College of Engineering
Department of Computer Science and Information Technology

SWE401
Software Engineering

Project Title: AI Wardrobe

Dr Mourad Al-Rajab

Mohammed Ali Kumo 1087491
Mohammed Al-Hammadi 1093829
Mohamed Amir Smail 1088172

Introduction

Fashion is a major part of daily life, yet many people still face the same problem every morning: deciding what to wear. Closets often end up full of clothes, but only a handful of items get used regularly. This makes people feel like they have “nothing to wear,” even though plenty of clothes are sitting unworn. It also leads to unnecessary shopping and adds to the growing issue of fashion waste. Studies show that the fashion industry is responsible for about **10% of global carbon emissions** and generates nearly **92 million tons of textile waste each year** (United Nations Environment Programme [UNEP], 2021). On an individual level, research has found that people wear only about **20% of their wardrobe** regularly while the rest goes untouched (Movinga, 2018). Our project, the **AI Fashion Wardrobe**, is designed to help people organize their clothing in a smarter way, reduce daily stress around choosing outfits, and promote more sustainable use of what they already own.

General Overview of the System

- The system works like a **digital wardrobe** where a user can:
  - Upload pictures of their clothes.
  - Categorize items into groups (tops, pants, shoes, accessories, etc.).
  - Keep everything in one place for easy access.
- Once set up, the system will **suggest outfits** based on:
  - The weather.
  - The type of event or occasion.
  - What the user has recently worn.
- Core features of the first version:
  - Wardrobe management.
  - Outfit suggestions.
  - Outfit history tracking.
  - Reminders for underused items.
- Future upgrades may include:
  - Automatic clothing recognition.
  - AR-based virtual try-ons.

    - Sustainability tracking to measure wardrobe eco-impact.

Intended Users and Stakeholders

- **Primary users**:
    - Students and professionals who need quick outfit suggestions.
    - Fashion lovers who enjoy experimenting with combinations.
    - Environmentally conscious individuals who want to reduce waste.
- **Future stakeholders**:
    - Fashion retailers.
    - Thrift stores.
    - Designers who may offer resale, rental, or shopping features through the platform.

Description of the Current System (“As-Is”)

- Most people currently manage their wardrobe by **memory** or a quick glance at the closet.
- A few may use **photos or notes**, but these are uncommon and not very effective.
- Existing apps mainly focus on:
    - **Online shopping**.
    - **Outfit sharing**.
- **Gap identified**: There is no widely adopted system that helps users manage their existing wardrobes, get personalized outfit recommendations, and track sustainability.

Key Problems and Challenges

- **Image handling & categorization** - managing clothing photos efficiently.
- **Recommendation accuracy** - making sure outfit suggestions are relevant and practical.
- **API integration** - connecting with weather APIs reliably.
- **User experience (UX)** - building a simple, intuitive interface.

- **Scalability & storage** - performance issues as wardrobes grow.
- **Time constraints** - focusing only on core features for this semester.

Technologies to Be Used

- **Frontend (UI)**: React.js (dynamic and responsive web interface).
- **Backend (Server-side)**: Node.js with Express.js OR Java Spring Boot (for REST APIs).
- **Database**: SQLite (prototype) or PostgreSQL (future scalability).
- **Image storage**: Local during development; cloud storage (e.g., AWS S3) later.
- **External APIs**: OpenWeatherMap API (weather-based outfit suggestions).
- **Design & mockups**: Figma or Adobe XD (UI/UX prototyping).
- **Version control**: Git & GitHub (for collaboration and tracking).

Team Members' Brief CVs

Mohammed Ali

- Background: Software Engineering student with backend and database experience.
- Skills: Java, Spring Boot, PostgreSQL, basic React.js.
- Role: Lead backend developer; responsible for database schema, APIs, and system architecture.

Mohammed Al Hammadi

- Background: IT (Cybersecurity) student with good Java programming skills and hands-on experience in system support and event operations.
- Skills: Java, IT troubleshooting, system security basics, leadership & team coordination, problem-solving.
- Role: Lead technical coordinator; responsible for Java development tasks, supporting secure system operations, and ensuring smooth execution of technical setups for projects and events.

Mohamed Amir Smail

- Strong background in programming (Java, web development) with solid academic performance and project experience.
- Experience in gaming-related work/projects, combining creativity and technical problem-solving.
- Proven sales and client management skills from roles in media marketing and technology sectors.
- Blend of technical expertise and business acumen, able to connect products with customer needs.

Market Potential

The AI Fashion Wardrobe has a promising market potential since it is a solution to an issue that most people nowadays are struggling with, they do not know what to wear. Time wasting is also one of the reasons why many people spend so much time in the morning selecting outfits, yet despite having so many clothes in their closets, majority of the same clothes would be repeated the following day. An option that provides rapid outfit recommendations based on the weather, the nature of the occasion, and what the user has put on in the recent past would make life easier. It would save time to students and professionals, and fashion lovers could experiment with new combinations. Environment-conscious people would also benefit because the system would promote reusing clothes rather than buying new clothes every time.

The system is also able to create a social and environmental difference. Among the largest issues in the industry is fashion waste, and most individuals only wear an approximate of 20 percent of their wardrobe. When reminding people about the clothes that they do not wear often, the AI Fashion Wardrobe would help in diminishing wastage and might make individuals more conscious of their purchasing practice. The slightest alterations in the use of clothes by people may make an enormous difference.

A few apps already provide wardrobe or outfit management such as Stylebook and Pureple, although they are not popular as they are mostly associated with shopping or sharing styles. The difference in our project is that we target the clothes the user already has and help him/her use them more often.

Comparison Table

Feature	Pureple / Stylebook (Existing Systems)	AI Fashion Wardrobe (Proposed System)	Advantage of Proposed System
Wardrobe management	Yes, manual upload	Yes, with categories + reminders	Saves time, adds organization
Outfit suggestions	Basic combinations	Context-based (weather, events, history)	More personalized and practical
Sustainability focus	Not included	Tracks usage + eco impact	Promotes eco-friendly choices
User experience	Can feel commercial	Simple and user-focused	More intuitive, less distracting
Integration	Limited (mostly standalone apps)	Planned integration with calendar + shopping apps	Enhances convenience and real-world usability

Feature	Pureple / Stylebook (Existing Systems)	AI Fashion Wardrobe (Proposed System)
Wardrobe management	Yes, manual upload	Yes, with categories + reminders
Outfit suggestions	Basic combinations	Context-based (weather, events, history)
Sustainability focus	Not included	Tracks usage + eco impact
User experience	Can feel commercial	Simple and user-focused

The Project Management Plan:

WorkPlan Schedule

Activity ID	Activity Description	Responsible Person	Start Date	End Date
1.0	Topic Selection & Initial Planning	All Members	28-Sep-2025	30-Sep-2025
1.1	Brainstorm possible project ideas (3–5 topics)	All Members	28-Sep-2025	28-Sep-2025
1.1.1	Each member proposes at least one idea	All Members	28-Sep-2025	28-Sep-2025
1.1.2	Discuss pros/cons of each idea	All Members	28-Sep-2025	29-Sep-2025
1.1.3	Vote and finalize the chosen topic	All Members	29-Sep-2025	30-Sep-2025
1.2	Assign initial roles (writer, editor, presenter, researcher)	All Members	30-Sep-2025	30-Sep-2025

