# FIND YOUR BEST OUTFIT

# AGENDA

# I. Introduction

## Problem Overview

Although the global online fashion sector shows strong growth prospects, offline fashion shopping still holds a significant share of 64%. This opens up an opportunity as well as a challenge for online shopping development.

Reviewing the advantages of shopping offline, we find key points for its solitary position in the market: Tangible Experience & Personal Assistance.

We thus define our mission is to blur the customers' experience boundary between offline shopping and online shopping.

# I. Introduction

## Solution Overview

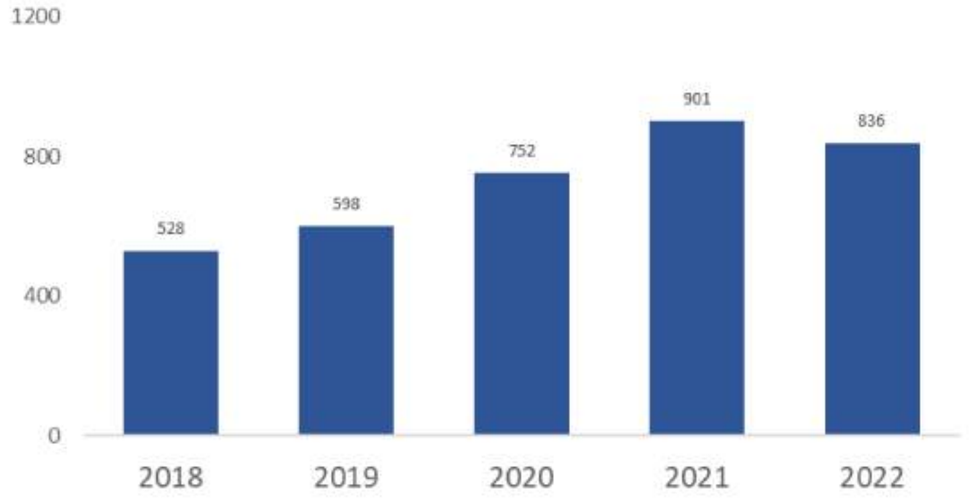
A multifunctional system that blurs the difference between online and offline shopping.

Rating System

Sematic
Recommendation
Search

Virtual Fitting

# II. Problem Statement




Fashion Ecommerce Revenue ($billion)

**64%**

of global fashion shopping still takes place offline

The global online fashion sector accounting for around a fifth of the world's ecommerce in 2022. Despite being down on 2021's peak of $901bn, the sector is significantly ahead of its pre-pandemic level of $590bn, showing strong evidence for growing trend.

Moreover, up to 64% of global fashion shopping still takes place offline, which is a very low penetration rate, implies huge room for development in this sector.

**Seizing the share of offline market will bring exponetial growth**

Source: RetailX analysis

# II. Problem Statement

## Online fashion shopping is a great choice for...

- Convenience: allows customers to purchase from the comfort of their own homes or anywhere they have internet access at any time, instead of visiting physical stores during specific operating hours.
- Wide Selection: Online stores often have a larger inventory compared to physical stores, offering customers a wider range of fashion options to choose from .

## ...but offfline fashion shopping has:

- **Tangible Experience:**

Physical stores allow customers to see and try on fashion items before making a purchase. This tactile experience can enhance customer satisfaction because they can determine whether this item is suitable for their body or their overall outfit.

- **Personal Assistance:**

In physical stores, customers can interact with knowledgeable sales staff who can provide guidance, recommendations to their outfit in real-time.

# II. Problem Statement

## PROBLEM

Majority of fashion shopping still take place offline due to its superiority in providing customers with: Tangible Experience, Personal Assistance.

## OUR MISSION

BLUR THE CUSTOMERS' EXPERIENCE BOUNDARY BETWEEN SHOPPING OFFLINE AND ONLINE

# II. Problem Statement

## OUR SOLUTION

An multifunctional system that named: “Find Your Best Outfit” which include 3 main features:

1. Sematic Recommendation Search
2. Virtual Fitting
3. Rating System

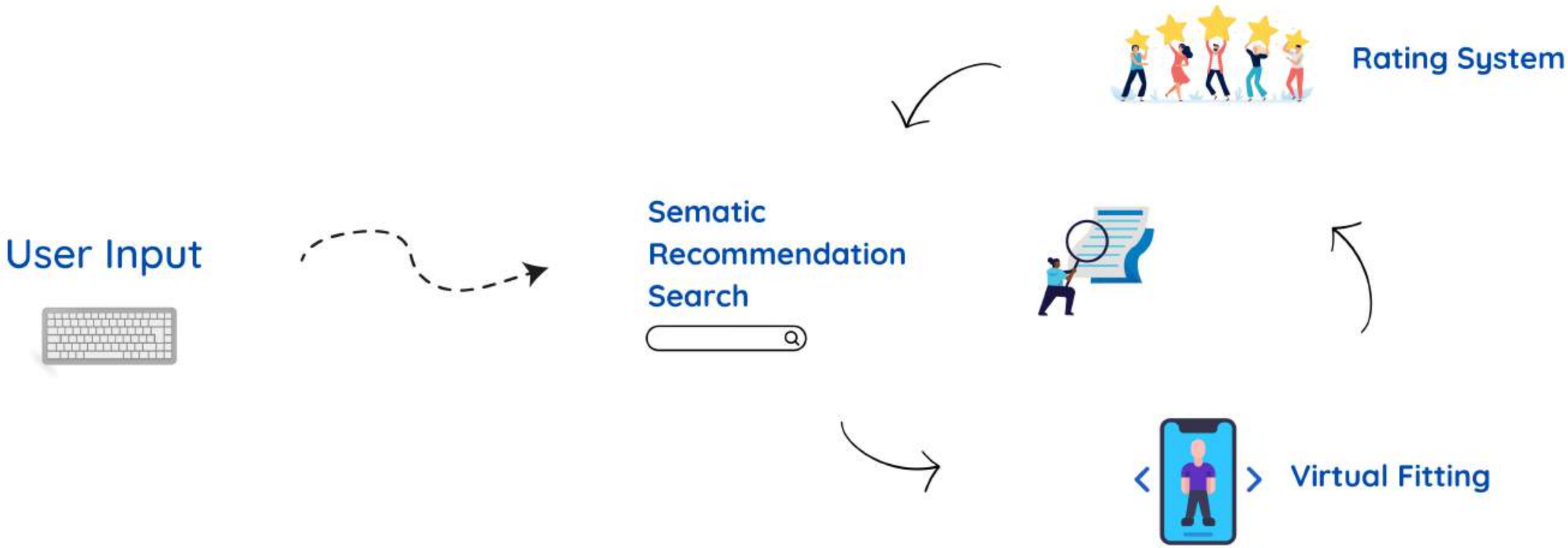
# III. Solution Overview

# III.Solution Overview

We found that **Dataset 1** has a vast array of Nike and Adidas products that can significantly expand the available options within the application. We are confident that Dataset 1 will be a valuable resource in creating a successful and user-friendly virtual fitting room for Nike and Adidas garments.

# III.Solution Overview

## Pipeline



# III.Solution Overview 1

Sematic Recommendation Search

## Semantic Recommendation Search

Helps users find exactly the outfits they want with just a few descriptive prompts

- Implement deep learning and LLM (Large Language Model) algorithms to provide outfit recommendations based on the input of users

- Offer options for users to receive enhanced suggestions for higher-suitable outfits

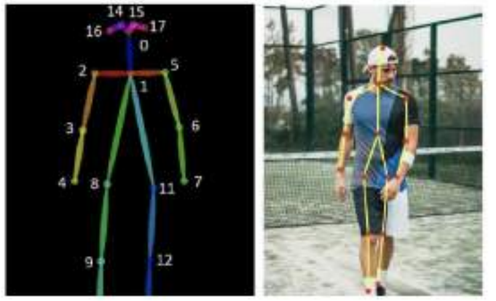
# III.Solution Overview 2

Virtual Fitting

## Image Processing and Virtual Try-On

helps users try on their chosen outfit just by uploading photos to the virtual fitting room

- Utilize image processing implied in model to recognize and analyze factors such as body shape, garments, and distractors.



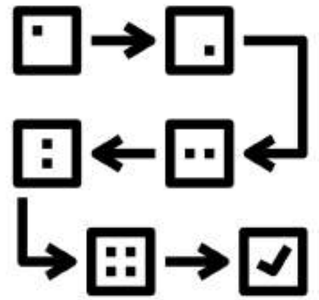

- Integrate a virtual try-on model for users to select and try on diverse clothing items

# III.Solution Overview 2

Virtual Fitting

## Customized Outfit Sequencing

This is the main function, bringing a breakthrough in being able to optionally overlap layers together to coordinate outfits



- Enable users to arrange the sequence of garments within their outfit

- Allow drag-and-drop functionality or numerical input for users to prioritize the order of clothing pieces

# III.Solution Overview 2

Virtual Fitting

## More about: Customized Outfit Sequencing

Enable users to arrange the sequence of garments within their outfit.

In this example, this girl has the flexibility to select different garments to determine the best fit for her outfit, which is more complexity to do in real-life.

# III.Solution Overview 3

Rating System

## Rating System

A system that helps improve the user's outfit through calculation and giving scores

- Develop an intelligent rating system capable of evaluating user images to assess the suitability of the chosen outfit.

- Integrate a rating system based on factors such as color coordination, style, and fashion trends

# III.Solution Overview 3

## Rating System

## More about Rating System: Real-Time Weather Integration

- Integrate an API for real-time weather data based on the user's location or specified preferences.

- Utilize weather information like temperature, humidity, and season at living place to refine outfit recommendations

# IV. Methodologies

1. Semantic Recommendation Search
2. Virtual Fitting Room called DiOr
3. Rate your Outfit

# IV. Methodologies

## 1. Semantic Recommendation Search - Overview

How to upgrade your recommendation

Traditional search engines which only find documents based on lexical matches.

Semantic search is to embed all entries in data, into a vector space. At search time, the query is embedded into the same vector space and the closest embeddings from your corpus are found. These entries should have a high semantic overlap with the query.

Traditional Search Engine

Semantic Search

# IV. Methodologies

## 1. Semantic Recommendation Search - Semantic Search Ranking

### How we build it

all-mpnet-base-v2

**Using Hugging Face Sentence Transformer** + **Finetuned Pretrained Model**

- SentenceTransformer🤗is a Python framework for state-of-the-art sentence, text and image embeddings.

- This model used the pretrained microsoft/mpnet-base model and fine-tuned on a 1B sentence pairs dataset.
- Model will map sentences & paragraphs to a 768 dimensional dense vector space and can be used for tasks like clustering or semantic search

# IV. Methodologies

## 1. Semantic Recommendation Search - Phase 1

- User will input the description of the product they want
- For example: "I want a comfortable T-shirt, which is formal but also easy to wear"
- This input will be embeded using all-mpnet-base-v2 model

all-mpnet-base-v2 embedding →

768 dimension dense vector

**User Input search**

- Using all-mpnet-base-v2 model to embed each product information (the description columns in dataset)
- Each product will now have a 768 dimension dense vector

all-mpnet-base-v2 embedding →

768 dimension dense vector

768 dimension dense vector

768 dimension dense vector

...

