|  |
| --- |
| **Project Title:** **SmartWardrobe** |
| **Project Summary:**  The project's goal is to develop a fashion e-commerce platform driven by AI that provides customers with a highly tailored and knowledgeable purchasing experience. The site will enable users to search for products using natural queries like "Show me red dresses for summer" or even upload photographs to find visually similar things as a result to the integration of powerful Natural Language Processing (NLP) capabilities. Additionally, based on user behavior, previous purchases, and preferences, the site will make individualized product recommendations using machine learning algorithms.  There will be two primary methods of recommendation: content-based filtering, which makes recommendations based on product features like color, style, or brand, and collaborative filtering, which makes recommendations based on the preferences of similar users. The platform will also have an outfit-matching feature, where AI makes recommendations for complimentary items to finish a look, improving the user experience while browsing and promoting more transactions.  NLP-based sentiment analysis will be used to examine customer evaluations and produce insightful analysis. Review summaries that are automatically generated by the system will assist users in rapidly identifying the most important comments, such as "Highly rated for comfort" or "Most users loved the fit."  The purpose of this project is to improve the online buying experience by offering a smooth, user-friendly, entertaining, and personalized platform. This platform will be useful for fashion followers, internet shoppers, and people seeking for personalized advice. In a typical use case, a user might perform a few clicks to search for a certain item, get outfit recommendations based on AI, and rapidly study shortened customer reviews before making a purchase. As a result, customers get a more personalized shopping experience, which boosts customer happiness.  Example Use Case:  Emily is a user who visits to the AI-powered fashion platform in search of summer festival attire. She enters the scenario as follows: "Show me eco-friendly, sleeveless jumpsuits for a summer festival under 75 euros, with reviews mentioning comfort and durability." After processing her request, the platform's natural language processing technology shows her a variety of eco-friendly jumpsuits that meet her requirements.  Emily uploads a photo of a dress she saw on social media because she is interested and the image recognition component of the site suggests outfits that are similar and may be purchased. Based on her past browsing behavior and preferences, the AI makes related product recommendations as she adds items to her cart, such as a dress from a favorite brand.  Emily reads the reviews before making her buy. Using the platform's sentiment analysis feature, she finds simplified highlights like "Most users loved the fit" and "Highly rated for quality" instead of manually going through hundreds of individual reviews. She feels more certain about her decision after reading this description. |
| **Project Development:**  **Front-end**  Frontend MVP  - Conversation based search for fashion products.  - Filter according to the description of the products.  - Save products in Collection and favorite products for later use.  - Recommended Fashion products for different events.  Frontend Technical Components  - React.  - MUI and Ant Design Components for rapid prototyping and responsive design.  - Redux + Redux Toolkit for State Management.  - React Router for app routing.  - Deployed on AWS Amplify.  - Retrieves data from designated endpoints on backend server through HTTP/S  - Indicative List of Views:   * Homepage * Search Bar * Cart Option * Recommended Categories of Fashion Dresses * View All Dresses Option * Navigation Links for Collection, Favorites Dresses * Search History * Conversation Screen * Conversational Search bar * Image Input * Conversation Result * Filters * Product Details Modal * Product Images * Product Details * Product Reviews * Similar Items * Outfit matching * Payment Screen * Payment Details * Collections/Favorite Screen * Saved Collection Products * Cart Screen * User add items * Login Page   Frontend Learning Outcomes:   * Responsive and Accessible UI * UX Design and Testing Methodology * JavaScript + React   **Back-end**  Backend MVP   * Basic CRUD operations and endpoints for users authentication and authorization. * Apis for product listing, and product detail pages. * Advanced Apis for searching app & filtering products.   Backend Technical Components   * Node.js * Implement authentication and authorization using Nest.JS auth features & JWT. * Implementing RBAC for all users. * Using Clean code architecture. * Deploy on AWS Lambda. * Amazon DynamoDB for database storage. * S3 for image storage.   **API Endpoints**        **Back-end Learning Outcomes**  Node.js Application Development and Deploy   * Develop and deploy a scalable e-commerce back-end using Node.js and Express. * Handle core features like product listings, filters, favourites, and checkout. * REST API Creation   + Build RESTful APIs for managing product data, search, and user features. * External API Integration   + Incorporate third-party services for checkout payments and delivery. * Authentication and Authorization   + Secure APIs using JWT-based authentication. * Learn Clean code architecture for making reusable components.   **Data sources**   * The H&M dataset from the Kaggle competition, which contains information on the product specifics, fashion recommendations, customer behaviour. * Product classification and analysis will make use of metadata from the dataset, including product\_type\_name, colour\_group\_name, detail\_desc, and garment\_group\_name. * The Kaggle dataset contains transaction logs and user engagement data, including consumer purchases, views, and clicks. * Product descriptions and images can be found in the corresponding fields of the Kaggle dataset. * The dataset offers pre-labeled data that can be used to train the model for product classification according to occasion or season and for rule-based labelling. * Download the data directly from the Kaggle. After logging in and agreeing to the competition's regulations, the data can be downloaded in CSV format from the data page.   <https://www.kaggle.com/competitions/h-and-m-personalized-fashion-recommendations/data>  **System Architecture**  **A diagram of a software development  Description automatically generated**  **Cloud Architecture** |
