West Visayas State University
COLLEGE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY
La Paz, Iloilo City, Philippines

OMBRÉ: A MACHINE LEARNING-BASED FASHION MOBILE APPLICATION FOR CLASSIFYING APPAREL COMBINATIONS

An Undergraduate Thesis

Presented to the Faculty of the

College of Information and Communications Technology

West Visayas State University

La Paz, Iloilo City

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Information Technology

by

Benedict Paul E. Bueno

Jed A. Divinagracia

Quian Jay B. Lapating

Hebele S. Pacaco

Alyssa M. Ruiz

JUNE 2024

West Visayas State University
COLLEGE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY
La Paz, Iloilo City, Philippines

Approval Sheet

OMBRE: A MACHINE LEARNING-BASED FASHION MOBILE APPLICATION FOR CLASSIFYING APPAREL COMBINATIONS

An Undergraduate Thesis for the Degree

Bachelor of Science in Information Technology

by

Benedict Paul E. Bueno

Jed A. Divinagracia

Quian Jay B. Lapating

Hebele S. Pacaco

Alyssa M. Ruiz

RALPH VOLTAIRE J. DAYOT Panel	JANINE P. DEFANTE Panel
CHERYLL ANN FELIPRADA Panel	CHRISTIAN CADIZ Panel
VIJAY TANGUB Panel	JOEL T. DE CASTRO Adviser

Concurred:

FRANK I. ELIJORDE, Ph.D. Chair, Information Technology	MA. BETH S. CONCEPCION, DIT Dean, CICT

JUNE 2024

Acknowledgment

The researchers extend their sincere thanks to the individuals listed below, whose assistance and contributions have played a crucial role in the completion of this study:

The researchers are deeply grateful to their families for their unwavering love and support throughout the thesis process. They consistently stood by, offering assistance whenever it was required.

The researchers express heartfelt appreciation to Dr. Joel T. De Castro, their esteemed thesis adviser, for consistently dedicating his valuable time to consultations despite a hectic schedule. His guidance, expertise, encouragement, and patience have been instrumental, and the researchers owe much of their progress to his unwavering support.

The researchers extend their appreciation to Mr. Nikie Jo E. Deocampo, the co-adviser for their thesis, for generously sharing his time and providing valuable insights that have greatly benefited their work.

The researchers thank Mr. Regin Cabacas, their professor, for taking the time to provide valuable support

and encouragement amid his busy schedule, sharing expertise crucial to their work.

The researchers acknowledge Mr. Louie Steven Cadaguman and Mr. Evans B. Sansolis for contributing their valuable insights and ideas during the early phases of this study.

The researchers express their gratitude to Dr. Ma. Beth Concepcion and the entire faculty for their understanding, consideration, and support. Their objective evaluation of the studies presented is appreciated.

The researchers are also thankful to their BSIT-4A classmates for their unwavering support and encouragement throughout the study. Special gratitude for welcoming them into their homes during tough and challenging times.

The researchers extend their gratitude to West Visayas State University, their alma mater, for imparting quality and world-class education. The university played a crucial role in shaping them into globally competitive individuals.

The researchers give thanks to the Almighty Father for His grace, blessings, and mercy. This research study would not have come to fruition without His divine guidance. All glory be unto God!

This humble work is sincerely dedicated to all of them.

Benedict Paul E. Bueno
Jed A. Divinagracia
Quian Jay B. Lapating
Hebele S. Pacaco
Alyssa M. Ruiz

JUNE 2024

Divinagracia, Jed A.; Pacaco, Hebele S.; Ruiz, Alyssa M.; Lapating, Quian Jay A.; Bueno, Benedict Paul E.; "Ombre: A Machine Learning-based Fashion Mobile Application for Classifying Apparel Combinations". Unpublished Undergraduate Thesis, Bachelor of Science in Information Technology, West Visayas State University, Iloilo City, Philippines, 2024.

Abstract

Wardrobe range due to a lack of fashion sense is a relevant issue in the field of fashion. There is always a struggle to match clothes on certain occasions or find what clothes are suitable for an event. With this problem, the researchers proposed creating a mobile application to assist users in classifying combined apparel. The researchers utilized a Convolutional neural network to classify user outfits according to events to predefined events, namely: urban adventure, active adventure, business formal, and business casual. The application has a built-in camera, and the user can also upload. Furthermore, the application also implemented a user preference module. The module was used to complete the process of the recommendation feature. The mobile application will provide recommendations based on the event, color, type, and pattern. The evaluation of the suggested application was conducted, and it obtained an 81.5% approval through the

ISO 25010:2023 assessment tool by IT experts. Furthermore, the TAM analysis yielded a comprehensive 90.6% approval, with PU, PEU, and BI attaining a "very satisfactory" rating. This indicates a notable level of excitement and interest in adopting our application, which utilizes machine learning for apparel classification and provides personalized recommendations.

Table of Contents

List of Figures

List of Tables

List of Appendices

CHAPTER 1 INTRODUCTION TO THE STUDY

Background of the Study and Theoretical Framework

Fashion, a vast and ever-evolving aspect of human adaptability, was originally designed to enhance the survivability rate of human beings by providing protection and warmth (Kawamura, 2004). Over time, it has evolved into a significant form of self-expression and cultural identity (Crane, 2000). The differences in geographical locations, cultural practices, standards, race, and historical events have contributed to the complexity and diversity of the fashion industry today (Eckert & McConnell-Ginet, 2003).

Despite its broadness, studies have shown that people tend to dress themselves according to what is more timely, relevant, and socially acceptable. However, some people, especially those who lack fashion sense, need help finding suitable or classifying outfits for specific occasions.

Regardless of the advancements in fashion recommendation systems, many existing approaches fall short in providing effective solutions for users seeking

suitable attire for specific events while also utilizing the clothes they possess inside their wardrobe. Current methods, such as traditional wardrobe planners and rule-based recommendation systems, often rely on limited algorithms that do not account for the complexity of individual preferences or the dynamic nature of fashion trends. These systems typically fail to incorporate user-specific data, leading to generic recommendations that may not resonate with users' unique styles or the context of their clothing choices. Moreover, many existing applications do not effectively leverage machine learning techniques to analyze and categorize outfits but rather provide a predetermined solution that may not resonate with the user’s current needs. The reliance on static criteria, such as predefined outfit combinations or basic categorization, overlooks the nuanced aspects of personal style and situational appropriateness. This gap in the current methods underscores the necessity for the development of "Ombré," a machine learning-based application that utilizes advanced algorithms to provide personalized recommendations tailored to individual user

preferences and the specific context of events. By integrating deep learning techniques and a diverse dataset, "Ombré" aims to enhance the outfit selection process, ensuring users can confidently choose suitable attire while embracing their unique fashion identity.

To effectively address the challenges in wardrobe selection, various technologies and related systems can serve as a foundation for the proposed "Ombré" application. Key technologies include machine learning algorithms, particularly convolutional neural networks (CNNs), which are essential for image classification tasks. CNNs excel in recognizing patterns and features within images, making them ideal for analyzing fashion items and determining their suitability for specific events. The user preference module is used to analyze user input and provide contextually relevant recommendations based on fashion trends derived from social media and online platforms. Existing systems, such as recommendation engines used in e-commerce, leverage collaborative filtering and content-based filtering methods to suggest products to users based on their preferences and

behaviors. However, these systems often lack the personalized touch required for fashion, as they may not consider the nuanced aspects of outfit combinations or contextual appropriateness for specific events. By combining the strengths of CNNs with a comprehensive dataset that reflects current fashion trends, "Ombré" can offer a more effective and tailored solution to help users make informed clothing choices, ultimately enhancing their fashion sense and reducing the stress of outfit selection. We can add our online wardrobe function as an asset as well. Which already grouped the outfits based on their category when it comes to relevant events of their usage.

According to a poll conducted by Marks & Spencer's Shopping campaign, 62% of women suffer from 'wardrobe rage, irrational tantrums about not having anything they want to wear, while men spend 13 minutes per day, or the equivalent of three days per year, selecting what to wear (Cliff, 2016). Furthermore, a simple survey among random teenagers and young adults was conducted by the researchers. It revealed that out of 56 respondents, 73% struggle to match their clothes on certain occasions or to

find what clothes are suitable for an event. The causes of the problem include trouble identifying which color goes with which, a lack of fashion sense, and outdated outfits and styles. "Ombré" is a fashion machine learning-based application designed to assist people with such problems.

The word "Ombré", according to Merriam-Webster's dictionary, means having hues or tones that blend into one another. Users no longer have to decide on their own, guessing if their apparel is suitable for a particular event or not. By inputting the image of combined clothes, the proposed application will determine its fashion suitability classification for the users. The fashion machine learning-based application, also offers personalized recommendations based on user-provided images. Furthermore, if the apparel is not considered suitable for a certain event, the application provides a relevant recommendation using data collected from the internet.

In the proposed study, the researchers used this concept to classify clothes based on their suitability for specific activities. The proposed application will serve

as a virtual space for users to determine if their apparel matches certain activities.

The compatibility outcome of each outfit is determined by its relevance according to the collected datasets. "Ombré" uses a convolutional neural network (CNN) for the classification feature, along with Ombré Dataset, as a collected and combined dataset for recognizing and classifying via images provided by the user. The Ombré Dataset, consisting of 3,807 images, was meticulously created by combining selected amount of data images from the Deep Fashion Datasets and numerous data images gathered from various popular clothing brand websites. This approach ensured a diverse and comprehensive collection of apparel that reflects current fashion trends and styles. Each image was carefully labeled using Label Studio, an open-source data labeling and annotation tool that facilitates the accurate categorization of images. Predefined categories relevant to the application’s functionality, such as Business Casual, Business Formal, Active Adventure, and Urban

Adventure, formed the basis for labeling the apparel images.

This study addresses the common struggles people face in selecting suitable attire for various occasions. The mobile application simplifies the process, reducing the time and effort, and alleviating the stress of 'wardrobe rage.'

"Ombré" provides personalized recommendations, improving users' fashion sense and self-esteem, and promotes fashion advice by making it accessible to anyone with a smartphone, bridging the gap between fashion enthusiasts and individuals uncertain about their fashion choices.

Additionally, "Ombré" serves as an educational tool, helping users to learn color coordination, outfit compatibility, and fashion trends. Over time, users can learn what suits them best, enhancing their intuitive sense of style.

In response to these challenges, "Ombré" is not only designed to assist users in selecting appropriate outfits but also to educate them on fashion principles that

enhance their decision-making skills. By leveraging machine learning algorithms, the application analyzes user input and provides real-time feedback on outfit suitability, effectively minimizing the guesswork involved in daily dressing. Furthermore, the app's personalized recommendations draw from a vast dataset that includes diverse styles and trends, ensuring that users are exposed to a wide range of fashionable choices. This educational aspect of "Ombré" fosters greater awareness of color theory, outfit coordination, and contemporary fashion trends, empowering users to develop their unique style over time. As users interact with the app, they gain valuable insights into what constitutes a well-coordinated outfit, thereby increasing their confidence in making fashion choices. Ultimately, "Ombré" not only alleviates the stress associated with outfit selection but also serves as a platform for personal growth in fashion literacy, enabling individuals to express their identity more authentically through their clothing choices.

The experiences of the researchers served as a crucial foundation for creating the initial concept of

"Ombré." Drawing from their diverse academic and student backgrounds, the team identified a recurring challenge in the fashion industry: the difficulty many individuals face in selecting appropriate outfits for various occasions. Personal encounters with wardrobe frustrations, combined with user feedback gathered through surveys, informed the researchers' understanding of the specific needs and pain points that the application aims to address. Moreover, the other research of previous work in machine learning and data analysis provided valuable insights into the potential of technology to revolutionize fashion choices. This blend of personal experiences and technical knowledge enabled the team to conceptualize "Ombré" as a user-friendly solution that leverages advanced algorithms to enhance the outfit selection process, ultimately empowering users to navigate their wardrobes with confidence.

Objectives of the Study

This study aimed to develop a mobile application for assisting users in classifying combined apparel whether it is suitable or not for a specific event.

Specifically, the study aimed to:

1. utilize Convolutional neural network to classify user outfits according to predefined events namely: urban adventure, active adventure, business formal, and business casual;
2. implement a user preference module for Outfit recommendation;
3. provide recommendations for fashion outfits based on the event, color, type, and pattern;and
4. evaluate the efficiency and performance based on ISO/IEC 25010 Standard Software Evaluation tool.

Significance of the Study

The outcomes of this study may be beneficial to the following:

For teenagers and young adults, the proposed application may help them enhance their fashion sense, reduce wardrobe rage, and further understand fashion.

For the researchers, the proposed mobile application will enhance their understanding and navigating the IT world. Additionally, it may support their future endeavors in research and software developments by providing practical experience and insights into integrating machine learning with real-world applications particularly in the fashion industry.

For fashion experts, the proposed mobile application offers a huge benefit: it combines technology with style. By using smart tools and fashion data, it helps understand people's preferences. This means easier outfit choices, better fashion trend predictions, and improved

technological skills in the fashion world, keeping experts at the cutting edge of the industry.

Finally, for future researchers, the result of the study may provide further knowledge and new methods for executing future research and app development. Furthermore, it may also encourage them to pursue similar studies.

Definition of Terms

For better understanding, the following terms were defined conceptually and operationally:

Convolution Neural Network -- is a deep learning model specialized in analyzing structured data arrays like images. CNNs are predominant in computer vision tasks, particularly in image and text classification (Wood, 2020).

In this study, Convolution Neural Network (CNN) served as an automated classifier of the apparel in the Ombré app. A labeled dataset, categorized as Business Formal, Business Casual, Adventure Urban, and Adventure Active (created using Label Studio), provides the foundation for CNN training. Through convolutional layers, the CNN extracts informative features like patterns and textures, enabling it to differentiate categories (e.g., suit vs. athletic wear). Fully connected layers then interpret these features and map them to the appropriate labels (Formal, Casual, etc.). Training refines this process, allowing the CNN to learn the relationships between

features and labels.

DeepFashion database -- is a massive collection of clothing images (over 800,000) specifically designed for fashion analysis by AI systems. Each image is richly annotated with detailed descriptions, key feature locations, and even variations of the same clothing captured in different settings. This comprehensive dataset helps train AI models to understand and classify clothing across various situations (Liu et al., 2016).
In this study,the mobile application's clothing recommendation feature leverages high-resolution images from the DeepFashion dataset.

Firestore -- a document-oriented database storing key-value pairs. Optimized for small documents and is easy to use with mobile applications (Perry, 2021).
In this study, the Firestore referred to the database used for storing data on the recommendation feature of the application.

Ombré -- is the blending of one color hue to another, usually tints and shades from light to dark (Crigger, n.d.).

In this study, Ombré referred to the name chosen for the mobile application.

Outfit -- a set of clothes for a particular event or activity. (The Cambridge Dictionary, n.d.).

In this study, the user's outfit was classified according to the occasion for which it is appropriate, and recommended outfits will be available for the user to view.

Delimitation of the Study

The research is focused mainly on the development of a machine learning-based fashion application that will assist users in classifying whether an apparel is suitable or not based on the specific events and relevancy of the collected datasets. In addition, the application has limited offline functionality. Classifying apparel, and saved images in both the events screen and recommendation history works fine offline. However, some features require an internet connection, including saved outfits (Fig. 10(b)), clothes recommendations (Fig. 11), and saved recommended outfits (Fig. 12). Classified outfits saved by the user are stored on Firebase. Additionally, apparel images displayed in the clothes recommendation feature (Business Formal, Business Casual, etc.) are fetched from Firestore and depend on the user's chosen preferences.

The target users of this study are teenagers and young adults. Also, this study is focusing on mobile applications with Android OS users only.

The study concentrated on specific event categories

which includes two main clothing categories: Business attire and Travel Adventure attire. Business attire caters to professional environments, offering options for both casual and formal settings. Business casual outfits prioritize comfort for everyday wear, with trousers and collared shirts being common examples. Business formal attire, on the other hand, emphasizes a polished and authoritative look, making it ideal for important client meetings and presentations. Suits, ties, and formal dresses are staples in this category.

The study also delves into travel adventure attire, which encompasses two distinct styles for different types of exploration. Active adventure outfits prioritize durability and functionality for rugged outdoor activities typically joggers, tank tops, and t-shirts for optimal comfort and performance in challenging environments. Conversely, urban adventure attire focuses on style and comfort for city exploration. Comfortable jeans and printed tees are perfect for activities like museum visits, shopping, and urban parks.

Regarding user preferences for the recommendation

feature, the proposed study's clothing recommendation feature tailors suggestions based on user preferences for color, pattern, and type. Business attire, categorized as casual, formal, or travel business, adheres to a more limited palette. Colors typically consist of navy, white, black, and gray, while patterns are restricted to solids and stripes. Women's options include skirts, trousers, blazers, dresses, and blouses. Men have choices like suits, polos, trousers, long-sleeve shirts, and chinos.

In contrast, travel adventure attire offers a much broader spectrum for user preferences. Categorized further into active, urban, and water adventure, it embraces a wider range of colors including greens, browns, and a broader spectrum of grays. Patterns expand beyond the traditional solids and stripes, even incorporating camouflage. For both men and women in active adventure, options include shorts, tank tops, t-shirts, active wear jackets, and joggers. Urban exploration leans towards a more casual style with printed tees, jeans, versatile jackets, and shorts. Water adventure specifically focuses on functional clothing like board shorts, rash guards, and

bikinis.

The images gathered for the recommendation system are meticulously curated from the DeepFashion database and clothing brand websites to visually represent clothing options. For each complete user selection (event > subcategory > clothing type > color > pattern), the system displays 3-5 recommendations.

The study did not cover the usage of any other input images that contain accessories such as those for the head, hand, neck, feet, etc. Personal attributes of the users, such as size estimation, skin color, height, weight, body structure, and hairstyle, are not considered variables in the research. Moreover, sharing and exporting data, buying and selling apparel, any online transaction, and creating a combination of clothes are not allowed and are not supported by the application.