2.0	**Research & Information Gathering**	**All Members**	**1-Oct-2025**	**18-Oct-2025**
2.1	**Divide research areas (background, data, case studies, visuals)**	**Mohamed Amir (coordination)**	**1-Oct-2025**	**1-Oct-2025**
2.1.1	**Research academic articles and journals**	**Mohammed Ali Kumo**	**2-Oct-2025**	**10-Oct-2025**
2.1.2	**Collect real-world case studies/examples**	**Mohammed Al-Hammadi**	**2-Oct-2025**	**12-Oct-2025**
2.1.3	**Find supporting visuals, graphs, and statistics**	**Mohamed Amir**	**2-Oct-2025**	**13-Oct-2025**
2.2	**Summarize findings in shared document**	**All Members**	**14-Oct-2025**	**18-Oct-2025**
2.2.1	**Write bullet-point summaries**	**All Members**	**14-Oct-2025**	**17-Oct-2025**
2.2.2	**Share document on Google Docs**	**Mohamed Amir**	**18-Oct-2025**	**18-Oct-2025**
3.0	**Report Writing & Drafting**	**Main Writer: Mohammed Ali Kumo**	**19-Oct-2025**	**31-Oct-2025**

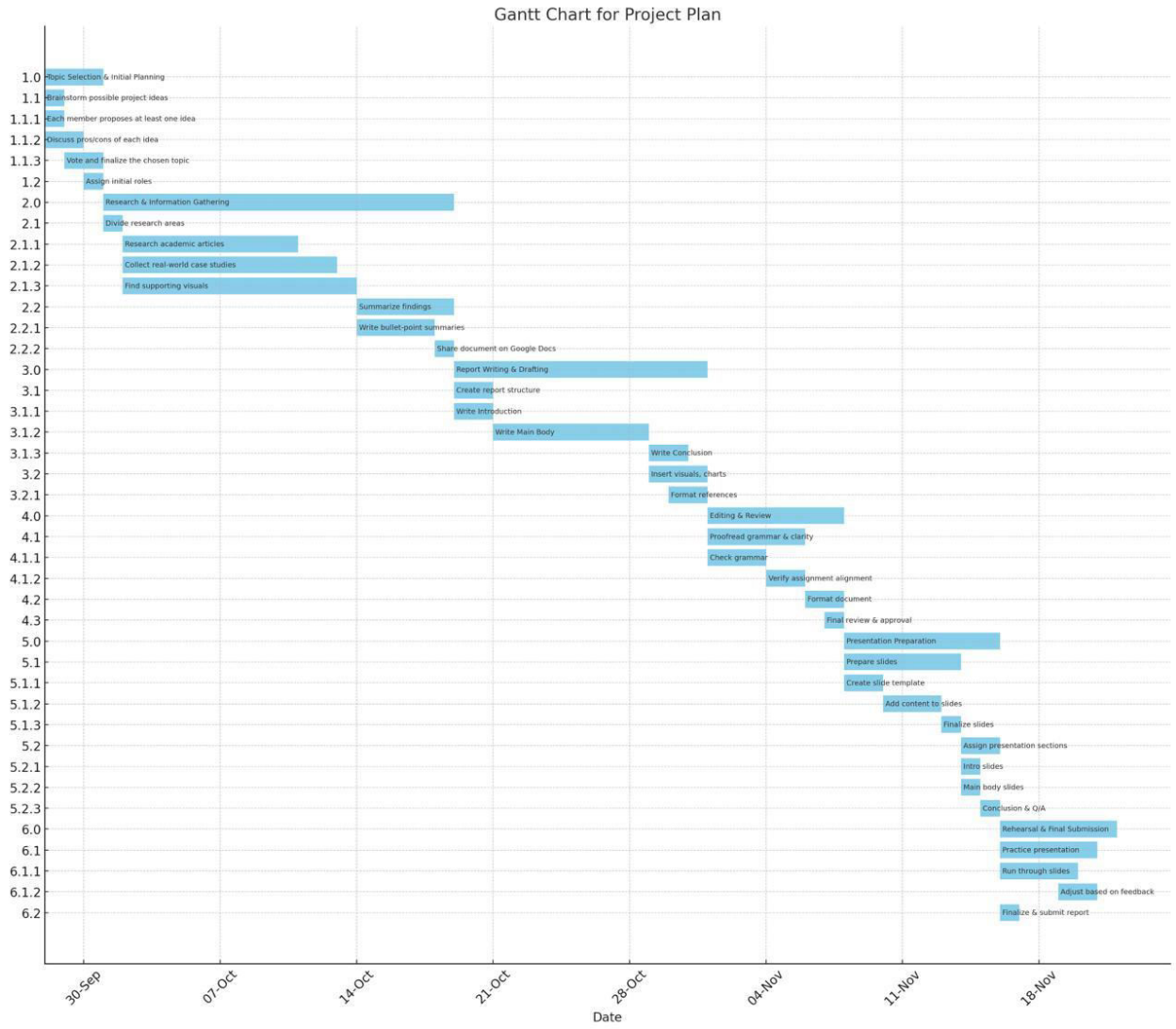
3.1	**Create report structure (intro, body, conclusion)**	**Mohammed Ali Kumo**	**19-Oct-2025**	**20-Oct-2025**
3.1.1	**Write Introduction (context + importance)**	**Mohammed Ali Kumo**	**19-Oct-2025**	**20-Oct-2025**
3.1.2	**Write Main Body (sections based on research)**	**All Members (each takes 1 section)**	**21-Oct-2025**	**28-Oct-2025**
3.1.3	**Write Conclusion (summary + future implications)**	**Mohammed Ali Kumo**	**29-Oct-2025**	**30-Oct-2025**
3.2	**Insert visuals, charts, and references**	**Mohamed Amir**	**29-Oct-2025**	**31-Oct-2025**
3.2.1	**Format references in correct citation style**	**Mohammed Al-Hammadi**	**30-Oct-2025**	**31-Oct-2025**
4.0	**Editing & Review**	**Editor: Mohammed Al-Hammadi**	**1-Nov-2025**	**7-Nov-2025**
4.1	**Proofread for grammar and clarity**	**Mohammed Al-Hammadi**	**1-Nov-2025**	**5-Nov-2025**
4.1.1	**Check grammar, spelling, sentence flow**	**Mohammed Al-Hammadi**	**1-Nov-2025**	**3-Nov-2025**

4.1.2	**Verify alignment with assignment requirements**	**Mohamed Amir**	**4-Nov-2025**	**5-Nov-2025**
4.2	**Format document (fonts, spacing, headers)**	**Mohamed Amir**	**6-Nov-2025**	**7-Nov-2025**
4.3	**Final review & approval**	**All Members**	**7-Nov-2025**	**7-Nov-2025**
5.0	**Presentation Preparation**	**All Members**	**8-Nov-2025**	**15-Nov-2025**
5.1	**Prepare presentation slides (10–12 slides)**	**Mohamed Amir (lead)**	**8-Nov-2025**	**13-Nov-2025**
5.1.1	**Create slide template (consistent design)**	**Mohamed Amir**	**8-Nov-2025**	**9-Nov-2025**
5.1.2	**Add content (summaries, visuals, key points)**	**All Members**	**10-Nov-2025**	**12-Nov-2025**
5.1.3	**Finalize and format slides**	**Mohammed Ali Kumo**	**13-Nov-2025**	**13-Nov-2025**