768 dimension dense vector

Using Dataset 1: Adidas and Nike Product

# IV. Methodologies

## 1. Semantic Recommendation Search - Phase 2

- Now we have text embedding vector of user input and embedding vector of each product information
- We will find which product information vector is closest to our user input
- We will do that simply using cosine similarity. Most suitable product will be closer together in the vector space, and queries that differ most will be farther apart.

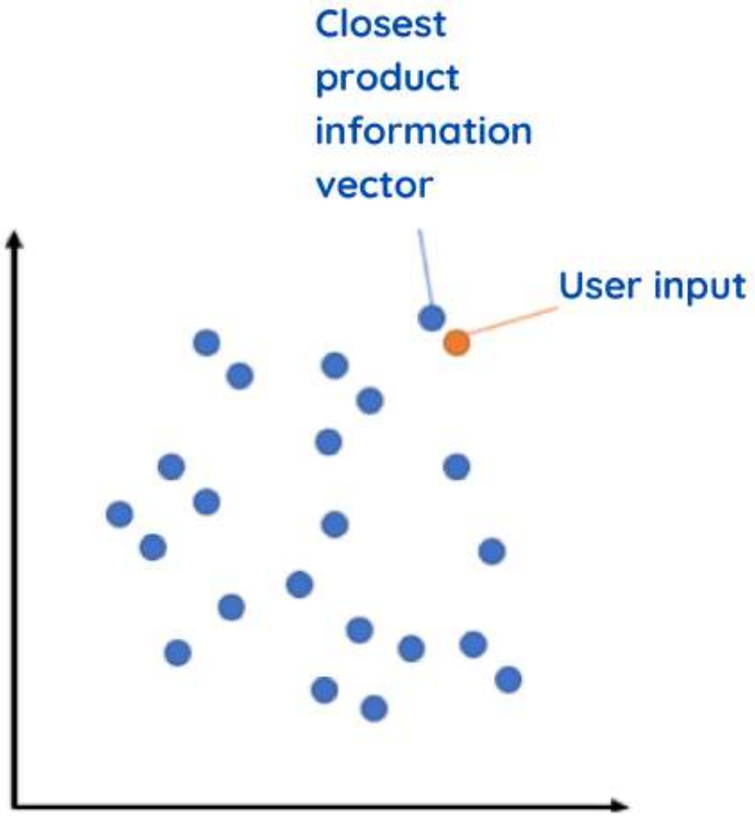
768 dimension dense vector of user

768 dimension dense vector of product

cosine similarity

a score tell how similar of 2 vector

The results of the cosine similarity function will detect which of the texts in the database are closest to our query in vector space.




- Representation of embeddings in a 2D vector space. Obtained from the Sentence Transformers documentation.

# IV. Methodologies

## 1. Semantic Recommendation Search - Further Improvement

**SPEED**

- Using model to embedding the all the information of all product will take a lot of time. So we will precompute the embedding of information of each product and store in the database. user will only need to wait for the model to embedding their input sentence
- Not compute score similarity for all data just a list of product information (say 100 or 1000 products) and output the best fit in the list. If user dont like the output product, we will contine do this with another different 1000 another product. Iterate until find what user need, This will save us quite a lot of time. (The number of product in the list need more experiment for the best trade - off

**DIVERSITY**

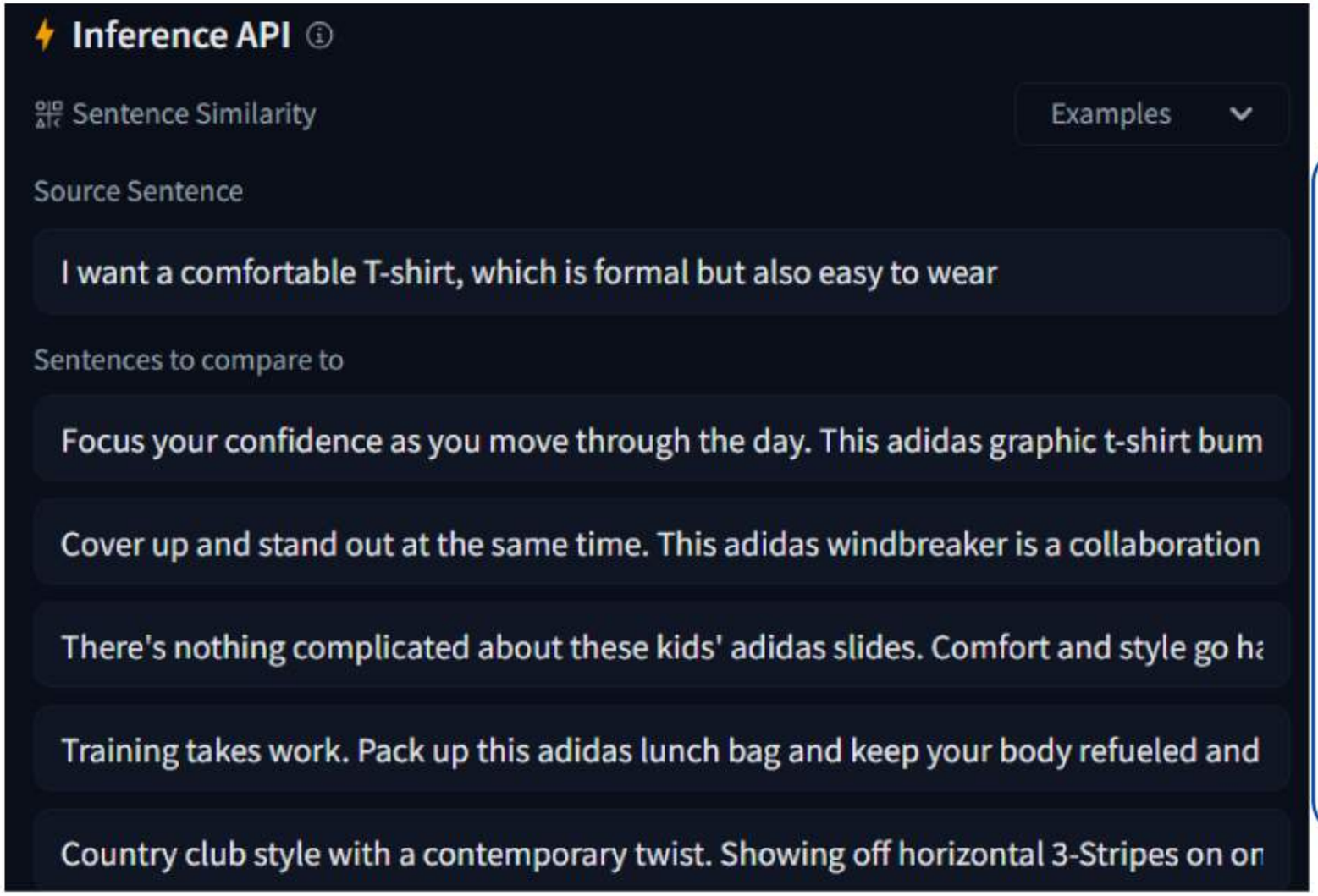
- The best fit might not be the best fit. So we will not only recommend the product with the closest similar with user input, we will recommend top 3 - 5 product, this will give user more choices and more freedom to try out clothes

**ACCURACY**

- More information of a product will result in better accuracy. So we not only will use the “description” column in the “nike and adidas products” dataset, we will concatenate all the other valuable column such as : brand. color. currency, price, availability, avg_rating, review_count. Use this as a general product information and send it to model embedding

# IV. Methodologies

## 1. Semantic Recommendation Search - Demo




Our team demo for semantic search - Part 1

- This demo using Hugging Face Inference API - Sentence Similarity, avalable in this link : https://huggingface.co/sentence-transformers/all-mpnet-base-v2
- The input consist of :
  - source sentence (this will be ) our user input
  - sentences to compare to: this is the product description of the first 5 product in the “nike and adidas products” dataset

# IV. Methodologies

## 1. Semantic Recommendation Search - Demo

| | |
|---|---|
| Focus your confidence as you move through the day. This adidas graphic t-shirt bumps up the style of any look with a foil-print mandala graphic. When it goes on, you'll look as good as you feel in the soft cotton jersey fabric.\r\n\r\nBy buying cotton products from us, you're supporting more sustainable cotton farming. | 0.524 |
| Cover up and stand out at the same time. This adidas windbreaker is a collaboration with FARM Rio and features a wildflower-inspired allover print that's just as vibrant as you are. Tighten up the hems for added protection if the wind really picks up.\r\n\r\nThis product is made with Primegreen, a series of high-performance recycled materials. | 0.381 |
| There's nothing complicated about these kids' adidas slides. Comfort and style go hand-in-hand in simple slip-on sandals that are perfect for a day at the pool, lounging at home or whatever other low-key plans your kid comes up with. | 0.035 |
| Training takes work. Pack up this adidas lunch bag and keep your body refueled and recharged. Insulated to keep your lunch cool, it's got a roomy main pocket and a front pocket for easy access to snacks. A clip on the top lets you attach it to another bag so you never leave it behind. \r\n\r\nThis product is made with recycled content as part of our ambition to end plastic waste. | 0.133 |
| Country club style with a contemporary twist. Showing off horizontal 3-Stripes on one sleeve, this adidas Club tennis tee is built for comfort. | 0.352 |

**Our team demo for semantic search - Part 2**

- With the user input that they want a comfortable t-shirt but also formal enough. The first product give the best similairty and as you can tell, this is in reality a good result fit the user need.
- For the ease of demo and the limit of a presentation, i just show the data limit with 5 products, but the result is quite good. With more products (say 1000) we can give user the exact match that they want

# IV. Methodologies

## 1. Semantic Recommendation Search - Demo code 1

Read Dataset 1 of VietNam Datathon

```
nike_adidas = pd.read_csv ("/kaggle/input/nike-adidata-dathon-data/adidas_nikes_products_snaphost_
nike_adidas.head(3)
```

| | name | description | brand | color | currency | price | availability | |
|---|---|---|---|---|---|---|---|---|
| 0 | Foil Mandala Graphic Tee | Focus your confidence as you move through the ... | adidas | White | USD | 20.0 | InStock | https://assets.adidas.com/images/w_600,f |
| 1 | FARM Rio Print Relaxed Lightweight Windbreaker | Cover up and stand out at the same time. This ... | adidas | Blue | USD | 52.0 | InStock | https://assets.adidas.com/images/w_600,f |
| 2 | Adilette Comfort Slides | There's nothing complicated | adidas | Black | USD | 24.0 | InStock | https://assets.adidas.com/images/w_600,f |

Convert to hugging face dataset

```
ds = Dataset.from_pandas(nike_adidas)
```

Call pretrained model using sentence transformer framework from hugging face

```
model = SentenceTransformer('sentence-transformers/all-mpnet-base-v2')
```

# IV. Methodologies

## 1. Semantic Recommendation Search - Demo code 2

Use pretrained model to embedding user input

```
["I want a comfortable T-shirt, which is formal but also easy to wear"])
```

Use pretrained model to embedding description of all product in data - this phase take only 3 seconds using GPU P100 from Kaggle

```
description_list = []
from tqdm.auto import tqdm
for item in tqdm ( ds) :
    description = item['description']
    description_list.append ( model.encode (description) )
```

Use cosine similarity to score the difference between the user input and description - this phase take only 24,6 ms using GPU P100 from Kaggle

```
score = []
for description in description_list :
    score.append ( util.pytorch_cos_sim(description, uer_input )[0][0].item() )
```

Now you just need to find the best score, this code will select the best score and retrieve the product data best score

```
ds [int (np.argmax ( score ) ) ]
```

# IV. Methodologies

## 1. Semantic Recommendation Search - Demo code result

User input

I want a comfortable T-shirt, which is formal but also easy to wear

Our recommendation output

### A CROPPED T-SHIRT WITH VERSATILE STYLE.

Whether it's a morning yoga flow, an afternoon run through the park or an evening of hanging with friends, the important thing is to do what makes you happy. This tee will match your energy every step of the way. Feel cool and confident with just the right amount of coverage. Pair it with tights to dial up the sports-inspired look or with your favorite denim for an everyday style that's on point for any occasion.