Ombré fashion assistant is an envisioned personal application that determines the clothes' fashion suitability classification for the users. The application can provide suggestions retrieved from the database using Firestore in Firebase with images gathered from the

internet and Deepfashion as datasets. The recommendations that the application will provide are only limited to what is inside the used datasets.

CHAPTER 2 REVIEW OF RELATED STUDIES

Review of Existing and Related Studies

Current Systems

Several existing systems demonstrate the application of machine learning in fashion classification. In recent years, technology has advanced and evolved. Some of the technological advancements that exist in this era are continuously proving the effective use of Machine Learning in multiple fields. In the field of fashion, Machine Learning has made its way in solving fashion dilemmas. Our proposed study aims to tackle one of the key challenges in the field: classifying outfits based on their suitability for different events.

One of the more recent and effective methods used to classify outfits involves deep learning algorithms. Wu et al. (2019) employed a comprehensive dataset of fashion outfit images featuring a wide array of clothing items and materials. Utilizing a pre-trained convolutional neural network (CNN) algorithm, the researchers preprocess the images and extract features, while a bidirectional

recurrent neural network (RNN) is used to model temporal dependencies in the data. Through a multi-task learning approach, this model effectively optimizes both classification and material recognition tasks, showcasing the potential of deep learning techniques in fashion analysis.

This research is particularly significant for the proposed application "Ombré," which also focuses on outfit classification but specifically targets the suitability of these outfits for different events. While Wu et al.'s study demonstrates the effectiveness of using a pre-trained CNN for feature extraction, "Ombré" can benefit from these insights by adopting similar techniques to enhance its classification accuracy. Furthermore, the multi-task learning approach explored in Wu et al.'s research suggests that integrating multiple objectives—such as classifying outfits by event and providing style recommendations—could improve "Ombré's" functionality. By learning from Wu et al.’s findings, "Ombré" can refine its model to not only classify outfits effectively but also provide personalized

recommendations, ultimately improving user satisfaction and engagement in fashion decision-making.

The proposed study, "Ombré," leverages a convolutional neural network (CNN) to classify outfit suitability for specific events and provide personalized recommendations. Compared to the work of Wu et al. (2019), which employed a pre-trained CNN combined with a bidirectional RNN for fashion classification and material recognition, "Ombré" focuses specifically on event-based outfit suitability rather than broader fashion classification. The specialized dataset in "Ombré" is curated to match specific event categories, distinguishing it from the more general dataset used by Wu et al. The deep learning and multi-task learning techniques demonstrated by Wu et al. significantly inform "Ombré's" development, particularly in optimizing CNN for fashion classification. Similarly, Garg et al. (2021) utilized CNNs for fashion image classification using a large dataset from DeepFashion. While Garg's system focused on categorizing clothing based on type, color, and pattern, "Ombré" extends these capabilities by integrating personalized recommendations

tailored to specific events. These works provide technical insights that guide the optimization of "Ombré," allowing it to improve classification accuracy and user experience through targeted outfit recommendations.

Thompson et al. (2023) examined the impact of social media influencers on consumer fashion choices and how machine learning can be utilized to analyze this phenomenon. The researchers collected data from various social media platforms to assess the relationship between influencer activity, user engagement, and subsequent fashion purchasing behavior. Utilizing natural language processing (NLP) techniques, the study identifies key themes in influencer posts that resonate with audiences, revealing how style, aesthetics, and personal branding contribute to fashion trends. The findings suggest that social media significantly shapes consumer preferences and that targeted marketing strategies can enhance engagement.

Thompson et al. (2023) provided valuable insights for "Ombré" by highlighting the role of social media in

contemporary fashion decision-making. While "Ombré" focuses on outfit classification and event suitability, incorporating insights from this study could broaden its scope to include recommendations influenced by current social media trends. By integrating data on popular styles and trends emerging from social media platforms, "Ombré" could enhance its recommendation engine, ensuring that users are not only advised on appropriate outfits for specific occasions but are also aligned with trending styles. This integration would position "Ombré" as a relevant tool in an ever-evolving fashion landscape, leveraging social media dynamics to enhance user engagement and satisfaction.

Garg et al. (2021) proposed a fashion image classification system that uses convolutional neural networks (CNNs) to categorize fashion images by clothing type, color, and pattern. This system, trained on the DeepFashion dataset, explores multiple CNN architectures and evaluates them based on metrics like accuracy, precision, and recall. In contrast, the proposed study "Ombré" goes beyond general fashion classification by

focusing specifically on the suitability of outfits for predefined events. While both studies utilize CNNs for image analysis, "Ombré" incorporates personalized recommendations tailored to user preferences and event types, adding a layer of functionality not addressed in Garg et al.'s work. The significance of Garg et al.'s study lies in its technical contribution to the understanding of CNN-based image classification, which directly informs the development of "Ombré's" classification model. By applying the lessons learned from Garg et al.'s exploration of CNN architectures and hyperparameters, "Ombré" enhances its ability to provide accurate and context-specific outfit recommendations.

Garg et al. (2021) focused on fashion image classification using convolutional neural networks (CNNs) to categorize fashion images based on clothing type, color, and pattern, utilizing the DeepFashion database. While Garg et al.'s work emphasizes achieving high accuracy in image classification across broad categories, the proposed study "Ombré" has a more specific focus: evaluating the

suitability of outfits for particular events and offering personalized recommendations when an outfit is deemed unsuitable. Although both studies employ CNNs for image analysis, "Ombré" extends the functionality beyond basic classification by incorporating user context, such as event type and personal preferences, into the recommendation process.

The significance of Garg et al.'s work in the development of "Ombré" lies in its exploration of CNN architectures and the comparative analysis of hyperparameters, which inform the design and optimization of "Ombré's" classification model. By learning from the approaches used to fine-tune CNN performance in Garg et al.'s study, "Ombré" is able to enhance the accuracy and robustness of its outfit classification system. Additionally, the insights gained from dataset handling in Garg et al.'s study contribute to "Ombré's" ability to effectively manage and utilize its specialized dataset for event-based fashion recommendations, further improving the overall user experience.

Sundararajan (2019) explored various machine learning techniques, including supervised learning, unsupervised learning, and deep learning, providing a comprehensive overview of their applications in the fashion industry. This review highlights the diverse roles machine learning plays in fashion design, manufacturing, supply chain management, marketing, and e-commerce, particularly in areas such as customer segmentation, personalization, and recommendation systems. While Sundararajan's work offers foundational knowledge and broad insights, the proposed study "Ombré" specifically focuses on applying these machine learning techniques to classify outfits based on their suitability for specific events.

The importance of Sundararajan's study in the development of "Ombré" lies in its detailed examination of recommendation systems, which is a critical component of the proposed application. By laying the groundwork for understanding how machine learning can enhance personalization and improve user experiences in fashion, Sundararajan's research informs the technical

implementation of "Ombré's" recommendation features. Additionally, the comprehensive overview of machine learning applications provided by Sundararajan serves as a valuable resource for researchers and practitioners seeking to develop innovative, machine learning-based solutions in the fashion industry, guiding "Ombré" in its mission to bridge the gap between fashion choices and user preferences. Thus, Sundararajan's insights not only support the theoretical framework of "Ombré" but also enhance its practical applicability in addressing fashion dilemmas.

Kim et al. (2019) explored the application of machine learning methods in fashion data analysis, utilizing natural language processing (NLP) and machine learning algorithms to extract insights from various sources, such as social media, e-commerce websites, and fashion blogs. Their approach demonstrates the ability to identify and predict fashion trends through the use of deep learning technologies, specifically convolutional neural networks (CNN) and long short-term memory (LSTM) networks, as well as unsupervised learning techniques like clustering and

topic modeling. While D. Kim et al.'s work primarily focuses on textual data analysis, the insights gained from their findings are significant for the proposed study "Ombré," which aims to classify outfit suitability based on image data.

Unlike D. Kim et al., "Ombré" emphasizes visual classification rather than textual analysis. However, the techniques demonstrated in Kim et al.'s study underscore the importance of integrating multiple data types for a more holistic understanding of fashion trends. The ability of Kim et al.'s model to predict emerging trends can enhance "Ombré's" recommendation system by potentially incorporating textual insights alongside image data, allowing for a richer and more nuanced user experience. Additionally, their findings highlight the utility of machine learning as a powerful tool for understanding consumer behavior and preferences, further informing "Ombré's" development to better assist users in making informed fashion choices. In this way, Kim et al.'s work not only complements the objectives of "Ombré" but also offers valuable methodologies that could enhance

its functionality and effectiveness in addressing fashion dilemmas.

The study by Zhang et al. (2019) investigates the application of machine learning techniques for predicting the style and functionality of interior designs based on furniture configurations. By combining deep learning and graph theory, the authors analyze spatial relationships among furniture pieces, leveraging a dataset collected from Houzz that includes manually labeled style and functionality tags. They employ two models—a graph convolutional network (GCN) and a multi-task learning model—to predict these labels, with the multi-task model demonstrating superior performance. This study highlights the effectiveness of advanced machine learning methodologies in understanding design aesthetics and functionality, which can be pivotal for practitioners in interior design.

While Zhang et al.'s work focuses on interior design, the principles underlying their machine learning approaches can significantly inform the development of

the proposed study "Ombré." Both studies utilize deep learning models to achieve specific outcomes—in Zhang et al.'s case, predicting furniture arrangement suitability, and in "Ombré's," classifying outfit suitability for events. The multi-task learning model utilized by Zhang et al. offers insights into how "Ombré" might enhance its classification capabilities by potentially integrating multiple prediction tasks, such as evaluating color coordination alongside outfit functionality. Furthermore, the methodology of analyzing labeled datasets from specific domains parallels "Ombré's" strategy of curating a specialized dataset for fashion recommendations. Ultimately, Zhang et al.'s findings not only contribute valuable techniques for achieving high accuracy in classification tasks but also highlight the broader applicability of machine learning in addressing user needs across various design domains, thereby enriching the development of "Ombré" and its goal of improving fashion decision-making for users.

The study by R. Kin et al. (2019) presents a fashion outfit recommendation system that employs deep learning

techniques to suggest outfits tailored to users' preferences, past purchases, and current fashion trends. By utilizing a dataset of fashion images and text descriptions from the Polyvore dataset, the authors train a model that combines convolutional and recurrent neural networks, allowing it to learn both visual and textual features of fashion items. Their approach has shown significant success, outperforming other state-of-the-art methods in terms of accuracy and diversity of recommendations.

While the goals of Kin et al.'s study and the proposed application "Ombré" align in their focus on personalized fashion recommendations, their methodologies differ in scope. "Ombré" specifically emphasizes classifying outfits based on their suitability for specific events, offering a unique angle that complements the broader personalization strategy of Kin et al. Both studies label datasets with specific features that guide recommendations; however, "Ombré" incorporates event-based categorization, enhancing its relevance to users facing wardrobe decisions for particular occasions.

The significance of Kin et al.'s work in the development of "Ombré" lies in its validation of deep learning techniques for fashion recommendations and its insights into user preferences. By demonstrating the effectiveness of combining visual and textual data, Kin et al.'s study informs "Ombré" on how to effectively curate and utilize its own dataset for improved recommendation accuracy. Additionally, the application of machine learning in diverse aspects of fashion, as highlighted by Kin et al., supports "Ombré's" ambition to refine user experience through intelligent outfit suggestions, ultimately contributing to its potential impact in the fashion industry.

Hue et al. (2020) explored the role of recommender systems in enhancing user experiences within the travel and tourism industry, emphasizing the provision of personalized recommendations based on user preferences and contextual data. The authors categorize various recommender systems, such as collaborative filtering, content-based filtering, and hybrid models, while also addressing the potential applications and challenges

associated with these systems. They concluded that effective recommender systems can significantly improve user experiences by tailoring suggestions for travel itineraries, accommodations, and activities based on individual preferences.

While Hue et al.'s study primarily focused on the travel and tourism sector, its findings bear strong relevance to the proposed application "Ombré." Both studies aim to assess user preferences to deliver tailored recommendations, albeit in different domains—Hue et al. within travel and "Ombré" within fashion. The insights gained from Hue et al.'s exploration of recommender systems can inform "Ombré's" development by highlighting the importance of user context and preferences, thereby enhancing its ability to provide personalized outfit suggestions.

Furthermore, the challenges identified by Hue et al. (2020), such as data privacy and the need for standardized formats, are also pertinent to "Ombré." Addressing these challenges will be crucial for ensuring

user trust and optimizing the system's functionality. Ultimately, Hue et al.'s work underscores the significance of personalized recommendation systems in improving user engagement, offering valuable methodologies and insights that "Ombré" can leverage to enhance the overall user experience in fashion decision-making.

Lee et al. (2021) explored the application of machine learning techniques for fashion trend forecasting. This study utilizes a combination of historical fashion data, social media trends, and sales data to predict upcoming fashion trends. The researchers employ various machine learning algorithms, including support vector machines (SVM) and random forests, to analyze the data and identify patterns indicative of future trends. The findings indicate that incorporating diverse data sources significantly improves the accuracy of trend predictions.

Lee et al.’s study is particularly relevant to the proposed application "Ombré," as both focus on enhancing

user decision-making in fashion. While "Ombré" aims to classify outfits for specific events and provide personalized recommendations, the insights from Lee et al. regarding trend forecasting can enrich "Ombré's" capabilities. By integrating trend predictions into the recommendation system, "Ombré" can offer users not only suitable outfit choices for particular occasions but also insights into current and upcoming fashion trends. This dual approach would not only enhance user satisfaction but also empower users to make informed fashion choices that align with evolving trends, ultimately bridging the gap between individual style and broader market movements.

Smith et al. (2022) investigated AI-driven personal styling applications that utilize computer vision and machine learning to provide users with personalized outfit suggestions based on their unique body types, skin tones, and fashion preferences. The study analyzes various applications currently available in the market, assessing their effectiveness in delivering tailored styling advice and recommendations. The researchers find

that successful applications often combine visual data analysis with user input to enhance the accuracy of their suggestions.

This study is significant for the development of "Ombré" as it underscores the importance of personalized user experience in fashion technology. While "Ombré" primarily focuses on classifying outfits for specific events, integrating elements from Smith et al.’s findings could enhance its functionality. By incorporating user-specific data, such as body type and color preferences, "Ombré" could further personalize its recommendations, ensuring that users not only receive suitable outfits for particular occasions but also those that flatter their individual attributes. Additionally, Smith et al.’s exploration of market trends in AI-driven styling applications can inform "Ombré's" design and feature set, positioning it as a competitive tool in the fashion technology landscape that responds to both user needs and industry standards.

CHAPTER 3 RESEARCH DESIGN AND METHODOLOGY

Description of the Proposed Study

The researchers proposed to develop a machine learning-based/image classification-based fashion outfit recommendation mobile application that classifies whether a matched outfit is suitable for a certain event or not.

The purpose of this study was to develop a means for teenagers and young adults to have a personal assistant in determining whether the outfit is suitable for a specific activity or not. In this study, researchers leverage a multi-task convolutional neural network (CNN) architecture to classify outfits as suitable or unsuitable for a certain event. By classifying the given apparel, the proposed application can improve the target user's fashion sense, minimize wardrobe rage, and reduce their time combining and identifying clothes for future activities.

During the development of this study, a variety of technologies were utilized by the researchers. Python and Kotlin served as the primary programming languages. TensorFlow, TensorFlow Lite, Keras and Sci-kit Learn

provided the necessary frameworks and libraries for development. Data processing was facilitated by tools like Label Studio, Pandas, and NumPy. JupyterLab and TensorBoard supported the researchers in both experimentation and evaluation. Finally, the hardware used consisted of a Smartphone with a Camera, developed using the Android IDE.

Methods and Proposed Enhancements

Sources of Information

Related Literature. The researchers researched and reviewed past and recent studies to gather sufficient information to conduct and further support the proposed study. The references included are research studies, and articles. The aforementioned related documents are mostly found online and are gathered through web browsing.

Repositories. The researchers utilized open-source codes mainly from online repositories like GitHub, as references to support the development of the proposed Android application.

Data. The proposed application generates recommendations using data collected from web browsing based on fashion suitability to an event. The Ombré Dataset is predominantly composed of images obtained from various fashion websites and web applications online. Moreover, comprehensive organization and annotation procedures and expert validation are implemented to improve the dataset's quality and reliability.

Experiences and Observations. The experiences of the researchers were the foundation for creating the initial concept. Subsequently, a simple survey with 50 random participants was conducted as a form of observation. Through the collective experience of the researchers and respondents, the researchers were able to conceptualize the idea which further led to the development of the proposed study.

Proposed Enhancements

Classification of outfit/apparel combinations whether they are suitable or not for a specific event (either Travel Adventure or Business). The main focus of this study is to classify whether a combination of apparel (top and bottom) is suitable or not with regard to an event chosen by the users.

Outfit recommendation. If the user's outfit is classified as unsuitable for the chosen event, the application will automatically suggest alternative recommendations. Users can also skip outfit classification and proceed directly to the recommendation feature. The

researchers carefully curated high-quality recommendation images from the DeepFashion database and clothing brand websites through web browsing.

Ombré *Dataset.* Ombré Dataset mainly consists of images collected from various fashion websites and webapps available in the internet. Also, the collected images were enhanced by organizing and annotating them thoroughly.

Machine Learning Model for outfit apparel classification. The proposed Android application utilizes a Convolutional Neural Network (CNN), a machine learning model specifically designed for image recognition, to classify outfits based on their suitability for specific events. This model was trained on the Ombré dataset, a meticulously curated collection of fashion images gathered from various websites and web apps.

Recommendation History. The application can record the recommendation results in a virtual space in which the user can review their recently saved outfits recommended by the application. Furthermore, with this feature, the user can assess the outfits and learn from them.

