

A photograph of a woman with curly brown hair, wearing a light blue blazer over a white shirt. She is looking towards the right side of the frame, possibly at a clothing rack. The background is blurred, showing other garments and lights, suggesting a retail environment.

Aug 21, 2024

## **Title: Virtual Wardrobe and Shopping Assistant**

Empowering Your Fashion Journey with Innovative Technology

### **Presented by:**

Balaji K (727821tucs026)

Dharanidharan S (727821tucs039)

Dhiyaneshwar R (727821tucs043)



# Base paper details

**Title of the Paper:** Developing a Smart Wardrobe System  
**Authors:** Yujie Lin , Pengjie Ren, Zhumin Chen,  
Zhaochun Ren, Jun Ma,  
**Year:** 2020

## Journal Details:

The paper presents the Neural Outfit Recommendation (NOR) framework, which simultaneously provides outfit recommendations and generates explanations in the form of comments. NOR utilizes a convolutional neural network for outfit matching and a gated recurrent neural network for comment generation, trained together in a multi-task learning framework. The approach achieves state-of-the-art results on outfit recommendation and comment generation tasks.

# Abstract :

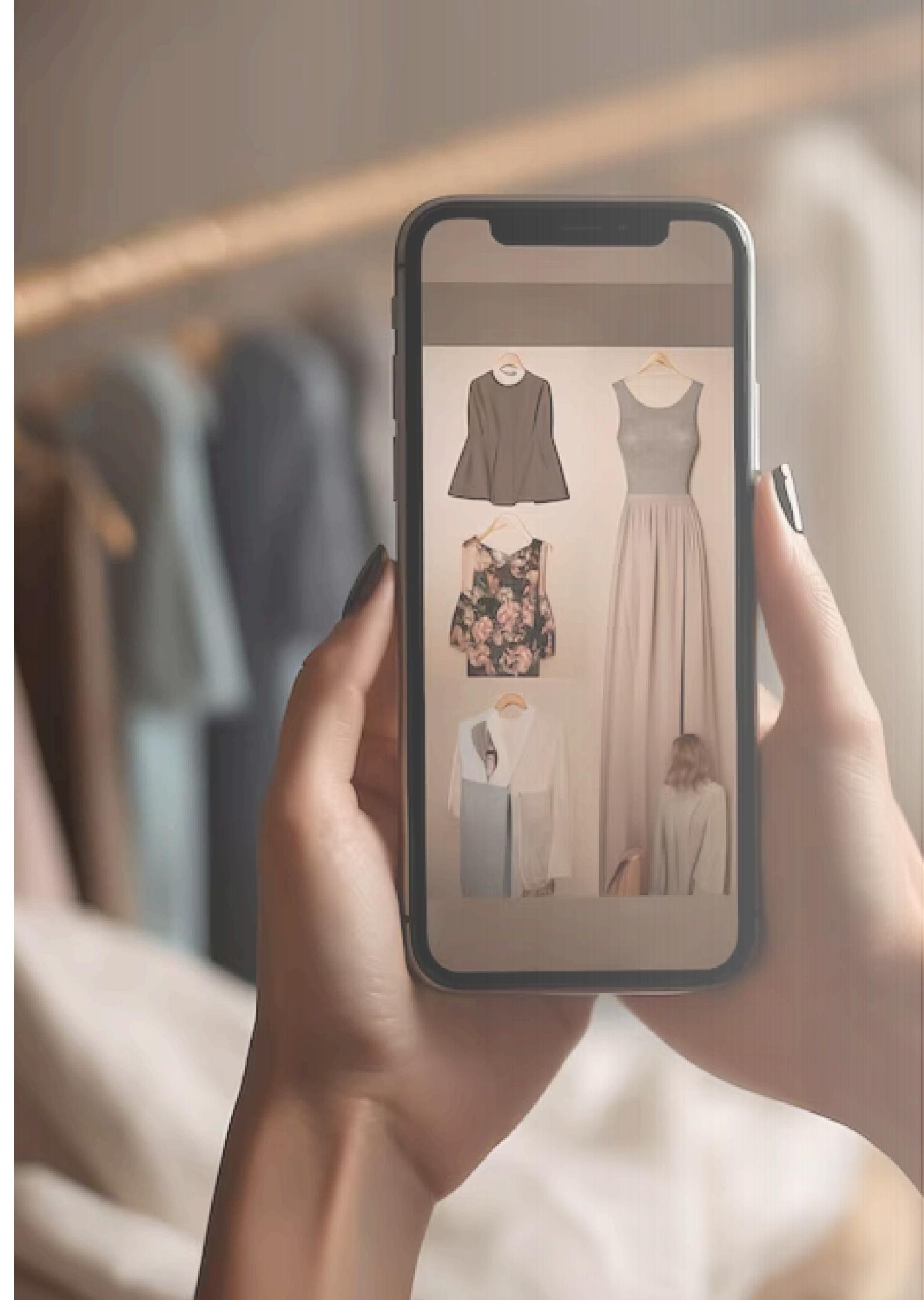
The Virtual Wardrobe and Shopping Assistant project aims to revolutionize the way users manage their fashion choices. This mobile app integrates advanced technologies such as Artificial Intelligence (AI) and Augmented Reality (AR) to offer personalized outfit recommendations and virtual try-ons. Users can digitally organize their wardrobe, explore AI-generated designs for custom clothing, and locate nearby stores. Additionally, the app provides curated shopping suggestions, exclusive deals, and connects users with a vibrant fashion community. This innovative solution addresses common challenges in fashion management, making it easier for users to stay stylish and organized in their daily lives.



