Spring 2024

Trend Tour

Aaron Ancheta

Khushboo Gupta

Xiomara Quinonez

SANTA CLARA UNIVERSITY
Department of Computer Science and Engineering

I HEREBY RECOMMEND THAT THE THESIS PREPARED
UNDER MY SUPERVISION BY

Aaron Ancheta
Khushboo Gupta
Xiomara Quinonez

ENTITLED

TREND TOUR

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
BACHELOR OF SCIENCE
IN
COMPUTER SCIENCE AND ENGINEERING

6-18-2024

Thesis Advisor date

6/18/2024

Department Chair date

TREND TOUR

by

Aaron Ancheta
Khushboo Gupta
Xiomara Quinonez

SENIOR DESIGN PROJECT REPORT

Submitted to
the Department of Computer Science and Engineering
of
SANTA CLARA UNIVERSITY
in Partial Fulfillment of the Requirements
for the degree of
Bachelor of Science in Computer Science and Engineering

Santa Clara, California
Spring 2024

TREND TOUR

Aaron Ancheta
Khushboo Gupta
Xiomara Quinonez

Department of Computer Science and Engineering
Santa Clara University
Spring 2024

ABSTRACT

Choosing outfits as a college student can prove challenging – especially for new college students and students new to the area where their college is. Trend Tour, our web-based application, displays what the fashion trends are in the user's college and near their area. This helps the user with ideas on how to dress. Furthermore, the user can change the color, silhouette, and/or accessories of outfits they want to see at that college and near their area. This web application aims to help college students feel more confident and comfortable.

Table of Contents

List of Figures and Tables

Chapter 1

Introduction

1.1 Problem Statement

A college student needs a way to style their outfits because they are new to college outfit trends. Fashion trends tend to change quickly and can vary greatly based on geography – for instance, a student coming from Minnesota who will begin their college career in California would have a different idea as to what the current fashion trends are – making the tracking of fashion trends difficult, especially when considering different seasons. In addition, people have different preferences and comfort levels when it comes to what they wear. This will allow students to explore different current trends custom to their individualized style, needs, and comfort in one streamlined interface as opposed to going through many forms of media.

1.2 Background

For our web application, we recognize that current college fashion trends are readily accessible through sources such as Instagram, TikTok, Pinterest, and more. However, we aim to consolidate this information in a more consolidated, user-friendly way. By going through the many sources, the user would have to manually keep track of what they are coming across most frequently – which can prove effortful and time-consuming. On the other hand, by using our web application, the user is able to more easily and quickly view what the current trends are based on the user's college and location.

1.3 Requirements

For our web application, we identified the functional requirements, non-functional requirements, and design constraints of our application. These are necessities of our application. The functional requirements are what the program must do, whereas the non-functional requirements describe how the functional requirements must be achieved. Meanwhile, the design constraints limit the methods in which the solution can be designed and implemented.

1.3.1 Functional Requirements

- Allows the user to input college/location
- Allows users to input preferences such as specific styles, colors, etc.
- The user interface displays trends and suggested outfits based on the user's college/location
- The user interface displays suggested outfits based on the user's preferences alongside college/location

1.3.2 Non-functional Requirements

- The user interface is intuitive
- The user interface is user-friendly

1.3.3 Design Constraints

- The application is web-based
- The application is compatible with common browsers

Chapter 2

Design

2.1 Conceptual Model

We created a quick mock-up to understand our project concept better. When the user opens the application, it takes them to the landing page that will show the name of the application along with a search bar to input your college. After that is entered, it will take you to the next page that will present the user with the current trends in each of the three categories. It will be simply presented as text. If they choose to continue, they will be taken to the next page where they have to pick a trend for each category or chase n/a if they do not want a certain category. The user will have a navigation bar so they can pick from the options. Once the user inputs the preferences, the user will be taken to the next page and will be able to click on each category to see the results. The results will be presented as pictures for a simple, clear interface.