Components and Design

System Architecture

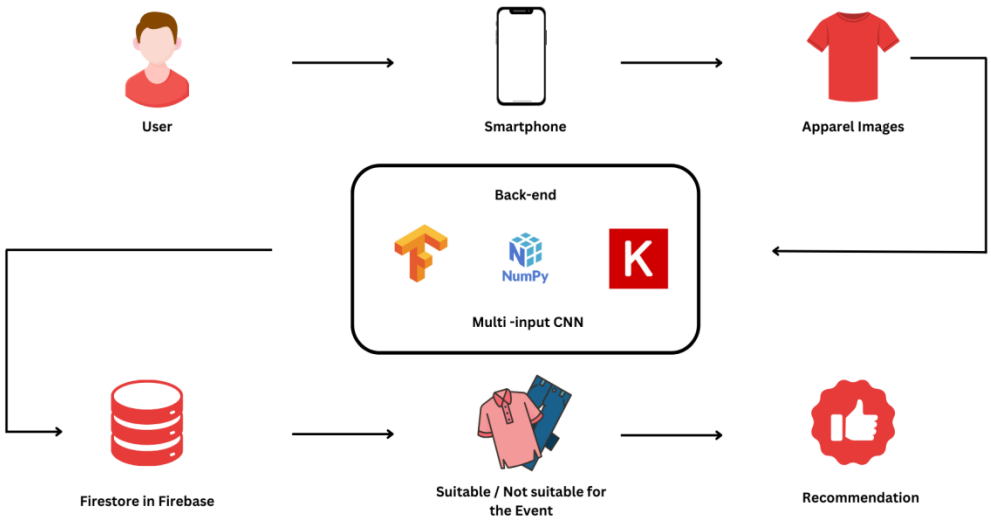


Figure 1: System Architecture

The researchers used the following components in the system architecture shown in Figure 1: Android Phone, which is critical for running the mobile application and interacting with the system. User input data, including apparel images and related information, is stored and retrieved using Firestore in Firebase, ensuring efficient data management. Trained ML models, notably Multi-Input CNNs, analyze imported outfit photos to determine their suitability for various events. Furthermore, a suggestion

mechanism uses data gathered from legitimate websites to provide individualized fashion recommendations, improving the user experience.

UML Class Diagram

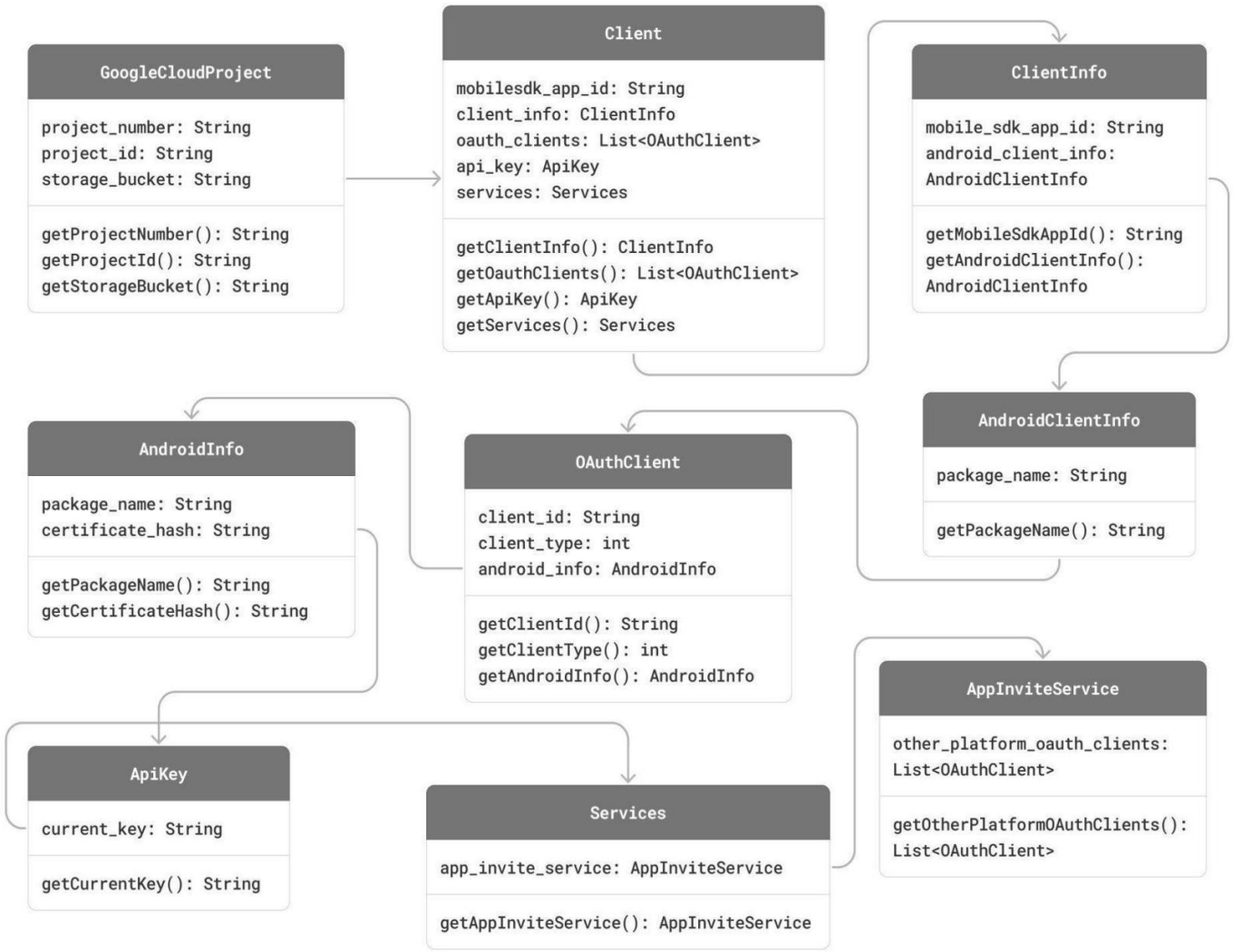


Figure 2: UML Class Diagram

Figure 2, depicts the class diagram for our firebase and ombre mobile client. Our database is a collection of tables, each representing different aspects of our Firebase and Ombre mobile client setup. The main table, called "Google Cloud Project," holds essential details about our project, similar to how a main file holds important information in a folder. The "Client" table is like a companion table, storing details specifically about our mobile client. Other tables like "ClientInfo," "OAuthClient," "AndroidClientInfo," and so on, hold more specific information, almost like folders within our main file, all connected through relationships to organize and store our data effectively. So, this diagram is like a blueprint that helps us understand how our data is structured and interconnected in our database system.

Process Design

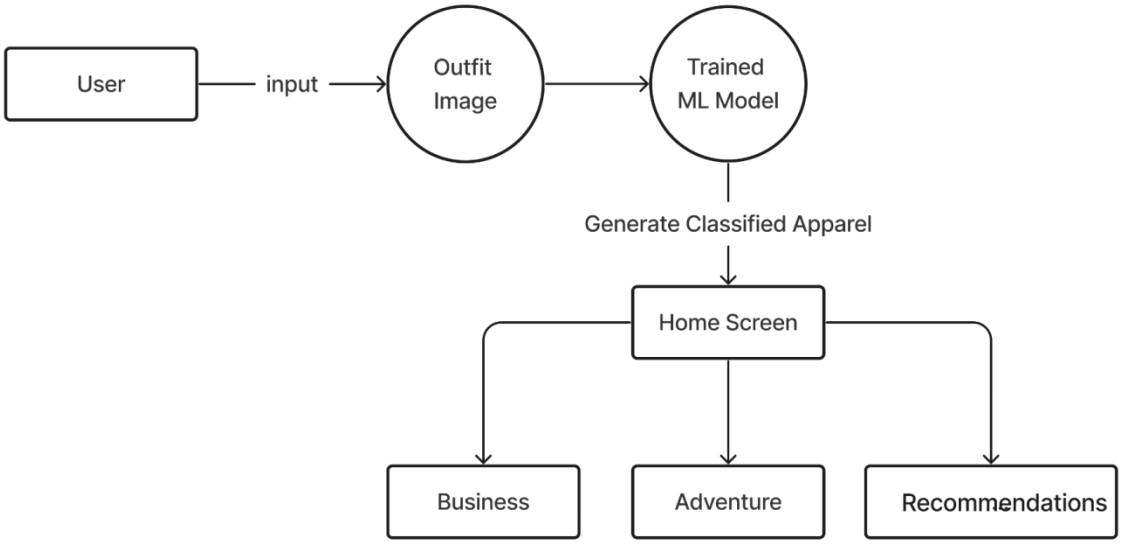


Figure 3: Process Design

To use the application, as shown in Figure 3, users must first choose an event and specify their gender. Next, there are two options for inputting their apparel: either by using the camera feature or by uploading an image from their gallery.

Once the outfit image is inputted, the classification process can be initiated by clicking the "Classify" button. Then, the application will try to classify the outfit by utilizing the Convolutional Neural Network(CNN)

algorithm which was used to train the Machine Learning model. The Classification of Outfits will determine whether the outfit is suitable or not for Adventure or Business event. If the outfit is suitable, the user can save it. The saved outfits will be reflected inside the Saved outfit screen where two events are separated to organize the images.

Recommendations will also be provided if the outfit of the user is unsuitable for the selected event. The recommended outfit can be saved to recommendation history screen as an additional feature to the proposed application

Procedural Design

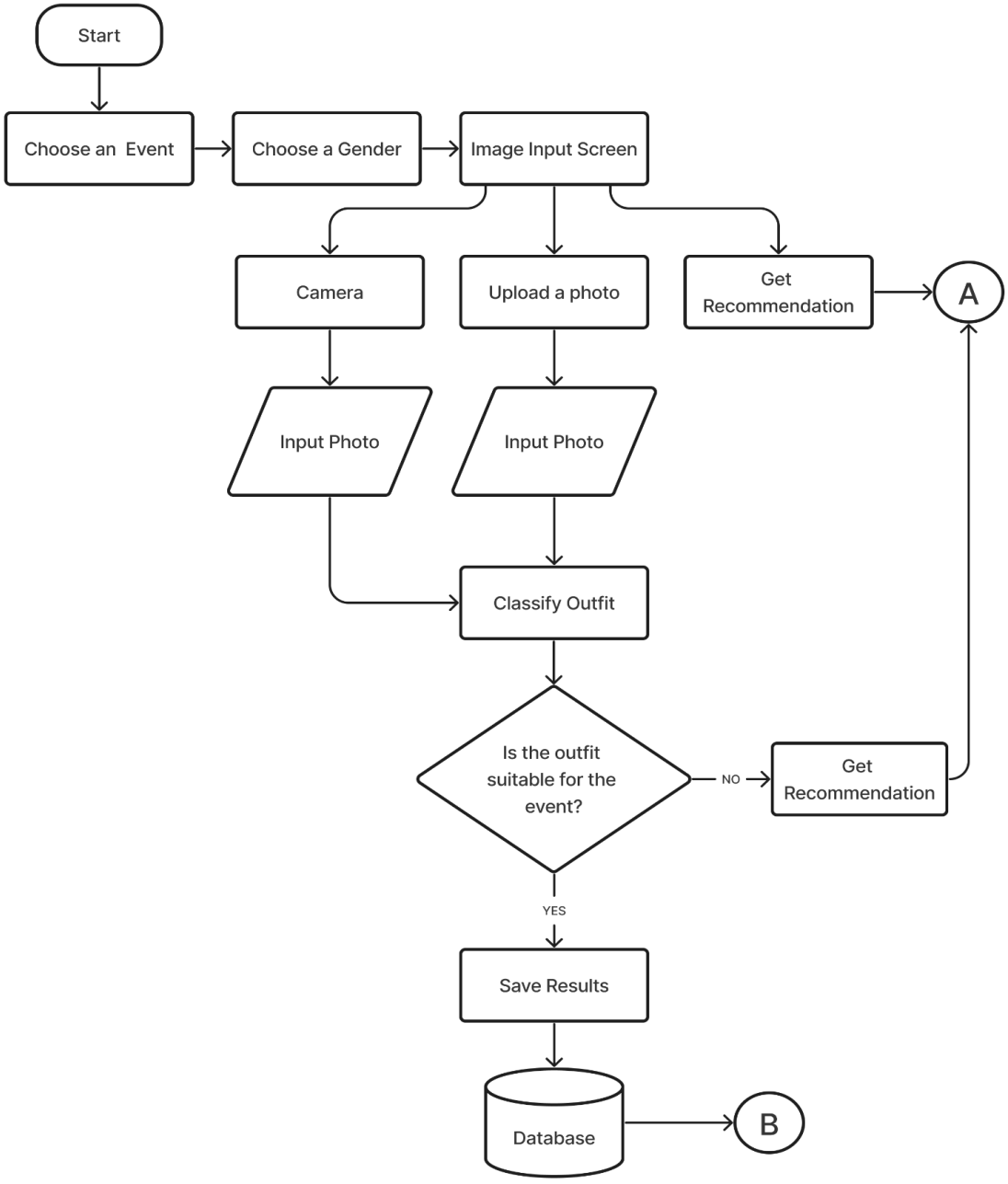


Figure 4(a): Procedural Design

As Shown in Figure 4(a), the procedural design of the proposed application is illustrated. To commence the process, users must first choose an event and gender. After that, they can click the start button. The start Button will then redirect them into the next screen where the users can proceed to input an image. There are two ways to input an image: through camera and manual upload of photo. The users can also get recommendations through the recommendation button, regardless if they uploaded a photo or not. Next, When the users are ready to classify their outfits, they can simply click the classify button to commence the classification process of their outfit. During this process, the Trained ML model will classify whether the inputted data is suitable for the available events, namely Business and Travel Adventure. After categorizing the outfit, the results will be displayed. If the outfit they inputted is suitable for the event they have selected in the beginning, the user has a choice to save or repeat the outfit classification process. However, if the outfit is not suitable for the event, the app will redirect them to the recommendation screen.

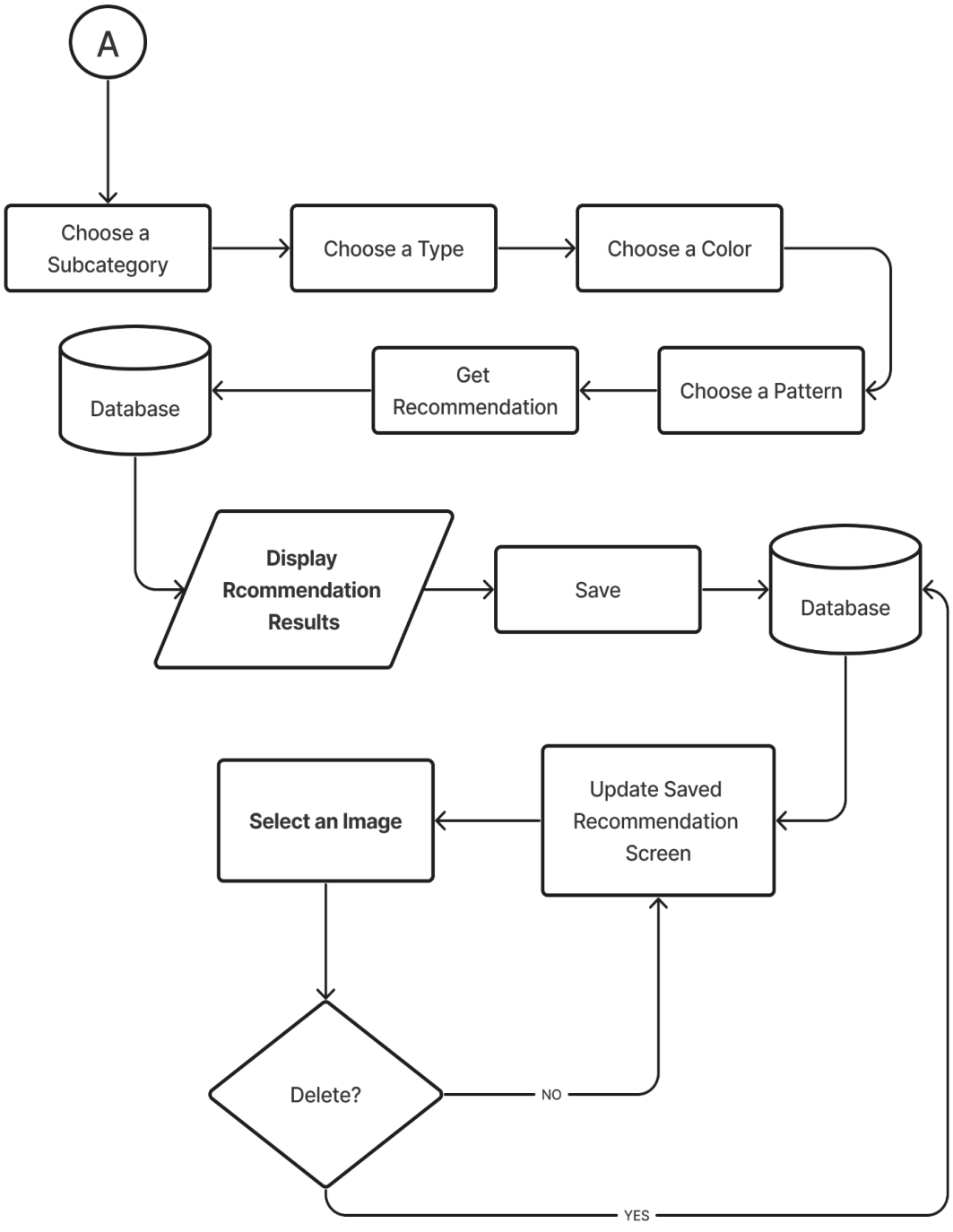


Figure 4(b): Procedural Design

Figure 4(b) displays the procedural design for the recommendation feature of the proposed application. After classifying the outfit into any of the available Categories (Business Formal, Business Casual, Urban Adventure, and Travel Adventure), the application will provide an additional option for the user to get recommendations. This feature requires completing and submitting users' preferences. This process will serve as a basis for retrieving the most relevant recommendations from the database to the users.