This information is retrieved using url column in dataset

# IV. Methodologies

## 1. Semantic Recommendation Search - Search upgrade

**Limitation**

Model can make a good representaion of product description but not true for another product feature like : brand, color, currency, price, availability, average - rating, the traditional search still do better on this feature

**Improve**

Give user the traditional search so they can filter the product base on product features like: brand, color, currency, price, avalability, average - rating

After that, user can input what the product they want

The most suitable product will be shown

# IV. Methodologies

## 2. Virtual Fitting Room

**Model**

Dressing in Order (DiOr): Recurrent Person Image Generation for Pose Transfer, Virtual Try-on and Outfit Editing

**Overview**

DiOr introduces a novel recurrent generation pipeline that sequentially dresses a person, allowing different looks to be achieved by trying on the same garments in different orders.

**Innovation**

The key innovation of DiOr lies in its ability to produce dressing effects that surpass existing methods. It enables different interactions of garments, such as wearing a top tucked into the bottom or over it, and supports layering of multiple garments of the same type, like a jacket over a shirt over a t-shirt.

# IV. Methodologies

## 2. Virtual Fitting Room

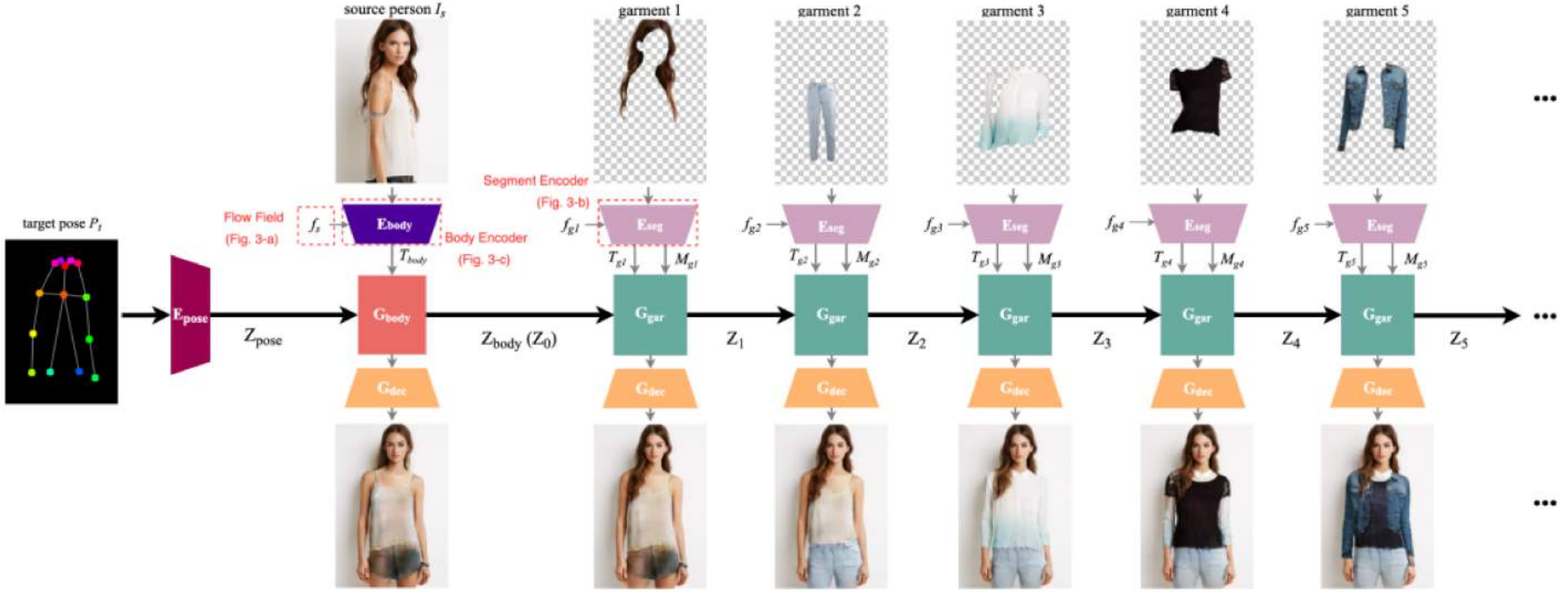
### Why DiOr ?

- In real life, people put on garments one by one, and can layer them in different ways (e.g., shirt tucked into pants, or worn on the outside).
- However, existing try-on methods start by producing a mutually exclusive garment segmentation map and then generate the whole outfit in a single step. This can only achieve one look for a given set of garments, and the interaction of garments is determined by the model.
- By contrast, our system incorporates a novel recurrent generation module to produce different looks depending on the order of putting on garments. This is why we call our system DiOr, for Dressing in Order.


Some examples applications supported by our DiOr system

# IV. Methodologies

## 2. Virtual Fitting Room




DiOr generation pipeline

# IV. Methodologies

## 2. Virtual Fitting Room called DiOr - Person Representation

We represent a person as a (pose, body, {garments}) tuple, each element of which can come from a different source image

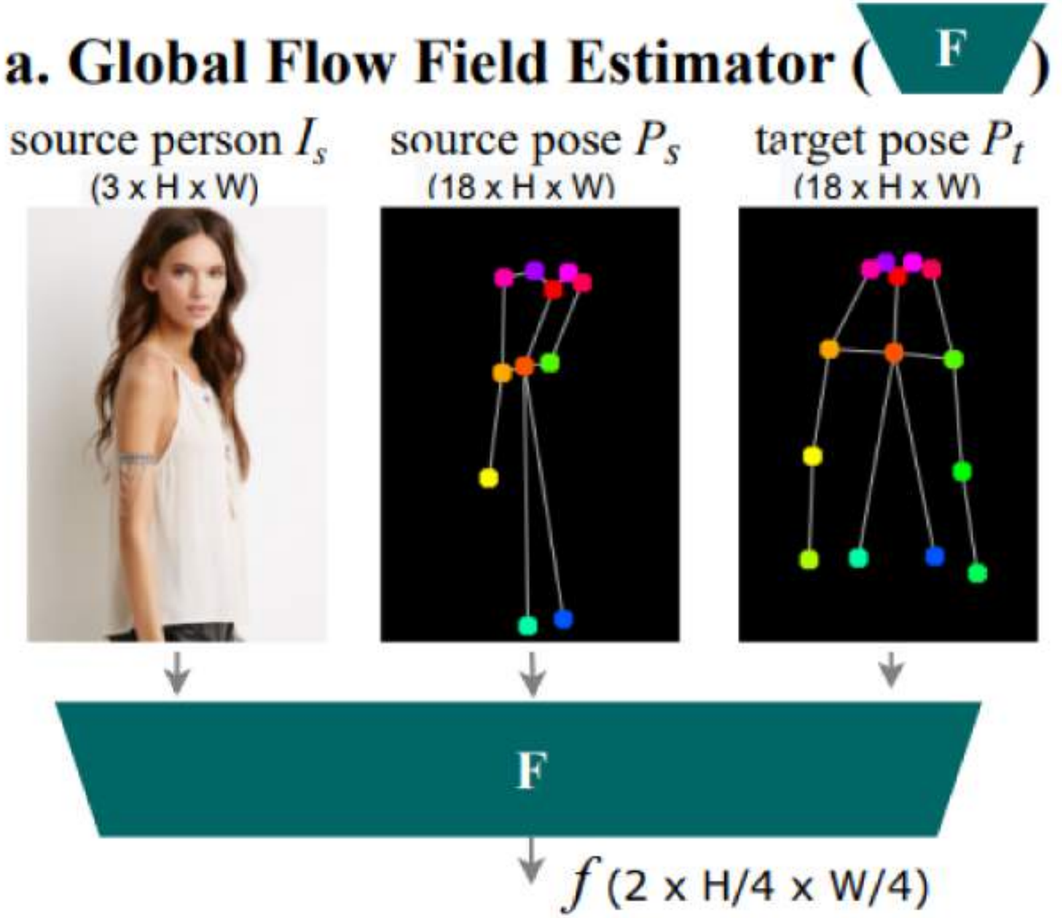
Garment representation

Body representation

# IV. Methodologies

## 2. Virtual Fitting Room - Garment Representation

- The garment representation process involves obtaining the masked garment segment and pose estimate from an image of a person wearing the garment.
- A flow field f is inferred to align the garment segment with the desired pose by using the Global Flow Field Estimator F.

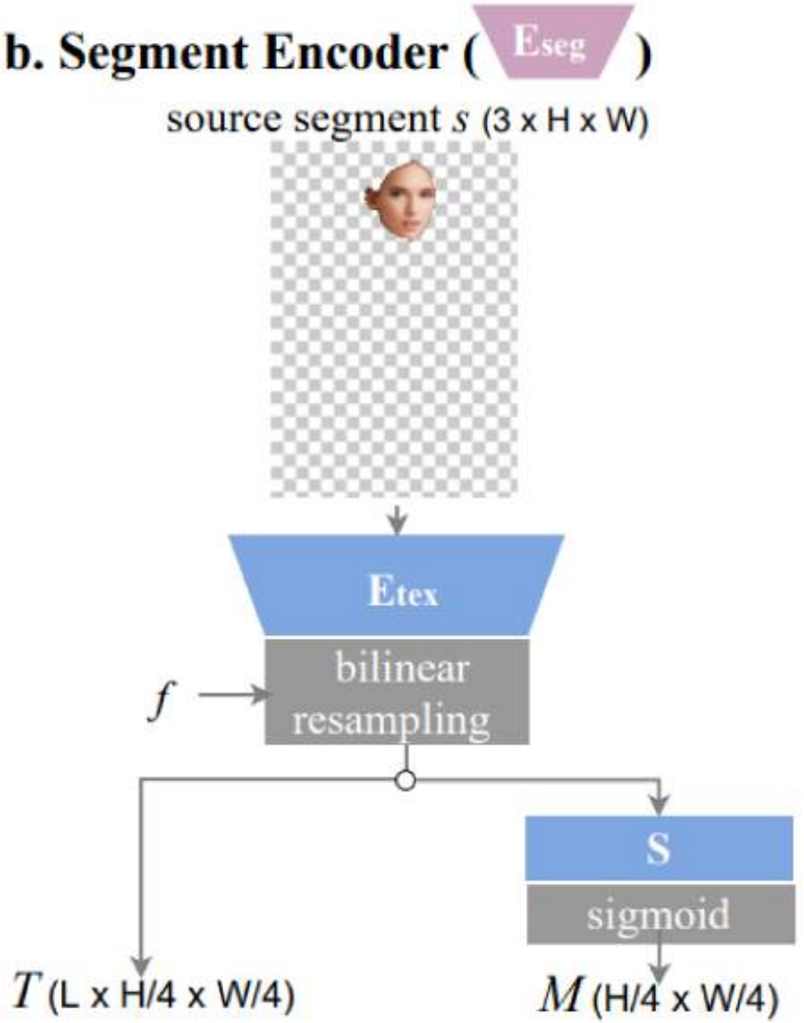



a. Global Flow Field Estimator

# IV. Methodologies

## 2. Virtual Fitting Room - Garment Representation

- The garment segment is encoded using a texture encoder and a shape mask is computed. The texture feature map is then mapped to the correct dimension for style blocks.