Figure 2.1.1: Landing page to enter college name

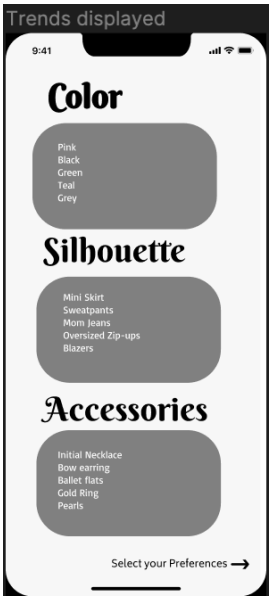

Figure 2.1.2: Current trends based on location

Figure 2.1.3: Input personal preferences

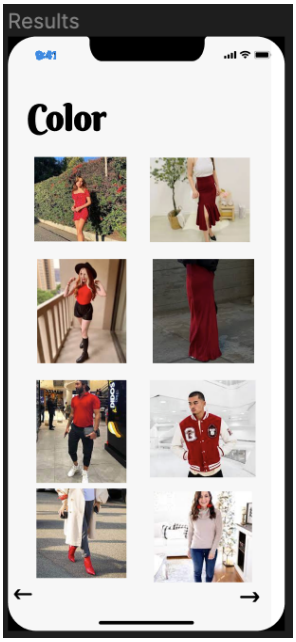

Figure 2.1.4: Results based on inputted preferences

2.2 Use Cases

In Figures 2.2.1 - 2.2.6 illustrated below, we depict our use case diagram features. In the created web application, when the user opens the application, the user inputs their college name. Once the application receives the location, it takes the user to the next screen.

Due to the timeline constraints, we implemented three categories. The three categories will be color, silhouettes (type of clothing), and accessories. The user will be able to see the top 5 trends of each of these categories on this screen.

At the bottom of the screen, the user will see the option “input your preferences”. Once the user selects that option, it will switch to the next screen.

In this screen, the user can choose their preference for each of those three categories using a navigation bar option that contains the five current trends. There will also be an option “n/a” in each of the three categories if the user does not want to pick from that category. After the user fills each of the three categories from the option, the user will hit “enter”.

After hitting enter, the user will be taken to another screen where all the three categories will become clickable to see 10 pictures of that chosen trend.

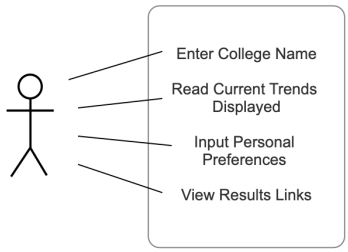

Figure 2.2.1: User options diagram

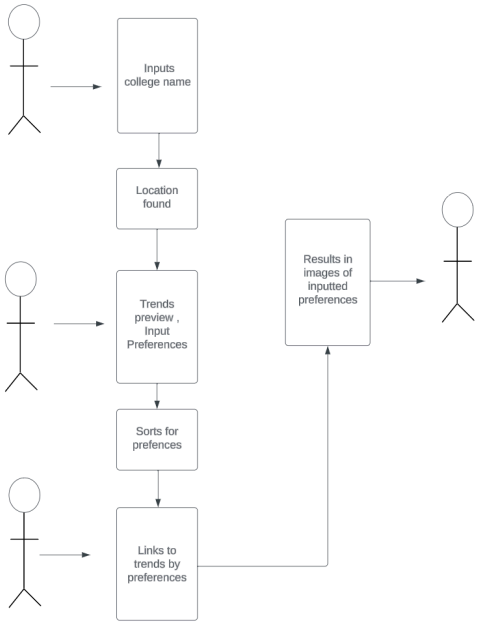

Figure 2.2.2: Use cases diagram

1. Enter College Name
    - Actor: User
    - Pre - Condition: A search bar that allows you to input your college
    - Post-Condition: The location is found and can be used given the input of the college by the user

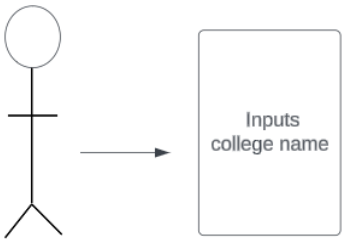

Figure 2.2.3: College input diagram

2. Current Trends Displayed
    - Actor: User
    - Pre-Condition: The location of the user is inputted and found. The application has been able to find trends specific to the location
    - Post-Condition: The user reads the trends and chooses to continue by clicking “input your preferences” at the bottom.