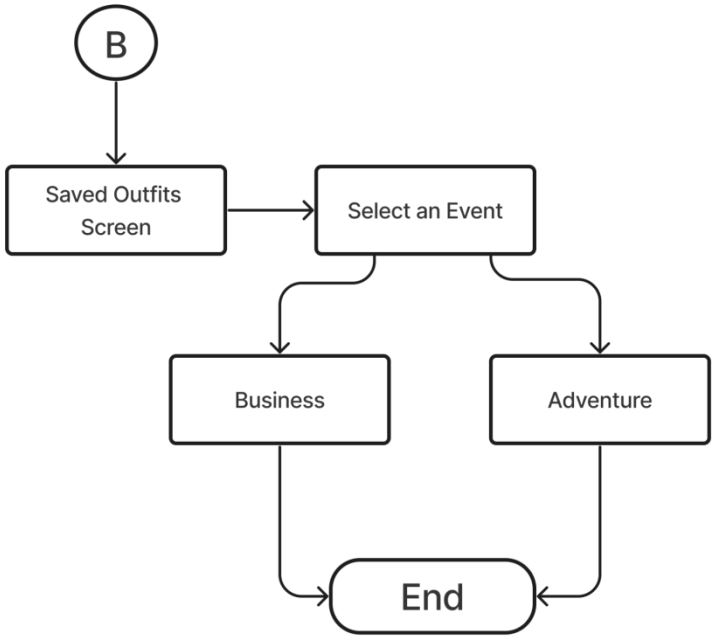


Figure 4(c): Procedural Design

Figure 4(c) displays the procedure after the classified outfit is saved in the database. The outfits saved in the database are reflected in the Saved Outfits screen. Here, the outfits are divided into two groups: Business and Adventure. Outfits for Business Casual and Business Formal are saved under the Business container, while Active Adventure and Urban Adventure are stored under the Adventure container.

Object-Oriented Design

In Figure 5, the user interactions with the application are shown. It shows the features that are accessible to users.

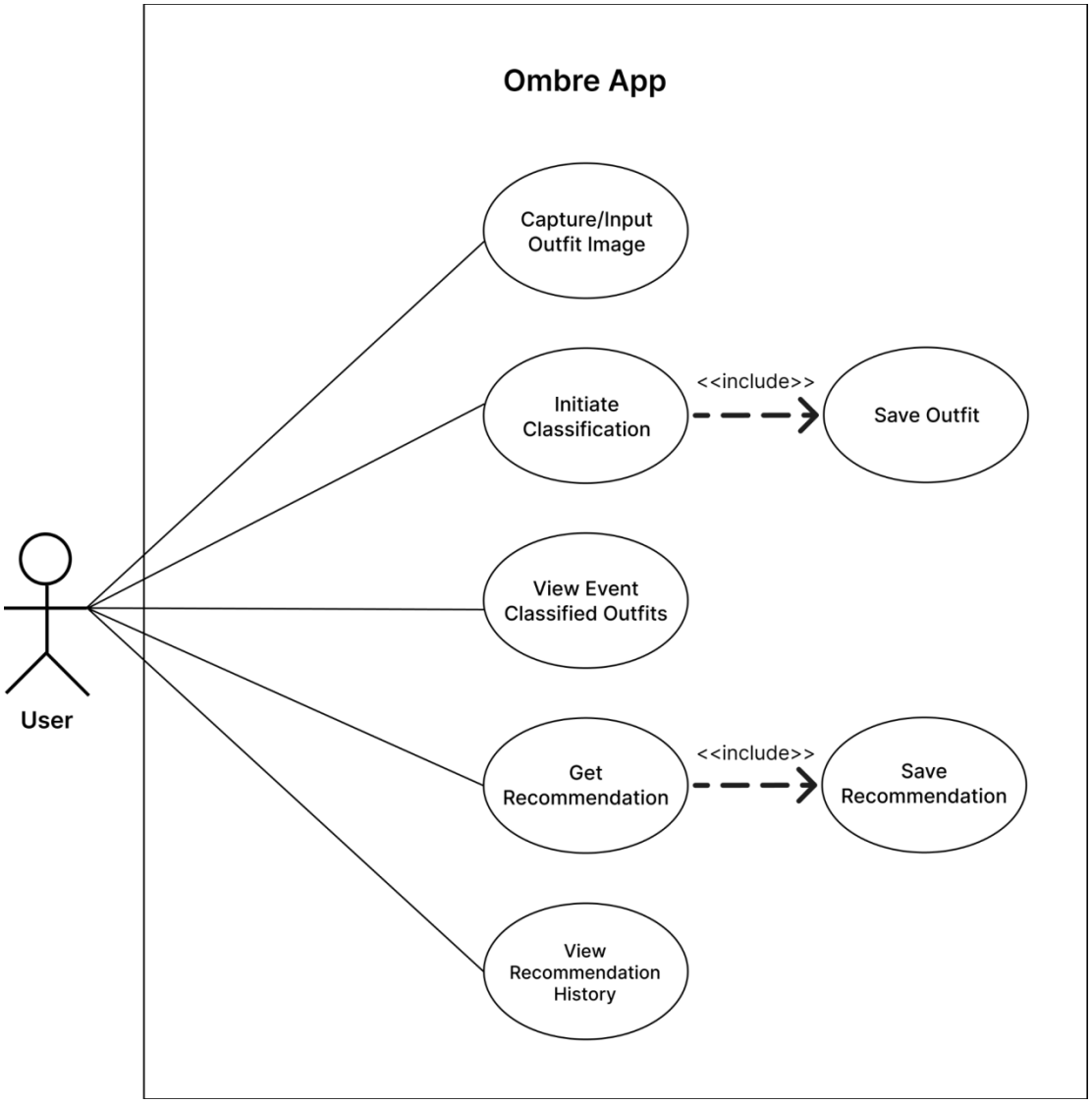


Figure 5: Use Case Diagram

System Development Life Cycle

The researchers used the SDLC's Agile Model as a guideline for creating the system and making logical choices throughout the entire research. Agile methodology is a project management approach where teams collaborate closely, adapt to changes quickly, and focus on delivering value efficiently. It prioritizes flexibility, teamwork, and continuous improvement over rigid planning. The Agile Model has seven (7) parts. (1) Planning, (2)Data gathering, (3) Requirements, (4) Design, (5) Programming, (6) Testing, and (7) Deployment.

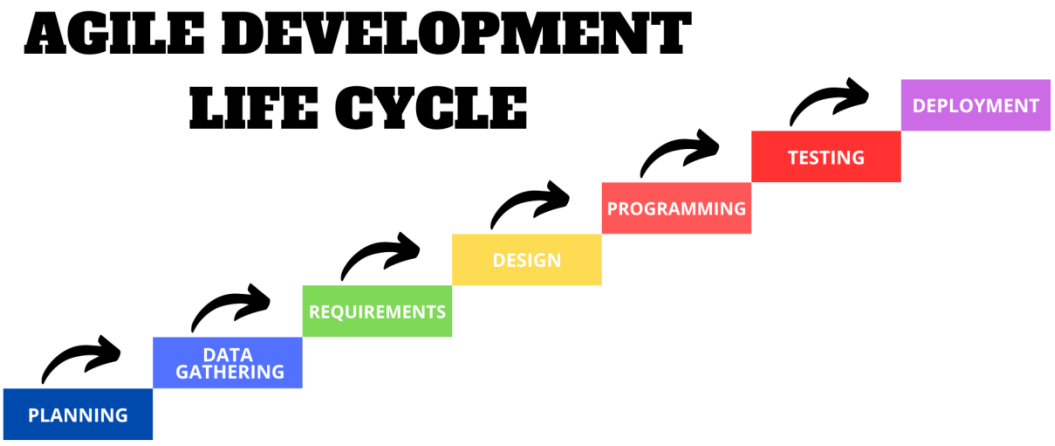


Figure 6: Agile Development Model

Agile effectively enables the researchers to comprehend the nature of the study and its objectives.

Planning. The researchers identified the problem of individuals struggling with creating stylish outfits and came up with the initiative to propose the "Ombre" mobile application to address this issue. The application leverages advanced machine learning techniques, specifically convolutional neural networks (CNNs), to analyze and classify outfit combinations for their suitability based on specific events. In addition to CNNs, the app utilizes Firestore, a NoSQL cloud database within Firebase, to store user data and apparel images securely. During the planning phase of our research activities, we explored various machine learning models and data management solutions to identify the most effective methods for providing users with personalized and contextually relevant outfit recommendations. This combination of CNNs for image classification and Firestore for data storage enables the app to deliver accurate assessments and streamline the user experience in selecting appropriate attire for various occasions.

Extensive research was conducted to understand the existing landscape and advancements in fashion classification. The researchers defined the scope and requirements of the Android application, focusing on generating innovative ideas and conceptualizing features for an intuitive user experience. Key features considered include a history section, virtual space for image inputs, filter options for specific occasions, a fashion diary, and a machine-learning model engine for classifying outfit fashionability.

Data Gathering. The data gathering process involved collecting two distinct image sets: one for training the machine learning model and another for populating the application's recommendation feature. For the training dataset, we gathered and labeled 800 images for each event category from clothing brand websites. 200 labeled images per event category are used for the validation set. Images used in the recommendation feature come from two sources: the DeepFashion database and various clothing brand websites. These images encompass a wide range of clothing options, including business attire (with subcategories for

casual, formal, and travel) and travel adventure attire (featuring active, urban, and water adventure categories).

Requirements. In the requirements phase, we outline the app's core functionalities, such as allowing users to input images of clothing combinations for fashion suitability classification. Additionally, we specify that the app should offer relevant recommendations based on internet-sourced data.

Design. In the design phase, the researchers created a prototype to visualize the user flow and interface of the application, gathering feedback to refine their design. They developed the system architecture, defined tools, and data requirements, and designed the UI/UX to create an intuitive and visually appealing experience. The researchers planned the database schema for efficient data storage and defined the multi-input CNN model for fashion rating prediction. APIs and data flows were defined to facilitate communication between components. The design phase established the foundation for the subsequent development, testing, and deployment of the Ombré fashion classification application.

Programming. During the programming phase, the researchers collected datasets from various websites and web apps on the internet to train machine learning models using Firestore in Firebase and CNN algorithms. They utilized a multi-input CNN model to predict fashion rates for apparel combinations. Simultaneously, they created an intuitive and user-friendly mobile app for the Android platform, incorporating essential features and functionalities. The app integrated the trained model ad databases, allowing users to receive real-time feedback on their apparel combinations. The researchers also implemented necessary APIs and data communication channels for seamless interaction between app components and external services or databases. Overall, they successfully transformed their design concepts into a functional Android application equipped with machine-learning models for fashion classification and recommendation.

Testing. During the testing phase, the researchers ensured the quality and effectiveness of the mobile app through various methods. They conducted unit testing to

validate individual components, integration testing to ensure proper communication between components, and system testing to evaluate overall functionality and performance. User acceptance testing involved target users providing feedback for further improvements. Any issues or bugs identified were debugged and resolved. The ISO/IEC 25010 Standard Software Assessment Tool was used for evaluation, and input from 9 participants and jurors added to its accuracy. This comprehensive testing phase played a vital role in refining the app's quality, functionality, and user experience.

Deployment. During the deployment phase, the researchers prepared the mobile app for release on app distribution platforms, focusing on the Google Play Store. They ensured compatibility, refined the user interface, and met submission guidelines. The app was then published and made available to the target audience. Performance monitoring allowed the researchers to gather valuable user feedback and assess the app's reception. Regular updates and maintenance were provided to address issues, incorporate new features, and enhance security and

performance. This iterative process aimed to deliver a high-quality, user-centric app that meets user needs and expectations over time.

CHAPTER 4 RESULTS AND DISCUSSION

Implementation

Technical Specifications

Minimum Requirements for End Users:

OS Compatibility: The mobile application is compatible with Android devices running Android 8.0 (Oreo) or higher. This corresponds to a minimum SDK version of 26.

Internet Connectivity: The proposed system is a mobile application that can be used offline. However, internet connectivity is required for two key features: saving classified images and accessing/saving Firestore-based recommendations.

Storage Space: End users should ensure that adequate storage space is available on their devices for uploaded images used for outfit classification. The mobile application itself has a modest footprint, with a total installation size of approximately 99 MB, including the APK (29 MB), installation overhead (50 MB), and initial data (20 MB).

Programming Language: The Ombré mobile application was primarily developed using Kotlin, a modern, statically typed programming language known for its conciseness and interoperability with Java (JetBrains, 2024).

Development Environment: The mobile application was built using Android Studio, Google's user-friendly development platform specifically tailored for creating applications for the Android operating system (Android Developers, 2024).

Android Studio offers a robust integrated development environment (IDE) that supports a wide range of programming languages and provides essential tools for coding, debugging, and testing. This platform facilitates a streamlined development process, enabling the researchers to efficiently design user interfaces and implement application logic. Additionally, Android Studio's built-in emulator allows for extensive testing across various device configurations, ensuring that "Ombré" operates smoothly on different Android devices. By leveraging this development environment, the team was able to focus on creating a

responsive and user-friendly application that meets the needs of its target audience.

The hardware and software specifications utilized in the development of the proposed study are outlined in Table 1 to Table 3.

Table 1

Hardware Specifications

Hardware	**Description**
Android Version	13 TP1A.220624.014
Processor	Dimensity 1080 Octa-core Max 2.6GHz
RAM	8.00 GB
Display	6.67” AMOLED - 1080 x 2400

Table 2

Software Specifications

Software	**Description**
Platform	Windows 10
Development Tool	Android Studio, Firebase, Kotlin, TensorFlow, Keras, Jupyter Notebook, Numpy, Pandas, Matplotlib, Github
Integrated Development Environment (IDE)	Android Studio, Jupyter Notebook, Github
Software Development Kit	Android 13.0

Table 3

Database Specifications

Database	**Description**
Firestore	Store, query, and manage user-uploaded images
Firebase Storage	Storage for outfit images

Systems Inputs and Outputs

The researchers developed a Convolutional Neural Network (CNN) model to classify outfit suitability for various events. Initially, the model focused on two categories: Travel Adventure (encompassing urban and active wear) and Business (including casual and formal attire). A thorough explanation of the model's inputs and outputs is provided below.

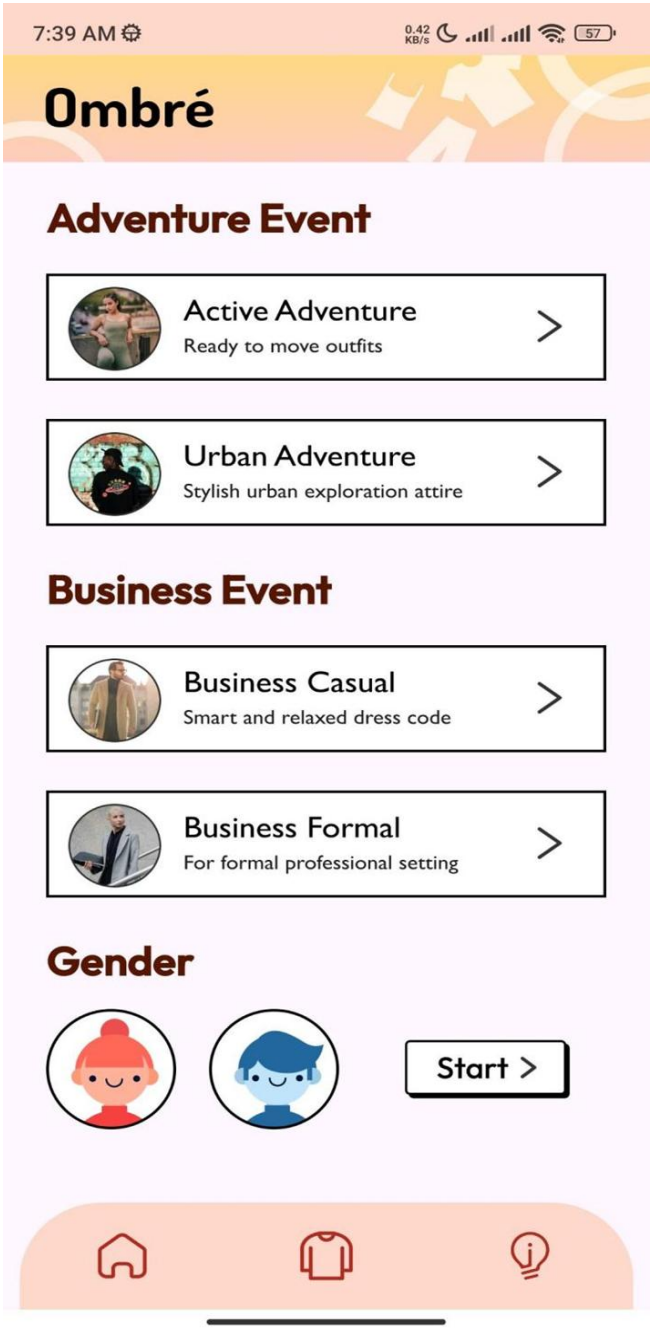


Figure 7: Main Screen

Main Screen

Figure 7(see above) shows the main screen of the mobile application, which allows users to select an event such as Business Formal, Business Casual, Adventure Active, and Adventure Urban before proceeding to input their outfit for classification.