5.2	**Assign presentation sections**	**All Members**	**14-Nov-2025**	**15-Nov-2025**
5.2.1	**Introduction slides**	**Mohammed Ali Kumo**	**14-Nov-2025**	**14-Nov-2025**
5.2.2	**Main body slides**	**Mohammed Al-Hammadi**	**14-Nov-2025**	**14-Nov-2025**
5.2.3	**Conclusion & Q/A**	**Mohamed Amir**	**15-Nov-2025**	**15-Nov-2025**
6.0	**Rehearsal & Final Submission**	**All Members**	**16-Nov-2025**	**21-Nov-2025**
6.1	**Practice presentation (timing & flow)**	**All Members**	**16-Nov-2025**	**20-Nov-2025**
6.1.1	**Run through slides together (2–3 times)**	**All Members**	**16-Nov-2025**	**19-Nov-2025**
6.1.2	**Adjust based on feedback**	**All Members**	**19-Nov-2025**	**20-Nov-2025**

6.2	**Finalize and submit project report**	**Mohammed Al-Hammadi**	**16-Nov-2025**	**16-Nov-2025**
6.3	**Deliver presentation on due date**	**All Members**	**17-Nov-2025**	**21-Nov-2025**

Gantt Chart



Project Methodology Selection

We have chosen the Throwaway Prototyping methodology for this project. This is the most appropriate methodology because we are looking to quickly build a working prototype of the AI Fashion Wardrobe to test basic features such as wardrobe management, outfit suggestion, and outfit history tracking.

Throwaway prototyping allows us to:

Quickly build a simplified version of the system for the sake of getting initial feedback from the users.

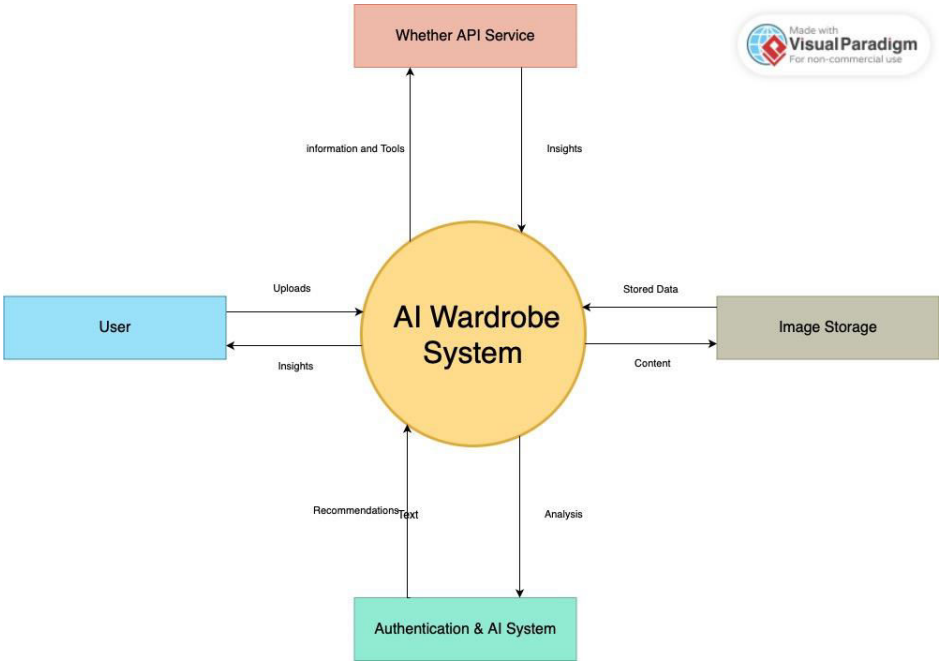
Detect any future image handling issues, recommendation quality, and user experience problems before investing in the full system.

Test UI designs and interaction flows to ensure the interface is intuitive.

Then, after evaluating the prototype and receiving the feedback of the users, we will complete our requirements and design, if necessary scrapping the initial prototype, and then proceed with the development of the actual system. The method ensures that we are focused on developing a user-friendly, user-centered application and minimize wasted time in the preliminary development process.

Phase 2: Requirements Analysis

1. System Context Diagram



The system context diagram illustrates AI Wardrobe's position within its operational ecosystem, identifying all external entities and their interactions with the core system. It shows how users, weather services, and cloud storage systems exchange data with AI Wardrobe, while internal components like the recommendation engine and database support core functionality.

2. Requiment Elicitation Technique

The AI Fashion Wardrobe project aims to solve one of the most common everyday challenges: choosing what to wear. To ensure that the proposed system meets real-world expectations and reflects the needs of potential users, two main requirements elicitation techniques were appliedsurveys and semi-structured interviews. Both methods were selected to gather a combination of quantitative and qualitative data. The survey helped us identify general trends and user priorities, while the interviews provided deeper insights into individual motivations, frustrations, and expectations. This blend of numerical and narrative data allowed the team to form a complete picture of what users truly need from an intelligent digital wardrobe.

Survey Analysis

1.To begin, a digital survey was designed and distributed via **Google Forms** to a group of **45 participants**, consisting mainly of university students, young professionals, and casual fashion enthusiasts aged between **18 and 35 years**. These participants were chosen as they represent the system's target audience individuals who regularly dress for social, academic, or professional occasions and often experience time pressure in making outfit choices.

2. The survey included **ten targeted questions** that focused on personal clothing habits, wardrobe organization methods, and attitudes toward sustainability. It aimed to measure how often users struggle to decide what to wear, how frequently they reuse their clothes, and whether they would be willing to trust a digital wardrobe system to assist them. The questions also explored opinions on environmental impact, fashion waste, and the role technology could play in improving these issues.

3. The responses highlighted some powerful trends. A striking **81% of respondents** admitted that they find it difficult to decide what to wear at least three times a week. This confirmed the existence of what many participants referred to as "decision fatigue," especially during busy mornings. Furthermore, **67% of respondents** said they often forget about certain items in their wardrobe and end up wearing the same limited selection of clothes repeatedly. This finding reinforces the idea that a digital wardrobe system capable of tracking and reminding users about underused items could provide genuine value.

4. Another important insight was that **72% of participants** expressed interest in using an application that could provide **weather-based outfit suggestions**, while **74%** mentioned concerns about **fashion waste and overconsumption**. They acknowledged that they buy new clothes unnecessarily because they forget what they already own or

lose track of their wardrobe. This shows that the AI Fashion Wardrobe could help users make better use of existing clothes and indirectly promote sustainable fashion practices.

5. Interestingly, several participants commented that they already use digital tools like Pinterest or Instagram for style inspiration but wish there were a more personal and practical app that considers their *actual* wardrobe instead of general online trends. This realization was key in defining our system's unique direction: rather than promoting shopping, the AI Fashion Wardrobe helps users rediscover their own clothes, encouraging creativity, convenience, and sustainability in everyday dressing.

Survey Summary: Overall, the survey confirmed a strong user demand for a system that reduces time pressure, minimizes clothing waste, and simplifies fashion decisions. It also provided a solid statistical foundation for defining our functional and non-functional requirements in the later stages of the project.