| **Evaluation:**  1.⁠ ⁠Front-end evaluation  We will evaluate and improve each key stage of UI/UX development, and make adjustments based on feedback, testing and analysis. Design the page wireframes and core components before development to ensure consistent design style. Invite users to experience and collect their feedback after each delivery. This will help us find design problems and improve through each iteration. Communicate regularly with experienced UX designers or front-end developers and ask them to provide more professional improvement suggestions.  For front-end code testing, each React component should be tested separately to ensure that its functions can run independently. Ensure that different components can work together correctly, especially AI-driven functions such as product search and recommendation. Ensure that the user's input can smoothly trigger the search and return the correct results. Check whether the UI can be displayed correctly and run normally on different browsers and devices.  2. Back-end evaluation   * Unit testing. * Test the reliability, security, and performance of all API endpoints. * Ensure seamless integration between different services and components. * Verify the effectiveness of access control mechanisms to ensure that only authenticated users can access certain resources. * Simulate a large number of concurrent users accessing the system to test the system's ability to handle high traffic. * Measure the response speed of the backend to front-end requests, with a special focus on the performance of AI-driven search and recommendation features, as these features are critical to user satisfaction.   3. Data- side evaluation   * Data evaluation: We will ensure that the data input into the AI ​​model is high-quality and fully pre-processed. We will clean, deduplicate, and process missing data to ensure that the data used for recommendations and search is accurate and updated in a timely manner, thereby improving the reliability of the model. * Model evaluation: We will conduct comprehensive tests on AI models to evaluate their performance, such as the accuracy of NLP models, sentiment analysis, and recommendation engines used for search. We will collect user feedback on the accuracy of search results and recommendations and analyze it accordingly. With this feedback data, we will continuously retrain the model to ensure that the model can better meet user needs. |
| **Project Management:**  Group Meetings  We will organize a 15-minute stand-up meeting every morning, which will be conducted online. Each team member will share their progress or any obstacles at the meeting. We will have a sprint planning meeting every week, where we will review the sprint and determine the goals for the next sprint. After important deliverables, we will arrange a meeting to analyze the feedback from the mentor to determine the next stage of planning.  Development Management and Progress Tracking - Jira  All online team meetings are conducted through Teams, and we choose Jira as our project management tool. We use Jira's sprint board to track task progress in real time, ensure that all tasks have deadlines, and create automatic reports to monitor project progress against the schedule. We keep the task list organized by organizing the backlog, and can adjust the priority according to the specific project progress.  Code Management: GitHub  We will implement a clear Git branching strategy to manage feature development, bug fixes, and releases. We will create separate branches for development and testing to ensure isolated environments. Make code reviews mandatory for all pull requests in GitHub. Team members must review and approve code before merging it into the main branch. This ensures high code quality and prevents potential bugs.  Deadlines   * Create timeline – see Gannt chart below. * Working prototype and interim presentation – week 6. * User/System evaluation – week 9. * Final demonstration – week 13. * Final report – week 14.   Success  **Functionality: All the functions of the front-end and back-end meet all the project requirements. The front-end and back-end interaction is very smooth, and users will not have doubts about the functionality during testing.**  **AI Model: The AI ​​model we use, whether it is for recommendation, natural language processing (NLP), or sentiment analysis, should perform well in terms of accuracy, speed, and relevance.**  **User Interface: The interface needs to be user-friendly, responsive, and visually appealing. During testing, users should be able to easily navigate the website and use features such as search, recommendation, and outfit matching without any problems.**  A screenshot of a computer  Description automatically generated  A screenshot of a computer  Description automatically generated |

|  |
| --- |
| **Team Name: SmartWardrobe** |
| **Team Members:**   |  |  |  | | --- | --- | --- | | **Name** | **Student Number** | **Contact Number** | | **Pavan Kumar Murugan** | **D23124670** | **0894447599** | | **Muhammad Osama Noor Uddin** | **D23124872** | **0892785440** | | **Jiaxin Zhang** | **D23127255** | **0894857887** | | **Salil Sunil Luley** | **D23124871** | **0899490172** | | **Lin Chen** | **D23125391** | **0892063142** | |
| **Team Meetings:**   * Does everyone have to attend all meetings?   Yes, everyone must attend all meetings. This is a team project, and every member needs to be involved in all stages of the project.   * How often will there be meetings?   We will have a daily online meeting in the morning where everyone shares what they did yesterday, what they will do today, and any issues they faced. Every Friday, we will have an in-person meeting. If more communication is needed during the week, we may schedule another face-to-face meeting.   * Are there topics that are out-of-bounds?   Personal life topics are off-limits and should not be discussed during meetings.   * Online or face-to-face?   There will be daily online meetings, with a fixed face-to-face meeting every Friday. Additional face-to-face meetings may be scheduled mid-week if necessary.   * Decision making (Majority rule/Unanimous/Expertise wins)?   We will use when a majority rule approach making decisions.   * How will turn taking happen?   Each week, there will be a team leader, rotating in this order:  Pavan – Jiaxin – Osama – Salil – Lin. |
| **Team Conflict:**    Discuss issues like:   * How do you deal with habits of individual members?   We will respect each person’s working habits, but if any habit affects the team’s progress, we will discuss it as a group and find a solution that works for everyone.   * How do you deal with unresolved issues?   We will hold a separate meeting to focus on the issue. Every team member will have an equal chance to speak and share their ideas. If we still can't solve the problem, the team will seek help from mentors.   * How will you deal with conflict?   All our team members are very nice, and we are all people who enjoy communicating. If conflict does happen, we will address it openly and respectfully. When needed, we will use voting to make decisions for the team. Some compromises will be unavoidable.   * How will you avoid it?   When assigning tasks, we will make sure everyone's responsibilities are clear. If any team member faces a difficult problem, they should bring it up in the daily meeting. This will help the team support each other and make the project run more smoothly.   * Who will have the ultimate veto?   The majority team will have the ultimate veto. |