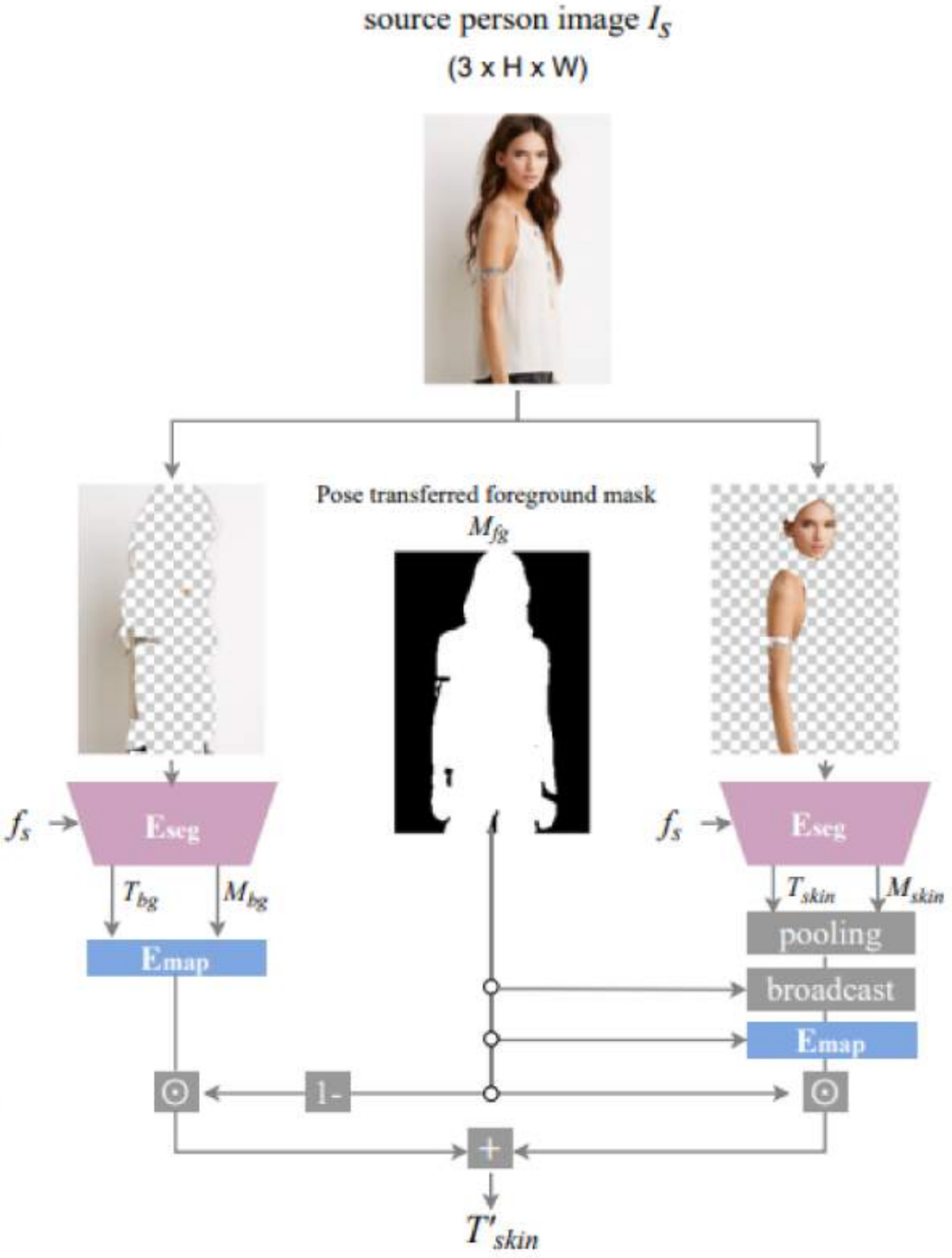



b. Segment Encoder

# IV. Methodologies

## 2. Virtual Fitting Room - Body Encoder

- The body of the source person is encoded using a human segmenter, resulting in masks for the background and skin (consisting of arms, legs and face) regions .
- The segment encoder is used to encode these masks, producing texture maps for the background and skin .
- A mean body vector is computed from the skin texture map over the region of interest defined by the skin mask .
- The mean body vector is broadcasted to the pose-transferred foreground region and mapped to the correct dimension .
- The mapped background feature is combined with the broadcasted feature map to obtain the final body feature map, accounting for both the foreground and background regions.



# IV. Methodologies

## 2. Virtual Fitting Room - Generation Pipeline

The main generation process include

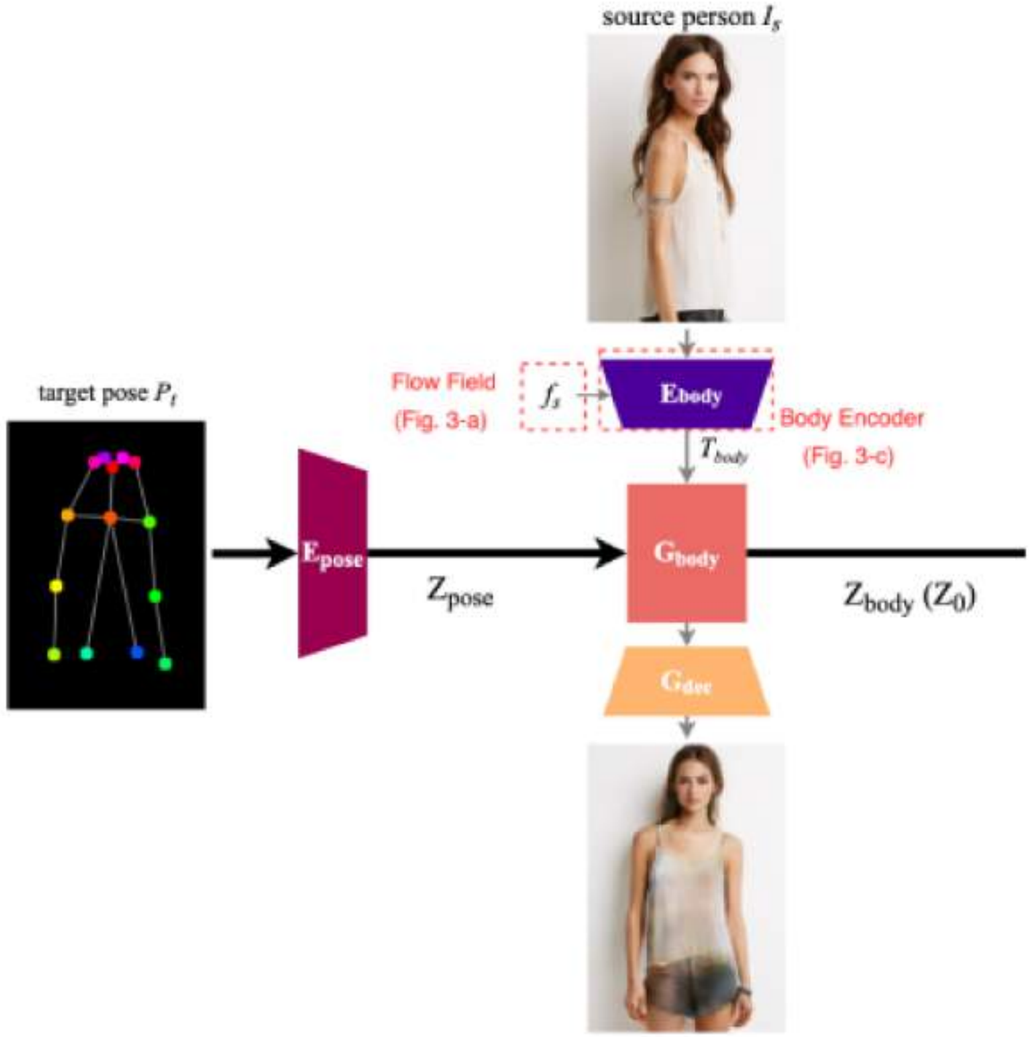
- Pose and skin generation

- Recurrent garment generation

# IV. Methodologies

## 2. Virtual Fitting Room - Generation Pipeline: Pose and skin generation

For generating the pose and skin, it uses a pose encoder called $E_{pose}$ with convolutional layers and instance normalization. This results in a hidden pose map $Z_{pose}$ .The body map $Z_{body}$ is generated from $Z_{pose}$ and the body texture map $T'_{body}$ using a body generator called $G_{body}$ which uses style blocks similar to ADGAN but with SPADE replacing adaptive instance normalization.