Interview Analysis

The semi-structured interviews confirmed that the core problem is the **"morning outfit problem,"** where users consistently lose time and experience stress deciding what to wear. A parallel issue is the **"overlooked wardrobe dilemma,"** with participants admitting they regularly wear only **20%** of their clothes, forgetting the rest exist. The interviews strongly emphasized that the ideal solution must be **quick, easy, and effort-saving**, capable of suggesting outfits based on real-world context (like weather and recency of wear). Crucially, the process revealed an **emotional link**(outfits impact confidence) and an opportunity for **sustainability impact**, as users expressed a willingness to use a feature that tracks usage efficiency to encourage re-wearing and reduce fashion waste.

3. System Users and Roles (One-Liner Specification)

3.1 Registered Users

Registered Users are the primary, everyday users (students, professionals) who actively manage their wardrobe and request intelligent outfit suggestions.

ID	Registered User Requirements
3.1.1	Registered Users shall be able to **upload, categorize, and manage** all clothing items in their virtual wardrobe.
3.1.2	Registered Users must be able to **edit, rename, or delete** any item from their virtual wardrobe at any time.
3.1.3	Registered Users shall be able to request **outfit suggestions** tailored by **weather, occasion, and usage history**.

3.1.4	The system must record a complete **outfit history log** of previously worn outfits to avoid repetition.
3.1.5	Registered Users shall receive **automatic reminders** for items that have not been worn for a long time, promoting clothing rotation.
3.1.6	Registered Users must have access to a customizable settings menu to **adjust preferences** (e.g., reminder frequency, interface color).
3.1.7	Registered Users shall be able to **view sustainability insights** regarding the usage rate of their clothing items.
3.1.8	Registered Users must be able to securely **log into and manage** their personal account profile.

3.2 Guest Users

Guest Users are visitors exploring the application before committing to full registration and wardrobe setup.

ID	Guest User Requirements
3.2.1	Guest Users shall be able to **browse a demo version** of the application interface and its core features.
3.2.2	Guest Users must be able to initiate the **account registration and sign-up process** from any page.
3.2.3	Guest Users must be presented with the **Privacy Policy and data consent terms** before registration.

3.3 System Administrators

System Administrators are responsible for backend operations, ensuring security, performance, and data integrity.

ID	Administrator Requirements
3.3.1	Administrators must have access to analytics for **monitoring system performance** (server load, response times, storage usage).
3.3.2	Administrators shall be immediately alerted to any **technical errors**, such as failed API connections or image upload errors.
3.3.3	Administrators must be able to **manage user accounts** (reset passwords, handle deletion requests) while maintaining user data privacy.
3.3.4	Administrators are required to ensure **data integrity** by performing regular backups and verifying encryption protocols.
3.3.5	Administrators must **oversee and test all external integrations** (e.g., weather API) to ensure continuous functionality and accuracy.

3.4 Fashion Partners (Future Stakeholder)

Fashion Partners are external entities (retailers, brands) whose requirements focus on future integration for business expansion and sustainability initiatives.

ID	Future Stakeholder Requirements
3.4.1	The system architecture must include **secure API endpoints** to facilitate future integration with external Fashion Partner services (e.g., retail links).

3.4.2 The system shall provide **aggregated, anonymized data reports** (with explicit user consent) on clothing usage trends for analysis by Future Stakeholders.

3.4.3 The system must have the **capacity to integrate donation or resale features** provided by eco-friendly clothing brands or thrift stores.

User Characteristics:

- Age range: 18-40
- Gender: All
- Tech proficiency: Average smartphone and browser user
- Access platform: Web browser (mobile and desktop)

4. Functional Requirements

1. User Registration and Authentication

FR-1.1: The system shall allow new users to create an account using their email, username, and a secure password.

FR-1.2: The system shall verify login credentials and provide secure authentication for users.

FR-1.3: The system shall allow users to recover their passwords through a registered email.

FR-1.4: The system shall internally differentiate between various user types, including Basic, Fashion-Conscious, Sustainability-Focused, and Administrator roles.

2. Wardrobe Management

FR-2.1: Users shall be able to upload photos of their clothing items with attributes such as color, type, brand, and fabric material.

FR-2.2: The system shall automatically categorize uploaded items into predefined categories such as shirts, pants, accessories, and shoes.

FR-2.3: Users shall have the option to edit, delete, or reclassify items in their wardrobe.

FR-2.4: The system shall store wardrobe data securely in a user-specific database.

3. Outfit Recommendation

FR-3.1: The system shall generate daily outfit recommendations based on user preferences, wardrobe data, and weather conditions.

FR-3.2: The system shall allow users to provide feedback (like, dislike, or replace) to refine future suggestions.

FR-3.3: The system shall provide outfit options for various occasions such as casual, formal, or seasonal events.

4. Analytics and Sustainability Tracking

FR-4.1: The system shall track item usage frequency and provide insights on underused clothing items.

FR-4.2: The system shall generate sustainability metrics, including clothing usage rate and environmental impact reduction.

FR-4.3: The system shall display analytics in visual formats such as charts and summaries within the user dashboard.

5. Weather Integration

FR-5.1: The system shall fetch real-time weather information through an external weather API.

FR-5.2: The system shall adjust outfit recommendations according to temperature, humidity, and precipitation.

FR-5.3: The system shall update weather data automatically at regular intervals.

6. Administrative Controls

FR-6.1: The administrator shall have access to monitor system performance and user activities.

FR-6.2: The administrator shall have permission to delete inappropriate content or images uploaded by users.

FR-6.3: The administrator shall manage database maintenance and software updates.

7. Notifications and Reminders

FR-7.1: The system shall send notifications to users regarding underused items and daily outfit recommendations.

FR-7.2: The system shall notify users about scheduled maintenance or new feature updates.

FR-7.3: The system shall send reminders for wardrobe updates or seasonal clothing recommendations.

5. Non-Functional Requirements (Weighted Quality Attributes)

Non-Functional Requirements define the quality and constraints on the system. Security and data integrity requirements use "shall" as they are mandatory, while performance, usability, and portability goals use "should" as they are desirable quality attributes.

1. Performance

NFR-1.1: The system should respond to user requests within 2 seconds for at least 95% of transactions.

NFR-1.2: Image uploads should be completed within 10 seconds under standard network conditions.

NFR-1.3: The system should maintain stable performance with up to 5,000 concurrent users.

2. Usability

NFR-2.1: The interface should be intuitive and user-friendly, requiring minimal learning time for new users.

NFR-2.2: The system should provide on-screen guidance and tooltips for key features.

NFR-2.3: The system should support both light and dark interface themes.

3. Reliability and Availability

NFR-3.1: The system shall maintain 99% uptime excluding scheduled maintenance.

NFR-3.2: All critical user data shall be backed up automatically every 24 hours.

NFR-3.3: The system should ensure quick recovery from failures within 10 minutes of downtime.

4. Security

NFR-4.1: All user credentials shall be stored using encryption and salted hashing techniques.

NFR-4.2: Data transmission between client and server shall be encrypted using HTTPS/TLS protocol.

NFR-4.3: The system shall implement role-based access control to restrict unauthorized actions.

NFR-4.4: Regular security audits should be conducted to ensure data protection.

5. Scalability

NFR-5.1: The system architecture should support horizontal and vertical scaling for future growth.

NFR-5.2: The system should handle increased user traffic without significant degradation of performance.

6. Maintainability

NFR-6.1: The system’s codebase should be modular and well-documented for ease of updates.

NFR-6.2: All software changes and fixes should be tracked using version control systems such as Git.

