FASHION EYE

CSD 334 MINI PROJECT

CSU19A05 MDL19CS013 Anagha S Menon
CSU19A18 MDL19CS032 Ashwin Girish
CSU19A48 MDL19CS092 Pranav Jayashankar
CSU19A54 MDL19CS104 Sidharth S
B. Tech. Computer Science & Engineering

Department of Computer Engineering
Model Engineering College, Ernakulam
Thrikkakara, Kochi 682021
Phone: +91.484.2575370
http://www.mec.ac.in
hodcs@mec.ac.in

August, 2022

Model Engineering College, Ernakulam
Dept. of Computer Engineering

C E R T I F I C A T E

This is to certify that, this report titled ***Fashion Eye*** is a bonafide record of the work done by

CSU19A05 MDL19CS013 Anagha S Menon
CSU19A18 MDL19CS032 Ashwin Girish
CSU19A48 MDL19CS092 Pranav Jayashankar
CSU19A54 MDL19CS104 Sidharth S

Sixth Semester B. Tech. Computer Science & Engineering

student, for the course work in **CSD 334 Mini Project** which is the Mini Project Work, under our guidance and supervision, in partial fulfillment of the requirements for the award of the degree, B. Tech. Computer Science and Engineering of **APJ Abdul Kalam University** .

Guide

Ms. Ameena K Ashraf
Asst. Professor
Computer Engineering

Coordinator

Mrs. Aysha Fymin Majeed
Asst. Professor
Computer Engineering

Head of the Department

August 5, 2022

Dr.Preetha Theresa Joy
Professor
Computer Engineering

Acknowledgement

We are profoundly grateful to Prof. Aysha Fymin Majeed for her expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. We would like to express deepest appreciation towards Dr. Jacob Thomas V,Principal, Govt Model Engineering College,Thrikkakara, Prof. Dr.Preetha Theresa Joy, Head of Department of Computer Engineering and Asst. Prof. Aysha Fymin Majeed, Project Coordinator whose invaluable guidance supported us in completing this project.At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

CSU19A05	MDL19CS013	ANAGHA S MENON
CSU19A18	MDL19CS032	ASHWIN GIRISH
CSU19A48	MDL19CS092	PRANAV JAYASHANKAR
CSU19A54	MDL19CS104	SIDHARTH S

Abstract

Fashion is a fast changing arena and it is often inspired by past designs. Fashion eye uses clothing semantic generation of fashion design using StyleGAN and GANSpace to generate new clothing designs that are inspired by existing designs. The ML model uses pre-existing design properties and uses those features to generate styles that can be customized according to user needs. The users can customize based on their measurements and fashion preferences.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Fashion is a fast changing arena. New fashion items are created ever so quickly in the recent years and the demand keeps on increasing.

1.1 Proposed Project

1.1.1 Problem Statement

Creating new design is a creative process that requires much contemplation and effort. Even after designing a new model to the specification provided, to know the fit and beauty of it on the person it was intended to, the designer have create the model and use up resources. It banks a lot on the creativity and imagination of the creator as well as their ability to help their client visualize and understand their idea. Moreover in the highly competitive arena such as fashion, designers sometimes burn out in phases of less inspiration.

1.1.2 Proposed Solution

Fashion Eye is a design and virtual try on solution that can help designers and common users alike. It can generate new cloth designs by combining 2 existing designs. For a designer, this tool can help them get more inspirations. Fashion eye also provide a platform for users to upload their image and the image of a cloth with clean white background and get a reliable representation of them wearing the given cloth. This can help designers communicate their ideas with clients more efficiently. Users can also use this tool to virtually try on a cloth they wish to buy online.

Chapter 2

Report of Preparatory Work

2.1 Literature Survey Report

1. **Paper 1: Generation of fashionable clothes using generative adversarial network, Montek Singh, 2019** - GAN architectures like image-to-image translation take image from one domain and generates an image in other domain. To check if the image lies in a particular domain or not, we use a discriminator and to force the generated image have elements of the original domain mode we use a reconstruction loss. Therefore, one-to-one mapping is developed. To generate image of other domain from a given domain while keeping its features intact can be done by utilizing cross-domain relationships. Generated clothes should have features of new as well as old-fashioned clothes. For the initial epochs the output majorly consists of noise. After running the system for about 5,000 epochs, one will get good outputs. As developed system works on the concept of GANs. Therefore, the loss and efficiency is measured using the visual perceptions of the newly formed design patterns. The main drawback of the system is, it is highly depend on the number of training images. The biggest limitation of the collected data set is that the clothes in the two domains do not belong to a specific category.

2. **Paper 2: A style-Based Generator Architecture for Generative Adversarial Networks, Tero Karras NVIDIA, 2019** - The generator in this model allows to control image synthesis by scaling different parameters associated with the style. Effects of each style is localized to network, ie, it only affects certain features of the image. To further encourage localization, style mixing is done by mixing regularization where only a percentage of two latent codes are mixed rather than one during training. The results of accuracy matrices show that StyleGAN produces better quality outputs than other GAN models image processing. It also ensures disentanglement of different features or subspaces of the input image. Elements or features of image that may occur randomly without following a common distribution (such as the freckles or position of hair on the image of a person's face/head) are called stochastic elements. This model has better performance with stochastic features. A major challenge is that this model is currently developed to work on human faces but can be transfer learned to objects such as clothing designs. The model requires big data with high quality images

having the subject in the foreground and a clean background.

3. **Paper 3: DeepFashion2: A Versatile Benchmark for Detection, Pose Estimation,Segmentation and Re-Identification of Clothing Images, Yuying Ge, 2019** - DeepFashion2 is a large-scale benchmark with comprehensive tasks and annotations of fashion image understanding. DeepFashion2 contains 491K images of 13 popular clothing categories. A full spectrum of tasks are defined on them including clothes detection and recognition, landmark and pose estimation, segmentation, as well as verification and retrieval. In the model, a novel Match R-CNN is also proposed to aggregate all the learned features from clothes categories, poses, and masks to solve clothing image retrieval in an end-to-end manner. DeepFashion2 possesses the richest definitions of tasks and the largest number of labels. It is large, versatile, expressive and diverse sample of clothing. Major disadvantages include decrease in the accuracy rate of detection of clothes of very small or very large sizes or in zoomed images. It also require longer training on a bigger data base.

4. **Paper 4: Readily Design and Try-On Garments by Manipulating Segmentation Images, Yoojin Jeong, 2020** - In this paper, a new apparel image is created using a generative model that can apply a new style to a desired area in a segmented image. It also creates a new fashion image by manipulating the segmentation image. Thus, interactive fashion image manipulation, which enables users to edit images by controlling segmentation images, is possible. Experts can easily prototype ideas and stimulate imagination. Anyone can design because it can be easily expressed with a simple touch. User-centered design is possible because users can quickly apply the style they want, so it is useful for choosing or recommending garments and fashion items. It is difficult to apply artificial intelligence to clothes because they have many changes in design and style. As real fashion is much more complex and diverse, there are some problems to be solved to apply this paper's methods directly to the fashion industry. Compared to the designer's ability, the completeness of the result is low.

5. **Paper 5: Virtual Clothing Try-on Using Generative Adversarial Networks, Rahul Singh, 2021** - Virtual clothing try-on aims at transferring a target clothing image onto a reference person. The traditional try-on task aims to align the target clothing item naturally to the given person's body and hence present a try-on look of the person. However, in practice, people may also be interested in their try-on looks with different poses. In this work, first the semantic layout of the reference image that will be changed after try-on is predicted, and then determines whether its image content needs to be generated or preserved according to the predicted semantic layout, leading to photo realistic try-on and rich clothing details. First, a semantic layout generation module utilizes semantic segmentation of the reference image to progressively predict the desired semantic layout after try-on. Second, a clothes warping module warps clothing images according to the generated semantic layout, where a second-order difference constraint is introduced to stabilize the warping process during training. Third, a content fusion module that integrates all information (e.g. reference image, semantic layout, warped clothes) to adaptively produce each semantic part of human body. This model succeeds in preserving both the character of clothes and details of human

identity, robustness of the implemented model and the high quality of the images generated. This model has potential to be employed in generating photo-realistic try-on images in online shopping and other websites. But the model has a long way to go before it can generate satisfactory results for all kinds of clothing and reference poses.