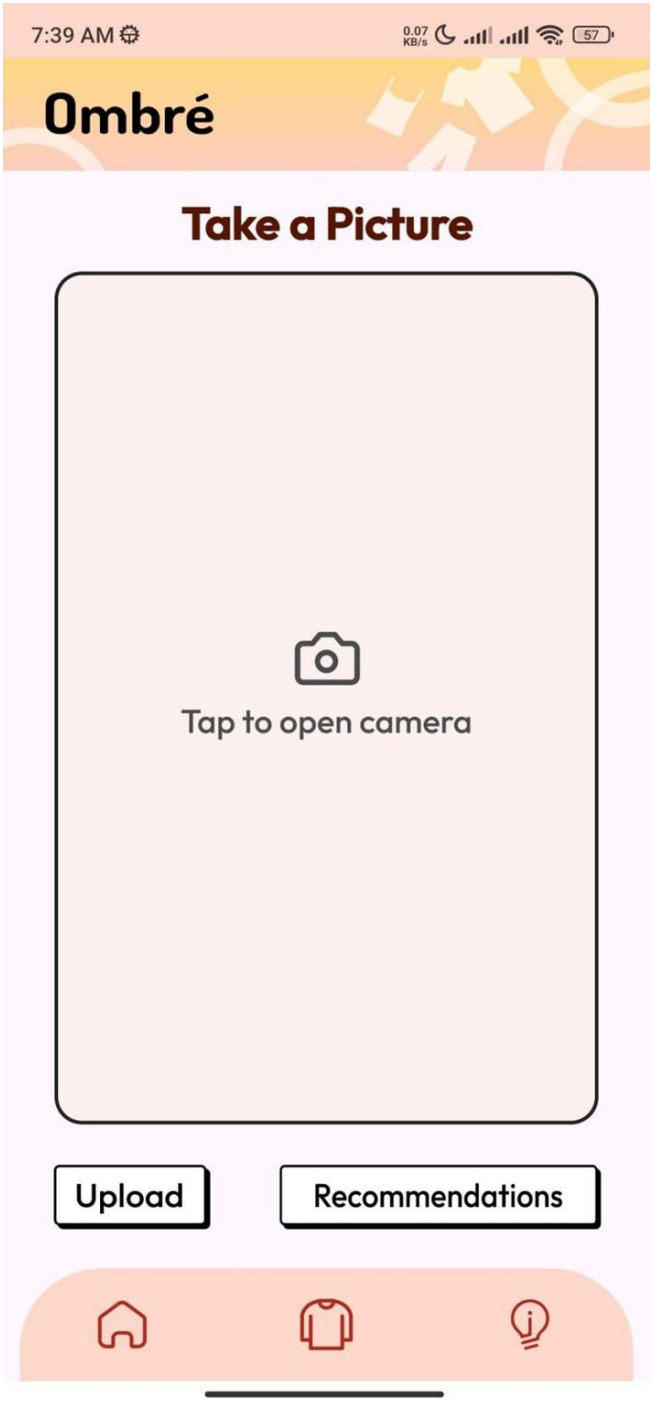


Figure 8: User Input

User Input

Following the selection of an event and gender on the main screen (Figure 7), users must click the "Start" button to proceed. This action directs them to the user input screen as shown in Figure 8, where they can upload an image of their outfit. Alternatively, users can bypass outfit classification by clicking the "Recommendation" button.

Figure 9(a): Clothes Classification

Figure 9(b): Clothes Classification

Clothes Classification

Figures 9(a) and 9(b) illustrate the outfit classification process. To assess the suitability of an outfit for a chosen event, users can upload a photograph of themselves wearing the desired attire. Clicking the "Classify" button initiates the analysis. Following classification, users can save the results using the "Save Results" button.

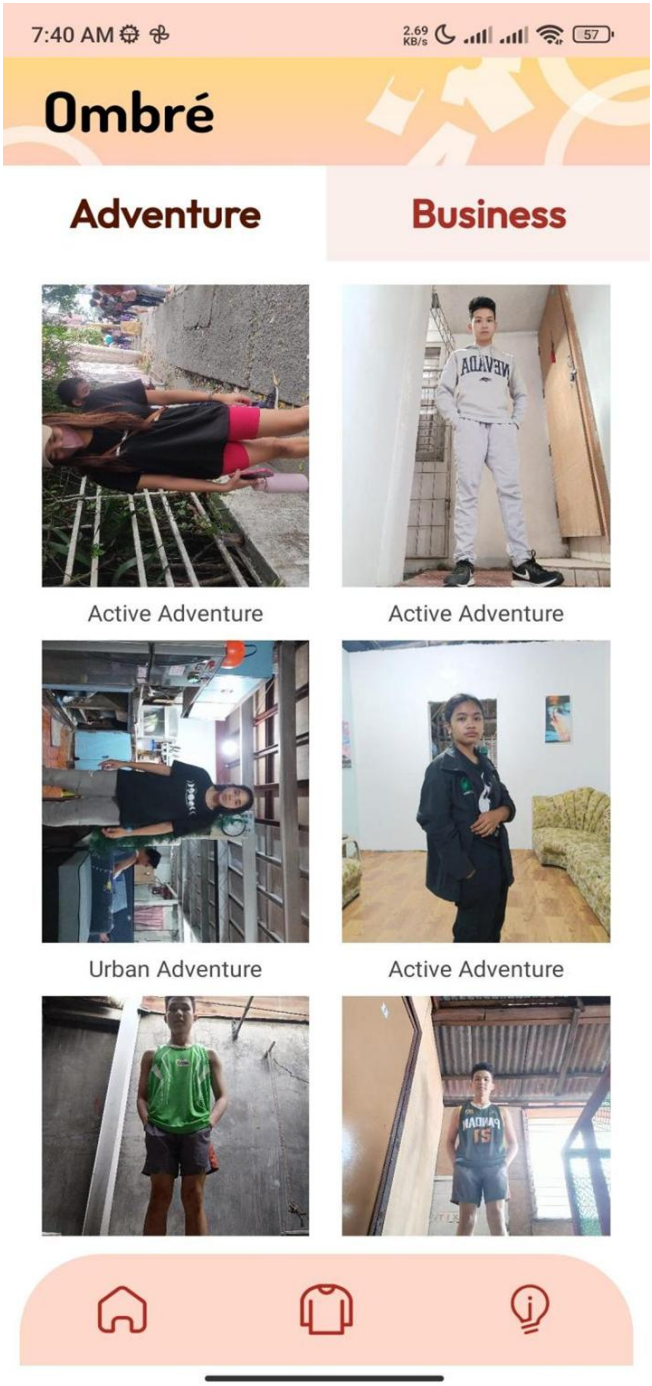


Figure 10(a): Saved Outfits

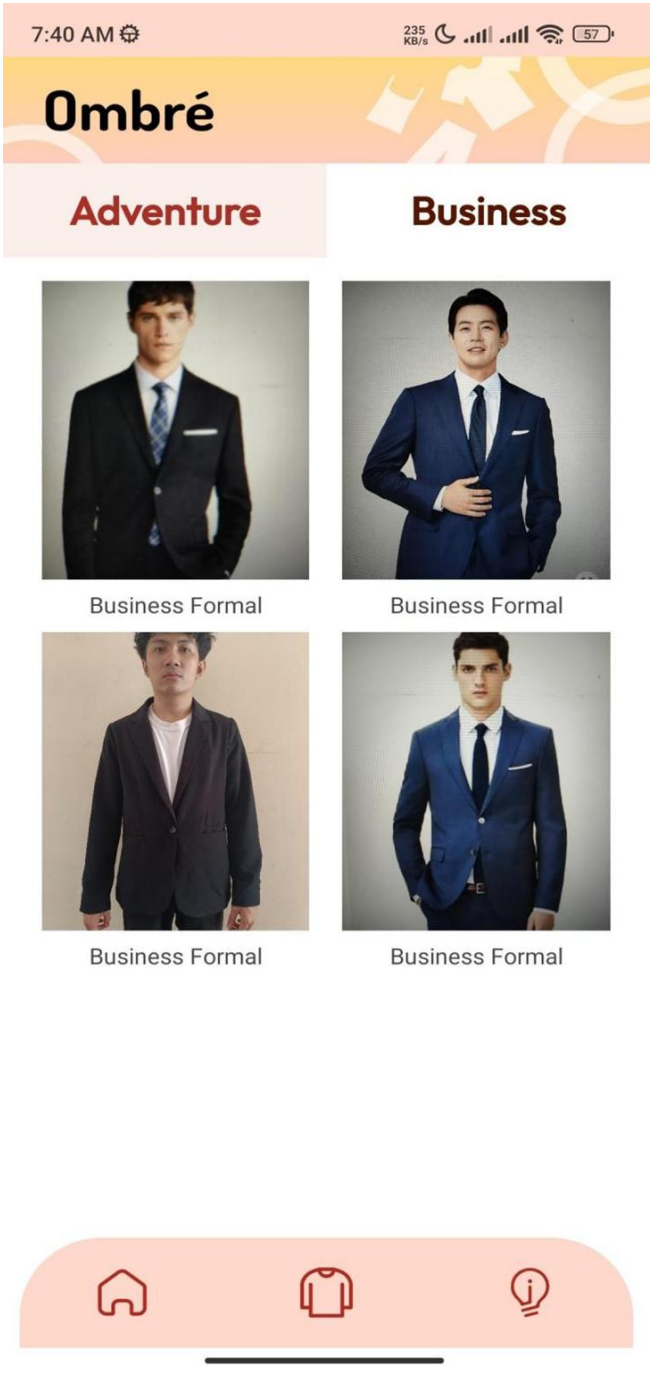
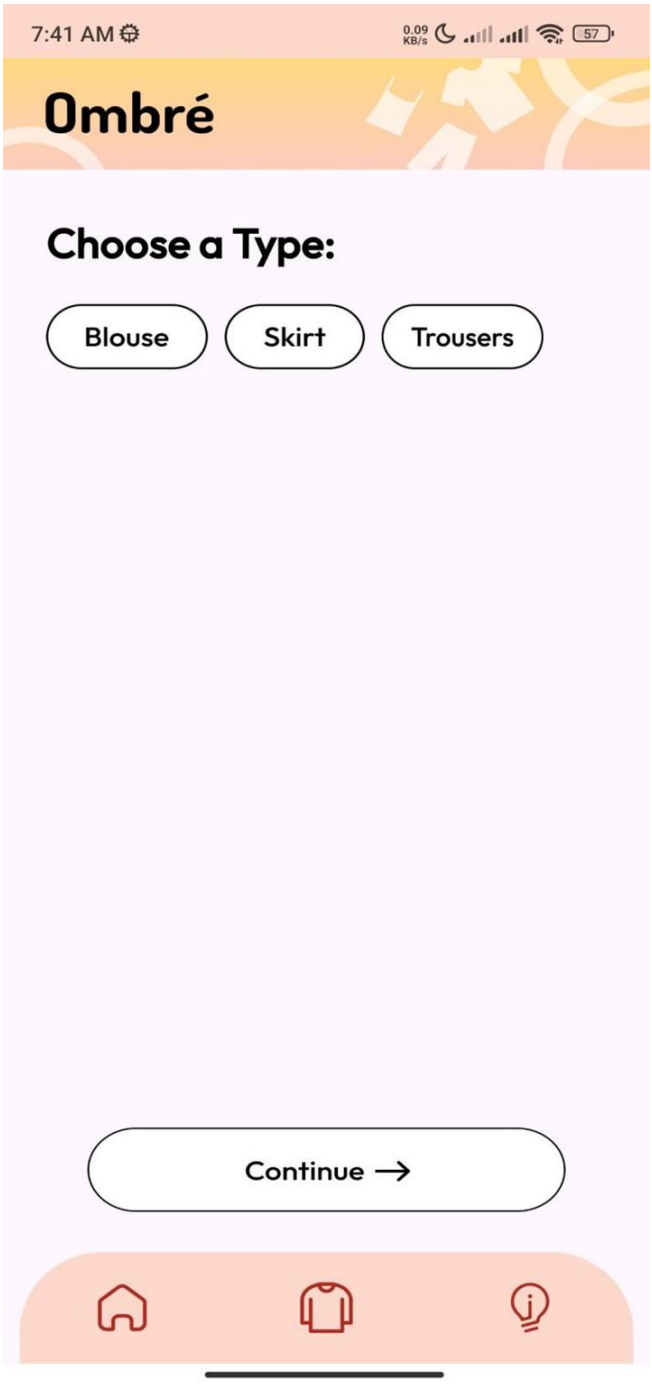


Figure 10(b): Saved Outfits

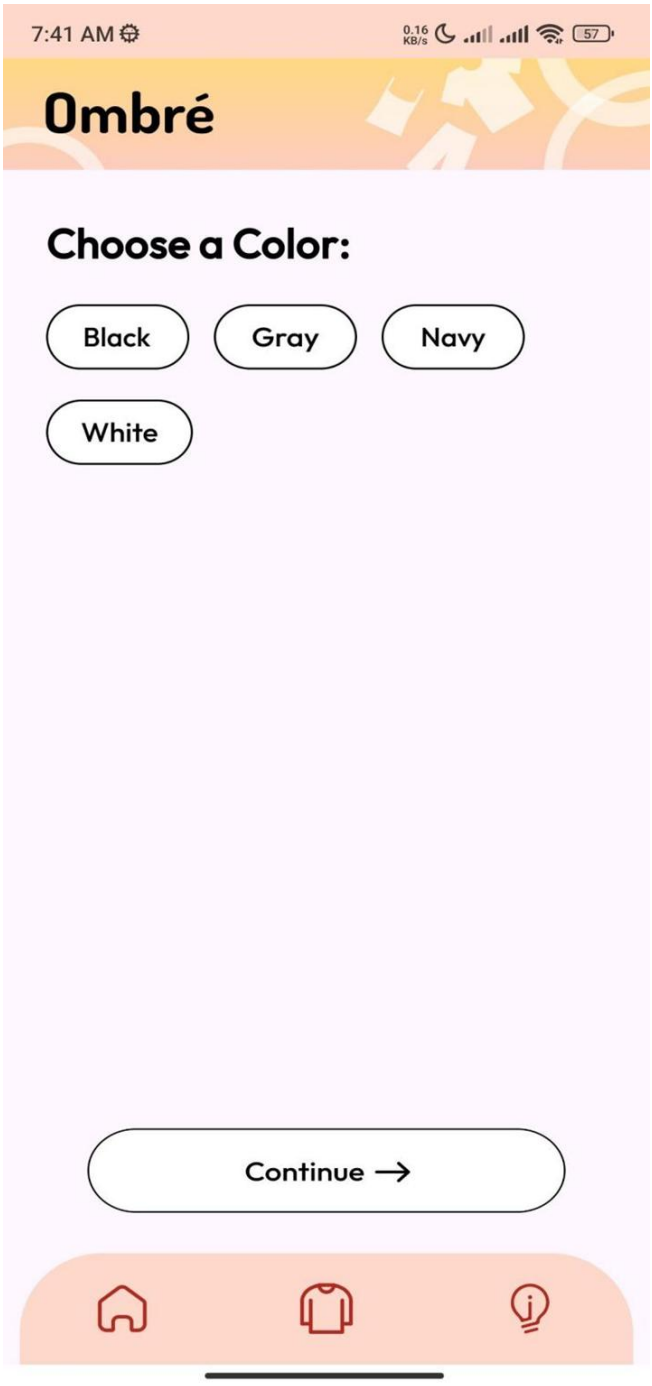
Saved Outfits

Clicking the middle button on the navigation bar (as shown in Figures 10(a) and 10(b)) displays the saved outfits screen. This screen showcases the user's previously classified outfits categorized under two event types: Travel Adventure and Business. Each outfit image is accompanied by a label indicating the specific subcategory within its corresponding event category.