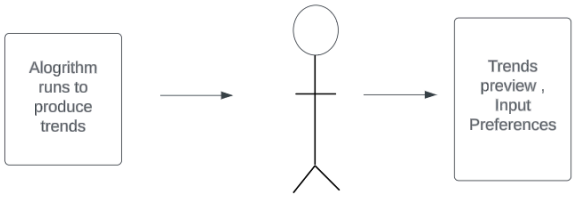

Figure 2.2.4: Current trends diagram

3. Personal Preferences Inputted
    - Actor: User
    - Pre - Condition: The screen has each of the three categories with populated navigation bars with the five predicted trends as options to choose from.
    - Post - Condition: The user chooses to continue by clicking “see the results” at the bottom

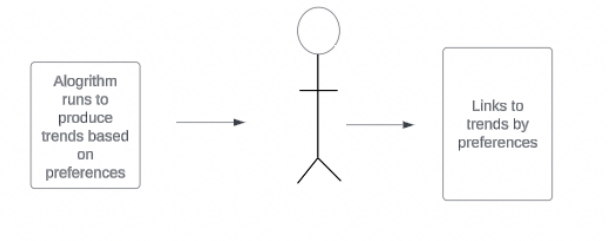

Figure 2.2.5: Personal preferences diagram

4. Personalized Results Displayed
    - Actor: User
    - Pre - Condition: The screen has each of the three categories with clickable links with your that takes you to the results page that has ten pictures presented created by the preference given by user
    - Post - Condition: The user clicks on the links and is switched to the results page with ten pictures of the chosen category with inputted preference.

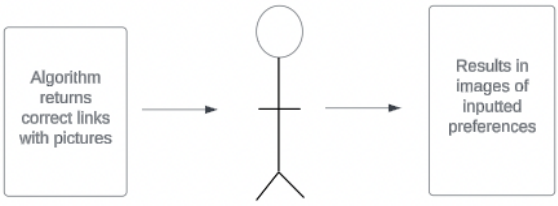

Figure 2.2.6: Personalized results diagram

2.3 Architecture Diagram

An architecture diagram is used to show the project as a whole. Using the mockups and the use cases, we were able to illustrate how the user of the application will be able to use the application. We created a simple high-level figure of the application where we showed that the user/client can access it using a device. The application will be on an online server that will display the results to the client on the device. We chose to use Ajax as it seems to be an efficient way of creating the connections between devices and the server. We believed that this was the most secure and fastest way to achieve the connection.

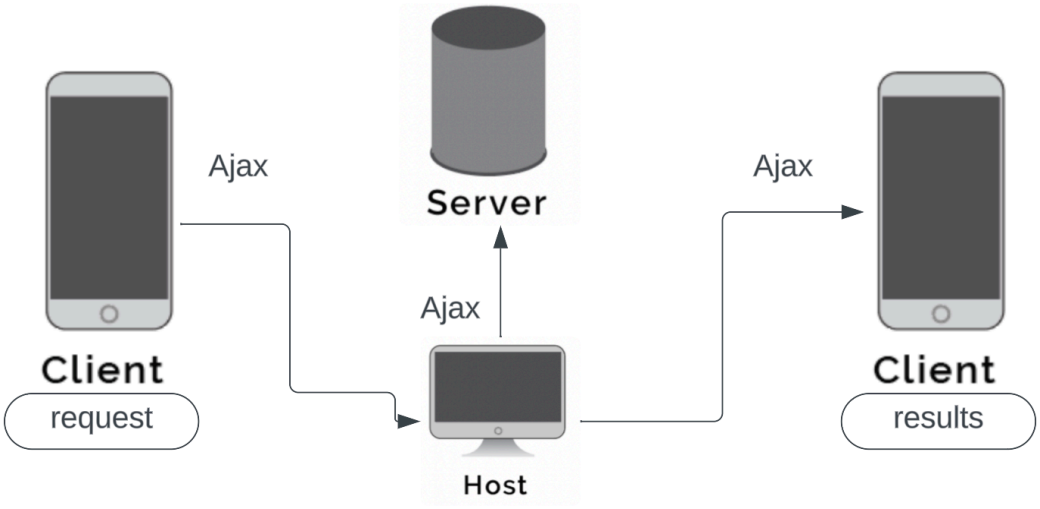

Figure 2.3.1: Architecture diagram

2.4 Technologies Used

To implement our website interface, we used the following technologies:

- Visual Studio Code[13]: a streamlined code editor for development, debugging, task running, and version control.
- Python[11][12]: a programming language used to build software, automate tasks, and analyze data.
- HTML[04]: a standard markup language used for creating web pages.
- CSS[01]: a style sheet language used for the presentation and styling of an HTML document.
- JavaScript[08]: programming language implemented into websites on the client side to manage website behavior.
- PyCharm[10]: an integrated development environment that offers a variety of tools for Python programming.
- YOLO Interface API[14]: API that enables object detection analysis on images without having to install YOLOv8 locally.
- Fashionpedia[02]: A YOLO object detection model for fashion-specific items.
- GitHub[03]: a platform used for code creation, storage, and management.
- Instaloader[05]: a Python library that we used to scrape images from Instagram accounts

2.5 Design Rationale

Before deciding on what kinds of designs and implementations to make for our project, we first thought about what would be the simplest way for our users to interact with our product in a way that was as informative and easy to understand as possible. To do this, we focused on creating a design that would be easy to navigate. The design choices and how they help with navigation and interactivity for users are listed below.

2.5.1 Image Scraping

We concluded that it would be best to scrape images, rather than videos due to the time constraints and skill level that we are at as a team. When doing research, it was difficult to find resources that would aid us in video scraping, and if we wanted to learn how to thoroughly video scrape, it would have taken up a lot of our time during the Winter Quarter. After doing some more research on how to scrape images from a variety of sources, we found that many libraries and APIs can help simplify the process. All available resources online also have different difficulty levels, so the team will have no problem finding different tools and resources to use for web scraping.

2.5.2 Scraping Instagram

Before agreeing on a scraping technique/platform to use for image scraping, we first researched what different platforms can be used. We found that TikTok would have been useful for going through videos of what users are posting as their outfits and what fashion trends they are following. However, TikTok often has unstable trends, and using it for scraping may be difficult to handle and expensive to access. YouTube was then another platform we considered to go through videos and understand what people are currently wearing and identify as fashion trends, similar to TikTok. However, we found that YouTube would be difficult to scrape from and that some videos may be too long. We then considered creating our own “source” through images that we could obtain of Santa Clara University students here on campus to get a more accurate representation of what is trending on college campuses, specifically in the Bay Area. However, for this, data is limited due to the limited amount of students that we would have access to. There would also be an issue with invasion of privacy and the issues of this being too expensive, timely, and difficult to collect. A more specific issue is that it would be biased for us to only collect images specific to Santa Clara University’s campus. One of the last few options we considered was fashion journals, which would not reflect the true fashion choices of everyday college students. We finally decided on Instagram as our source to scrape images from. With this tool, images can be found based on location, age, and following, which would allow us to find data for different filters in our final

implementation. With Instagram, we will use processing along with additional information to scrape and understand the data.

2.5.3 User Interface

We considered creating an app to present our final product; however, we realized that an app may be viewed as inconvenient to some people. For example, many people do not like the idea of having to download an app and have it installed on their phones to be able to use a product. This would make it much more difficult to persuade students to use our product in the future. So, we agreed on a website as the best final user interface to implement our solution. This made it easier to navigate through different parts of the user interface with no problem and move back and forth between pages easily. With a website, we were also able to implement and display different recommendations to users in a way that may feel less overwhelming to users.

2.5.4 Web App Design

Our web app consists of four main components. For the first component, we wanted some sort of “barrier” that is easily identifiable by users so that they are able to enter their college names easily and receive more accurate outputs based on their locations. This is the first thing the user sees so that they are able to get started using the web app to receive outputs. We also decided to create tabs so that there are predetermined categories that a user can choose from to filter out the trends. We chose the categories for the tabs to be clothing pieces, colors, and accessories because we found that this would be a bit simpler and less abstract to code out when scraping the web. Preferences inputted will be the third component of the web app, where users can choose a preference and narrow down the results. However, even if users choose to not input preferences, they will still receive general trends by default, as soon as they choose their school. The last component will be the results tab, where results are outputted both when the user inputs preferences and when they do not. This is how users are able to view the outputs of current trends that exist at their schools.

2.5.5 Visual Programming Environment

Visual programming environments are used to develop and edit source code. Through visual programming environments, users can perform anything from app to web development, through a simplified process. Visual programming environments also allow debugging and collaboration to occur more simply. The visual programming environment that the team chose to go with was Visual Studio Code, commonly known as VS Code. VS Code is the visual programming environment that the team is most familiar with and had no problem getting started with installing the different libraries needed for a variety of languages, due to the easy interface to navigate through in this environment.