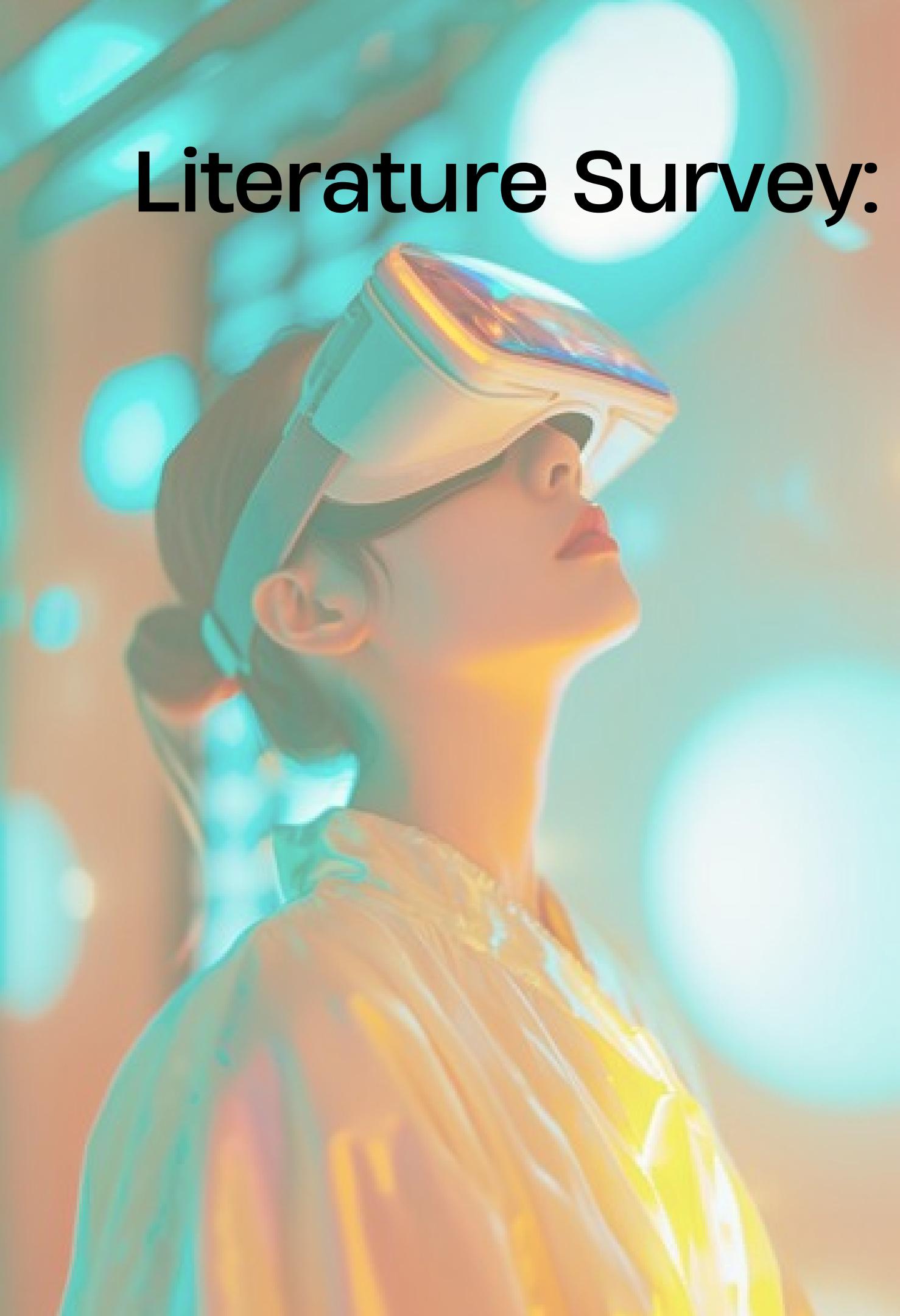
# Introduction:

Fashion management often poses challenges such as keeping track of clothing items, selecting suitable outfits for various occasions, and staying updated with current trends. Traditional solutions lack the personalization and advanced technology integration needed to address these issues effectively.

The Virtual Wardrobe and Shopping Assistant is designed to fill this gap by offering a comprehensive platform that leverages AI and AR. This app not only helps users organize their wardrobe but also provides tailored outfit recommendations based on weather, style preferences, and upcoming events, enhancing the overall fashion experience and making daily outfit selection effortless.



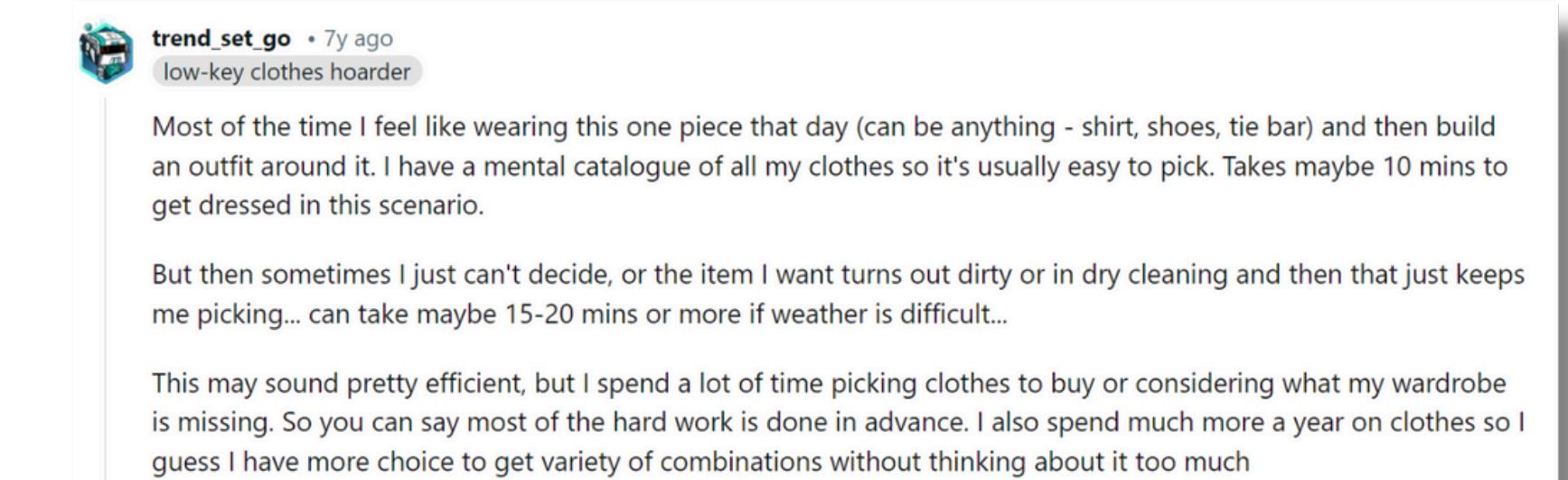
# Literature Survey:



Outfit recommendation research has evolved from simple visual-based models to more complex systems incorporating user comments for better explainability. Early works, such as those by Iwata et al. (2011) and Liu et al. (2012), focused on visual features and occasion-based recommendations using small datasets. Recent advances leverage larger datasets from platforms like Polyvore, enabling the use of deep learning models like CNNs. Lin et al. (2020) introduced the Neural Outfit Recommendation (NOR) framework, which combines outfit matching with comment generation, enhancing both recommendation accuracy and explainability through user-generated textual feedback.

# Problem Definition:

- The primary problem addressed by this project is the lack of a comprehensive tool for managing personal fashion.
- Users struggle with organizing their wardrobe, selecting outfits that match their style and the weather, and finding new clothing that suits their preferences. Existing solutions are either too simplistic or focus on only one aspect of the fashion process.
- The Virtual Wardrobe and Shopping Assistant aims to solve these issues by integrating AI for personalized recommendations, AR for virtual try-ons, and a robust database to keep track of wardrobe items, shopping preferences, and the latest trends.



trend\_set\_go • 7y ago  
low-key clothes hoarder

Most of the time I feel like wearing this one piece that day (can be anything - shirt, shoes, tie bar) and then build an outfit around it. I have a mental catalogue of all my clothes so it's usually easy to pick. Takes maybe 10 mins to get dressed in this scenario.

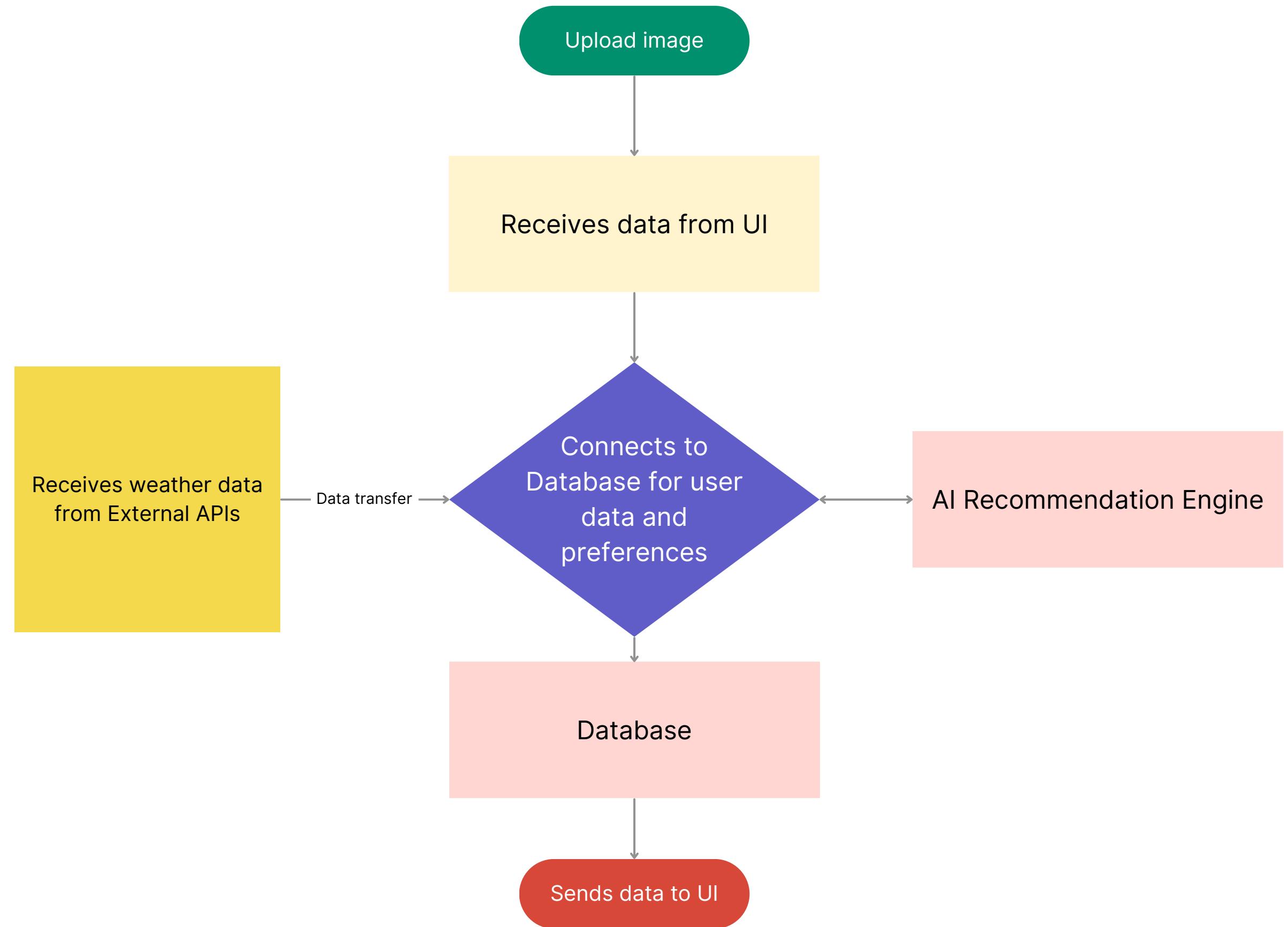
But then sometimes I just can't decide, or the item I want turns out dirty or in dry cleaning and then that just keeps me picking... can take maybe 15-20 mins or more if weather is difficult...