NFR-6.3: The system should allow configuration changes without impacting core functionality.

7. Portability

NFR-7.1: The system should operate across major web browsers and mobile platforms.

NFR-7.2: The interface should automatically adjust to varying screen resolutions and device orientations.

NFR-7.3: The system should maintain consistent functionality across Android, iOS, and desktop environments.

6. Text-Based Use Cases

Use Case Name: Upload Outfit Image	**ID:** UC - 1	**Priority: High**
Actor: User		
Description: The user uploads an image of their outfit to the AI Wardrobe system for analysis and storage.		
Trigger: The user selects the “Upload Outfit” option from the main menu.		
Type: ☑ External ☐ Temporal		
Preconditions: 1. The user is logged into their account. 2. The user has a stable internet connection.		

Normal Course:

1. The user clicks the “Upload Outfit” button.
2. The system prompts the user to select an image from their device.
3. The user selects the desired outfit image.
4. The system uploads the image to the server.
5. The system analyzes the image using AI to identify clothing items and colors.
6. The system stores the outfit details in the user’s virtual wardrobe.
7. The system displays a confirmation message that the outfit has been uploaded successfully.

Postconditions:

1. The uploaded image and its details are saved in the database under the user’s profile

Use Case Name: Generate Outfit Recommendation	**ID:** UC - 2	**Priority: High**
Actor: User		
Description: The user requests the AI Wardrobe to generate an outfit suggestion based on weather, occasion, and wardrobe items		
Trigger: The user selects the “Get Outfit Recommendation” option.		
Type: ☑ External ☐ Temporal		

Preconditions:

1. The user has uploaded at least one outfit to their wardrobe.
2. The system has access to weather data or event details.

Normal Course:

1. The user clicks on “Get Outfit Recommendation.”
2. The system prompts the user to enter occasion details (e.g., casual, formal, night out).
3. The user inputs the occasion and optionally allows access to weather data.
4. The AI analyzes wardrobe items and filters suitable outfits.
5. The system generates and displays the top 3 outfit recommendations.
6. The user selects one outfit or requests another suggestion.
7. The system confirms the user’s selection and logs the recommendation for analytics

.

Postconditions:

1. The chosen outfit is saved under “Recommended Outfits.”

Use Case Name: Edit Wardrobe Items	**ID:** UC - 3	**Priority: High**
Actor: User		
Description: The user edits or deletes an existing wardrobe item from their virtual closet.		
Trigger: The user selects the “Edit Wardrobe” option.		
Type: ☑ External ☐ Temporal		

Preconditions:

1. The user has uploaded wardrobe items previously.
2. The user is logged into their account.

Normal Course:

1. The user opens the “My Wardrobe” section.
2. The system displays a list of all uploaded outfits.
3. The user selects an item to edit or delete.
4. If editing, the system displays editable fields (name, category, color, etc.).
5. The user updates the details or confirms deletion.
6. The system saves the updated information or removes the item.
7. The system displays a confirmation message.

Postconditions:

1. The wardrobe database is updated with the new item information, or the deleted item is removed permanently.

7. Requirements Specifications

FR-3.1: Outfit Recommendation Functionality	
Description	When the user requests an outfit recommendation, the system shall use the stored wardrobe data, current weather information, and user preferences to generate a suitable outfit suggestion.
Inputs	User request for outfit recommendation Weather data from external API User’s wardrobe and preferences
Source	User.
Outputs	Outfit recommendation displayed to the user

Pre-condition	User has an existing wardrobe and internet connection
Post-condition	Recommended outfit is shown and stored in user's outfit history

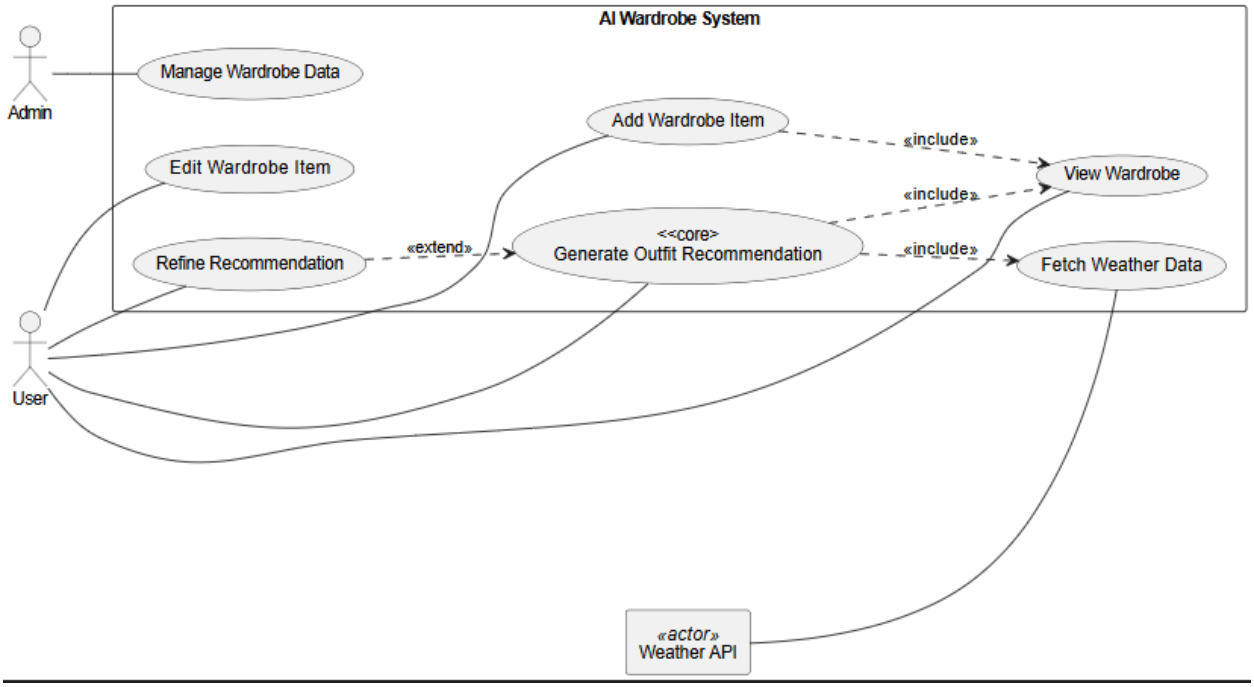
FR-1.2: Secure Authentication	
Description	The system shall allow users to log in using their email and password, verifying credentials securely before granting access.
Inputs	Email and password entered by the user
Source	User.
Outputs	Access granted to user dashboard or error message if credentials are invalid
Pre-condition	User must have a registered account
Post-condition	User session is established and tracked

FR-2.3: Edit or Delete Wardrobe Items	
Description	Enables the user to modify or remove existing items in their virtual wardrobe.
Inputs	Selected wardrobe item, updated item details or delete command.
Source	User.
Outputs	Updated wardrobe list or confirmation of item deletion.
Pre-condition	The wardrobe must contain at least one item.
Post-condition	The wardrobe database reflects the updated or deleted item accordingly.