2.5.6 Scraping Methods

Upon doing research, we found that PyCharm is an IDE often used for web scraping through Python and being able to create different scripts. The team had no experience with PyCharm; however, we found that this tool was the most interactive and easiest to navigate. There are many tutorials on how to get started using PyCharm, from installation to the actual coding portion. The tutorials are fairly easy to follow with copy-and-paste abilities that are compatible with all of our devices. To directly scrape the images from Instagram accounts, we used a library called Instaloader. By inputting usernames, we would be able to download all of the user's captions, videos, and pictures displayed on their profile.

2.5.7 Technologies

We used standard languages such as HTML[6], CSS[01], and JavaScrip[08] to create the website. We chose these languages to use because they are what are usually used for website creation and because this is what the team had the most experience with. Additionally, for the back end of actual web scraping, we used tools such as Python[11][12] and YOLO[14]. Python[11][12] and YOLO[14] will work together to provide us with the frameworks to begin web scraping and integrate any scraping needed into our solution. With the Fashionpedia[02] model, we will be able to access any data sets needed.

We also leveraged the use of the Fashionpedia YOLO[02] model, which focuses around creating bounds that allow us to detect objects in images, specifically fashion related items. After doing research on how the model works and using it ourselves, we found that there are five general steps that go into the performance of Fashionpedia[02]. First, various images that contain fashion items are fed into the model. In our case, we fed images to the model through the creation of folders that held different images from various content creators in different colleges. We had a folder for each college we focused on for our project. Second, preprocessing occurs to adjust the dimensions of each image. Third, object detection occurs, where bounds are created that determine what objects represent the fashion items. Fourth, thresholding applies a confidence score to filter out low confidence scores. Lastly, a final output for the pipeline is created to include a list of the detected objects with specific fashion items.

Chapter 3

Timeline

3.1 Spring Quarter 2023

- Ideas for what we wanted to do for our project
- Getting an advisor for the upcoming year

3.2 Fall Quarter 2024

- Creation of initial Design Document, outlining:
    - Planned implementation
    - Requirements
    - Architecture of prototype
    - User experience

3.3 Winter Quarter 2024

- Initial implementation of the platform
    - Beginning of user interface and connection with backend functionality
- Researching web scraping, machine learning algorithms

3.4 Spring Quarter 2024

- Completion of front end and back end implementation
- Usability testing
- Senior Design Conference
- Final Design Document

Chapter 4

Risk Table

We conducted a risk analysis in order to identify possible risks that would threaten our progress during the year. Each risk is assigned a probability from zero to one and a severity from zero to ten, in which the impact of each risk is calculated by multiplying the probability and severity together. By doing this, we attempted to prevent such risks and associated consequences.

In regards to the lack of coding knowledge, we thought our mitigation strategies would help. However, we hit multiple roadblocks. For instance, we adjusted how we would scrape our data from Instagram multiple times – was a big part of our project – ultimately slowing us down for a big part of Winter Quarter 2023. In addition, we were unable to successfully create a completely new model and train it from the ground up like we wanted to, in which we spent a lot of time trying to figure out, also slowing us down. Thankfully, in the end we completed a functioning project with all of the necessary aspects we determined. By doing incremental tests, we were able to finish on time.

Risk	Consequence	Probability	Severity	Impact	Mitigation strategies
Lack of coding knowledge	Doing tasks that we have not yet learned in class	1	10	10	Doing research on the languages and technologies we will be writing our program in, working together on each section, and consistently meeting for help
Running out of time	All of the necessary functions of our project are not completed by the conference	0.3	10	3	Using extensive time management strategies, using Trello, and having regular meetings in order to keep up
Bugs and errors	We would have to investigate what part of the code is not working correctly and fix it, taking away time from work	1	5	5	Incrementally test each step of the way in order to find bugs and solve them as soon as possible
Incomplete requirements	We do not implement a program according to the requirements	0.2	10	2	Review our requirements and ensure they meet the necessary outcomes and regularly discuss the requirements with our advisor

Table 4.1: Risk table

Chapter 5

Test Plan

Our test plan for the web application included three parts to ensure that our application was functional in a multitude of environments and settings.