2.2 System Study Report

2.2.1 System Functions

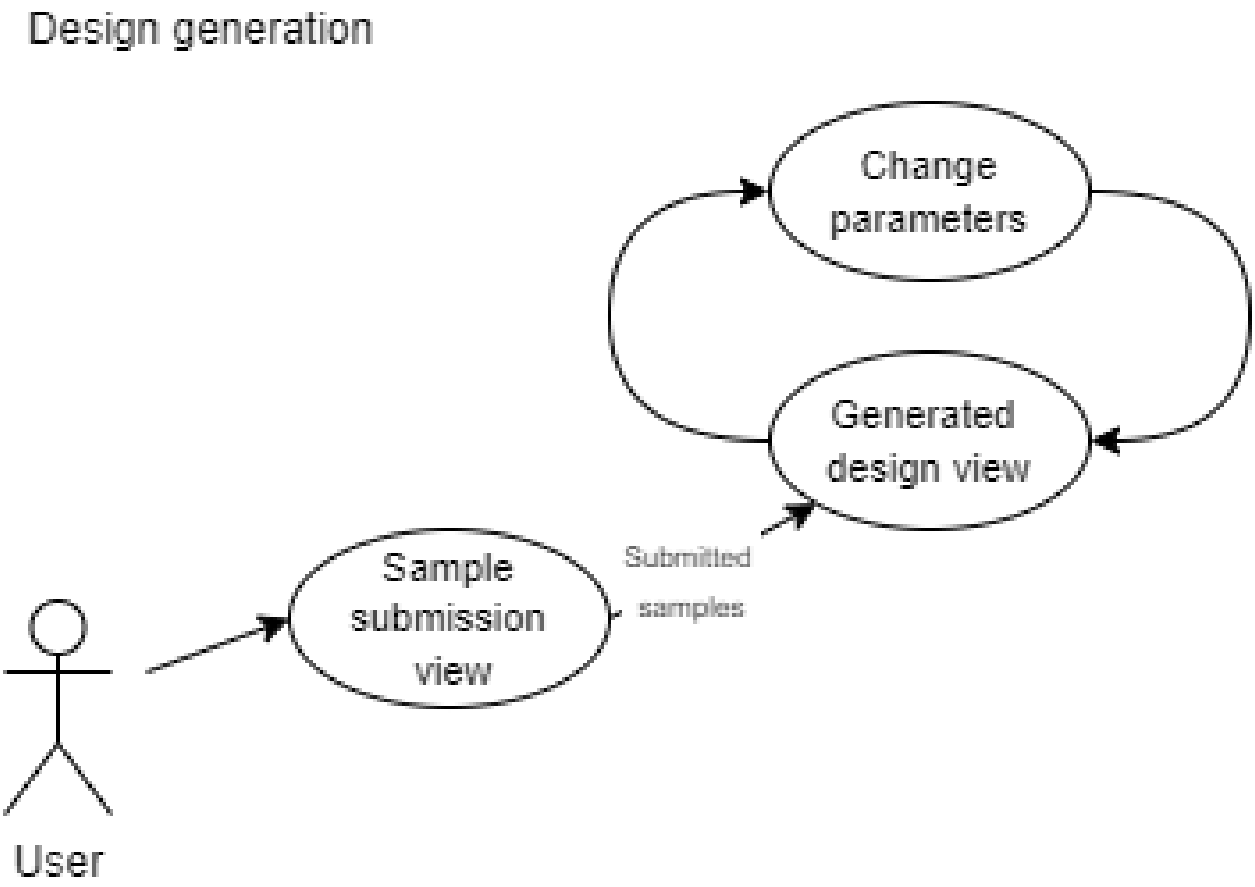

Figure 2.1: Use case diagram: Cloth Generation

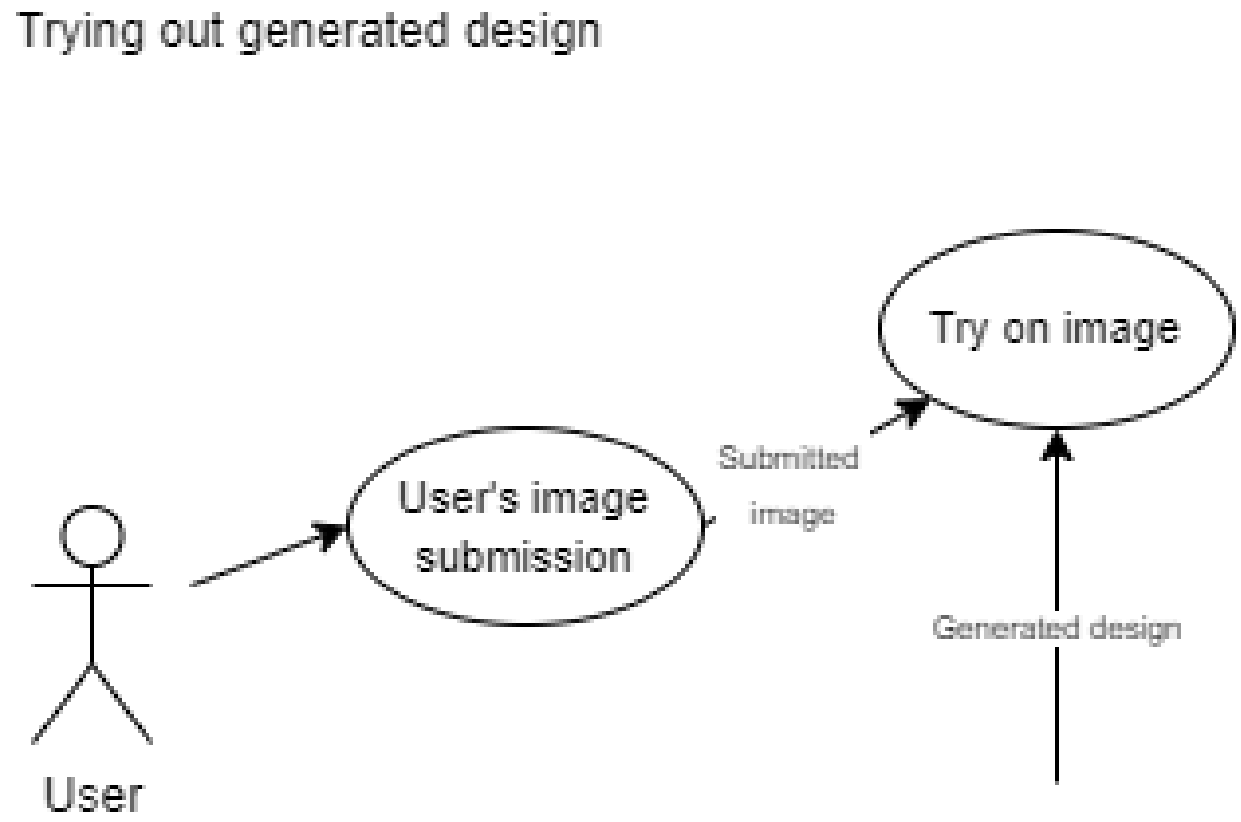

Figure 2.2: Use case diagram: Virtual Try On

2.2.2 Technological Stack

- Front-end / Web App

  1. **HTML** The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.
  2. **CSS** CSS stands for Cascading Style Sheets, and it's used to add style to a web page by dictating how a site is displayed on a browser. CSS is unique in that it doesn't create any new elements, like HTML or JavaScript. Instead, it's a language used to style HTML elements.
  3. **React.js v18** React is a free and open-source front-end JavaScript library for building user interfaces based on UI components. It is maintained by Meta and a community of individual developers and companies.

4. **React Router Dom v6** React Router is a lightweight, fully-featured routing library for the React JavaScript library. React Router runs everywhere that React runs; on the web, on the server (using node.js), and on React Native.
5. **Vite v3.0** Vite is a frontend build tooling that significantly improves the frontend development experience. It consists of two major parts: A dev server that serves the source files over native ES modules, with rich built-in features and astonishingly fast Hot Module Replacement (HMR). A build command that bundles code with Rollup, pre-configured to output highly optimized static assets for production. In addition, Vite is highly extensible via its Plugin API and JavaScript API with full typing support.

- Back-end interface

1. **Gradio v3.1.2** Gradio is an open-source Python library that is used to build machine learning and data science demos and web applications. With Gradio, it is possible to quickly create a beautiful user interface around machine learning models or data science workflow and let people "try it out" by dragging-and-dropping in their own images, pasting text, recording their own voice, and interacting with your demo, all through the browser.

- Back-end

1. **TensorFlow v1.x (For training)** TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML-powered applications. TensorFlow was originally developed by researchers and engineers working on the Google Brain team within Google's Machine Intelligence Research organization to conduct machine learning and deep neural networks research. The system is general enough to be applicable in a wide variety of other domains, as well. TensorFlow provides stable Python and C++ APIs, as well as non-guaranteed backward compatible API for other languages.
2. **PyTorch v1.12** PyTorch is a Python package that provides two high-level features: Tensor computation (like NumPy) with strong GPU acceleration Deep neural networks built on a tape-based autograd system. Python packages such as NumPy, SciPy, and Cython can be reused to extend PyTorch when needed.
3. **Torchvision v0.13** The torchvision package consists of popular datasets, model architectures, and common image transformations for computer vision. Transforming and augmenting images. Scriptable transforms. Compositions of transforms. Transforms on PIL Image and torch.Tensor.
4. **Torchgeometry v0.1.2** The PyTorch Geometry package is a geometric computer vision library for PyTorch. It consists of a set of routines and differentiable modules to solve generic geometry computer vision problems. At its core, the package uses PyTorch as its main backend both for efficiency and to take advantage of the reverse-mode auto-differentiation to define and compute the gradient of complex functions.

5. **Paddle GPU v2.3** PaddlePaddle has been officially open-sourced to professional communities since 2016. It is an industrial platform with advanced technologies and rich features that cover core deep learning frameworks, basic model libraries, end-to-end development kits, tools and components as well as service platforms. PaddlePaddle is originated from industrial practices with dedication and commitments to industrialization. It has been widely adopted by a wide range of sectors including manufacturing, agriculture, enterprise service, and so on while serving more than 2.3 million developers. With such advantages, PaddlePaddle has helped an increasing number of partners commercialize AI.
6. **CUDA Toolkit v11.3** The NVIDIA® CUDA® Toolkit provides a development environment for creating high performance GPU-accelerated applications. With the CUDA Toolkit, it is possible to develop, optimize, and deploy your applications on GPU-accelerated embedded systems, desktop workstations, enterprise data centers, cloud-based platforms and HPC supercomputers. The toolkit includes GPU-accelerated libraries, debugging and optimization tools, a C/C++ compiler, and a runtime library to deploy your application. Using built-in capabilities for distributing computations across multi-GPU configurations, scientists and researchers can develop applications that scale from single GPU workstations to cloud installations with thousands of GPUs.