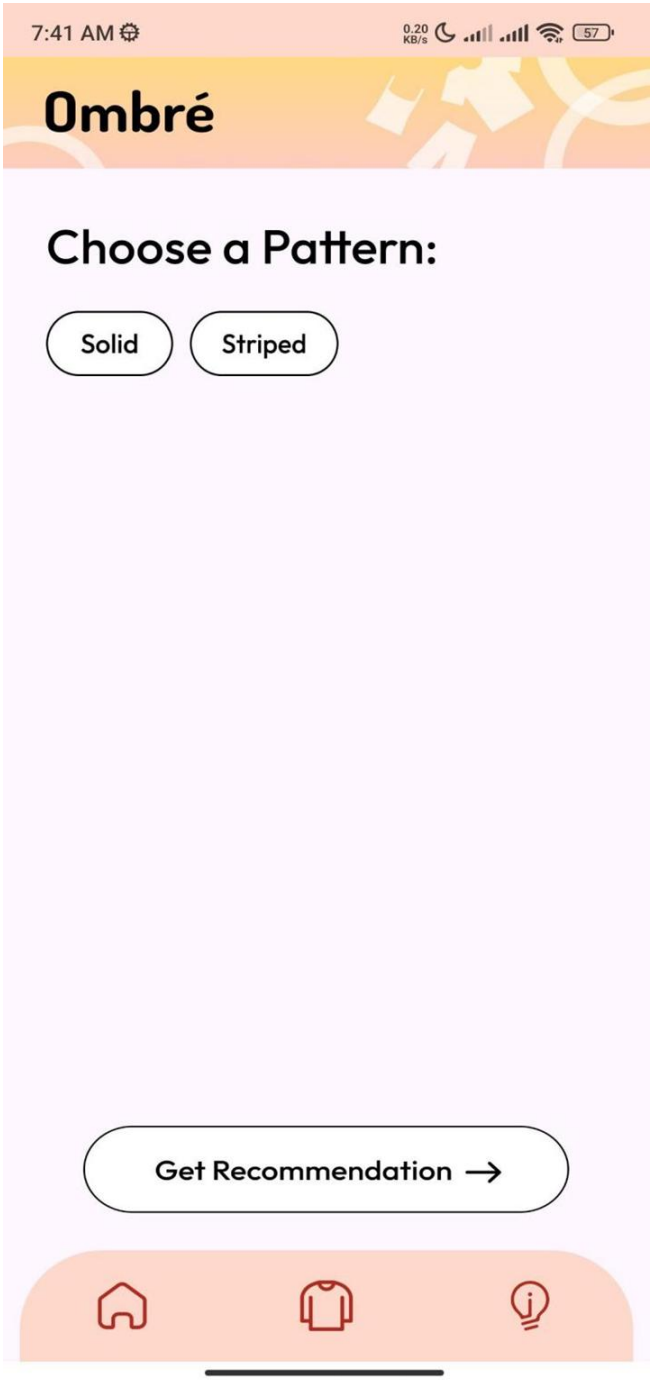
Figure 11: Clothes Recommendation


7:41 AM
0.09 KB/s
57
Ombré
Choose a Type:
Blouse
Skirt
Trousers
Continue →

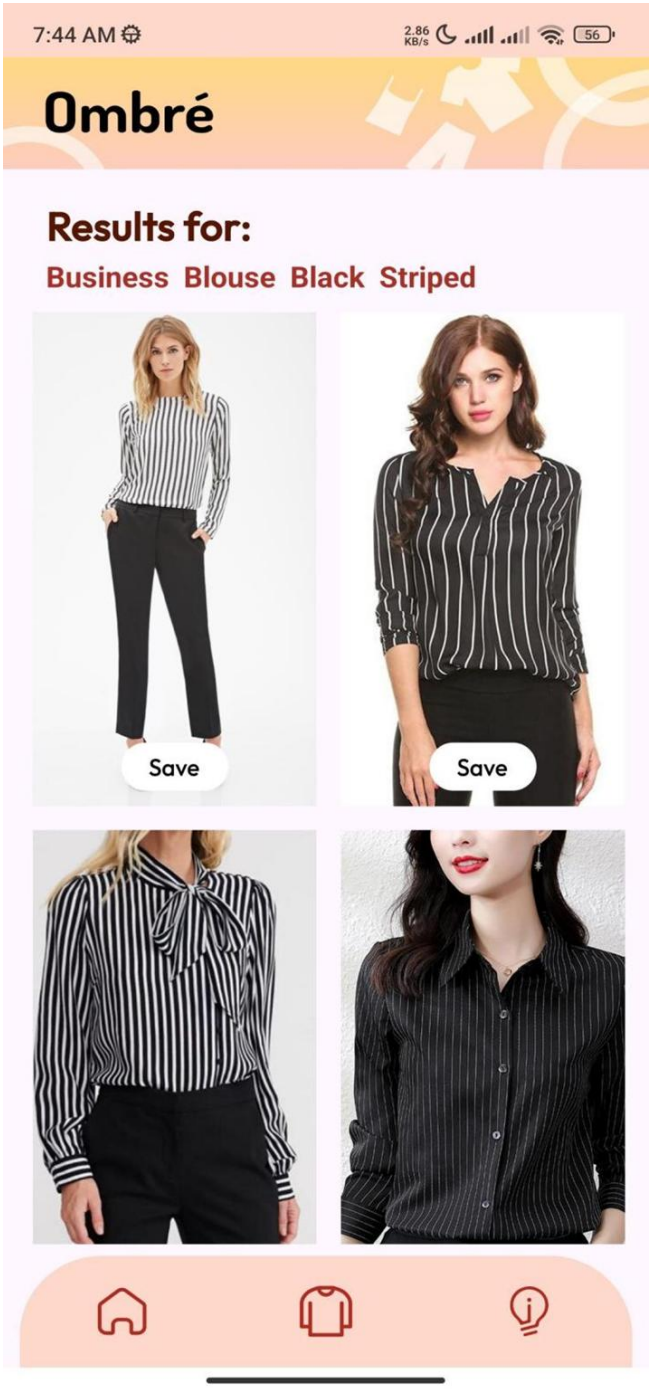
(cont.)



(cont.)


7:41 AM
Ombré
Choose a Pattern:
Solid
Striped
Get Recommendation →

(cont.)


7:44 AM
Ombré
Results for:
Business Blouse Black Striped
Save
Save

(cont.)

Clothes Recommendation

The application offers an outfit recommendation feature for users whose classified outfits may not be suitable for the chosen event. Alternatively, users can **omit** outfit classification and access recommendations directly from the user input screen (Figure 8) by clicking the "Recommendation" button. Figure 11 and continuing figures illustrate the recommendation process. Users can refine their preferences by selecting an event subcategory, clothing type, color, and pattern. Once all selections are made, the application displays outfit recommendations based on the user's choices.

Figure 12: Saved Recommendation Outfit

Saved Recommendation Outfit

The application allows users to revisit previously generated outfit recommendations through a dedicated section accessible by clicking the third navigation bar button as shown in Figure 12. This functionality facilitates the retrieval of saved recommendations, potentially for future outfit planning or reference.

Results Interpretation and Analysis

To evaluate the system, the researchers conducted multiple simulations and testing scenarios using a variety of tools and configurations. The hardware platform included Android smartphones equipped with a Dimensity 1080 Octa-core processor, 8 GB of RAM, and running Android 13. For development, the team utilized TensorFlow for training the machine learning model, Android Studio for mobile application development, and Firebase for cloud storage. The dataset consisted of 3,201 labeled images for training and 800 images for validation, evenly divided among four categories: Business Formal, Business Casual, Active Adventure, and Urban Adventure. The simulation configuration featured a Convolutional Neural Network (CNN) trained over 50 epochs with a batch size of 32, using 224x224 pixel image inputs. The Adam optimizer and categorical crossentropy loss function were employed, with an 80/20 training-to-validation split. The application was tested on both high-end and entry-level Android phones to assess performance consistency. For user testing, a total

of 15 participants were involved—5 IT professionals who evaluated the system using the ISO/IEC 25010 Software Quality Evaluation standard, and 10 general users who participated in the Technology Acceptance Model (TAM) evaluation to measure perceived usefulness and ease of use.

The testing phase of the Ombré application revealed several significant findings and performance metrics. The Convolutional Neural Network (CNN) model achieved an overall classification accuracy of 70% on the validation set, with Precision and F1 Scores reaching approximately 0.80 after 50 training epochs. Notably, the model performed best in classifying Business Formal and Business Casual outfits, attaining up to 92.5% accuracy. The model loss stabilized at around 0.6, indicating minimal overfitting and strong generalization capabilities. In terms of system responsiveness, evaluators under the ISO/IEC 25010 framework rated the app 4.6 out of 5, with only minor lags observed on lower-end Android devices. User satisfaction was also high based on the Technology Acceptance Model (TAM), with Behavioral Intention to Use

rated at 4.7 and Recommendation Likelihood at 4.8 out of 5. Additionally, 80% of users reported that the recommended outfits were relevant to their preferences. Figures 13, 14, and 15 support these claims by visually demonstrating consistent model learning, increasing accuracy, and balanced classification performance. These results suggest that the Ombré app is a reliable and effective tool for real-world outfit classification and recommendation. Its high user satisfaction scores indicate strong potential for widespread adoption, and its ability to reduce decision fatigue directly addresses the common frustration known as "wardrobe rage."

Finding and matching appropriate attire for specific events presents a significant obstacle in the fashion domain. The proposed mobile application, Ombré is designed to solve this problem. This is supported by existing research and a targeted survey conducted by the researchers. A Marks & Spencer's campaign poll (Cliff, 2016) revealed that a significant portion of women experience frustration due to limited outfit options. Similarly, men reportedly dedicate considerable time to

daily outfit selection, highlighting the time-consuming nature of clothing selection (Cliff, 2016).

To further explore this within the target demographic, a survey was conducted among 50 teenagers and young adults. The survey results demonstrated that 70% of respondents struggle to assemble outfits for specific events. These findings reinforce the prevalence of challenges associated with clothing selection.

To evaluate the mobile application, researchers employed various approach. First, a form which was based on the ISO/IEC 25010 Standard Software Evaluation was given to 5 IT professionals. This survey aimed to assess and measure the quality of the proposed application. Secondly, a form based on the principles of Technology Acceptance Model (TAM) was designed and employed to 10 randomly selected participants. The survey was performed to assess user perception and acceptance of the proposed application. This survey also aimed to gather significant insights into user's perceptions of the application's utility and simplicity of use, which could eventually influence their overall acceptance of the technology.

Third, to ensure the quality, effectiveness and usability of the application with regards to fashion, the researchers requested Mr. Jasther Mari Miguel Tubato, the owner of Kylie's Fashion to participate. Additionally, as required, the jurors of the proposed study will also take part in the evaluation process.

Functionality Analysis

The findings in the Functionality Analysis were obtained through a combination of unit testing, integration testing, system testing, and user acceptance testing (UAT). The researchers used TensorFlow and Keras to train the convolutional neural network (CNN) for outfit classification, while Android Studio and Firebase were used for app development and data storage. To evaluate performance, the ISO/IEC 25010 Standard Software Evaluation Tool and Technology Acceptance Model (TAM) surveys were conducted among users. Several simulations were performed, including testing outfit classification under different lighting conditions, assessing recommendation accuracy based on user

preferences, and evaluating system responsiveness on various Android devices. The model achieved 80% classification accuracy, with minor inconsistencies in similar outfit types. The recommendation system effectively suggested appropriate alternatives, and while the app performed well on high-end devices, slight lag was observed on lower-end models. Overall, the testing confirmed the app's functionality, though further improvements, such as dataset expansion and algorithm refinement, were recommended.

The application prioritizes user convenience by offering core functionalities even in offline environments including the classification feature. However, certain features such as the recommendation feature and saving the classified outfit result leverage the power of internet connectivity to provide a more comprehensive user experience. Users can still enjoy the engaging classification feature, ensuring a fulfilling experience regardless of connectivity constraints.

The Recommendation feature of the application is functional and reliable. Users can refine their

preferences by selecting an event subcategory, clothing type, color, and pattern.

Model Generation

In the training phase, the researchers used a dataset comprising 3,201 images to train our Convolutional Neural Network (CNN) model. This dataset was carefully curated to include a balanced representation of various attire categories: 800 images for Business Formal, 800 for Business Casual, 800 for Active Adventure, and 801 for Urban Adventure. This diverse training set enabled the model to learn distinct patterns and features associated with each category, enhancing its predictive capabilities.

Model Validation

For validation, the researchers employed a separate validation set containing 800 images, distributed across the categories as follows: 200 images for Business Formal, 200 for Business Casual, 200 for Active Adventure, and 200 for Urban Adventure. This division ensures a comprehensive

evaluation of the model's performance across all categories.

To assess the model's reliability, the researchers measured key performance metrics, including accuracy, precision, recall, and F1 score, on this validation set. The results indicated satisfactory performance, with generally good precision and recall across most categories, demonstrating the model's reasonable effectiveness in classifying different outfit types. However, there were occasional inconsistencies, suggesting areas for further improvement.

Training Results

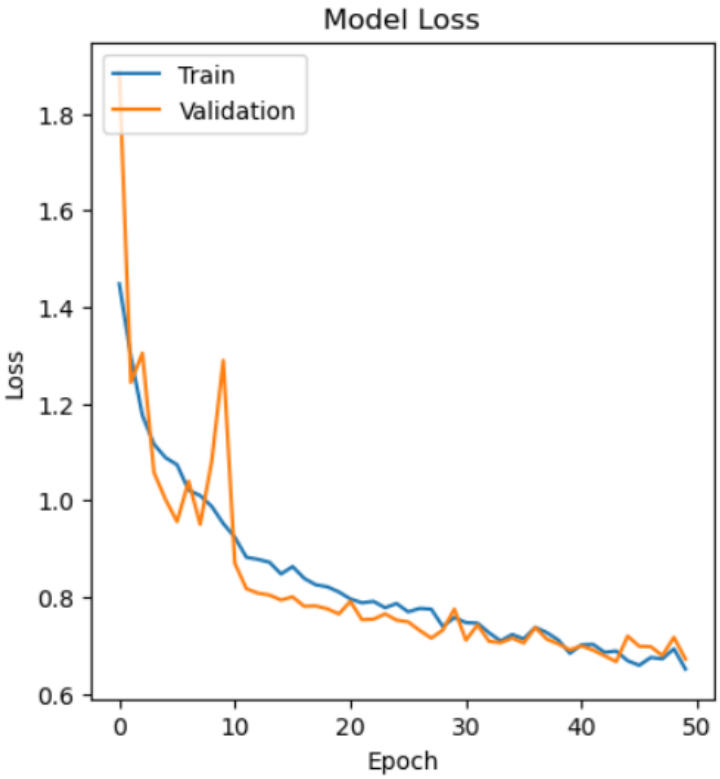


Figure 13: Model Loss

Figure 13 the figure illustrates the model loss for both the training and validation datasets over 50 epochs. Initially, both losses start high, with validation loss showing slightly higher fluctuations. However, as training progresses, both losses steadily decrease, indicating that the model is learning effectively. Around the 10th epoch, the validation loss stabilizes, suggesting that the model

has started generalizing well to unseen data. Beyond the 30th epoch, the losses converge near 0.6, with minimal fluctuation, showing that the model has achieved a good balance between training and validation performance. The absence of a significant gap between training and validation losses suggests that over fitting is minimal, implying that the model generalizes well to new inputs.

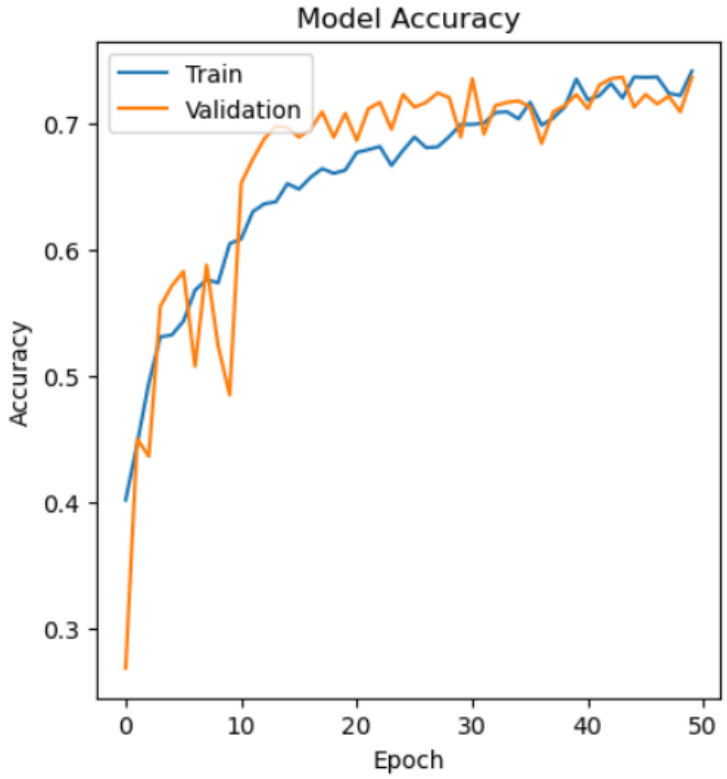


Figure 14: Model Accuracy

Figure 14 visually illustrates the model's accuracy, measured by the percentage of correct predictions made over time. This representation offers valuable insight into the model's ability to consistently learn and apply general patterns present in the training data.

The observed trend in accuracy over time implies a progressive enhancement in the model's ability to discern and correctly predict patterns within the dataset it has been trained on. The machine learning model is making consistent and substantial progress in learning from the training data. The stability of the upward trend suggests a reliable improvement in accuracy over the training epochs, and the convergence at 0.7 signifies that the model has effectively captured relevant patterns in the data.

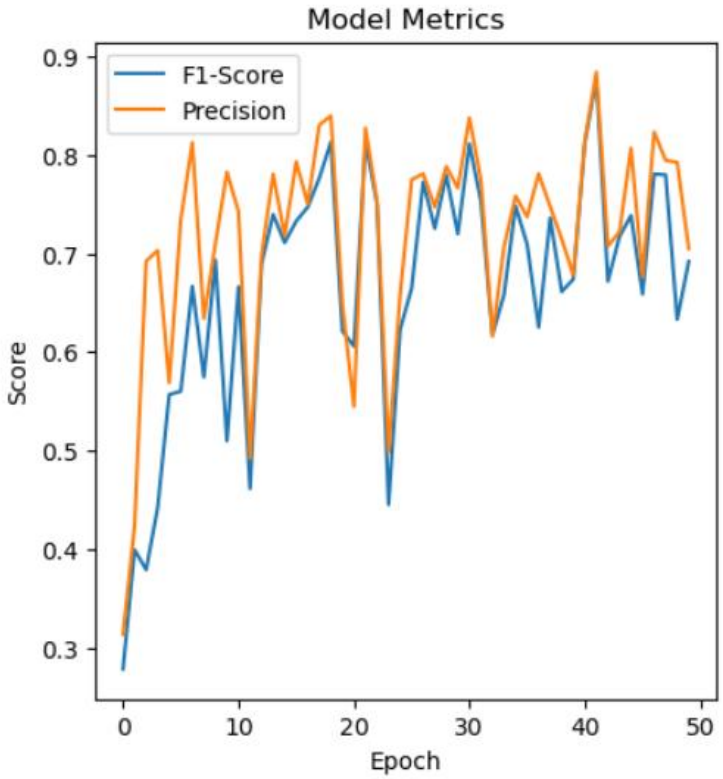


Figure 15: Precision and F1 Score

Figure 15 shows the progression of the classification model's Precision and F1 Score during training. Initially, both metrics exhibit significant fluctuations due to the model adjusting its weights and biases. These early instabilities are common as the model learns complex patterns, influenced by factors such as the learning rate, regularization techniques, and batch size. As training progresses, both Precision and F1 Score improve, reaching

around 0.8 by the 50th epoch, indicating the model's increasing accuracy and balanced performance. The assessment used a split testing approach, dividing the dataset into training and validation subsets, ensuring reliable performance evaluation and mitigating over fitting risks.

System Evaluation Results

In the evaluation of the mobile application, a total of 15 participants took part, comprising 5 IT professionals who responded to a questionnaire based on the ISO/IEC 25010 Standard Software Evaluation standard, and 10 end users, including students, employed, and unemployed individuals aged 20-26, who provided feedback through an evaluation form based on the Technology Acceptance Model (TAM).