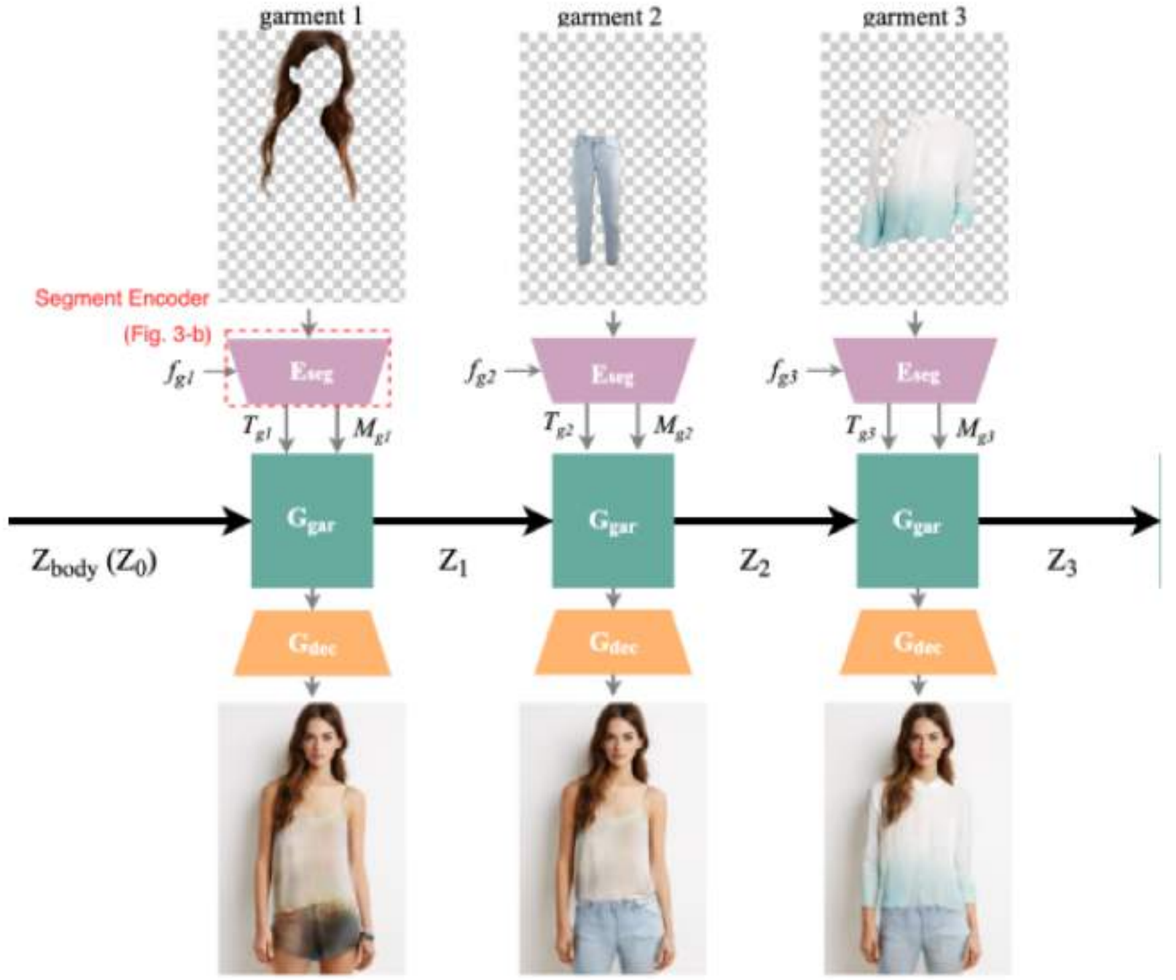

Pose and skin generation

# IV. Methodologies

## 2. Virtual Fitting Room - Generation Pipeline: Recurrent garment generation

In the recurrent garment generation phase, garments are generated one by one. For the k-th garment, garment generator $G_{gar}$ takes a texture map $T'_{gk}$, soft shape mask $M_{gk}$, and the previous state $Z_{k-1}$ to produce the next state $Z_k$ using a conditional generation module. The soft shape mask effectively controls garment opacity - a novel feature of our representation.

After the person is dressed, the final hidden feature map $Z_k$ is obtained, and the output image Igen is generated using a decoder ($G_{dec}$) similar to the one in ADGAN, which includes residual blocks, upsampling, convolutional layers, and normalization. This process results in the final image of the dressed person.




Recurrent garment generation

# IV. Methodologies

## 3. Rate your Outfit

### Introduction

Rate My Fit - A revolutionary AI that rates people's outfits based on color coordination, mood/aesthetic, appropriateness for the current weather, and the combination of complementary textures. We wanted to create a tool that not only enhances people's fashion sense but also helps them make the best outfit choices for any occasion and weather.

# IV. Methodologies

## 3. Rate your Outfit

### How we build it

To train our computer vision model, we collected a dataset of over 200,000 images of various outfits. We carefully curated the dataset to ensure a diverse representation of styles, body types, and occasions. Using this dataset, we were able to train our model to accurately recognize and analyze different aspects of an outfit such as color coordination, mood/aesthetic, appropriateness for the current weather, and the combination of complementary textures.

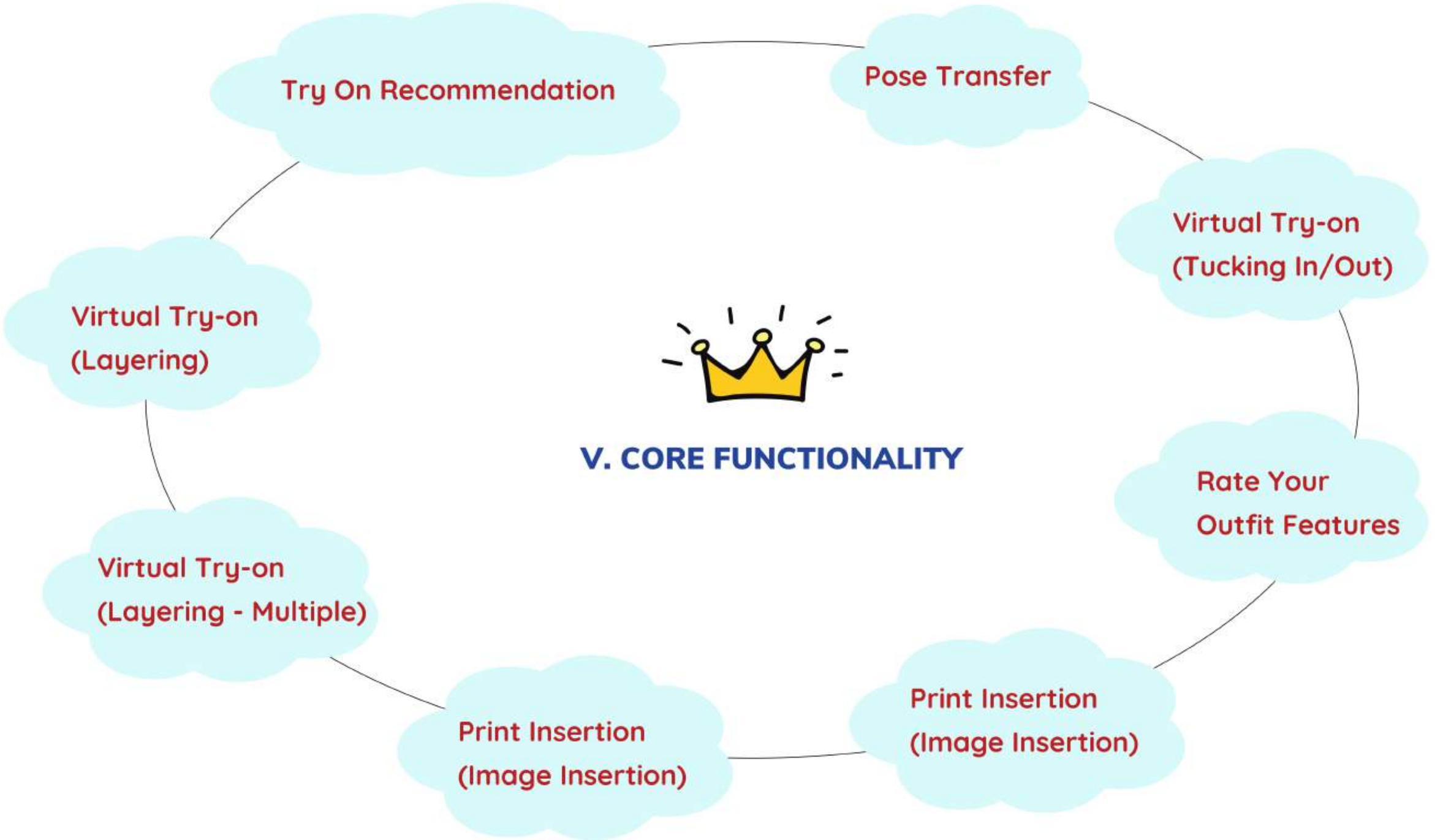
### Weather fit

To know the weather of user location, we use an api:

```
api_key = "88cd16a2bf4b36c85acaac28009e7dbf"
weather_base_url = "http://api.openweathermap.org/data/2.5/weather?"
```

# V. CORE FUNCTIONALITY

Revolutionizing virtual fitting rooms by enabling layered dressing for seamless outfit combinations.


Try On Recommendation
Pose Transfer
Virtual Try-on
(Tucking In/Out)
Rate Your
Outfit Features
Print Insertion
(Image Insertion)
Print Insertion
(Image Insertion)
Virtual Try-on
(Layering - Multiple)
Virtual Try-on
(Layering)
V. CORE FUNCTIONALITY

# V. Core Functionality

## 1. Pose transfer

Pose transfer allows users to change the pose or stance of the virtual model showcasing the clothing. Users can select from a range of predefined poses or customize the pose to see how the clothing looks in various positions.

This feature helps users assess the comfort, flexibility, and appearance of the clothing in different postures, ensuring it meets their needs.

# V. Core Functionality

## 2. Virtual Try-on (Tucking In/Out)

Users can virtually tuck in or untuck clothing items like shirts, blouses, or t-shirts. This feature helps users visualize how a specific item can be styled to achieve different looks.

It enables users to experiment with different outfit styles and find the best look for their preferences and occasions.

# V. Core Functionality

## 3. Virtual Try-on (Layering)

Layering allows users to mix and match multiple clothing items, such as pairing a shirt with a jacket or sweater, and see how they complement each other looks.

Users can create versatile outfits, ensuring their choices match their personal style and the weather conditions.

# V. Core Functionality

## 4. Virtual Try-on (Layering - Multiple)

This advanced feature enables users to layer multiple clothing items to create complex outfits. Users can combine various garments to design a complete look.

It offers a more comprehensive understanding of outfit combinations, helping users experiment with multiple layers and styles.

# V. Core Functionality

## 5. Print Insertion (Image Insertion)

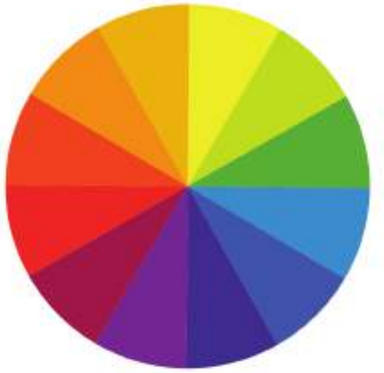
Users can insert images or custom graphics into clothing items, such as t-shirts, hoodies, or accessories. This feature personalizes the clothing with custom designs or artwork.

Users can design unique and custom clothing items, making the AdiStyle AI Assistant a versatile platform for both fashion enthusiasts and those looking for personalized apparel.

# V. Core Functionality

## 6. Rate Your Outfit Features

The AI assistant includes an outfit rating feature that provides a numerical score and qualitative feedback on user-selected outfits.



The AI evaluates the outfit's overall aesthetic, taking into account factors like color coordination, style, and seasonal appropriateness.

# V. Core Functionality

## 7. Try on Recommendation

User input

I want a comfortable T-shirt, which is formal but also easy to wear

Our recommendation output

### A CROPPED T-SHIRT WITH VERSATILE STYLE.

Whether it's a morning yoga flow, an afternoon run through the park or an evening of hanging with friends, the important thing is to do what makes you happy. This tee will match your energy every step of the way. Feel cool and confident with just the right amount of coverage. Pair it with tights to dial up the sports-inspired look or with your favorite denim for an everyday style that's on point for any occasion.

This information is retrieved using url column in dataset

# VI. Performance Metrics

# VI. Performance Metrics

## We employ these key metrics to access the product's performance:

Sales Growth: assessing the financial performance of the website, reflecting the effectiveness of the user experience in driving revenue. Key metrics include:

- Revenue: The total income generated through online sales.
- Average Order Value (AOV): The average amount spent by customers per order.
- Conversion Value: The total value of completed transactions

Conversion Rates: the percentage of visitors who take a desired action. Tracking conversion rates helps evaluate the effectiveness the overall user experience. Key metrics could be:

- Add-to-Cart Rate: The percentage of visitors who add products to their shopping cart.
- Conversion Rate: The percentage of visitors who make a purchase.