- Testing

1. **PyTest v7.1** The pytest framework makes it easy to write small, readable tests, and can scale to support complex functional testing for applications and libraries.
2. **Vitest v0.20** A Vite-native unit test framework. Vite's out-of-the-box support for common web patterns, features like glob imports and SSR primitives, and its many plugins and integrations are fostering a vibrant ecosystem.Vitest cares a lot about performance and uses Worker threads to run as much as possible in parallel. Some ports have seen test running an order of magnitude faster. Watch mode is enabled by default, aligning itself with the way Vite pushes for a dev first experience. Even with all these improvements in DX, Vitest stays lightweight by carefully choosing its dependencies. Vitest aims to position itself as the Test Runner of choice for Vite projects, and as a solid alternative even for projects not using Vite.

2.2.3 Constraints

- The system requires NVIDIA TESLA v100 GPU(s) for training and testing.
- The system requires CUDA toolkit v11.3 or greater.
- The system requires uninterrupted internet for runtime.

2.2.4 Non-Functional Requirements

- Performance Requirements

1. Users should have good Internet connection for smooth working of the software.
2. The server should be backed by NVIDIA TESLA v100 GPU x 1 for drawing inferences in short time.
3. The user input should be a clear and good quality image with a clean background.

- Safety Requirements
  1. Since the application does not store any user data, there are no safety concerns.
- Security Requirements
  1. Since the application does not store any user data, there are no security concerns.