This may sound pretty efficient, but I spend a lot of time picking clothes to buy or considering what my wardrobe is missing. So you can say most of the hard work is done in advance. I also spend much more a year on clothes so I guess I have more choice to get variety of combinations without thinking about it too much

# **Proposed System/Methodology:**

The proposed system is a mobile application that combines AI, AR, and a robust database to deliver a personalized fashion management experience. The app allows users to digitize their wardrobe, receive AI-powered outfit suggestions based on weather and style, and virtually try on clothes using AR. Additionally, users can explore AI-generated custom clothing designs and locate nearby stores. The methodology involves developing algorithms for outfit recommendations, implementing AR features for virtual try-ons, and integrating a community platform for users to share and receive fashion tips. This approach aims to create a seamless, user-friendly experience.

# Block Diagram:



# Modular Description:

The system is divided into several modules:

- 1) Wardrobe Management Module allows users to digitize and organize their clothes
- 2) Recommendation Module uses AI to suggest outfits based on user preferences and weather conditions
- 3) AR Try-On Module enables users to virtually try on clothes
- 4) Custom Design Module generates AI-based custom clothing designs
- 5) Store Locator Module helps users find nearby stores and offers
- 6) Fashion Community Module connects users with a community for sharing tips and inspirations. Each module interacts seamlessly with others to provide a holistic fashion management experience.