User Engagement: the level of interaction and involvement of visitors on the website, indicating the effectiveness of our MVP. Key metrics we use are:

- Pageviews: The total number of pages viewed by visitors.
- Time on Page: The average duration of time visitors spend on each page.
- Bounce Rate: The percentage of visitors who leave your website after viewing only one page.

# VI. Performance Metrics

## To effectively measure those metrics, we use following tools:

User Feedback: Encouraging users to provide feedback through surveys, feedback forms, or user testing sessions to gain valuable insights into their experience and satisfaction.

A/B Testing: Conducting controlled experiments by presenting different versions of your website to users and comparing the performance of different elements or features.

Analytics Tools: Utilizing web analytics platforms, such as Google Analytics, to track and analyze user behavior, engagement, conversion rates, and other relevant metrics.

# VII. Timeline and Roadmap

**25/11**

Using dataset URL to scrape image of product from web (about 50 - 100 images)

**28/11**

Complete run all models demo on scraped dataset

**30/11**

Complete web UI design

**8/12**

Complete the web functionality (not include model yet)

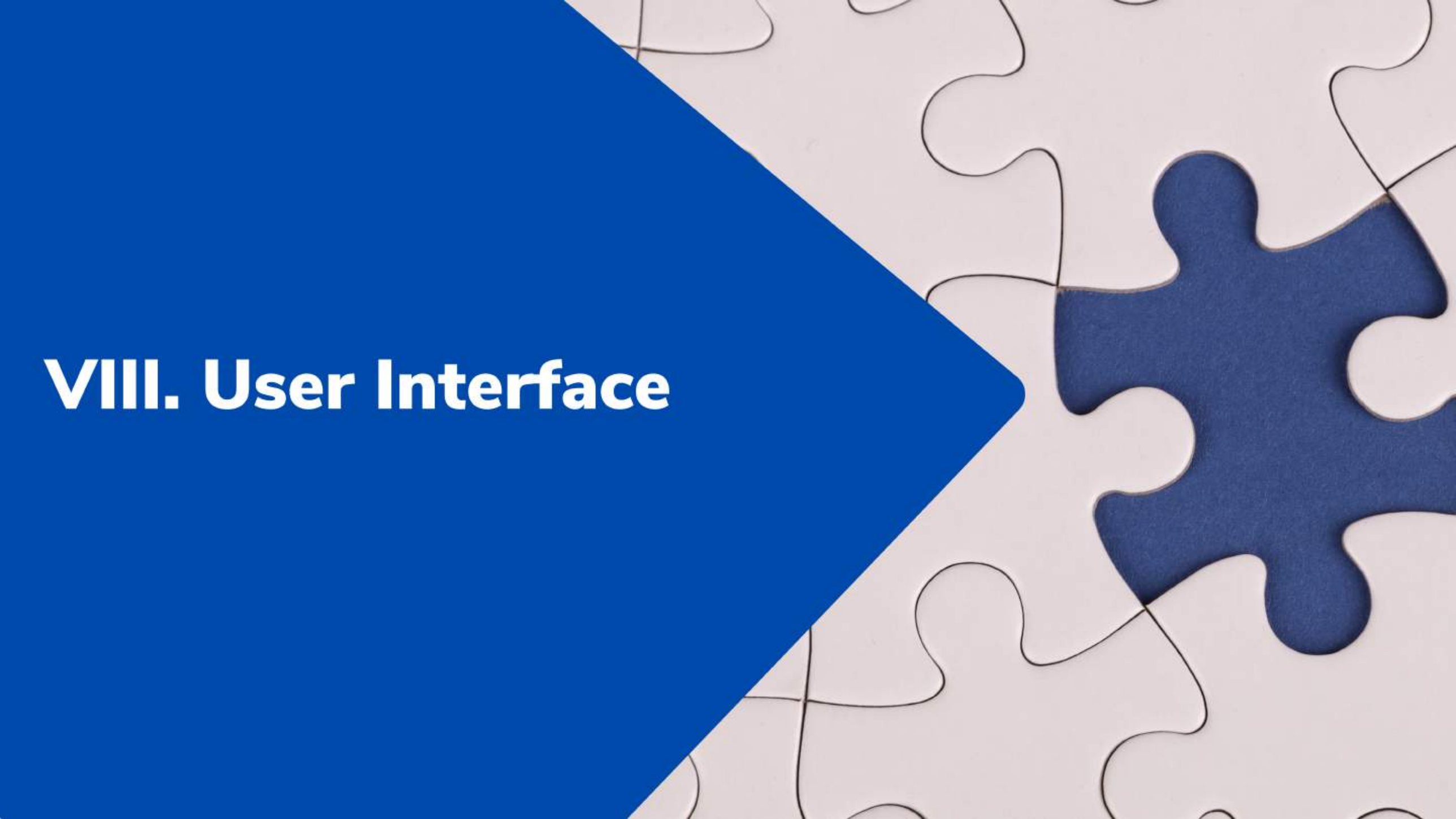
**15/12**

Complete deploy model on web

**16/12**

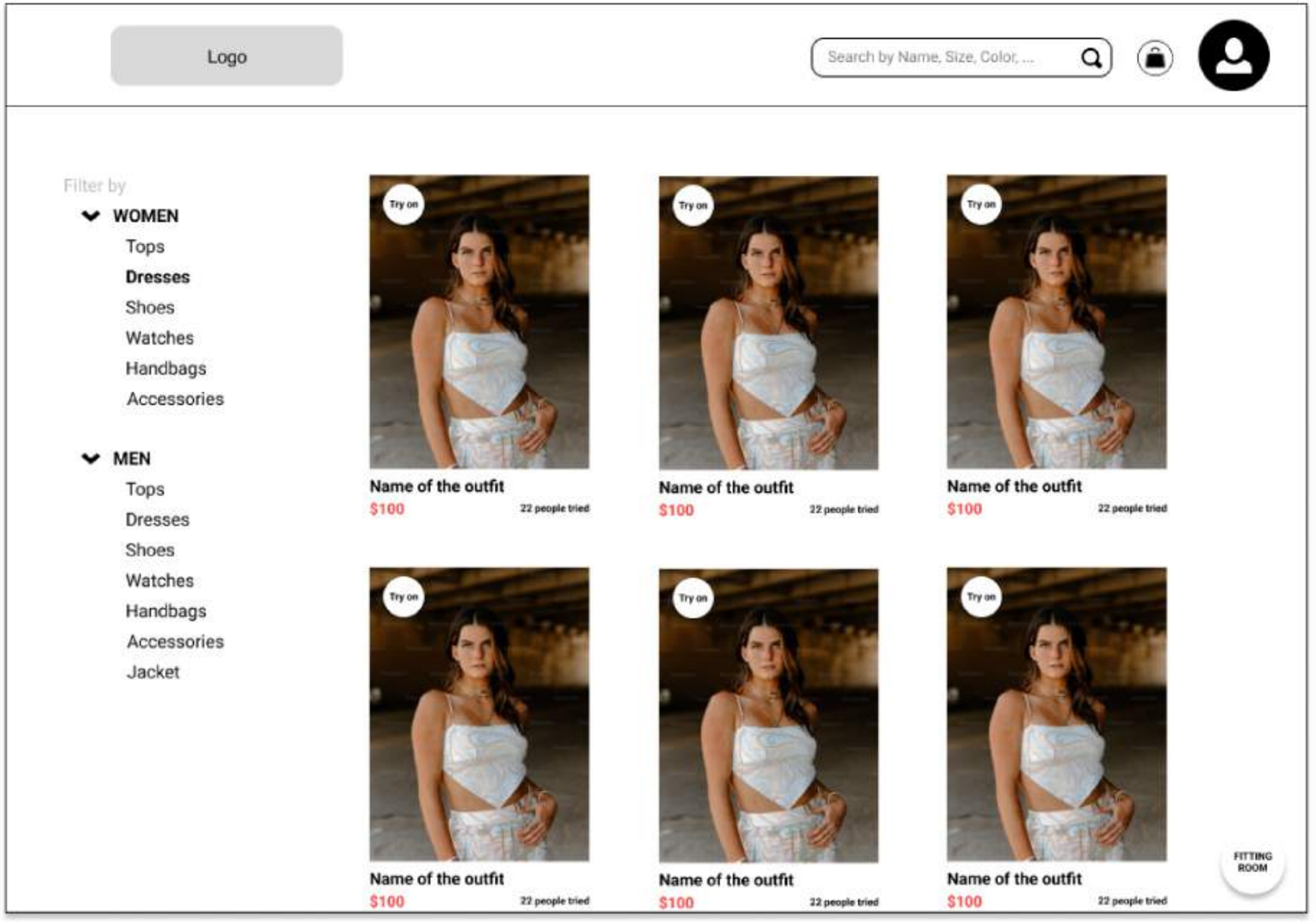
Finish set up website on datathon server