Chapter 3

Project Design

3.1 Block Diagrams

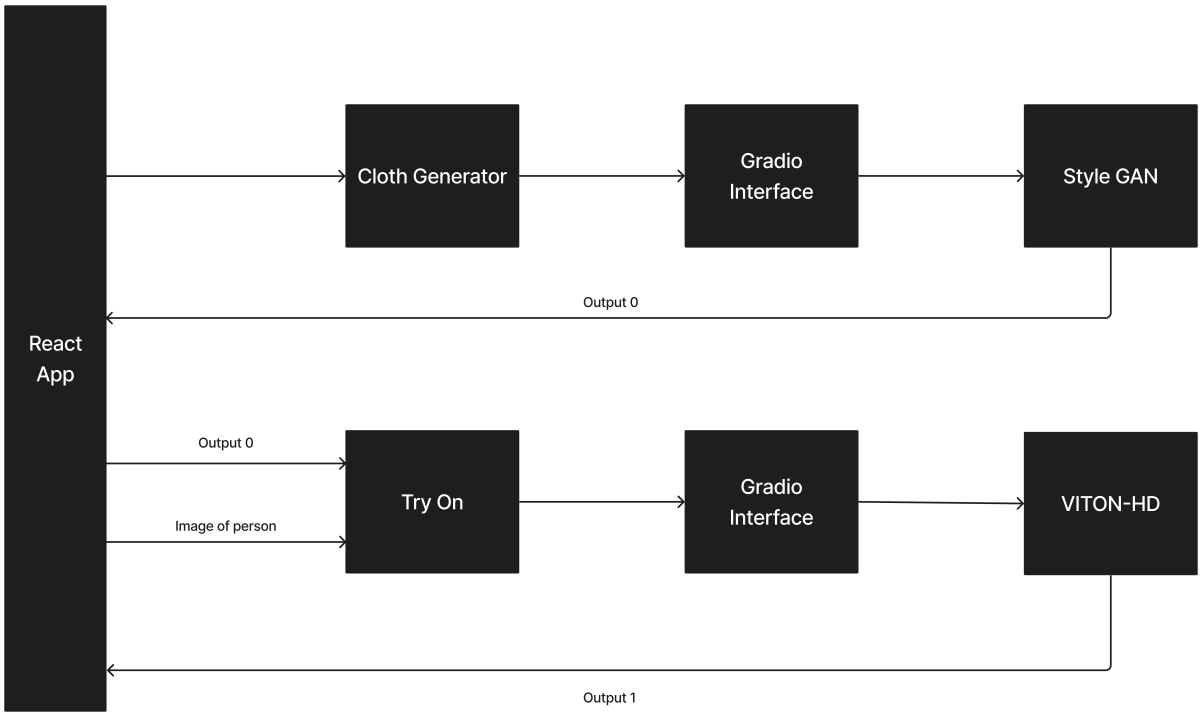

Figure 3.1: Architectural diagram of System

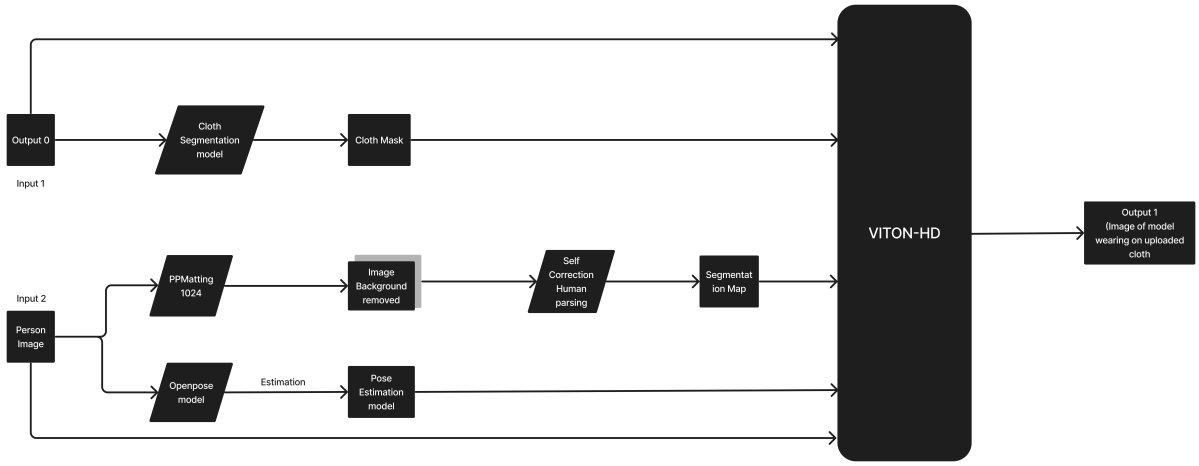

Figure 3.2: Architectural diagram of Design Generation

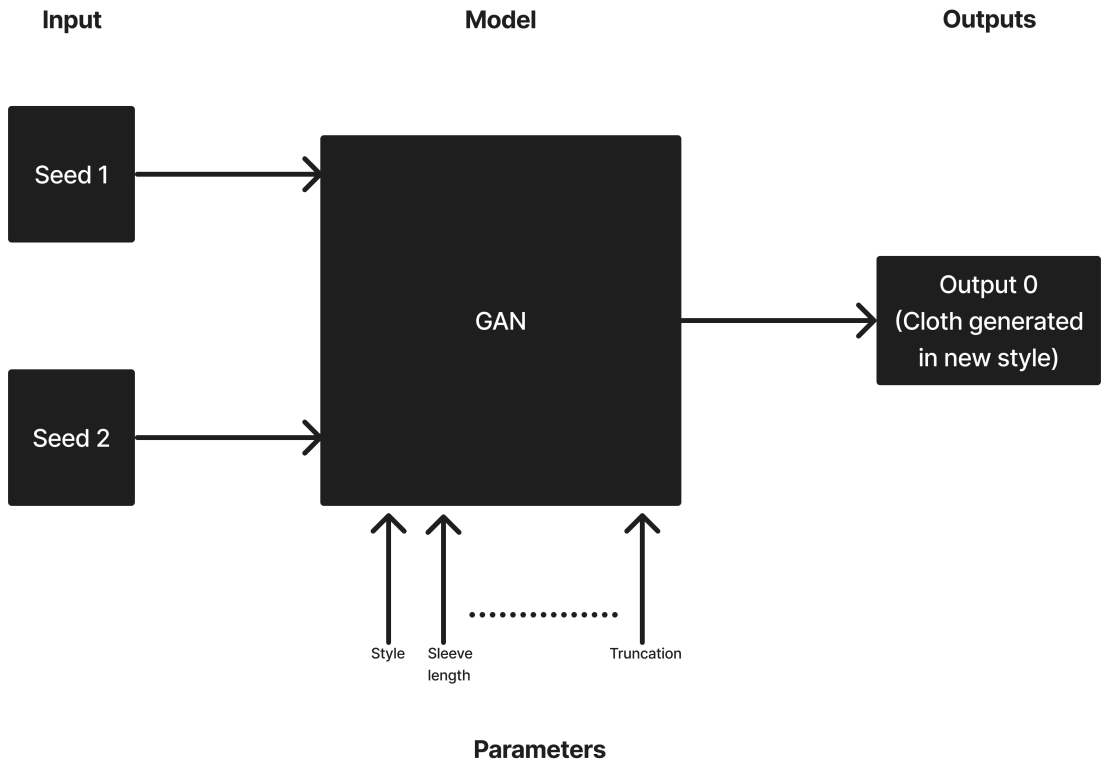

Figure 3.3: Architectural diagram for Try On

3.2 High Level Design

3.2.1 System Design

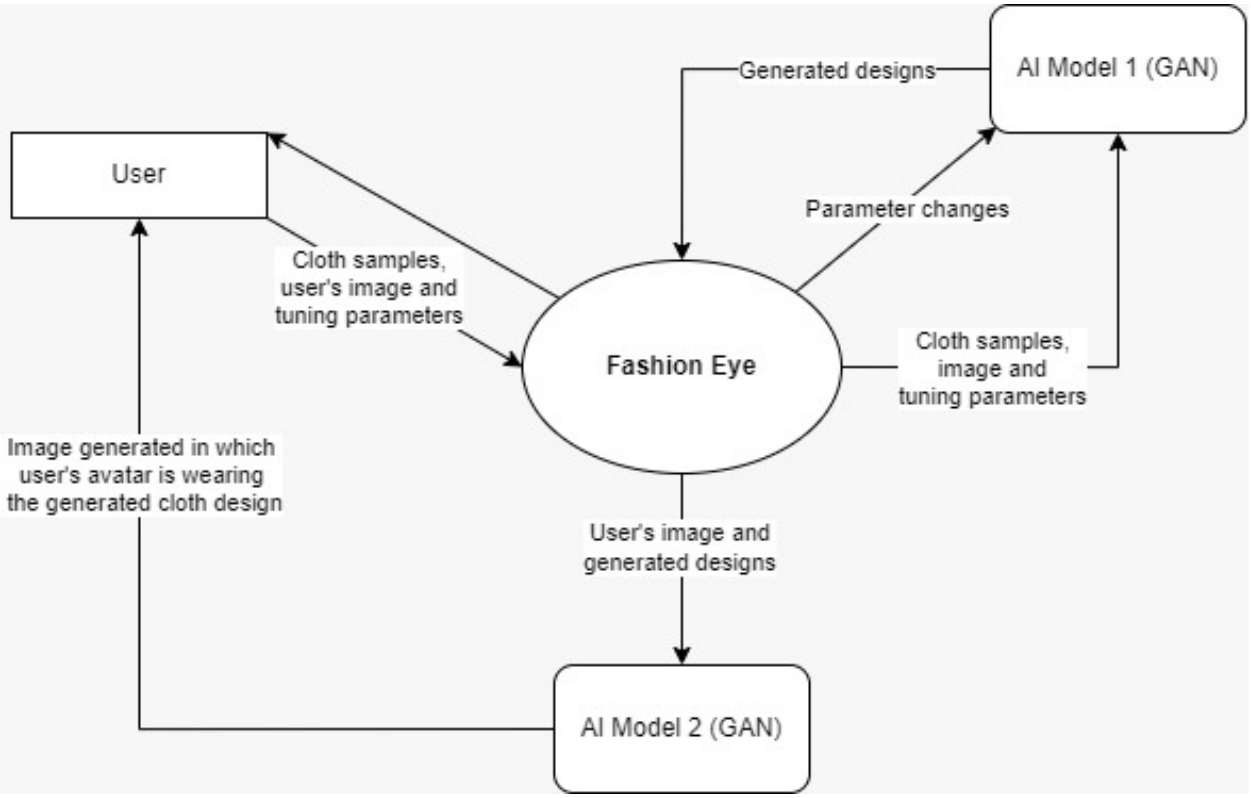

Figure 3.4: Data Flow Diagram Level 0

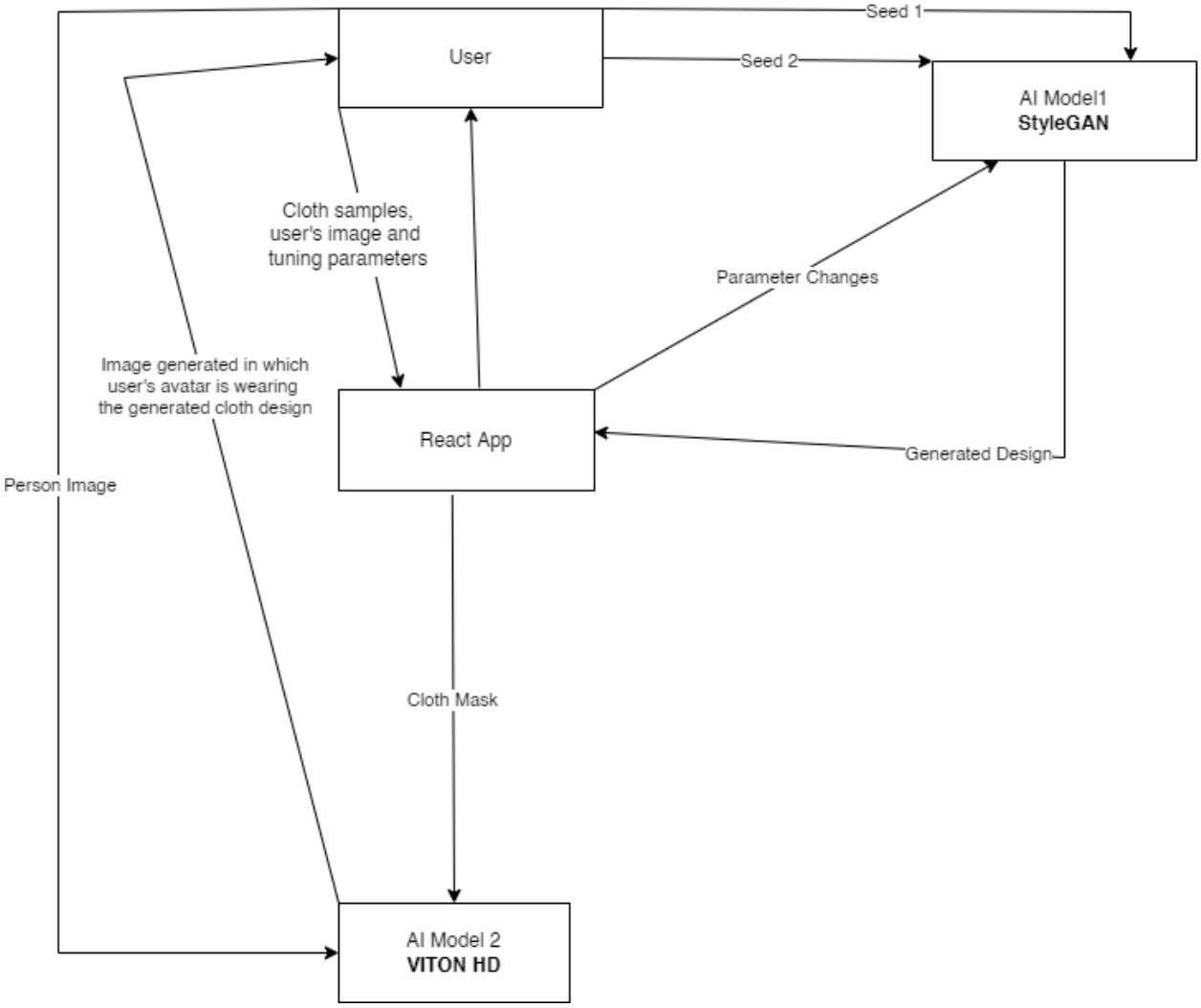

Figure 3.5: Data Flow Diagram Level 1

3.2.2 User Interface

Fashion Eye is a web-application that can be used to generate new clothing designs and let users virtually try them on. The primary function of our website is to give users a brief overview of Fashion Eye, create new clothing styles and also help users try on these styles or any other cloth of their choice as long as the cloth image has a clean background. To create various parent designs, we can alter the quantity of seeds. To create new designs, you can choose between Seed 1 and Seed 2. Additionally, there is the ability to alter the dress-jacket, female coat, graphics, dark, less cleavage, and structure, style, truncation, sleeve and size. Moreover, we get access to a preview of the generated designs from where we can download the image to our device. Users can also upload cloth images and the try-on model produce an accurate image of the wearer of the specified clothing.