5.1 Part A Testing

- Unit testing for edge cases, issues with the user interface, and other issues that could arise with the front-end
- Testing issues of our front-end and back-end connecting

5.2 Part B Testing

- Bringing the user interface to friends and peers and getting feedback on what works and what does not
- Further testing issues of our front-end and back-end connecting

5.3 Part C Testing

- Ensuring concurrent requests from users are able to be handled
- Checking for accurate photos of outfits

Chapter 6

Constraints and Standards

In recognizing constraints and standards, we will have a better idea as to what a realistic implementation of our design will look like. This allowed us to make a more socially and ethically responsible technological product.

6.1 Constraints

During our timeline, we were met with the constraint of time. We chose to implement this project as a way to learn and exercise machine learning and artificial intelligence skills that we were not experts. Since this project was implemented from scratch, not only time was spent on learning the new technology but also starting the project from scratch. Given that all of us had a full load of classes and our commitments, we found time to ensure that we are learning along the way and being mindful of our project and its implications. Brainstorming an idea, researching the market, understanding what can be done, and discussing with our advisors was all part of the project timeline which gave limited time for implementation.

During the months of implementation, we were met with multiple obstacles. We had created a script that downloaded the needed data/photographs but due to updates in social media, the script no longer was supported. This forced us to pivot and work on another way of gathering our data.

Since we were using personal computers, we soon learned that large language models were slow and unstable on our sized computers. To ensure that we could continue to make progress on our project, we pivoted to make sure the size of our model could be handled by our personal computers.

Since we were not allotted funds for the project, we were not able to host it online making it difficult to upkeep and maintain after deployment. New data sets would have to be continually scraped in order to keep up with modern trends. Furthermore, the foundation of new colleges after deployment would further complicate maintenance.

Because we are using public users' photos off of social media, ethical constraints exist. For instance, initially we wanted to ensure that we are keeping who we get the photos from and their information private to ensure safety. In addition, we wanted to ensure that we are not violating any of Instagram's policies. As we were working on the project, we decided to pivot to influencers and content creators as it seemed to multiple advantages. Content creators and influencers benefit from their photos being used and publicize their photos themselves. This increased their audience and also aligned with our ethical beliefs as they chose to keep the photos/datas open sourced and want to reach bigger audience. Thus, we only scraped content creators' and influencers' photos off of social media.

6.2 Standards

In regards to standards, we plan on following standards for our programming languages and design methodologies. Using HTML [04] and JavaScript [08], we will follow ISO standards. Using CSS, we will follow W3C standards [01]. Finally, as Python is not considered a standardized language, we will follow recommended practices and standards based on its documentation[11][12].

Chapter 7

Societal Components

As engineers and as students designing a project at Santa Clara University, it is important to recognize how the applications we create affect the communities around us – thus we must conduct ourselves ethically. In doing so, below we discuss how Trend Tour interacts with society and our communities.

7.1 Ethical

An ethical issue that we found most prevalent was using Instagram pictures for our data. We knew that we wanted to use Instagram pictures in our system in order to know that the respective pictures were at the user's college or near their area, but we wanted to ensure that we did so ethically. To ensure that we did this ethically, we plan on only using public Instagram pictures and keeping the data (username, location, etc.) private. We also only used influencers' and content creators' photos to ensure that we only used photos by users who wanted their photos to be public. Another ethical issue was the possibility of promoting fast fashion. We tackled this by using photos from only college influencers and content creators and not including any direct links to what was being worn. Finally, the last ethical issue we considered was the possibility of users feeling pressured to dress a certain way or wear certain clothing. To address this, we included a disclaimer stating that our purpose was to allow others to explore fashion with convenience and inspire ideas and that no sponsorships were ever used – instead of promoting certain products and people that would directly benefit us.

7.2 Social

We view our project as a tool to help college students pick out outfits more easily and feel more confident. We feel that our project is a tool that not only connects the community over a shared love of fashion and brings the community together, but a tool that also makes the members of a community feel more comfortable and confident.

7.3 Political, Economic, Health and Safety, and Manufacturability

As our project is a largely free, web-based application, we did not find that there are any relevant political, economic, health and safety, or manufacturability issues to discuss.

7.4 Sustainability and Environmental Impact

Because our project is a web-based application, we found it to be sustainable and environmentally friendly. Furthermore, by helping college students find ideas for outfits they will like, our project helps communities live more sustainably and benefits the environment, especially when compared to fast fashion.