# VIII. User Interface



# VIII. User Interface

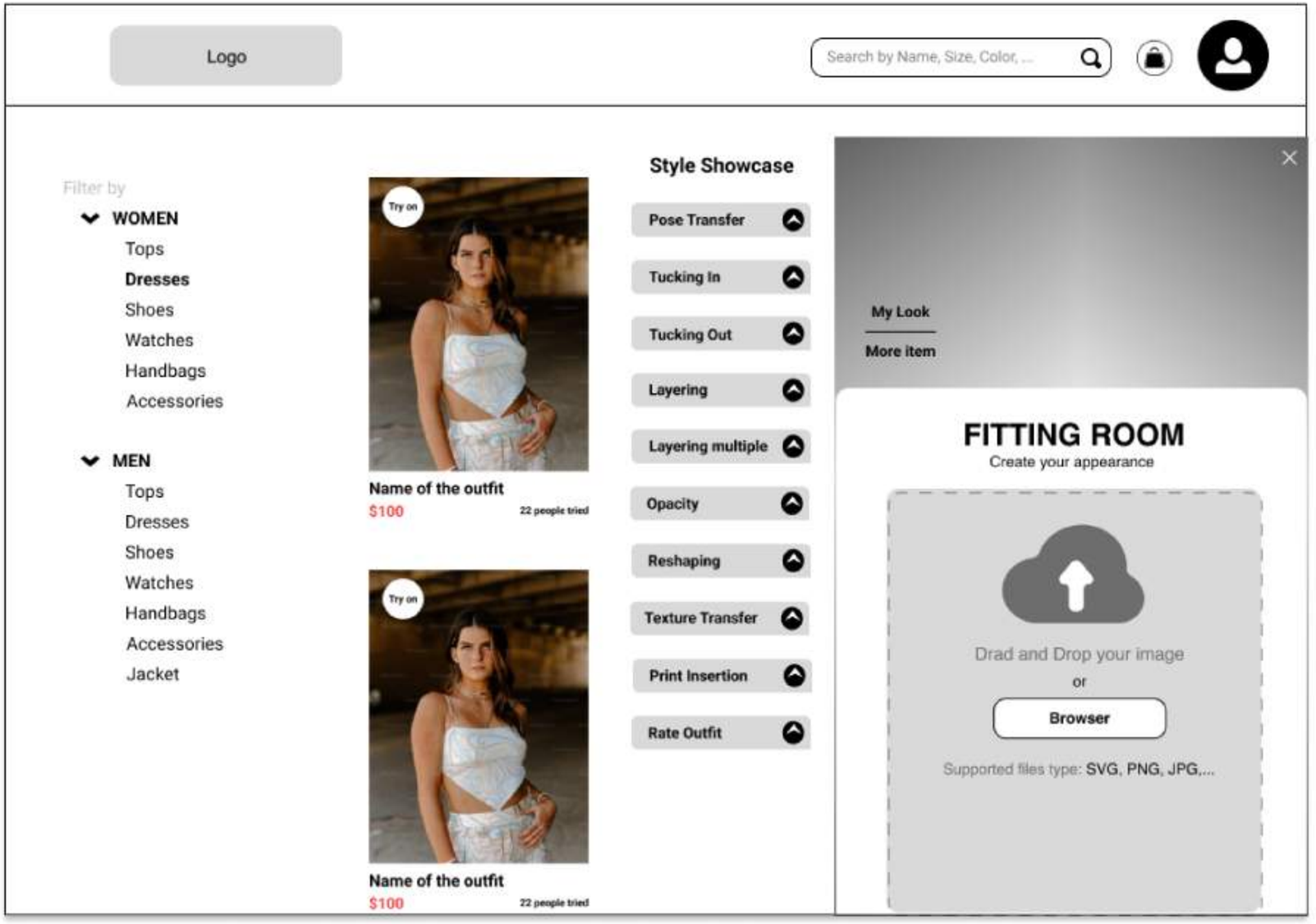
## HomePage



# VIII. User Interface

# Virtual Fitting Room

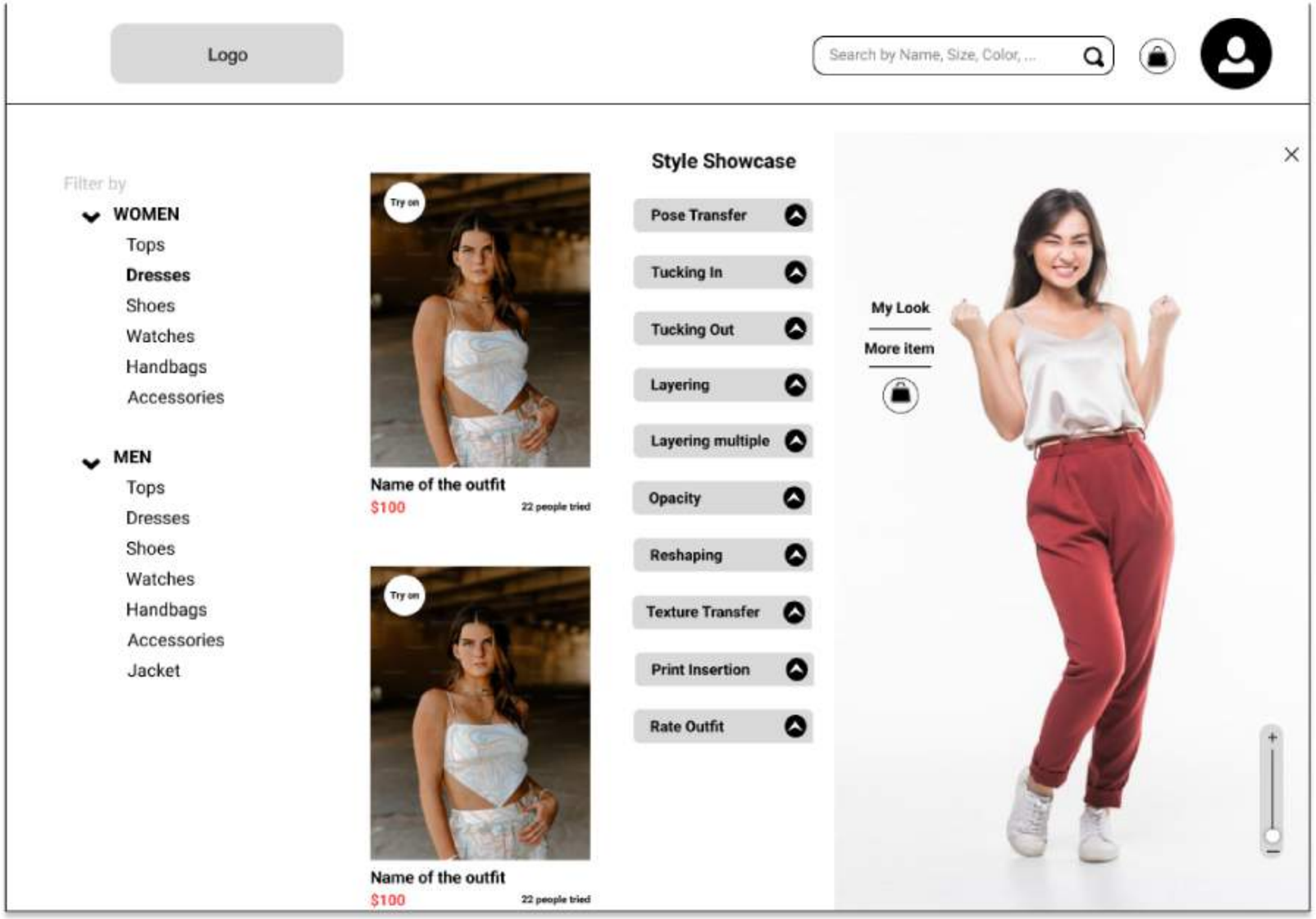
**The fitting room before uploading the image**



# VIII. User Interface

## Virtual Fitting Room

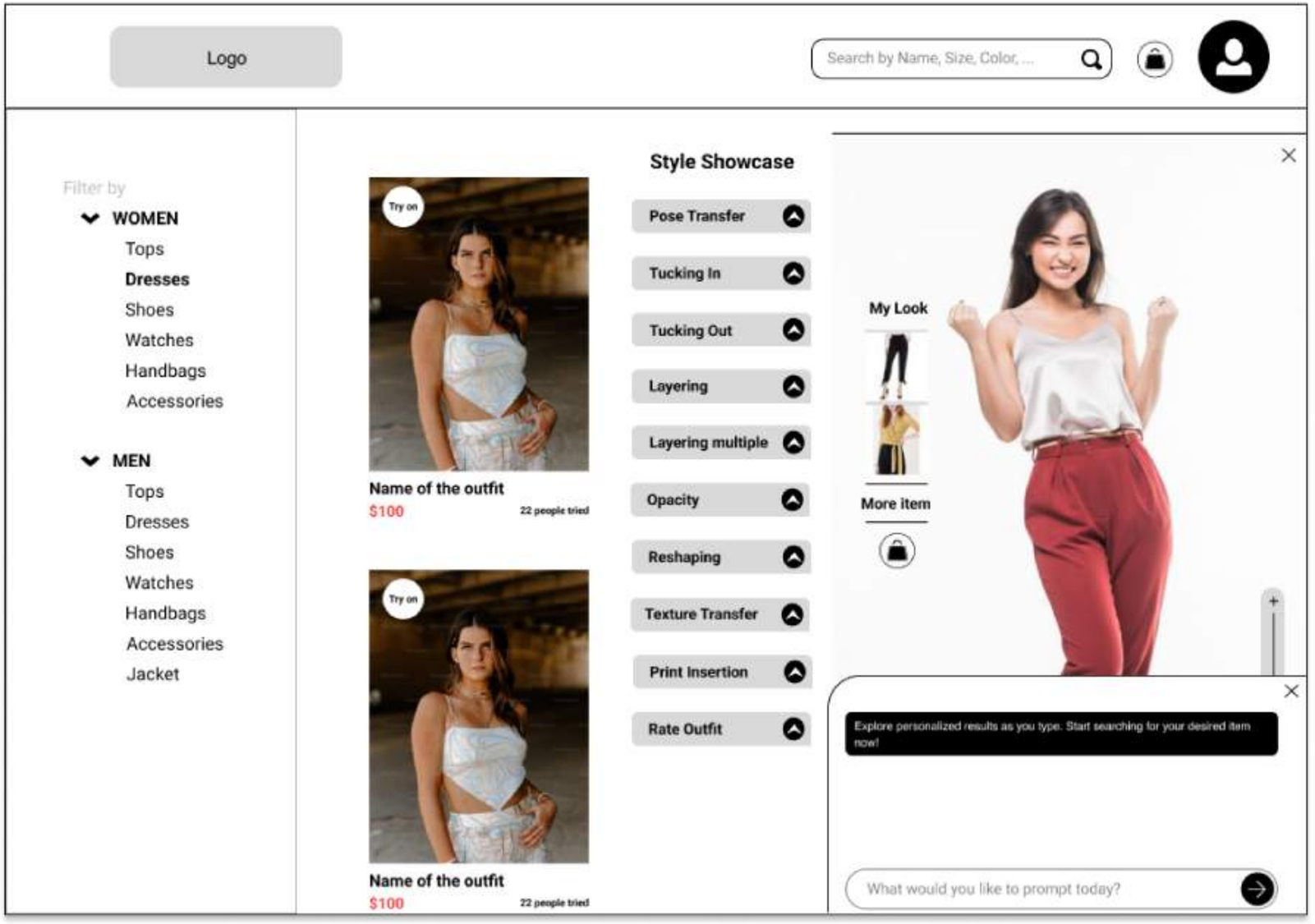
The fitting room after uploading the image



# VIII. User Interface

## Optimized Search

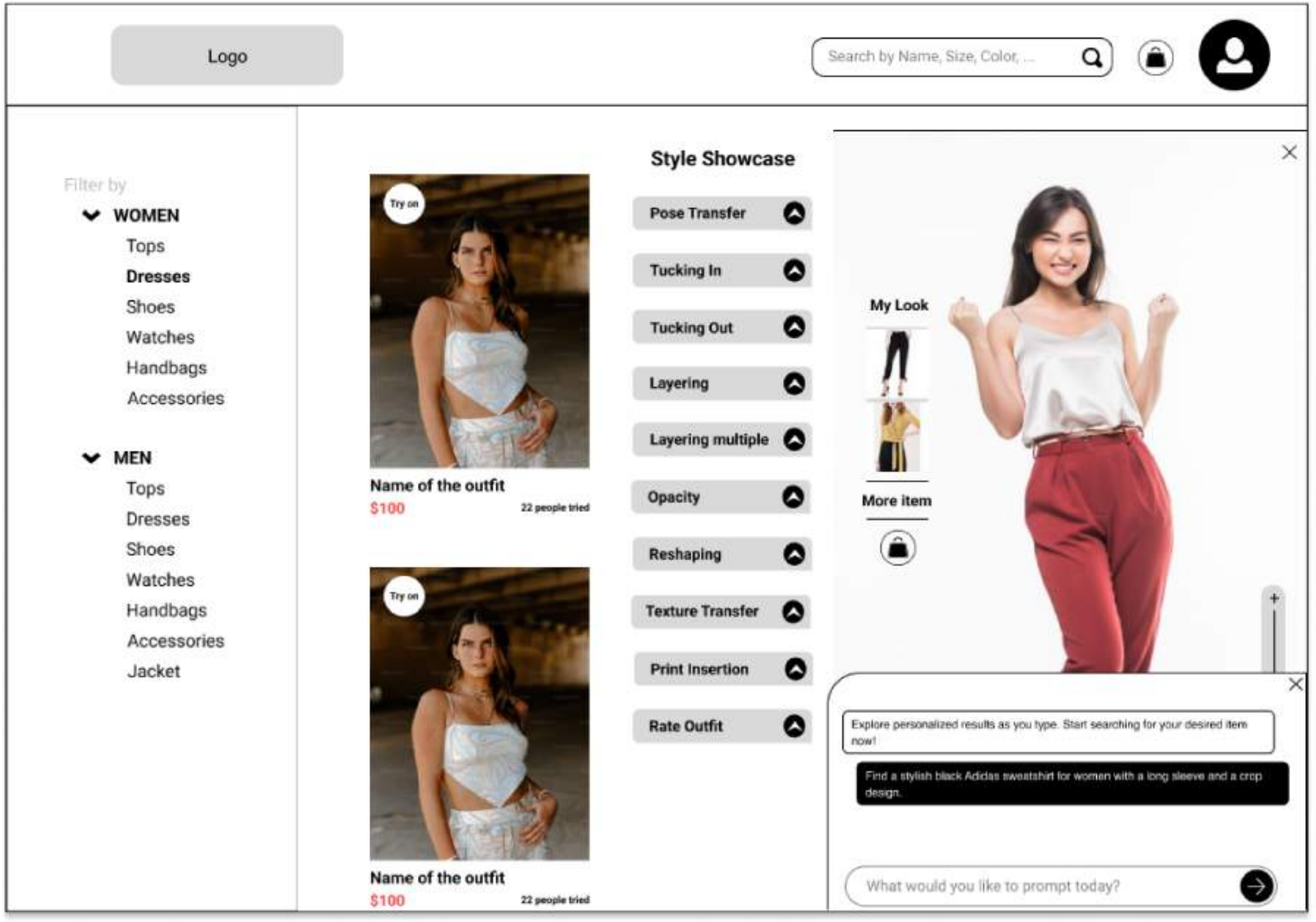
The Optimized Search appear after press “More Item”