3.2.3 Implementation

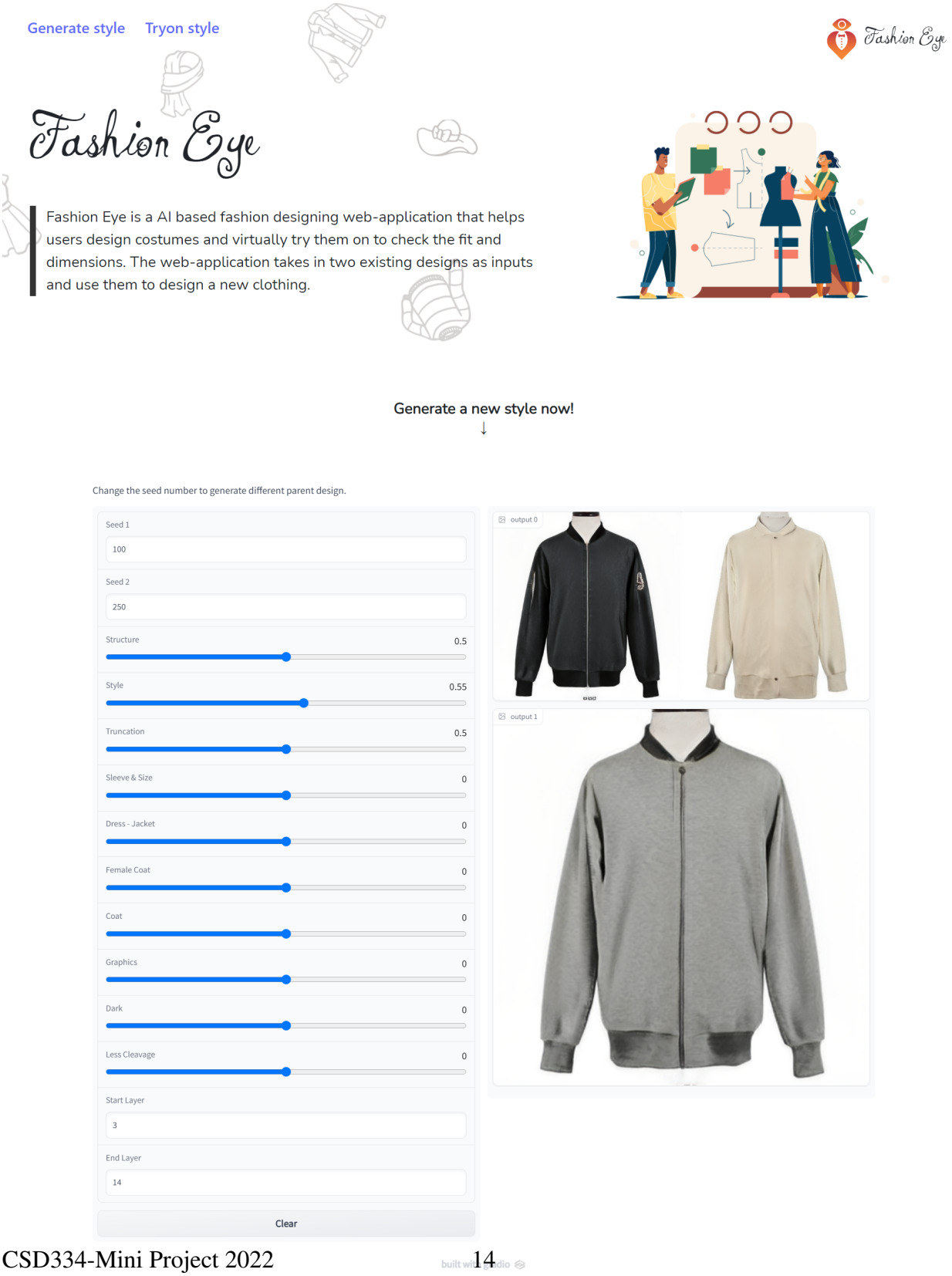

Figure 3.6: Screen Image for Cloth Generation

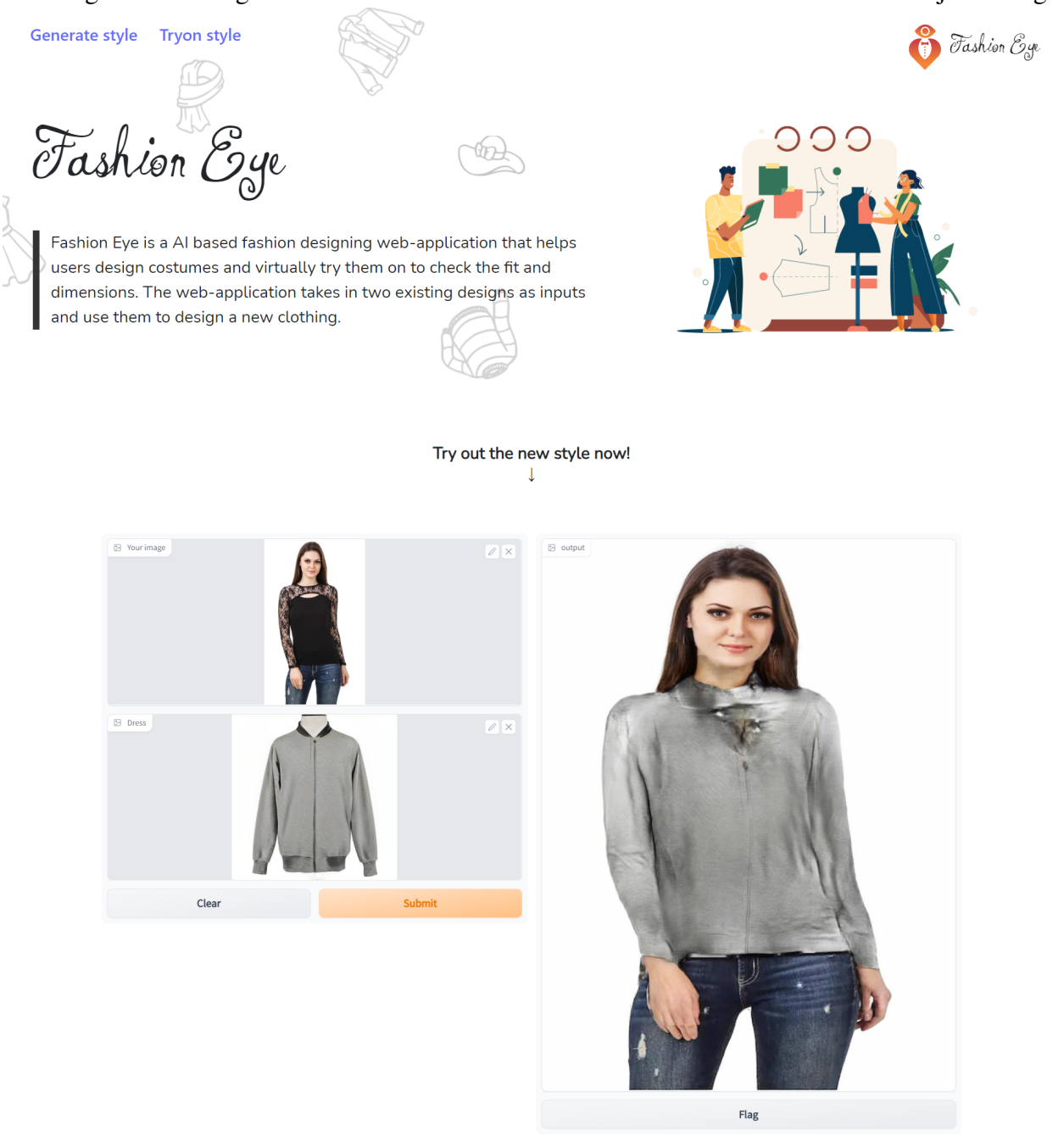

Figure 3.7: Screen Image for Try On

Figure 3.8: Responsive mobile view of the web app

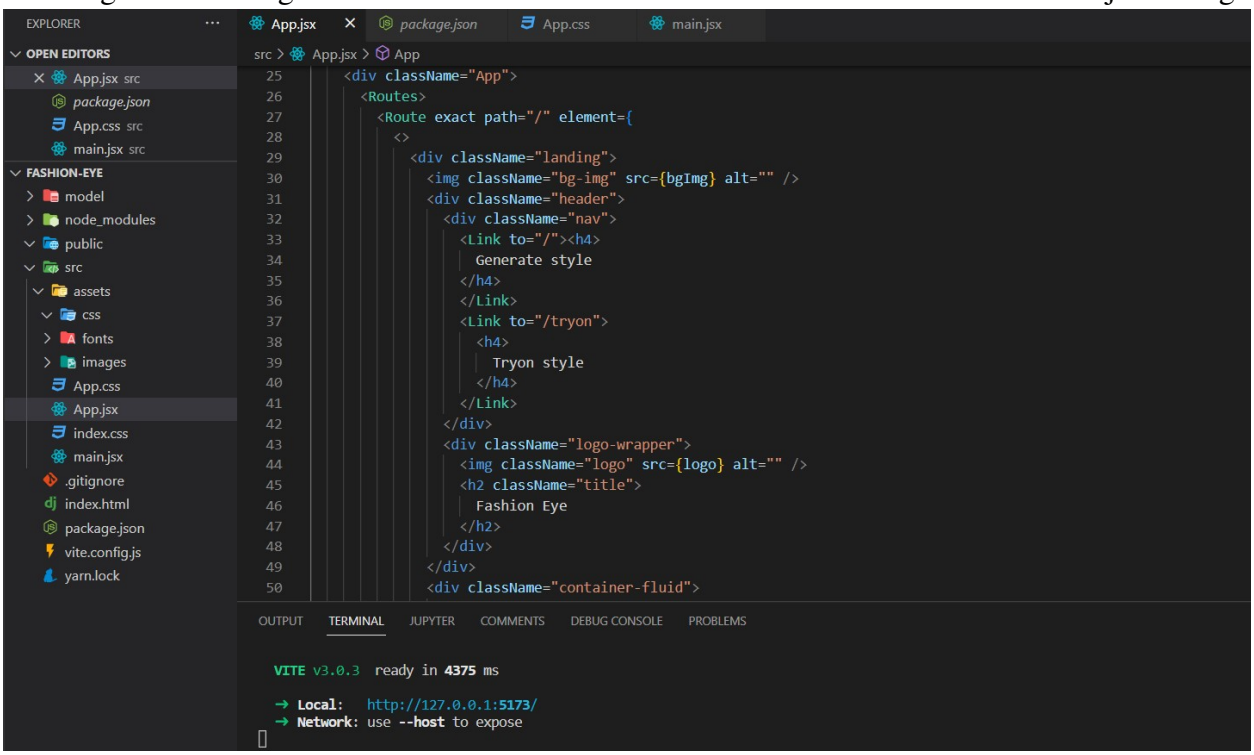

Figure 3.9: Web app running info

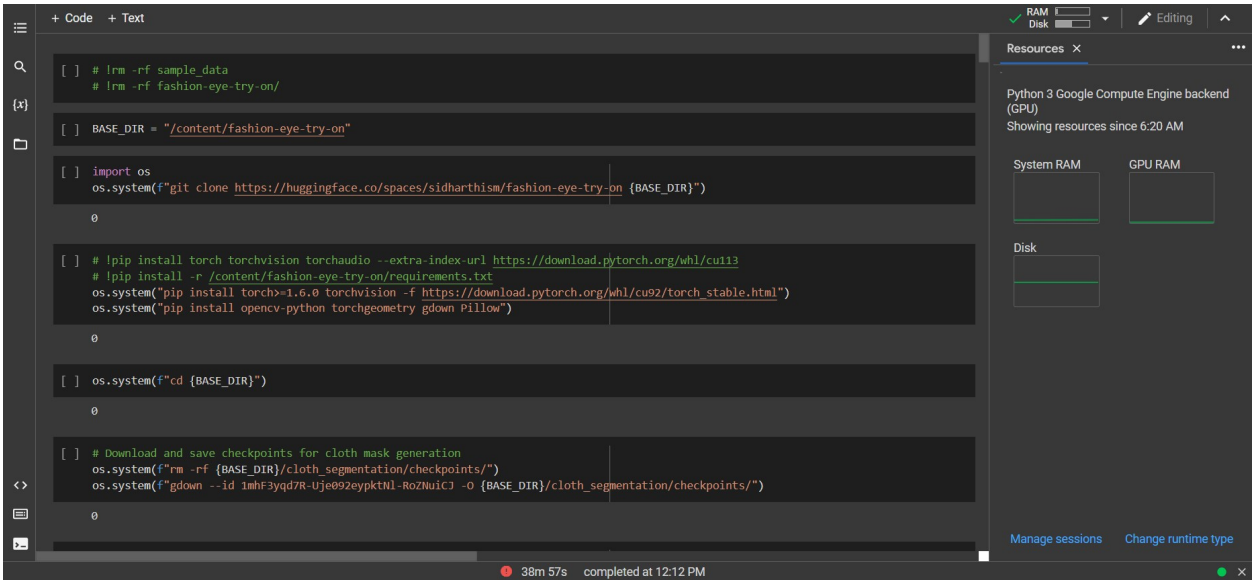

Figure 3.10: ML models running info

3.3 Algorithms

3.3.1 Cloth Generation

- The StyleGan2-ADA was trained on a subset of the Lookbook dataset containing a total of 8726 clothing images with clean background. It was transfer learned from FFHQ model and trained the model for a day.
- After Training the GAN model, the result proceedes to use GANSpace method to find important directions in the latent space. It tries to guess what these directions represent and labeled them accordingly. GANSpace is an unsupervised model and does not need an attribute classifiers.

3.3.2 Virtual Try On

- **Preprocessing**:
  1. Remove background of the person's image - It is modified using paddle segmentation: PP-Matting - 1024
  2. Predict segmentation map S using self correction human parsing
  3. Predict pose map P using the pose estimation model openpose model
  4. P and S tells us where the dress and hands of the person are. we use both to remove them from image
- **Segmentation generator**:
  1. Uses the segmentation map without arms and cloth segments, P and cloth to generate a new segment suitable for the cloth. here the model determines the parts of arm or body that will be visible new cloth is used. The segGenerator model is currently used for this purpose.
- **cloth Image deformation**:
  1. The cloth image is deformed to match the new segmented image.
  2. The geometric matching module is used for this purpose. A matching model uses both person representation and cloth image to warp the cloth using a correlation matching.
  3. Uses the synthesized segmentation, pose map P, Image with cloth and arm segment removed, warped cloth and a misalignment binary map to generate an ALIgnment - Aware Segment(ALIAS). The misalignment binary map is the difference between initial cloth map and input cloth map.