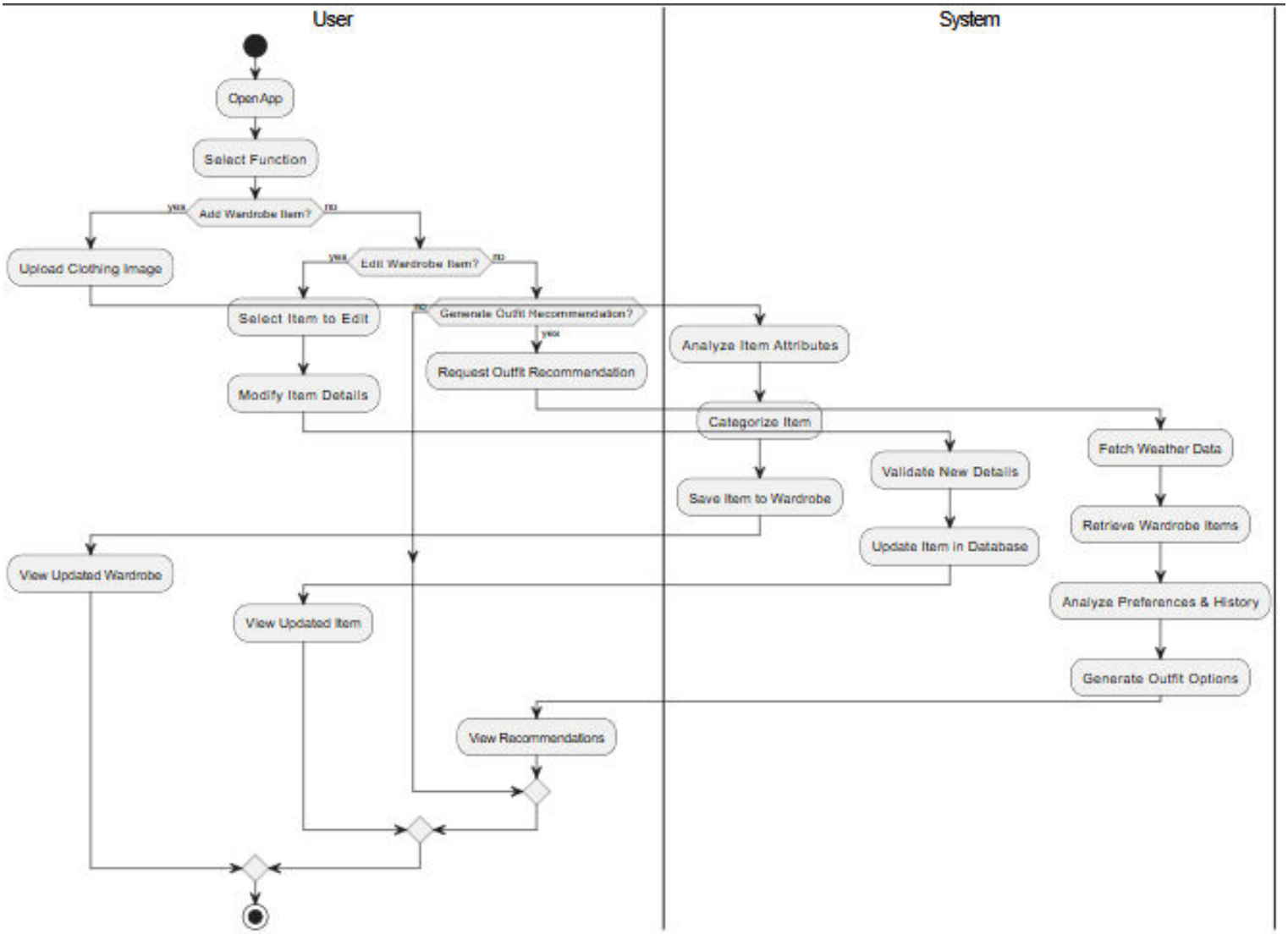
8. Traceability Matrix

Requirement ID	Requirement Description	Generate Outfit Recommendation	Add Wardrobe Item	Edit Wardrobe Item	View Wardrobe	Fetch Weather Data	Refine Recommendation	Manage Wardrobe Data
FR-1.1	User account creation							
FR-1.2	Secure login & authentication	X	X	X	X			
FR-1.3	Password recovery							
FR-1.4	User role differentiation	X	X	X	X			X
FR-2.1	Upload photos of clothing items		X		X			
FR-2.2	Auto-categorize clothing items		X		X			
FR-2.3	Edit/delete wardrobe items			X	X			
FR-2.4	Secure storage of wardrobe data	X	X	X	X			X
FR-3.1	Generate outfit recommendations	X				X	X	
FR-3.2	Refine outfit recommendations	X					X	
FR-3.3	Occasion-based suggestions	X						
FR-4.1	Track clothing item usage	X	X	X	X			
FR-4.2	Sustainability metrics	X	X	X	X			
FR-4.3	Analytics dashboard	X						
FR-5.1	Fetch real-time weather	X				X		
FR-5.2	Weather-based adjustments	X				X		
FR-5.3	Auto-update weather	X				X		
FR-6.1	Admin monitors system							X
FR-6.2	Admin deletes content		X	X				X
FR-6.3	Admin DB maintenance							X
FR-7.1	Notifications (underused items)	X						
FR-7.2	Maintenance notifications	X	X	X	X			X
FR-7.3	Seasonal reminders	X						
NFR-1.1	Response time < 2s	X	X	X	X			X
NFR-1.2	Image upload < 10s		X					
NFR-1.3	Concurrent users (5000)	X	X	X	X			X
NFR-2.1	Usability / intuitive UI	X	X	X	X			
NFR-3.1	99% uptime	X	X	X	X			X
NFR-4.1	Encrypted credentials	X	X	X	X			X
NFR-4.2	HTTPS	X	X	X	X			X
NFR-4.3	RBAC	X	X	X	X			X
NFR-5.1	Scalability	X	X	X	X			X
NFR-6.1	Maintainability	X	X	X	X			X
NFR-7.1	Cross-platform support	X	X	X	X			X