# Platform, Dataset, and Tools:

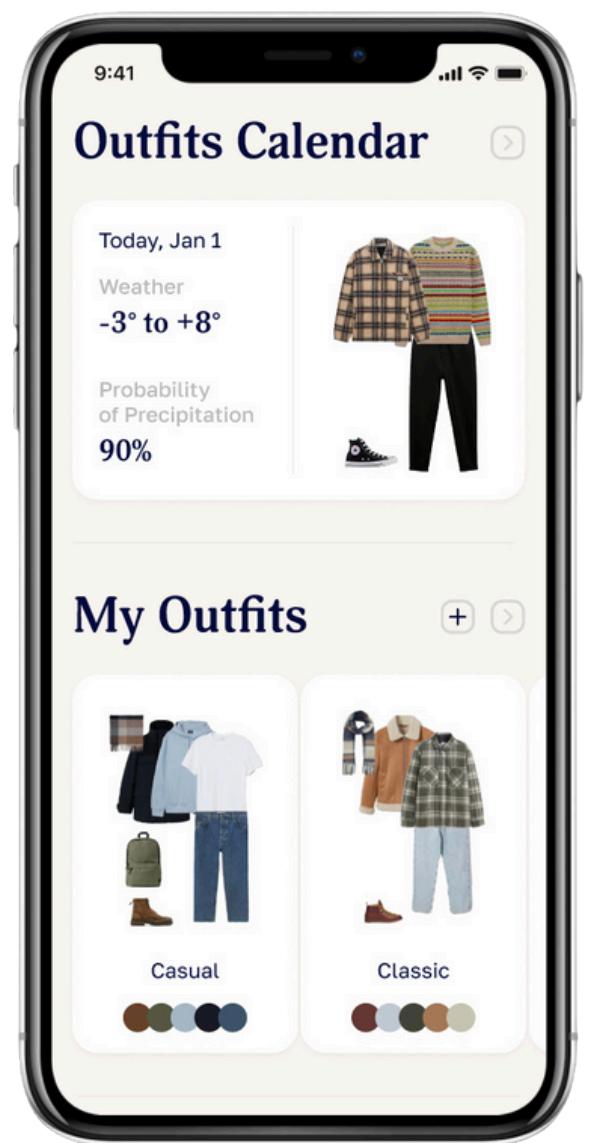
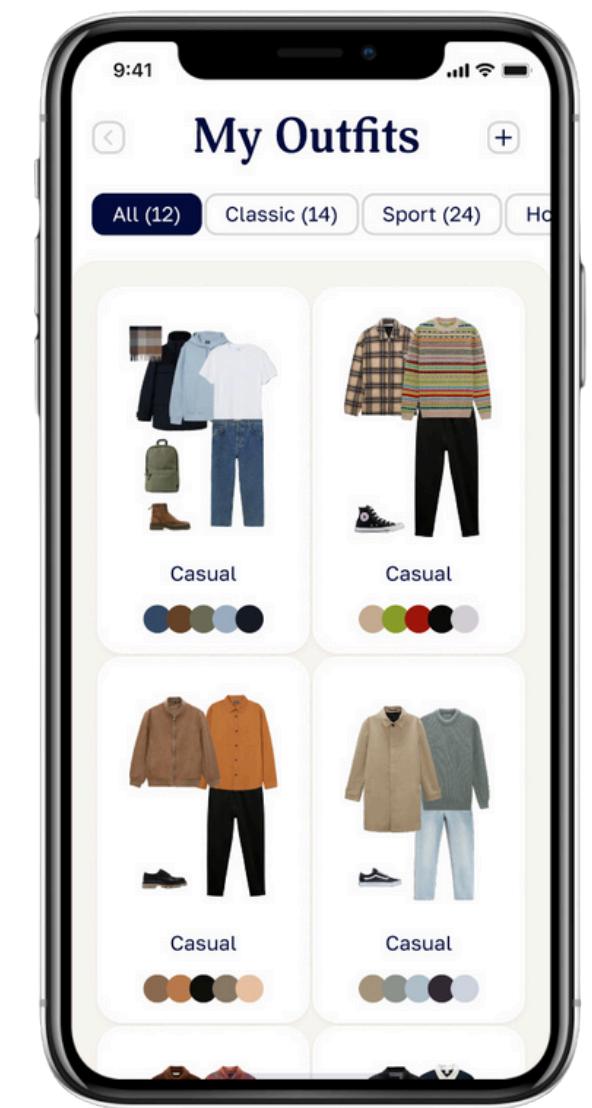
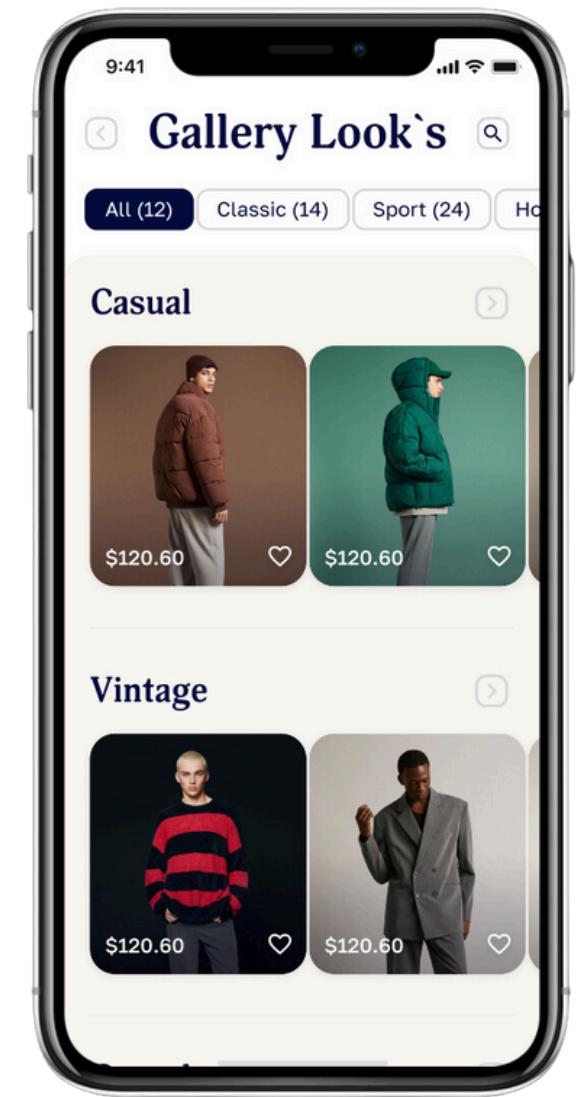
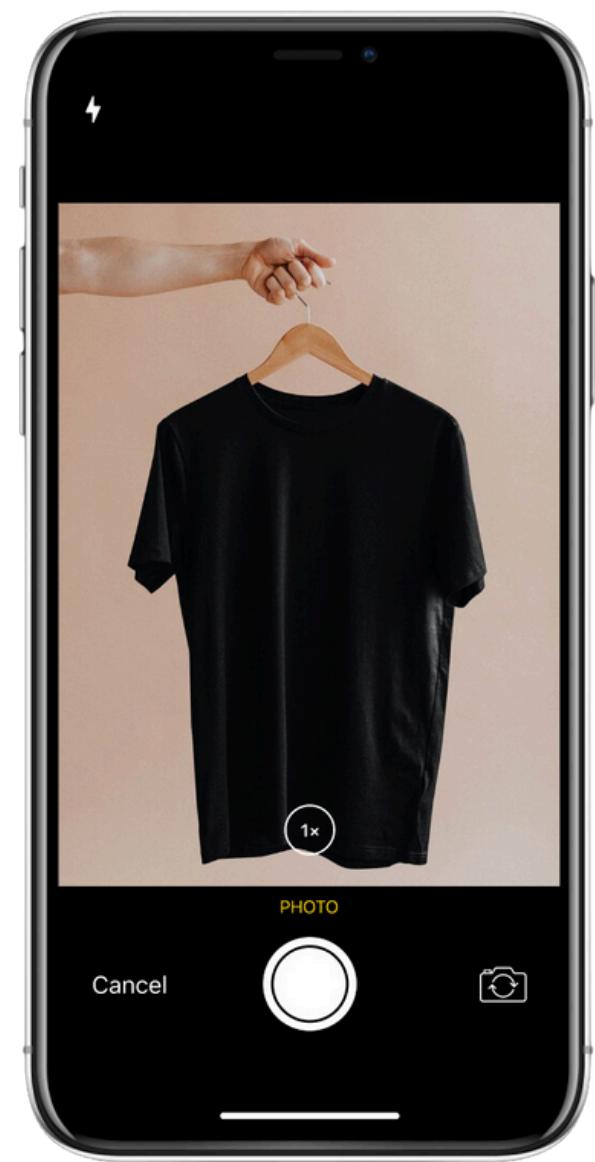
The app is developed for both Android and iOS platforms, ensuring broad accessibility. The dataset consists of a comprehensive collection of fashion items, styles, and trends, which is regularly updated to keep the app's recommendations relevant. The tools used include React Native for cross-platform development, TensorFlow for AI model training, and Unity for AR implementation. The backend is powered by Firebase for real-time database management and user authentication. The integration of these technologies ensures a responsive and scalable application that delivers a seamless user experience across devices.