- **Try-On Synthesis**:

1. Uses the cloth image, person's image, posemap, cloth mask, segmentation map and misalignment binary map to produce a reliable representation of the model wearing the given cloth using machine learning.

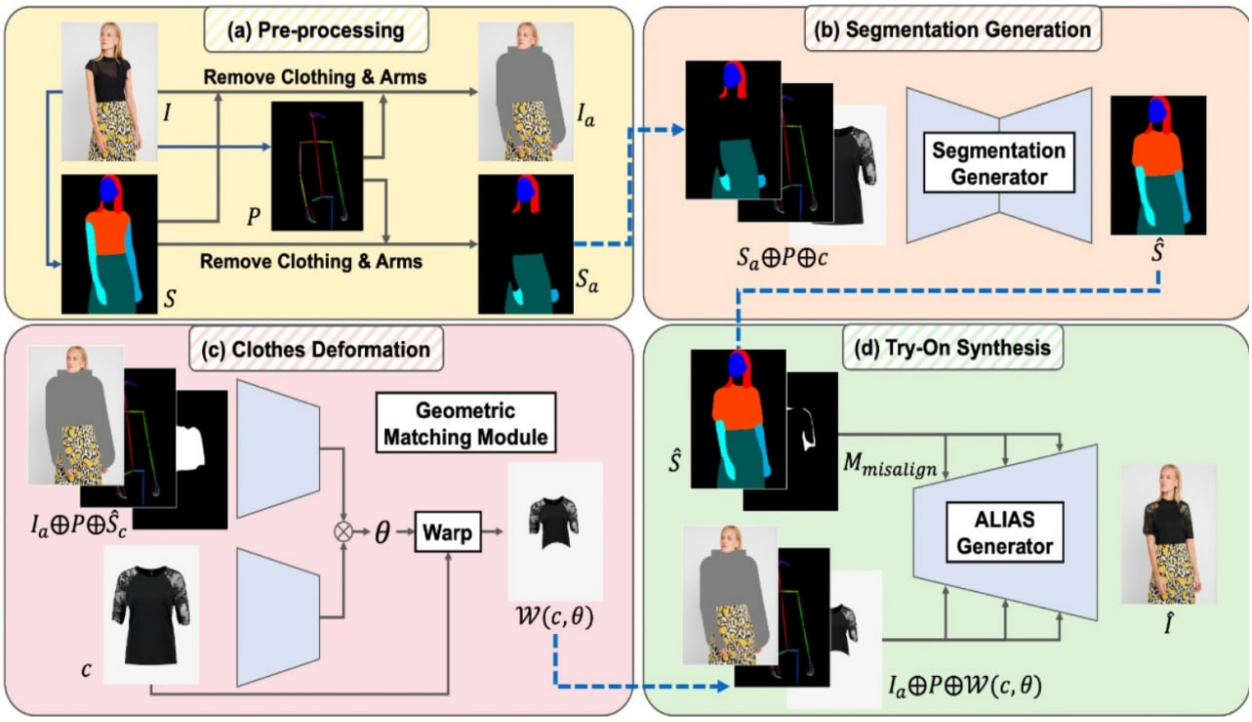

Figure 3.11: Internal Data Processing

3.4 Hardware And Software Requirements

3.4.1 Hardware Requirements

1. **Internet Connectivity**: The system requires uninterrupted internet for system runtime.
2. **Storage**: The user should have enough storage to save the output images.
3. **GPU** One or more high-end NVIDIA GPUs, NVIDIA drivers, CUDA 10.0 toolkit and cuDNN 7.5. To obtain accurate results, an NVIDIA GPU with CUDA 11.3 support and at least 16 GB of DRAM is required.

3.4.2 Software Requirements

1. **Web Browser**: Since the product is a web-application, the users must have an internet browser installed in their device. Mozilla Firefox or Chromium based browsers are recommended.

2. **Operating System** Both Linux and Windows are supported. Linux is recommended for performance and compatibility reasons.

3. **Python** 64-bit Python 3.6 installation. Anaconda3 with numpy 1.14.3 or newer is recommended.

4. **TensorFlow for training** TensorFlow 1.14 is recommended, but TensorFlow 1.15 is also supported on Linux.
   On Windows TensorFlow 1.14 should be used, as the standard 1.15 installation does not include necessary C++ headers.

Chapter 4

System Testing

The web app is to be tested by using files of different sizes. The issues of using different file formats have to be checked The image data is checked. Proper working of GUI components should be ensured. The working of intermediate stages of the ML models are tested. Memory (RAM) and GPU usage (DRAM) when the inference is running has to be checked. Performance and rendering times of the Web app under lower internet speeds have to be checked.