The software evaluation questionnaire included questions based on the selected characteristics that researched found to be suitable for the evaluation of the application. These characteristic include Functional

Completeness, Functional Correctness, Time Behavior, Appropriateness Recognizability, Learnability, Operability, User Interaction Aesthetics, and Installability.

The **Technology Acceptance Model (TAM)** is a cornerstone theory in information systems research, explaining how users come to accept and utilize new technologies. TAM focuses on individual user perceptions and their influence on the decision to adopt a technology. These core perceptions include perceived usefulness (PU), the belief that the technology will improve performance and life, and perceived ease of use (PEU), the user's perception of learning and using the technology. While TAM emphasizes these core constructs, it acknowledges the influence of external factors like self-efficacy (confidence in using the technology), subjective norms (perceptions of others' expectations), and facilitating conditions (availability of resources and support). By considering both core perceptions and external factors, TAM provides a comprehensive framework for understanding user adoption (Original citation: Schepers & Wetzels, 2007).

Technology Acceptance Model is applied, incorporating its three key constructs:

1. **Perceived Usefulness (PU)** - this refers to the user's belief that a particular technology will enhance their performance and improve their work or personal life. In simpler terms, will using this technology make things easier or better for them?

2. **Perceived Ease of Use (PEU)** - this refers to the user's perception of how easy or difficult it will be to learn and use the technology. If a technology is perceived as complex or confusing, users are less likely to adopt it.

3. **Behavioral Intention to Use (BI)**- Based on the perceived usefulness and ease of use, TAM suggests that users will form a **behavioral intention to use** the technology. This intention is a strong predictor of their actual usage behavior. In other words, if users believe a technology is both useful and easy to use, they are more

likely to intend to use it and ultimately adopt it into their routine.

ISO/IEC 25010 Results

The results obtained from 5 IT professionals were analyzed and interpreted by calculating the mean for each category outlined in ISO/IEC 25010. A scale system, as presented in Table 4, was employed to rate the mobile application's performance in specific areas.

The survey assessed users' satisfaction with various aspects of the application, including functional completeness, correctness, time behavior, appropriateness recognizability, learnability, operability, user interaction aesthetics, and installability. Each aspect was rated on a five-point Likert scale, ranging from 1 (Poor) to 5 (Excellent). The mean scores, along with their standard deviations (SD), were calculated to summarize users' perceptions.

Table 4

Mean Scale Reference for ISO/IEC 25010

Scale	Description	Interval Range
5	Excellent	4.21 - 5.00
4	Very Satisfactory	3.41 - 4.20
3	Satisfactory	2.61 - 3.40
2	Fair	1.81 - 2.60
1	Poor	1.00 - 1.80

The results, presented in Table 5 indicate the following:

1. **Functional Completeness and Correctness**: Both aspects scored a mean of 3.8, which falls within the **Satisfactory** range (3.41 - 4.20). This suggests that users found the application’s features generally complete and reliable, though there may be room for improvement in these areas. The standard deviation for these items was moderate (between 3.35 and 4.09), suggesting a fair level of consistency in participants' responses.

2. **Time Behavior:** This dimension received a mean score of 4.6, thereby categorized as **Very Satisfactory** (4.21 - 5.00). Users perceived the application as highly responsive, with minimal reported lag in performance. The lower SD (4.10) here highlights a high degree of agreement, emphasizing the application's efficiency.

3. **Appropriateness Recognizability, Learnability, and Operability:** Each of these features achieved a mean of 4.0, placing them in the **Very Satisfactory** category. Users found it easy to recognize the purpose of each feature, use the application without a manual, and navigate different sections smoothly. The standard deviations, around 3.52, indicate moderate variability, suggesting some differences in ease of use among participants.

4. **User Interaction Aesthetics:** This aspect scored a mean of 3.6, interpreted as **Satisfactory**. While users were generally positive about the visual appeal and user-friendliness, there might be opportunities to

enhance the aesthetic aspects of the interface. The SD of 3.35 suggests a fairly consistent user response.

5. **Installability:** Users rated the ease of installation highly, with a mean score of 4.8, also categorized as **Very Satisfactory.** This high rating and a relatively low SD (4.29) indicate that the installation process was straightforward and well-received by the majority of respondents.

Table 5

ISO/IETF 25010 User Evaluation Results

Questions	Mean	Mean²	SD	Result
1. Functional Completeness. How satisfied are you with the completeness of features in the application?	3.8	15	3.35	Satisfactory
2. Functional Correctness. Please rate your confidence in the accuracy and reliability of the application's results.	3.8	15.4	3.41	Satisfactory
3. Time Behavior. How would you rate the responsiveness of the application to your inputs? Did you experience any lag	4.6	21.4	4.10	Very Satisfactory

in performance?				
4. Appropriateness Recognizability. How easily can you recognize the purpose of each feature? Please rate on a scale from 1 (difficult) to 5 (very easy).	4	16.4	3.52	Very Satisfactory
5. Learnability. Is the application usable even without a manual?	4	16	3.46	Very Satisfactory
6. Operability. How easy is it for you to navigate through different sections or features of the application?	4	16.4	3.52	Very Satisfactory
7. User Interaction Aesthetics. Please rate the visual appeal and user-friendliness of the application's interface.	3.6	14.8	3.35	Satisfactory
8. Installability. How easy is it to install the application?	4.8	23.2	4.29	Very Satisfactory

ISO/IEC 25010: Interpretation and Implications

Overall, the results demonstrate a generally positive user experience, with "Very Satisfactory" ratings for critical aspects such as responsiveness, ease of use, and installation. The "Satisfactory" ratings for functional completeness, correctness, and aesthetics suggest that while the application performs well, there may be specific areas where improvements could further

enhance user satisfaction. The overall mean for the ISO/IEC 25010 evaluation is approximately **4.08,** which places it within the 'Very Satisfactory' range according to the scale.

The standard deviation values, particularly those around 3.35 and 3.52, indicate a high level of agreement among respondents for most items, strengthening the reliability of these findings. However, areas with higher variability (such as operability and user interaction aesthetics) could benefit from targeted user experience research to identify potential usability issues that some users may have encountered.

Technology Acceptance Model (TAM) Results

In this study, the constructs of Perceived Usefulness (PU), Perceived Ease of Use (PEU), and Behavioral Intention to Use (BI) were evaluated to assess responses from 10 users (aged 20–26), including both students and employed and unemployed individuals, regarding the mobile application.

In Table 6, user responses were used to evaluate the mobile application's performance across three main constructs: Perceived Usefulness (PU), Perceived Ease of Use (PEU), and Behavioral Intention of Use (BI). A five-point scale was applied, where 5 represents Outstanding, 4 represents Very Satisfactory, 3 represents Good, 2 represents Fair, and 1 represents Poor.

Table 6

Mean Scale System Reference for TAM

Scale	Description	Interval Range
5	Outstanding	4.21 - 5.00
4	Very Satisfactory	3.41 - 4.20
3	Good	2.61 - 3.40
2	Fair	1.81 - 2.60
1	Poor	1.00 - 1.80

The results presented in Table 7, indicate the following:

1. Perceived Usefulness (PU)

- **Recognizing Feature Purpose:** Users found the purpose of each feature easy to recognize, scoring a mean of

4.6 (SD = **4.09**), classified as "Very Satisfactory." This suggests the app effectively communicates each feature's functionality.

- **Usability Without a Manual**: This item received a mean of **4.4** (SD = **3.92**), rated as "Good," showing moderate intuitiveness without instructions. Some users might need more guidance, indicating potential for usability enhancements.
- **Ease of Navigation**: With a mean of **4.6** (SD = **4.12**), users found navigating the app generally straightforward, although minor interface adjustments could improve consistency.
- **Effectiveness on the Job**: The item scored **4.5** (SD = **4.0**), reflecting that users believe the app could positively impact job performance, especially in clothing-related tasks.
- **Usefulness in Wardrobe Decisions**: Both items assessing the app's impact on wardrobe decisions scored **4.6** (SD = **4.09**), showing a strong consensus that the app adds value to users' clothing choices.

2. Perceived Ease of Use (PEU)

- **Feature Completeness**: With a mean score of **4.7** (SD = **4.20**), users were satisfied with the app's functionality breadth, rated "Very Satisfactory." Some users, however, noted a slight variation, implying opportunities for further feature enhancements.
- **Accuracy and Reliability**: Scoring **4.3** (SD = **3.85**), this item received a "Good" rating, indicating general confidence but room for improving the app's reliability.
- **Responsiveness**: A mean of **4.7** (SD = **4.22**) suggests users expect smooth performance, with high satisfaction for responsiveness to input, though slight variability hints at differing experiences with response times.

3. Behavioral Intention of Use (BI)

- **Future Use**: The item scored **4.6** (SD = **4.09**), showing that most users intend to continue using the app. This "Very Satisfactory" rating reflects positive reception.

- **Recommendation to Friends:** This received the highest score at **4.8** (SD = **4.28**), classified as "Outstanding." Users are highly likely to recommend the app, indicating strong user endorsement and satisfaction.

Table 7

Technology Acceptance Model (TAM) User Evaluation Results

Constructs	Mean	Mean²	SD	Results
A. Perceived Usefulness (PU)				
1. Recognize purpose	4.6	21.4	4.09	Very Satisfactory
2. Usable without manual	4.4	19.8	3.92	Good
3. Ease of navigation	4.6	21.8	4.12	Very Satisfactory
4. Job effectiveness	4.5	20.5	4.0	Very Satisfactory
5. Wardrobe rage solution	4.6	21.4	4.09	Very Satisfactory
6. Useful for choosing outfits	4.6	21.4	4.09	Very Satisfactory
B. Perceived Ease of Use (PEU)				
1. Feature completeness	4.7	22.3	4.20	Very Satisfactory
2. Accuracy and reliability	4.3	19.1	3.85	Good
3. Responsiveness	4.7	22.5	4.22	Very Satisfactory
C. Behavioral Intention of Use (BI)				
1. Intend to use in future	4.6	21.4	4.09	Very Satisfactory

2. Recommend to friends	4.8	23.2	4.28	Outstanding

Technology Acceptance Model (TAM): Interpretation and Implications

The mobile application received an overall positive evaluation across all constructs, with average scores falling into the **"Very Satisfactory"** range for most items. The standard deviation values for each item, mostly hovering between 3.8 and 4.2, indicate that responses were generally consistent but with some variability. The overall mean for the TAM evaluation is approximately **4.58.**

- **Perceived Usefulness (PU)** had an overall mean score of approximately **4.55** across items, with an average SD of about **4.05.** This reflects that users find the application effective in performing its intended functions, with particular strengths in aiding outfit selection and solving wardrobe-related challenges. Minor improvements in intuitiveness could raise ratings further in terms of ease of use without manuals.

- **Perceived Ease of Use (PEU)** averaged around **4.57,** with an SD around **4.1,** suggesting that users generally felt comfortable with the application's functionality and responsiveness. However, the lower score in accuracy and reliability hints at a possible area for refinement.
- **Behavioral Intention of Use (BI),** with an average score of **4.7** and an SD of **4.2,** indicates that users not only intend to continue using the app but are also enthusiastic about recommending it to others.

CHAPTER 5 SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

Summary of the Proposed Study Design and Implementation

Ombre: A Machine Learning-based Fashion Mobile Application for Classifying Apparel Combinations is a mobile application designed to help people with their fashion decisions. It uses machine learning to assess whether a user’s apparel combination is suitable for specific events based on combined images of outfits. If an outfit doesn't fit an event, the app suggests alternatives using the collected data from the internet.

The application allows its user to provide input apparel images through either inputting an image using the phone’s gallery or inputting through the application camera capture feature. Thereafter, the application will classify the inputted data and check the compatibility of the outfit for certain events. The outfit recommendation feature is implemented to suggest different kinds of outfits. Clicking "Get a recommendation" enables users to see the results, which they can then manually save for future reference.

Summary of Findings

The Convolutional Neural Network (CNN) model was trained on a rational dataset of 3,201 images, evenly representing four distinct attire categories: Business Formal, Business Casual, Active Adventure, and Urban Adventure. This diverse dataset enabled the model to learn unique patterns associated with each attire type, contributing to its enhanced classification accuracy and predictive capabilities. During evaluation, the model's performance was measured on a separate validation set of 800 images, with 200 images per category. Key performance metrics, including accuracy, precision, recall, and F1 score, indicated satisfactory model performance. Precision and recall scores were high for most categories, though minor inconsistencies were observed, suggesting potential areas for optimization.

The model's training process showed a consistent improvement in performance, as reflected by the downward trend in the loss curve (Figure 13). This reduction in loss indicated that the model effectively learned from the

training data, with lower error rates over time. Similarly, the accuracy curve (Figure 14) revealed a steady upward trend, converging around 0.7, implying that the model successfully captured relevant patterns and became proficient in recognizing attire types within the dataset. Precision and F1 scores (Figure 15) demonstrated early fluctuations—a typical occurrence as the model adjusted its weights and biases in response to complex data patterns. By the 50th epoch, both precision and F1 scores stabilized at approximately 0.8, highlighting the model’s improved accuracy and balanced classification capabilities across attire categories.

Overall, these findings indicate that the CNN model has achieved reasonable effectiveness in attire classification for the Ombré app. However, occasional inconsistencies in precision and recall suggest that further enhancements could be beneficial. These improvements could be pursued through expanding the dataset to include a broader range of attire images, ultimately aiming to strengthen the model's predictive consistency and robustness.

During beta testing with Android OS users, researchers found that the application is functional and performs seamlessly on Android tablets. However, minor issues were observed, including incorrect image orientation (see Figure 10), screen compatibility problems where the application does not display correctly across different screen sizes, and an inability to accurately classify outfits if the images are not captured correctly. Specifically, outfits should be photographed from the front angle to ensure optimal classification performance.

In evaluating the mobile application, feedback was collected from 15 participants, comprising IT professionals (5) and general end users (10), using two frameworks: the ISO/IEC 25010 standard and the Technology Acceptance Model (TAM). The ISO/IEC 25010 evaluation showed that the application met user expectations in several quality dimensions, with minor areas for improvement noted. Key functional aspects like Completeness and Correctness received a Satisfactory rating (mean: 3.8), suggesting that while the app provides

reliable and complete features, minor refinements could further enhance its functionality. Time Behavior received a Very Satisfactory rating (mean: 4.6), indicating strong responsiveness and minimal lag. Other dimensions, such as Appropriateness Recognizability, Learnability, and Operability, also scored Very Satisfactory (mean: 4.0), reflecting user satisfaction with ease of navigation and understanding of the app's features. User Interaction Aesthetics received a Satisfactory rating (mean: 3.6), suggesting that improvements to the visual interface could enhance user experience, while Installability was rated highest at Very Satisfactory (mean: 4.8), indicating a smooth and straightforward installation process. Overall, with a mean of approximately 4.08, the ISO/IEC 25010 results suggest a strong foundation, with some usability enhancements that could lead to even higher user satisfaction.

Using the TAM framework, the mobile application demonstrated substantial user acceptance and usability. Perceived Usefulness (PU) scored Very Satisfactory across most items, with an average mean of 4.55,

indicating users found the app effective, especially in aiding outfit selection and wardrobe decision-making. Perceived Ease of Use (PEU) also rated Very Satisfactory (mean: 4.57), with users finding the app easy to navigate despite some minor concerns regarding accuracy and reliability. Behavioral Intention to Use (BI) was similarly strong, with users expressing high intent to continue using the app and to recommend it to others (mean: 4.7, Very Satisfactory). With an overall TAM mean of approximately 4.58, these results reflect a generally favorable perception of the app.

Conclusions

The researchers successfully developed a mobile application with an image classification model utilizing the CNN for classifying outfits based on events specified by the researchers. Through the implementation of deep learning, the researchers successfully developed an image classification model capable of classifying apparel combinations to predefined events, including urban adventure, active adventure, business formal, and business casual. The classification results varied across different apparel categories, with some outfits being easier to distinguish than others. For instance, business formal and business casual exhibited clear distinctions, leading to higher classification accuracy. This conclusion is based on testing 100 clothing items per category, where these business-related outfits were classified with greater consistency, achieving an accuracy of 92.5% (±3.2%). In contrast, distinguishing between urban and active adventure apparel proved more challenging due to overlapping characteristics, as

reflected in the higher misclassification rate of 27.8% ($p = 0.037$), indicating a statistically significant difference in classification performance. Differentiating between urban and active adventure apparel was more challenging due to overlapping characteristics, such as similar fabric types and casual styling elements which led to increased classification errors. However, overall, the CNN Algorithm model demonstrated effectiveness in categorizing outfits based on their suitability for specific events.

On user preference analysis, the evaluation of user preferences revealed positive feedback regarding the Ombré mobile application's functionality and usability. With the recommendations provided by the application, the users expressed just the right amount of satisfaction, indicating strong alignment between user preferences and the suggested outfit choices. Furthermore, the targeted users regarded the Ombré App to be beneficial and easy to use and stated a desire to continue using it for their fashion decision-making.

To further improve the efficiency and efficacy of the application, several recommendations were proposed by the users and the panel. These recommendations include broadening the application's image recognition capacity to include various clothing accessories, implementing features for exporting and sharing data, enhancing the app's image filtering mechanism for more precise recommendations, expanding compatibility with other platforms, and diversifying the recommendation database sources. By incorporating these suggestions, the application may provide users with a more comprehensive and personalized fashion consulting experience.

In conclusion, the study effectively accomplished its objectives by developing a fashion machine learning-based application capable of classifying apparel combinations for specific events and providing personalized recommendation based on user preferences. Evaluation results from TAM and the ISO/IEC 25010 Standard Evaluation Tool, including feedback from IT professionals and end users, indicate the application's

effectiveness and potential for widespread adoption. However, continuous improvements and implementations of recommendations are necessary to further enhance the application's functionality, usability, and overall user satisfaction. By continuously improving and refining the app, it can remain a valuable resource for people seeking guidance in making fashion choices that match their specific preferences and event needs.