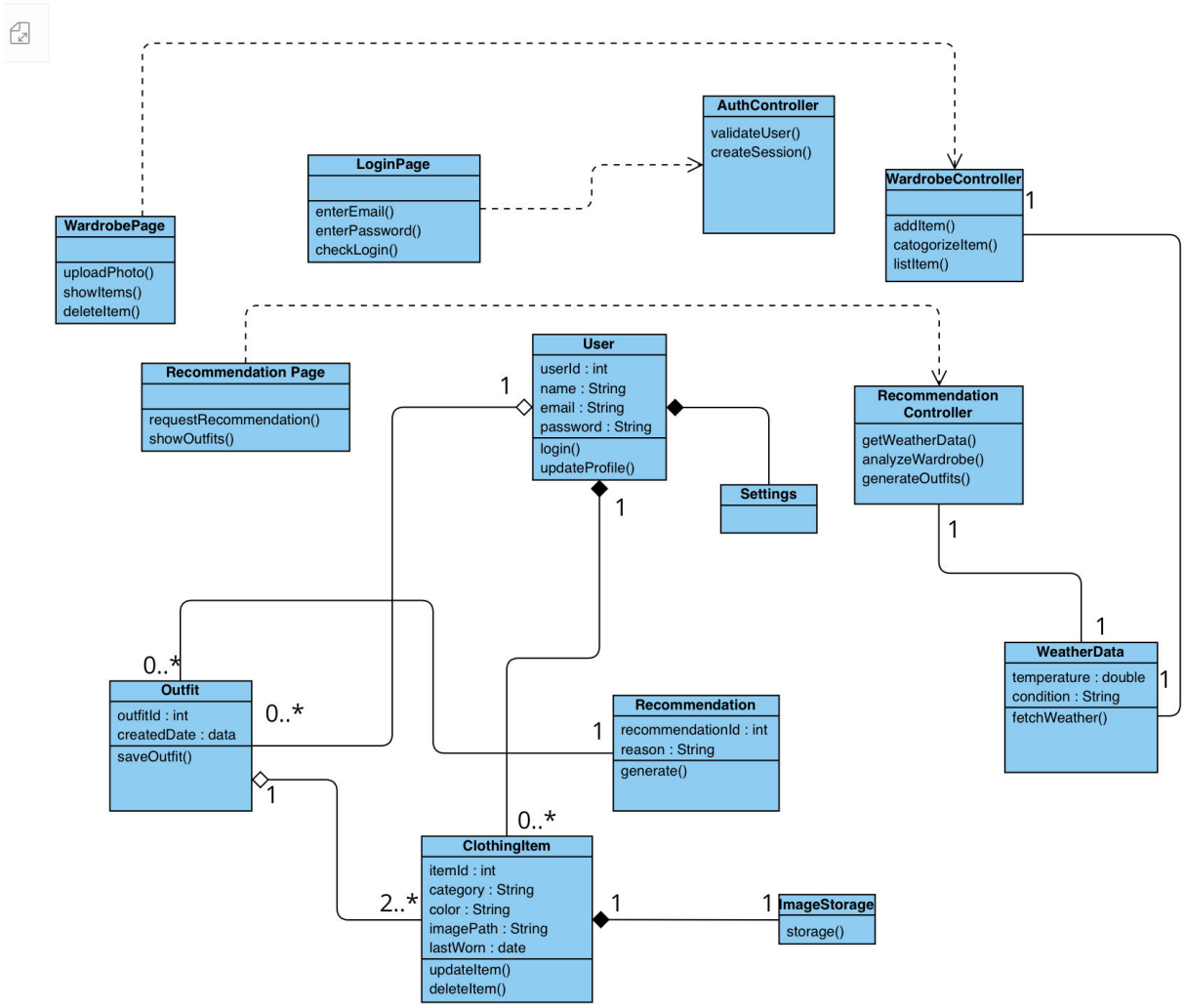
9. UML Use Case Diagram ‘



10. Activity Diagram



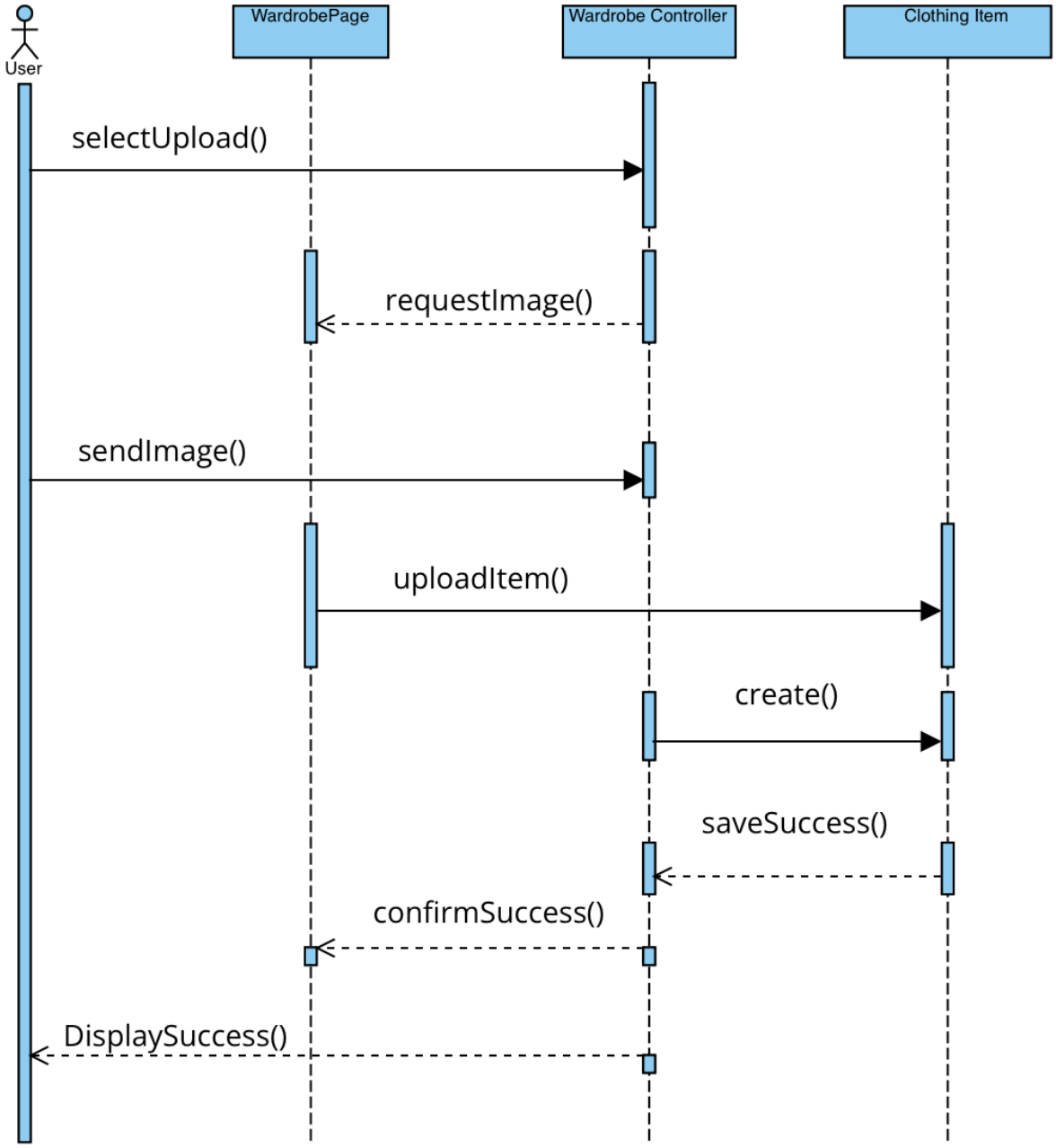
11. Class Diagram



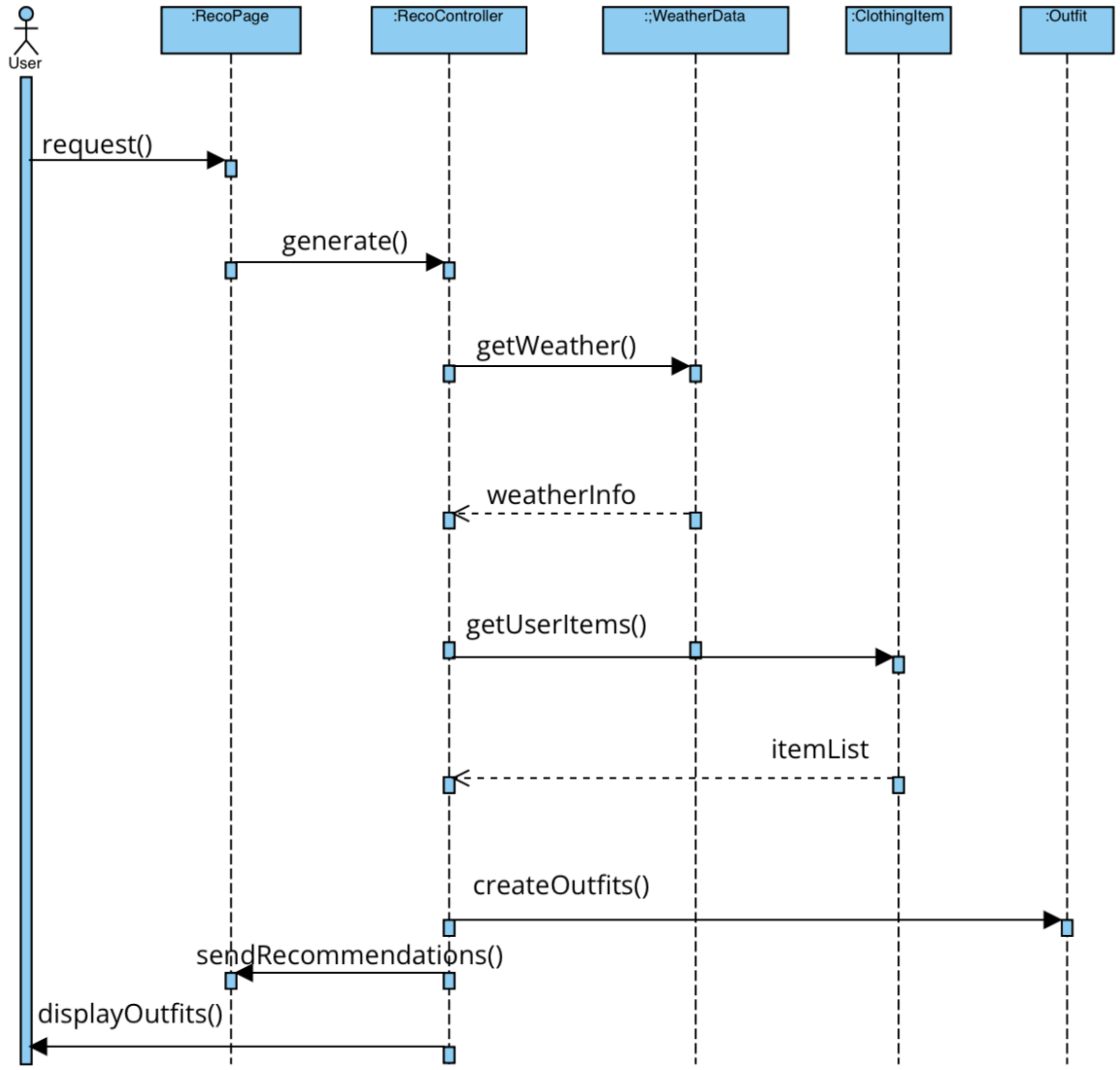
12. Sequence Diagrams

12.1

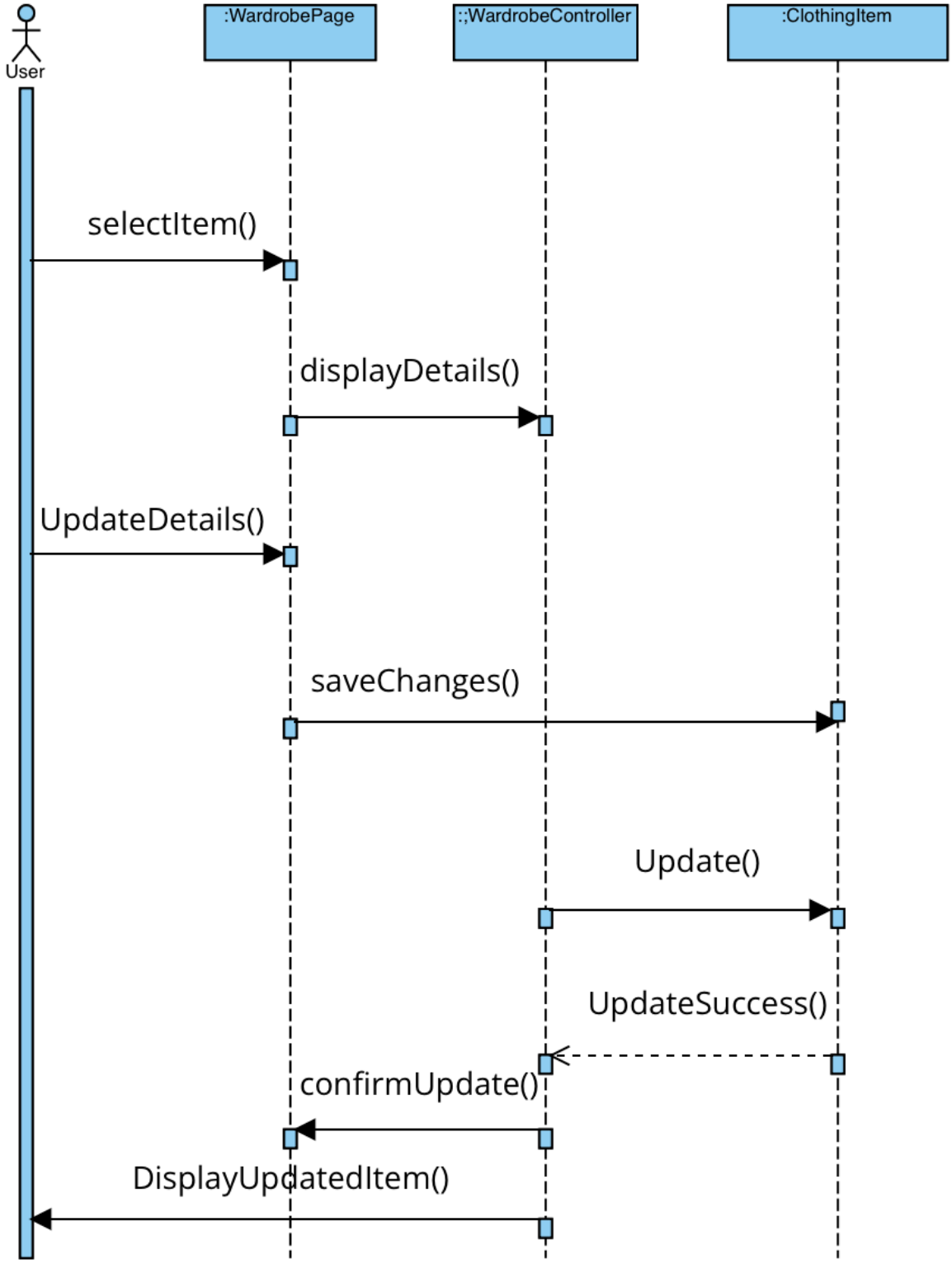
Upload Clothing Items



12.2
Get Outfit Recommendation



12.3
Edit Wardrobe Item


User
:WardrobePage
:;WardrobeController
:ClothingItem
selectItem()
displayDetails()
UpdateDetails()
saveChanges()
Update()
UpdateSuccess()
confirmUpdate()
DisplayUpdatedItem()

References

- United Nations Environment Programme (UNEP). (2021). *Sustainability and circularity in the textile value chain: Global stocktaking.* Retrieved from https://www.unep.org/
- Movinga. (2018). *The unused wardrobe: Global survey results.* Retrieved from https://www.movinga.com/