4.1 Test Cases and Test Results

Test Case Title	Test Condition	System Behaviour	Expected Result
Image Uploading	Person and cloth image should be uploaded.	The uploaded image was saved in .jpg format.	The uploaded image should be saved in .jpg format.
Image Size	Person and cloth image should be uploaded.	The uploaded image was converted to 768 x 1024 pixels.	The uploaded image should be converted to 768 x 1024 pixels.
Cloth mask generation	Cloth image should be uploaded.	Cloth mask was be saved to the datasets/test/cloth-mask folder.	Cloth mask should be saved to the datasets/test/cloth-mask folder.
Background removal	Person image should be uploaded.	Person's image background was removed and was saved to the datasets/test/image folder.	Person's image background should be removed and was saved to the datasets/test/image folder.
Pose estimation - Image	Background should be removed from person's image.	Pose estimation of person's image was generated and was saved to the datasets/test/openpose-img folder.	Pose estimation of person's image should be generated and saved to the datasets/test/openpose-img folder.
Pose estimation - JSON	Background should be removed from person's image.	Pose estimation JSON output of person's image was generated and was saved to the datasets/test/openpose-json folder.	Pose estimation JSON output of person's image should be generated and saved to the datasets/test/openpose-json folder.
Self Correction Human Parse	Background should be removed from person's image.	Segmentation map of person's image was generated and was saved to the datasets/test/image-parse folder.	Segmentation map of person's image should be generated and saved to the datasets/test/image-parse folder.
VITON-HD Output generation	Background removed person's image, pose estimation data, segmentation map and cloth image with cloth mask should be in the datasets/test folder.	Try on image of person's image was generated and was saved to the output folder.	Try on image of person's image should generated and saved to the output folder.

Table 4.1: Test Cases and Test Results

Chapter 5

Application Performance Testing and Analysis

The application is accessible as a web app. It is necessary to test and analyse the rendering time of the application under different network speeds. It is also required to perform resource utilization testing and analysis to improve the loading and rendering time and identify any possible memory leaks and eliminate them.