7.5 Usability

Usability was a key point for this project, as we wanted to ensure our program was intuitive. To do so, we planned our design using diagrams and charts first before implementation. Additionally, user testing by those unfamiliar with the program would help us determine if our program is intuitive.

7.6 Lifelong Learning

Trend Tour would reiterate the importance of lifelong learning, as we would have to learn new languages and technologies in order to implement the project. Thus, this project will help us in how to better and quickly learn important, relevant parts of a language.

7.7 Compassion

While our program is modest and does not necessarily relieve suffering around our community, it can still prove as a source of positivity by helping college students feel more comfortable and confident with themselves. Additionally, it is inclusive and brings people interested in fashion together in a positive environment.

Chapter 8

Conclusion

Through Trend Tour, we learned valuable lessons and skills. First, we learned how much time, effort, and careful planning it takes before even implementation when creating a project. We also learned much more about machine learning algorithms and web scraping techniques, which was an important part to us. Finally, we learned how to think ethically when it comes to making a project – especially in terms of how it ties into society and the globe.

As of now, some disadvantages of our current design are that it can be difficult to maintain in the future as it would have to be continually updated in order to keep up with the current trends and that it only scrapes from Instagram instead of multiple sources.

However, we feel that our project is a unique one that is polymorphic – while our focus was college students, it could be adapted and applied to fashion designers, fashion analysts, content creators, influencers, people moving, and people looking for a wardrobe change.

Chapter 9

Appendix A: User Manual

How to enter the page displaying clothing recommendations:

1. On the authentication page, host clicks on textbox
2. Host enters their school
3. System generates default outputs of clothing that is currently trending in host's inputted school

How to generate outputs based on color:

1. Host selects drop-down menu
2. Host clicks on a button that says "color" on drop-down menu
3. Host clicks on a specific color they would like to see
4. System generates new set of clothing outputs based on specific color that host chosetech

How to generate outputs based on accessory:

1. Host selects drop-down menu
2. Host clicks on a button that says "accessory" on drop-down menu
3. System generates new set of outputs based on accessory button that host chose

How to generate outputs based on specific clothing item:

1. Host selects drop-down menu
2. Host clicks on a button that says "clothing" on drop-down menu
3. Host clicks on a specific clothing they would like to see
4. System generates new set of clothing outputs based on specific clothing item that host chose

References

1. CSS. (n.d.). Cascading Style Sheets (CSS). World Wide Web Consortium (W3C). Available at: https://www.w3.org/Style/CSS/
2. Fashionpedia. (n.d.). Fashionpedia: A Dataset for Fashion Image Analysis. Available at: https://github.com/ashrafminhajul/Fashionpedia
3. GitHub. (n.d.). GitHub: Where the World Builds Software. GitHub, Inc. Available at: https://github.com/
4. HTML. (n.d.). HyperText Markup Language (HTML). World Wide Web Consortium (W3C). Available at: https://html.spec.whatwg.org/
5. Instaloader. (n.d.). Instaloader: Download pictures (or videos) along with their captions and other metadata from Instagram. Available at: https://instaloader.github.io/
6. ISO/IEC 15445:2000. (2000). *Information technology — Document description and processing languages — HyperText Markup Language (HTML)*. International Organization for Standardization. Available at: https://www.iso.org/standard/28234.html
7. ISO/IEC 16262:2011. (2011). *Information technology — ECMAScript language specification*. International Organization for Standardization. Available at: https://www.iso.org/standard/55755.html
8. JavaScript. (n.d.). JavaScript Guide. Mozilla Developer Network (MDN). Available at: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide
9. PEP 8. (2001). *Style Guide for Python Code*. Python Software Foundation. Available at: https://peps.python.org/pep-0008/
10. PyCharm. (n.d.). PyCharm: The Python IDE for Professional Developers by JetBrains. JetBrains. Available at: https://www.jetbrains.com/pycharm/
11. Python. (n.d.). Python Programming Language. Python Software Foundation. Available at: https://www.python.org/
12. Python Official Documentation. Python Software Foundation. Available at: https://docs.python.org/3/

13. Visual Studio Code. (n.d.). Visual Studio Code - Code Editing. Redefined. Microsoft. Available at: https://code.visualstudio.com/
14. YOLO Interface API. (n.d.). YOLO: Real-Time Object Detection. Available at: https://github.com/ultralytics/yolov5