hub for navigating through available product categories and accessing various features of the app.

FIGURE 2.53: Home Page

2.3.1.13 ItemScreen

Figure 2.54 illustrates the Item Screen, where users can view and select specific items from a category allowing users to see themselves in virtual environment, saving pictures of their selections, and changing the background to enhance their experience while trying on items.

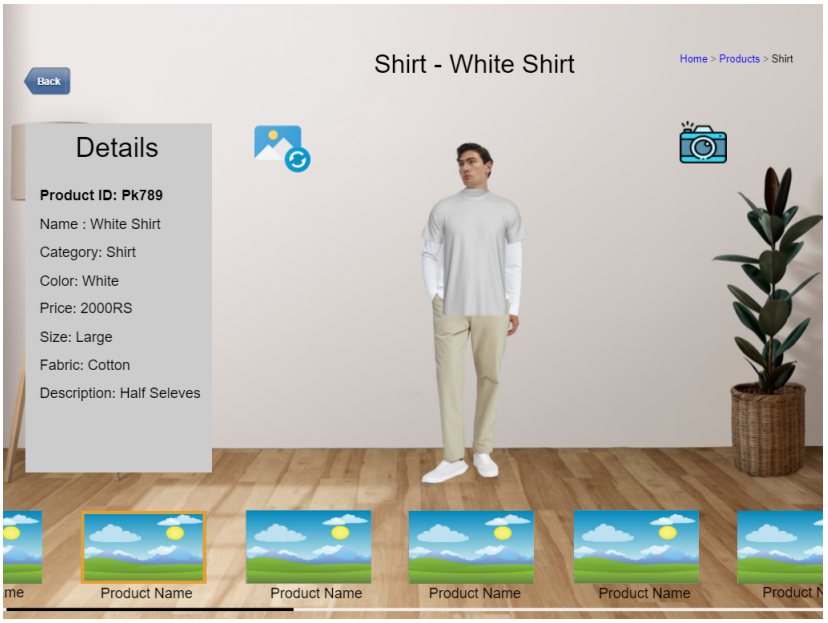

FIGURE 2.54: Main Screen

2.4 Database Design

Database design involves creating an organized and efficient database schema to effectively store and manage data.

2.4.1 ER Diagram

An Entity-Relationship (ER) diagram visually depicts the logical structure of a database by depicting entities, attributes, relationships, and constraints. The significance of ER diagrams lies in their ability to provide a clear and structured view of the data model of a system. Figure 2.55 presents the ER diagram for our project, depicting various entities, their attributes, and the relationships that exist among them.

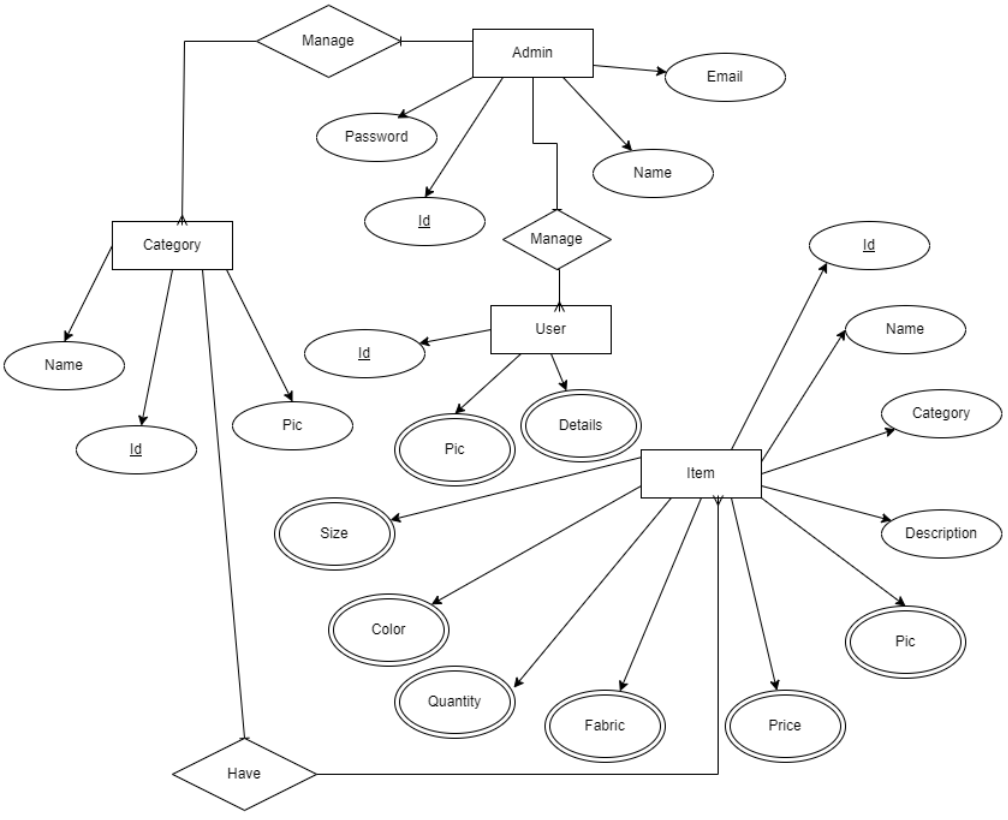

FIGURE 2.55: ER Diagram

References

[1] Abdul Moiz Farooq Izhar ul Haq. TryOn: An Augmented Reality Fitting Room. In *Mobile Devices and Smart Gadgets in Human Rights*, 2019.