# VIII. User Interface

## Optimized Search

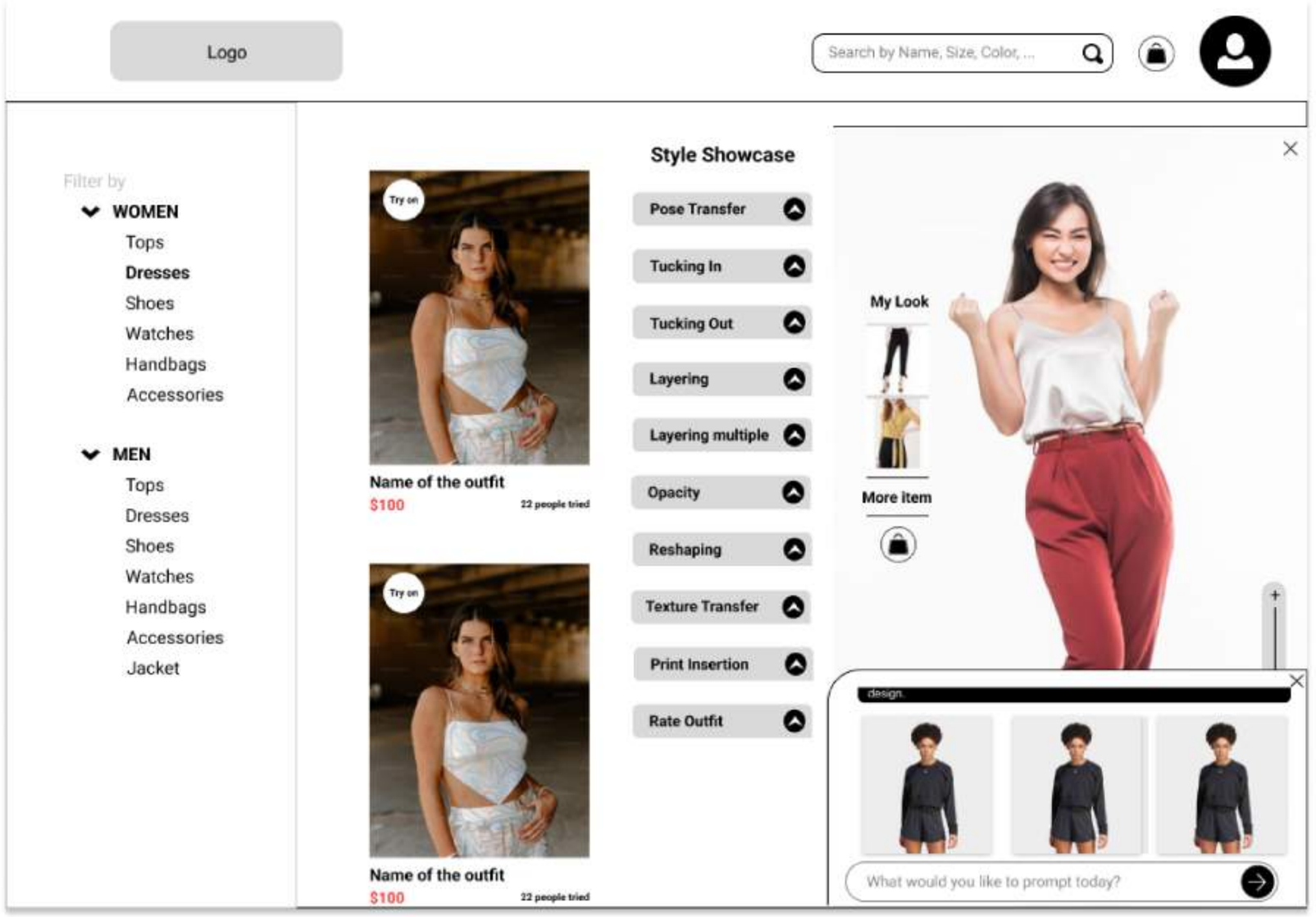
**Represents what the user entered at the prompt.**



# VIII. User Interface

## Optimized Search

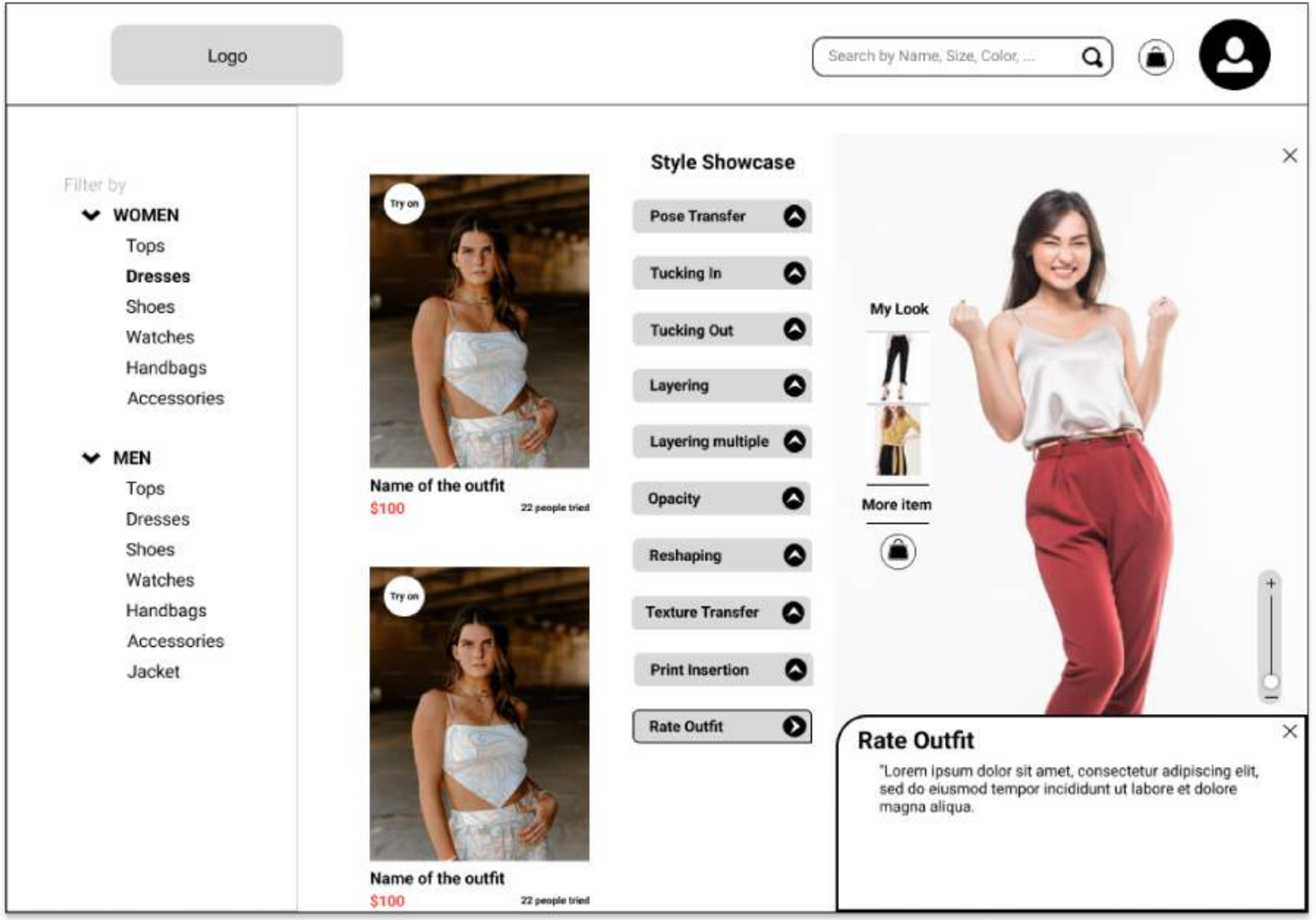
**The Optimized Search will return results based on the user's prompt.**



# VIII. User Interface

# Rate Outfit

**The function is used to evaluate the outfit that the user has tried on.**



# IX. Conclusion

Our solution provides customers  the easiest way to discover their best outfit

An advanced search algorithm
ultilize advanced AI technique

- Time saving
- Accuracy searching

A virtual fitting room which can
sequentially put garments on a person

- Provide a realistic view of how clothing will look on them
- give user freedom tocreatively custom their set of outfit

A clothing scoring system

- give customer idea how to improve their outfit

# IX. Conclusion

With our solution online shopping will be

More convenient

More interactive

More personalize

More creative

Enhancing user experience

Boosting sales figures

# X. Limitations and Future Enhancements

# X. Limitations and Future Enhancements

Limitations of Recommend System model:

1. The model is currently limited to utilizing only the latest input from the user and lacks the ability to leverage past conversations (similar to GPT chat) for providing suggestions.

Limitations of DiOr model:

1. Inaccurate rendering of complex or rarely seen poses.
2. Failure to preserve unusual garment shapes.
3. While better than other recent methods, garment features are still not entirely realistic.

Limitations of Rate you Outfit model:

1. It is just a simple model, so it cannot produce high accurate scores
2. The model needs a lot of pre-labeled data, as well as highly constrained input requirements

# X. Limitations and Future Enhancements

Future enhancements of all projects:

1. Improve the quality of garment representation and body representaion.
2. Create more functionality for virtual fitting room.
3. leverage past conversations (similar to GPT chat) for providing suggestions.
4. Apply a more advanced scoring model for fashion outfit clothing.

# Thank You