5.1 Under Different Network Speeds

5.1.1 Slow 3G Network

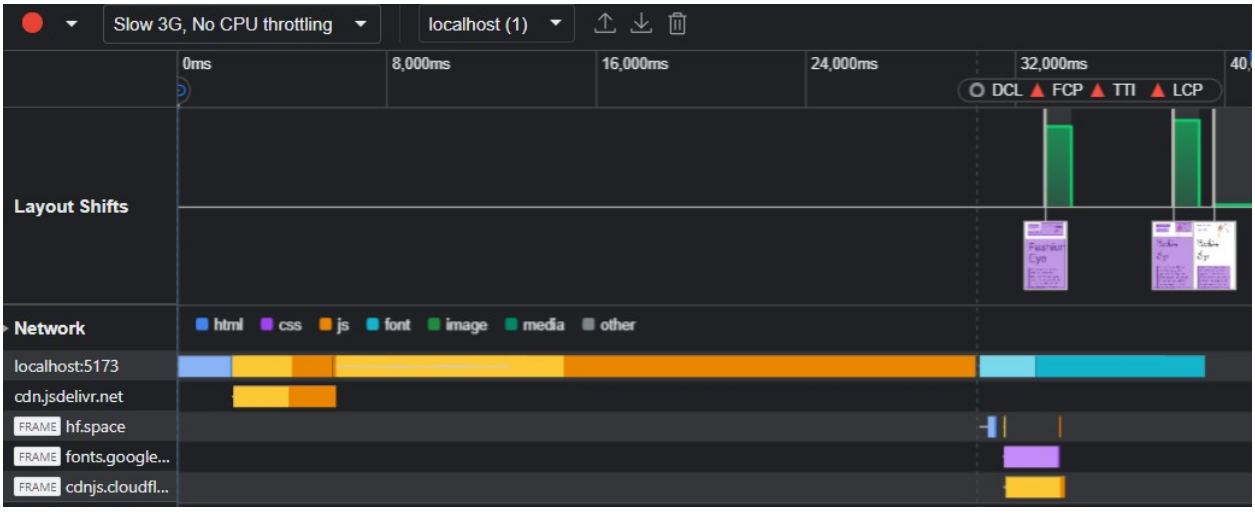

Figure 5.1: Slow 3G Network

5.1.2 Network At Normal Speeds (30 Mbps Connection)

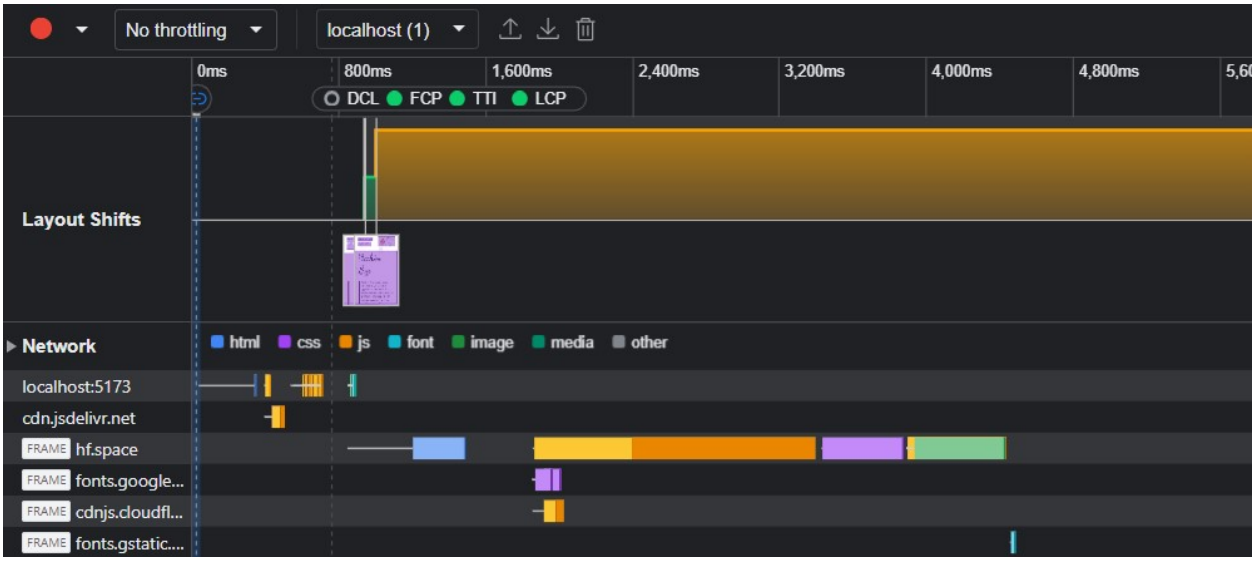

Figure 5.2: Under Normal Speeds

5.2 System Resource Utilization

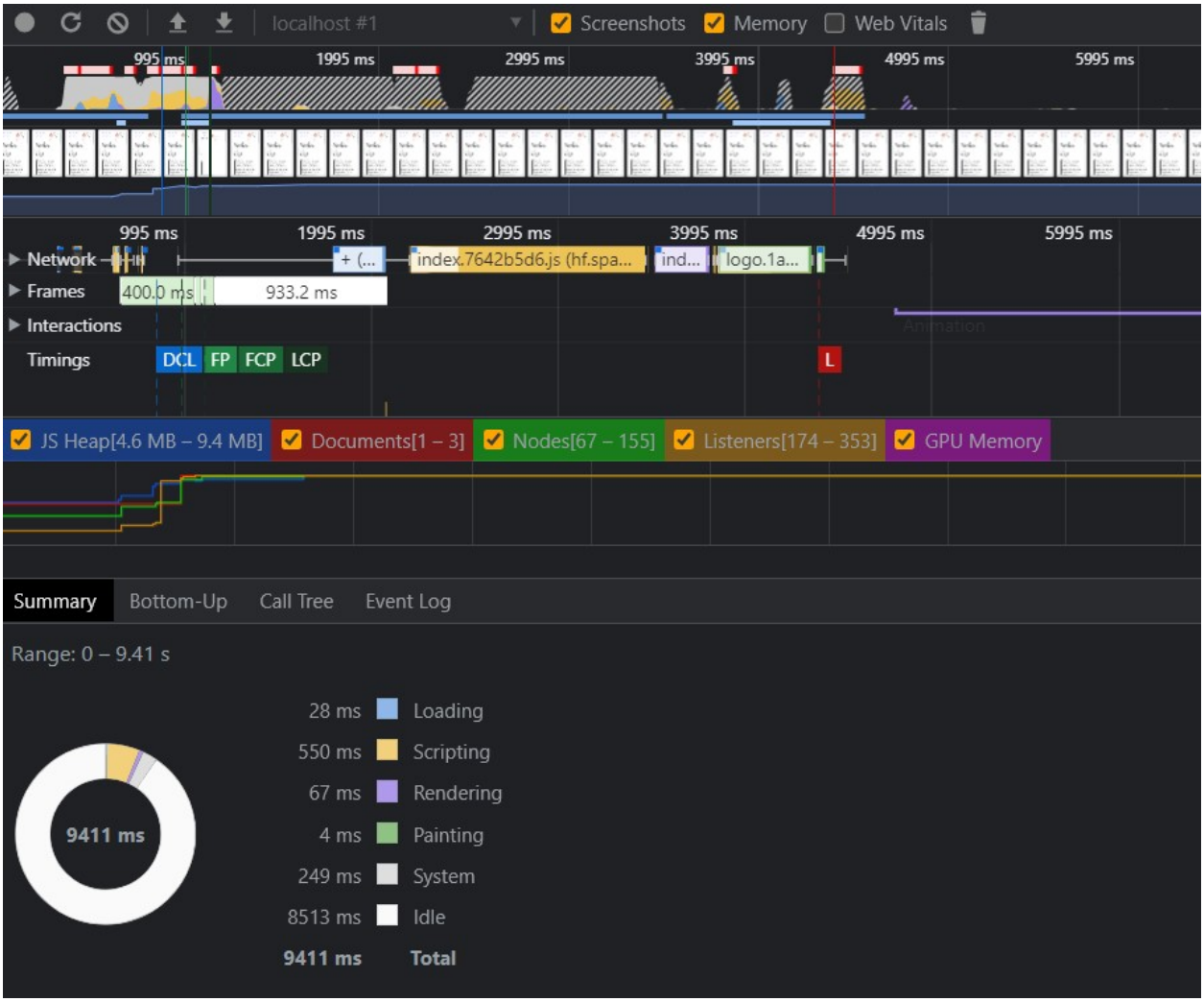

Figure 5.3: Resource Utilization

5.3 Application Performance Testing Report

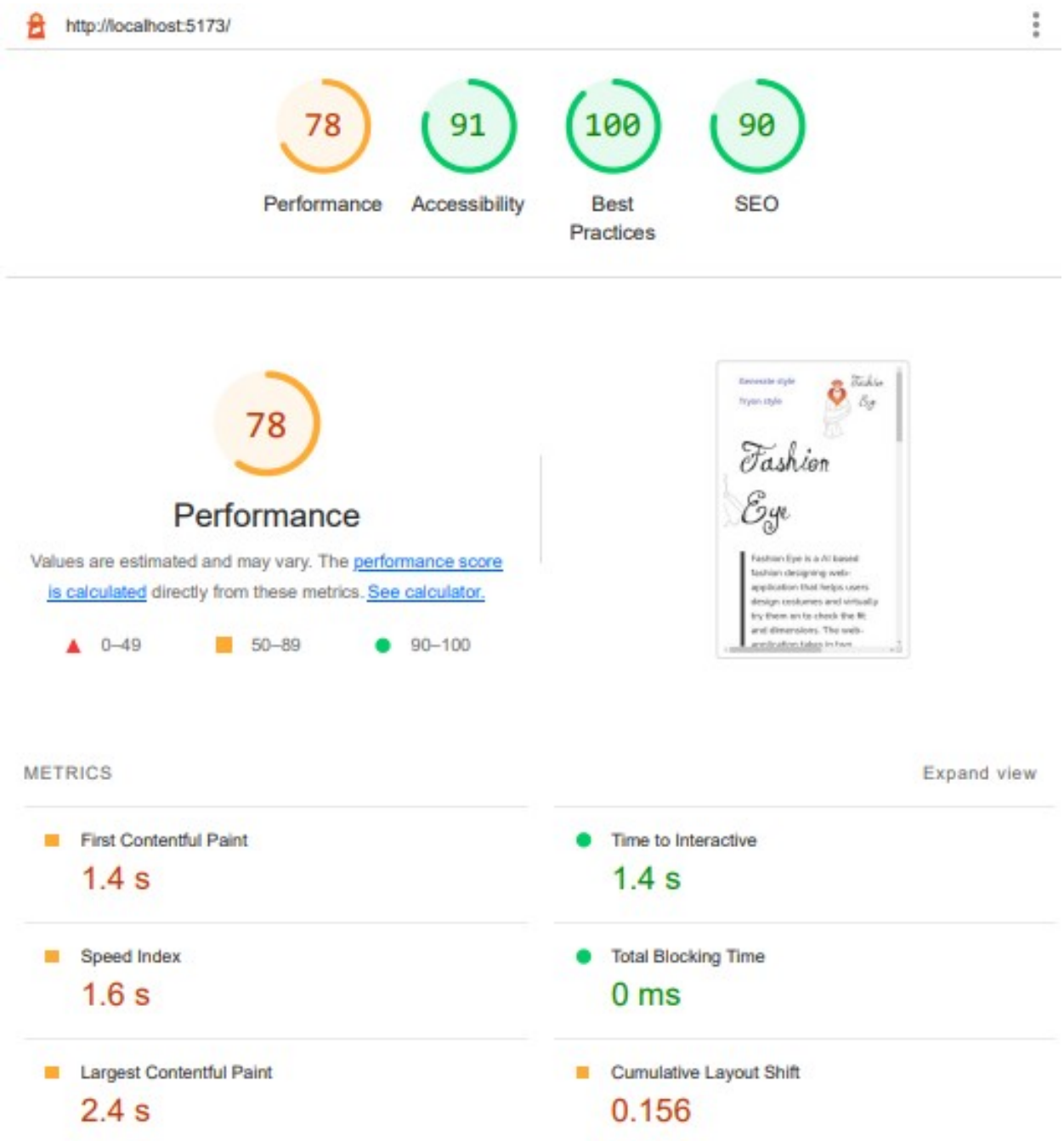

Figure 5.4: Application Performance Testing Report

5.4 Result

It is evident from the report that the inclusion of best development practices has lead to the improved performance of the web application. It is also observed that the application performance can be further improved by minimifying the JavaScript files and hosting the web application on a dedicated server.

Chapter 6

Conclusion

The proposed model allows users to generate and virtually try on clothing styles. It can inspire designers and also help users visualize themselves better wearing a dress they would like to buy. It is also a study into the ability of the neural network to be creative and imaginative in the field of fashion. AI is making great leaps with the increase in system capabilities and available data. Also the demand for AI based solutions for many human tasks is increasing. The current model was able to create a faithful representation of user wearing the given dress with an accuracy comparable to humans. With more training on better data, using more technology, and processors with higher computation power, AI can be used to generate new styles and designs.

6.1 Future Scope

The project has a great scope in the field of fashion. With further development, this model can develop better designs and better representations for virtual try on. Currently the dataset used to develop new designs contains uncategorized tops, coats, dresses etc. This dataset doesn't contain bottoms. Also the current model was trained using unsupervised learning. Users are currenly not able to upload inputs as seeds for the model. With further training and larger and categorized datasets, the model will be able to generate new styles at par with designers around the world. In the virtual try on model, better processing power can enable dynamically taking user images through the webcam. In the future, we aim to provide in-app storage. This can allow users to track and store their preferences and history.

Chapter 7

Appendix A: Glossary

Machine Learning: Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so.Machine learning is important because it gives enterprises a view of trends in customer behavior and business operational patterns, as well as supports the development of new products. Many of today's leading companies, such as Facebook, Google and Uber, make machine learning a central part of their operations.

Google Collab: Collab is a data analysis and machine learning tool that allows you to combine executable Python code and rich text along with charts, images, HTML, LaTeX and more into a single document stored in Google Drive. Google Collab have a prominent role in the field of machine learning and deep learning. Google Collab provides 12 hours of free access to K-80 GPU power.The best part of that is, there is no need to install any ML or DL library. Without installing we can use very easily.

Hugging Face: Hugging Face is a community and data science platform that provides 1) Tools that enable users to build, train and deploy ML models based on open source (OS) code and technologies. 2) A place where a broad community of data scientists, researchers, and ML engineers can come together and share ideas, get support and contribute to open source projects. It's a central place where anyone can share and explore models and datasets.

User Interface: A user interface is what you have to learn to operate a machine. For examples,the graphical user interfaces (GUIs) – windows, icons, and pop-up menus have become standard on personal computers.The goal of effective UI is to make the user's experience easy and intuitive, requiring minimum effort on the user's part to receive the maximum desired outcome.

Generative Adversarial Network (GAN): Generative modeling is an unsupervised learning task in machine learning that involves automatically discovering and learning the regularities or patterns in input data in such a way that the model can be used to generate or output new examples that plausibly could have been drawn from the original dataset. GANs are a clever way of training a generative model by framing the problem as a supervised learning problem with two sub-models:

the generator model that we train to generate new examples, and the discriminator model that tries to classify examples as either real (from the domain) or fake (generated).

VITON-HD: VITON-HD dataset is a dataset for high-resolution (i.e., 1024x768) virtual try-on of clothing items. Specifically, it consists of 13,679 frontal-view woman and top clothing image pairs. VITON-HD highly surpasses the baselines in terms of synthesized image quality both qualitatively and quantitatively.These methods follow two processes in common: (1) warping the clothing image initially to fit the human body; (2) fusing the warped clothing image and the image of the person that includes pixel-level refinement. Also, several recent methods add a module that generates segmentation maps and determine the person's layout from the final image in advance.

Chapter 8

Appendix B: Acronyms

ML:- Machine Learning

GAN :- Generative Adversarial Network

CUDA :- Compute Unified Device Architecture

GPU :- Graphics Processing Unit

FFHQ :- Flickr-Faces-HQ

ALIAS :- ALIgnment - Aware Segment

RAM :- Random Access Memory

DRAM :- Dynamic random-access memory

References

[1] Montek Singh, *Generation of fashionable clothes using generative adversarial networks: A preliminary feasibility study*, International Journal of Clothing Science and Technology 2020

[2] Karras, Tero and Laine, Samuli and Aila, Timo, *A style-Based Generator Architecture for Generative Adversarial Networks*, arXiv 2019

[3] Yuying Ge, Ruimao Zhang, Wu Lingyun, Xiaogang Wang, *DeepFashion2: A Versatile Benchmark for Detection, Pose Estimation, Segmentation and Re-Identification of Clothing Images*, arxiv 2019

[4] Yoojin Jeong and Chae-Bong Sohn, *VITON-GAN: Virtual Try-on Image Generator Trained with Adversarial Loss*, Eurographics 2019