# Recommendation Module

The Recommendation Module uses weather data to provide personalized suggestions for clothing and accessories. By integrating weather forecasts into the recommendation engine, this module dynamically adjusts its recommendations based on current and upcoming weather conditions. For instance, if the forecast predicts rain, the module might suggest waterproof jackets or umbrellas.

On a sunny day, it could recommend light, breathable fabrics. This approach ensures that users receive relevant and practical outfit suggestions that align with their immediate environmental conditions, enhancing both comfort and convenience. The module's ability to offer tailored advice not only improves user satisfaction but also optimizes wardrobe choices based on real-world factors.

# Screenshots of the Output:



```
● ● ●
1 import requests
2 import random
3
4 API_KEY = '1c3cd4d05f6949ac20ef50c13f8e91d9'
5 BASE_URL = 'https://api.openweathermap.org/data/2.5/weather'
6
7 def get_weather(lat, lon):
8     """Fetch weather data from OpenWeatherMap API."""
9     params = {
10         'lat': lat,
11         'lon': lon,
12         'appid': API_KEY,
13         'units': 'metric' # Use 'imperial' for Fahrenheit, 'metric' for Celsius
14     }
15     response = requests.get(BASE_URL, params=params)
16     if response.status_code == 200:
17         return response.json()
18     else:
19         raise Exception("Error fetching weather data")
20
21 def suggest_dress(weather_data):
22     """Suggest a dress based on the current weather conditions."""
23     temp = weather_data['main']['temp']
24     weather_main = weather_data['weather'][0]['main']
25     description = weather_data['weather'][0]['description']
26
27     dresses = {
28         'Clear': [
29             'A light summer dress with floral patterns.',
30             'A casual tank top and shorts.'
31         ],
32         'Clouds': [
33             'A comfortable sweater with jeans.',
34             'A light jacket over a casual shirt.'
35         ],
36         'Rain': [
37             'A waterproof raincoat and waterproof boots.',
```

```
38             'An umbrella and a cozy hoodie.'
39     ],
40     'Snow': [
41             'A warm winter coat and snow boots.',
42             'A thermal sweater with insulated pants.'
43     ],
44     'Drizzle': [
45             'A light rain jacket and comfortable shoes.',
46             'An umbrella with a casual outfit.'
47     ],
48     'Thunderstorm': [
49             'A heavy raincoat and durable boots.',
50             'Protective gear and a warm sweater.'
51     ]
52 }
53
54 # Default suggestion for unknown weather conditions
55 default_suggestion = 'Check the weather conditions and dress appropriately.'
56
57 # Suggest dress based on weather
58 for condition, suggestions in dresses.items():
59     if condition in weather_main:
60         return random.choice(suggestions)
61
62 return default_suggestion
63
64 def main():
65     # Latitude and Longitude for the location
66     lat = input("Enter latitude: ")
67     lon = input("Enter longitude: ")
68
69     try:
70         weather_data = get_weather(lat, lon)
71         dress_suggestion = suggest_dress(weather_data)
72         print(f"Weather Description: {weather_data['weather'][0]['description']}")
73         print(f"Temperature: {weather_data['main']['temp']}°C")
74         print(f"Suggested Dress: {dress_suggestion}")
75     except Exception as e:
76         print(f"An error occurred: {e}")
77
78 if __name__ == "__main__":
79     main()
80
```

# Conclusion:

The Virtual Wardrobe and Shopping Assistant app is a comprehensive solution for modern fashion management. By integrating AI, AR, and a robust database, it addresses common challenges like wardrobe organization, outfit selection, and staying updated with trends. The app's modular design ensures scalability and adaptability, making it a valuable tool for users seeking a personalized and interactive fashion experience. The project demonstrates the potential of combining cutting-edge technologies to solve real-world problems, paving the way for future innovations in the fashion industry.