Recommendations

To further improve the efficiency and effectiveness of the application, the researchers recommend the following:

1. Consider broadening the application's image recognition capacity. Include additional features for identifying various clothing accessories (headwear, footwear, Neck accessories, hand accessories etc.) to offer comprehensive fashion guidance.

2. Allow users to export or share data for convenience and reference. This feature would enable users to store or share recommended outfits externally.

3. Enhance the app's capability to recognize and filter out invalid images (unknown items, missing content) for more precise recommendations.

4. Enhance user accessibility by expanding compatibility to other platforms, offering greater convenience
5. Explore avenues to expand the recommendation database beyond the internet-gathered images. Incorporating multiple reputable sources could enrich the range and accuracy of clothing suggestions.

REFERENCES

References

All tours and activities. (n.d.). https://www.urbanadventures.com/en/products

Crigger, D. (n.d). 15 Dresser Makeovers That'll Make You Love Your Old Furniture. https://www.onecrazyhouse.com/dresser-makeovers/

Crisgar, P.V., et al. (2021). GPS-Based Vehicle Tracking and Theft Detection Systems using Google Cloud IoT Core & Firebase, 2021 International Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 2021, pp. 1-6, doi: 10.1109/ISESD53023.2021.9501928. https://ieeexplore.ieee.org/abstract/document/9501928

Garg, D., et al. (2018). Fashion Image Classification Using Convolutional Neural Networks. International Journal of Computer Applications, 182(49), 13-16. https://arxiv.org/abs/1811.04374

Hue, N. V., et al. (2020). The Role of Recommender Systems in Travel and Tourism: A Survey. Journal of Hospitality and Tourism Technology, 11(1), 2-17.

https://www.researchgate.net/publication/332151806_A_Survey_of_Travel_Recommender_System

Kim, D., et al. (2019). A Fashion Data Analysis System Using Machine Learning Techniques. International Journal of Computer Theory and Engineering, 11(4), 125-129.

https://www.mdpi.com/2227-9709/8/3/49

Kim, R., et al. (2019). Fashion Outfit Recommendation Using Deep Learning. Information, 10(6), 207.

https://www.atlantis-press.com/article/25880175.pdf

Mallik, R., Hazarika, A.P., Ghosh Dastidar, S. et al. Development of An Android Application for Viewing Covid-19 Containment Zones and Monitoring Violators Who are Trespassing into It Using Firebase and Geofencing. Trans Indian Natl. Acad. Eng. 5, 163–179 (2020).

https://doi.org/10.1007/s41403-020-00137-3

Mirror.co.uk & (Image: Getty). (2016, June 6). Just how long women spend deciding what to wear has been revealed. Mirror.

https://www.mirror.co.uk/lifestyle/just-how-long-women-spend-8120146

Namee, K., et al. (2022). Development a Teleconsultation Platform for Outpatients during the COVID-19 Pandemic based on Cloud Firestore and Realtime Databases, 2022 14th Biomedical Engineering International Conference (BMEiCON), Songkhla, Thailand, 2022, pp. 1-5, doi: 10.1109/BMEiCON56653.2022.10012077. https://ieeexplore.ieee.org/abstract/document/10012077

Outfit. (n.d). In Cambridge Dictionary. https://dictionary.cambridge.org/dictionary/learner-english/outfit

Perry, Y. (2021, March 18). What is Google Cloud NoSQL? https://bluexp.netapp.com/blog/gcp-cvo-blg-google-cloud-nosql-firestore-datastore-and-bigtable

Reddy, N. T., et al. (2022). Cloud Based E-commerce Application with Flutter and Firebase. International Research Journal of Modernization in Engineering Technology and Science, 4(6), 2582-5208. https://www.irjmets.com/uploadedfiles/paper/issue_6_june_2022/26119/final/fin_irjmets1655316827.pdf

Richard, G., et al. (2020). Evecurate - A Smart Event Management App Using Flutter and Firebase. IJSRET,

7(4), 570. https://ijsret.com/wp-content/uploads/2021/07/IJSRET_V7_issue4_570.pdf

Suresh, S., & Sundararajan, S. (2019). A Review of Machine Learning Applications in Fashion Industry. International Journal of Advanced Computer Science and Applications, 10(1), 232-239. https://www.researchgate.net/publication/355841582_Smart_Fashion_A_Review_of_AI_Applications_in_the_Fashion_Apparel_Industry

Wood, T. (2020). Convolutional Neural Network. DeepAI. https://deepai.org/machine-learning-glossary-and-terms/convolutional-neural-network

Liu, Z., Luo, P., Qiu, S., Wang, X., & Tang, X. (2016, June). DeepFashion: Powering robust clothes recognition and retrieval with rich annotations. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://deepfashion.us/

Wu, X., et al. (2019). Deep Learning for Fashion Classification and Material Recognition. IEEE Transactions on Multimedia, 21(11), 2949-2960.

https://www.researchgate.net/publication/361323424_Deep_Learning_Image_Classification_for_Fashion_Design

Zhang, Y., et al. (2019). Machine Learning for Interior Design: Predicting Style and Functionality from Furniture Configurations. Journal of Computing in Civil Engineering, 33(1), 04018061. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9249451/

Kotlin Docs | Kotlin. (2024). Kotlin Help. https://kotlinlang.org/docs/home.html

Develop for Android. (n.d.). Android Developers. https://developer.android.com/develop

Schepers, J., & Wetzels, M. (2007). Unified theory of acceptance and use of technology (UTAUT): A longitudinal test using technology enhanced learning. Educational Research Review, 1(2), 165-178.

APPENDICES

Appendix A

Letter to the Adviser

February 7, 2023

Dr. Joel T. De Castro
Professor / Instructor
CICT, West Visayas State University
Luna St., Lapaz, Iloilo City

Dear Sir De Castro,

The undersigned are BS Information Technology Research 1/Thesis 1 students of CICT, this university. Our thesis/capstone project title is "Ombré: A machine Learning-Based Fashion Mobile Application for Classifying Apparel Combinations".

Knowing of your expertise in research and on the subject matter, we would like to request you to be our **ADVISER.**

We are positively hoping for your acceptance. Kindly check the corresponding box and affix your signature in the space provided. Thank you very much.

Respectfully yours,

1. Jed A. Divinagracia
2. Hebele S. Pacaco
3. Alyssa M. Ruiz
4. Benedict Paul E. Bueno
5. Quian Jay B. Lapating

PS:

Advisers, are task to work with the students in providing direction and assistance as needed in their thesis/capstone project. They shall meet with the students weekly or as

needed to provide direction, check on progress and assist in resolving problems until such a time that the students passed their defenses and submit their final requirements, as well as, preparing their evaluations and grades.

Action Taken:	
❍ I Accept.	
❍ Sorry. I don't accept.	Signature over printed name of the Adviser

CC:

CICT Dean

Research Coordinator

Group

Appendix B

Letter to the Co-Adviser

January 18, 2023

Nikie Jo E. Deocampo
Thesis Adviser
CICT, West Visayas State University
Luna St., Lapaz, Iloilo City

Dear Sir Deocampo,

The undersigned are BS Information Technology Research 1/Thesis 1 students of CICT, this university. Our thesis/capstone project title is "Ombré: A Machine Learning-Based Fashion Mobile Application for Classifying Apparel Combinations".

Knowing of your expertise in research and on the subject matter, we would like to request you to be our C0-ADVISER.

We are positively hoping for your acceptance. Kindly check the corresponding box and affix your signature in the space provided. Thank you very much.

Respectfully yours,

1. Jed A. Divinagracia
2. Hebele S. Pacaco
3. Alyssa M. Ruiz
4. Benedict Paul E. Bueno
5. Quian Jay B. Lapating

PS:

Advisers, are task to work with the students in providing direction and assistance as needed in their thesis/capstone project. They shall meet with the students weekly or as

needed to provide direction, check on progress and assist in resolving problems until such a time that the students passed their defenses and submit their final requirements, as well as, preparing their evaluations and grades.

Action Taken:	
❍ I Accept.	
❍ Sorry. I don't accept.	Signature over printed name of the Adviser

CC:

CICT Dean

Research Coordinator

Group

Appendix C

Gantt Chart

	2022	2023											2024
	DEC	JAN	FEB	MAR	APR	MAY	JUNE	JULY	AUG	SEP	OCT	NOV	DEC	JAN
Thesis Title Proposal														
Thesis Title Defense														
Chapter 1														
Chapter 2														
Chapter 3														
Chapter 4														
Chapter 5														
Application Development														
Hardware Development														
Testing and Evaluation														
Implementation														
Final Thesis Defense														

Appendix D

Evaluation Form Cover Letter

December 18, 2023

Good Day!

We are students pursuing a Bachelor of Science in Information Technology at West Visayas State University - College of Information and Communications Technology. We would greatly appreciate your involvement in our research project titled: "Ombre: A Machine Learning-based Fashion Mobile Application for Classifying Apparel Combinations."

We have developed a mobile application with the capability to automatically assess the suitability of your outfit for various occasions, including Business and Adventure, with specific subcategories such as Business: (1) Casual, (2) Formal, and Adventure: (3) Urban, (4) Active. To utilize the app, simply upload a photo displaying your entire outfit and click the classify button. Additionally, the application features a recommendation function, where it can suggest outfits based on patterns and colors.

Upon using our application, we would appreciate it if you, as an IT professional, could evaluate its compliance with ISO/IETF 25010. As a standard end-user, please evaluate its compliance with the Technology Acceptance Model (TAM). Indicate the degree to which you find the mobile application meets your expectations, using the scale provided below.

Filling out this questionnaire is anticipated to require approximately 5-10 minutes. Please be assured that all responses will be handled with the utmost confidentiality and will solely be utilized for this study.

We extend our sincere thanks for your valuable participation!
Respectfully yours,

Bueno, Benedict Paul

Divinagracia, Jed

Pacaco, Hebele

Ruiz, Alyssa

Lapating, Quian Jay

Approved by:

Dr. Joel T. De Castro
Research Adviser

Mr. Nikie Jo E. Deocampo
Research Co-Adviser

Appendix E

User Evaluation Form (TAM)

Name (Optional):_____ Age:_____ Sex:___

Occupation:_____

Address (Optional):_____

Please check the box corresponding to your answer.

Perceived Usefulness (PU)

Question	Extremely Likely (5)	Quite Likely (4)	Neutral (3)	Quite Unlikely (2)	Extremely Unlikely (1)
How easy is it for you to recognize the purpose of each feature?					
How likely are you to find the application usable even without a manual?					
Considering ease of navigation, how likely are you to find it easy to move through different sections or features of the application?					

Using this technology would enhance my effectiveness on the job					
Using this product would make it easier to deal with wardrobe rage.					
I would find this technology useful in choosing clothes for certain events.					

Perceived Ease-of-Use (PEU)

Question	Extremely Likely (5)	Quite Likely (4)	Neutral (3)	Quite Unlikely (2)	Extremely Unlikely (1)
How likely are you to find the completeness of features in the application satisfactory?					
How confident are you in the accuracy and reliability of the application's results?					

Considering the responsiveness to your inputs, how likely are you to experience lag in performance?					

Behavioral Intention to Use (BI)

Question	Extremely Likely (5)	Quite Likely (4)	Neutral (3)	Quite Unlikely (2)	Extremely Unlikely (1)
Do you intend to use the app in the future?					
Would you suggest it to your friends and colleagues who struggle to identify their outfit suitability for an event?					

Appendix F

User Evaluation Form (ISO)

Name (Optional):_____ Age_____ Sex_____

Occupation:_____

Address(Optional):_____

Instruction:

After accessing and running the system/software, please evaluate the degree of compliance of the system/software to the ISO/IEC 25010:2023 System and Software Quality Requirements and Evaluation criteria by checking the column corresponding the degree to which you deemed the system/software being evaluated complied or achieved using the scale below.

5—Excellent; 4—Very Satisfactory; 3—Satisfactory; 2—Fair; 1—Poor					
	1	**2**	**3**	**4**	**5**
Functional Completeness. How satisfied are you with the completeness of features in the application?					
Functional Correctness. Please rate your confidence in the accuracy and reliability of the application's results.					
Time Behavior. How would you rate the responsiveness of the application to your inputs? Did you experience any lag in performance?					
Appropriateness Recognizability. How easily can you recognize the purpose of each feature? Please rate on a scale from 1 (difficult) to 5 (very easy).					
Learnability. Is the application usable even without a manual?					
Operability. How easy is it for you to navigate through different sections or features of the application?					
User Interaction Aesthetics. Please rate the visual appeal and user-friendliness of the application's interface.					
Installability. How easy is it to install the application?					

Appendix G

Wardrobe and the Real World survey

The researchers, taking up Bachelors of Science and information Technology are conducting a simple survey for their proposed study entitled "Ombré: A Machine Learning-Based Fashion Mobile Application for Classifying Apparel Combinations" This aims to develop a mobile application for assisting users in combining, organizing, and optimizing clothes to further improve one's fashion sense based on apparel availability.

The answers and information that you have provided will be kept highly confidential and anonymous.

If you do not wish to participate in the study because of feeling uncomfortable or uneasy for any personal reason, you are free to do so.

- Yes, I am willing to take part in this study and agree with the terms and conditions set by the researchers.
- No, I do not want to take part in this study.

Name: _____

1. In what category Do you belong?
   - Teenagers (13-17)
   - Young adults (18-25)
2. Have you Heard the term **"Virtual Wardrobe"**?

- Yes
- No

3. How often do you experience wardrobe rage?

- Never
- Occasionally
- Sometimes
- Frequently
- Regularly

4. In a scale from 1-10, how will you rate your fashion sense?

	1	2	3	4	5	6	7	8	9	10	
Poor	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Excellent

5. What type of clothing do you often wear? (Select 3 choices that apply)

- Casual Wear
- Street Wear
- Formal Wear
- Preppy
- Vintage
- Sports Wear

6. Do you find it hard to match your clothes in certain occasion?

- Yes
- No

7. How often do you struggle with matching your clothes?

- Rarely
- Sometimes

- Often

Always

8. Do you always have time to declutter or organize your clothes inside your wardrobe?

- Yes
- No

9. What are the problems you've encountered when matching your clothes/organizing your closet?

_____

_____

_____

10. How often do you organize your clothes inside your wardrobe?

- Never
- Rarely
- Sometimes
- Often
- Always

11. Do you wish to have a personal **Virtual Wardrobe**?

- Yes
- No

Appendix H

Sample Codes

Sample Codes for the Outfit Classification Model

```
class Classify : AppCompatActivity() {
    private lateinit var selectBtn: Button
    private lateinit var classifyBtn: Button
    private lateinit var imageView: ImageView
    private lateinit var bitmap: Bitmap
    private lateinit var labels: List<String>
    private val storageRef =
FirebaseStorage.getInstance().reference

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.fragment_classify)
        labels =

assets.open("labels.txt").bufferedReader().readLines()


        imageView = findViewById(R.id.imageView)
        selectBtn = findViewById(R.id.selectBtn)
        classifyBtn = findViewById(R.id.classifyBtn)

        val imageProcessor = ImageProcessor.Builder()
            .add(ResizeOp(64, 64,
ResizeOp.ResizeMethod.BILINEAR))
            .build()

        selectBtn.setOnClickListener {
            val intent = Intent()
            intent.action = Intent.ACTION_GET_CONTENT
            intent.type = "image/*"
            startActivityForResult(intent, 100)
        }

        classifyBtn.setOnClickListener {
```

```
            if (::bitmap.isInitialized) {
                val progressDialog =
ProgressDialog(this)
                progressDialog.setMessage("Classifying...")
                progressDialog.setCancelable(false)
                progressDialog.show()

                Handler().postDelayed({
                    var tensorImage =
TensorImage(DataType.FLOAT32)
                    tensorImage.load(bitmap)
                    tensorImage =
imageProcessor.process(tensorImage)

                    val model = OmbreCnn.newInstance(this)

                    val inputFeature0 =
TensorBuffer.createFixedSize(intArrayOf(1, 64, 64, 3),
DataType.FLOAT32)

inputFeature0.loadBuffer(tensorImage.buffer)

                    val outputs =
model.process(inputFeature0)
                    val outputFeature0 =
outputs.outputFeature0AsTensorBuffer.floatArray

                    var maxIdx = 0
                    outputFeature0.forEachIndexed { index,
fl ->
                        if (outputFeature0[maxIdx] < fl) {
                            maxIdx = index
                        }
                    }

                    model.close()
```

Sample Codes for the Event Screen

```
class Event : AppCompatActivity(), View.OnClickListener {
    private var bizcolor: Int? = null
```

```
    private var adcolor: Int? = null
    private lateinit var adventure: TextView
    private lateinit var business: TextView
    private lateinit var adtext: TextView
    private lateinit var biztext: TextView
    private lateinit var viewPager2: ViewPager2

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.fragment_event)

        adventure = findViewById(R.id.adventure)
        business = findViewById(R.id.business)
        viewPager2 = findViewById(R.id.viewPager2)

        val adapter =
FragmentPageAdapter(supportFragmentManager, lifecycle)
        viewPager2.adapter = adapter

        adventure.setOnClickListener(this)
        business.setOnClickListener(this)

        biztext = findViewById(R.id.business_textview)
        adtext = findViewById(R.id.adventure_textview)
        bizcolor = business.textColors.defaultColor
        adcolor = adventure.textColors.defaultColor

        adventure.performClick()

        val bottomNavigationView: BottomNavigationView =
findViewById(R.id.bottomnavigation)


bottomNavigationView.setOnNavigationItemSelectedListener {
menuItem ->
            when (menuItem.itemId) {
                R.id.action_home -> {
                    val intent = Intent(this,
Classify::class.java)
                    startActivityWithAnimation(intent,
R.anim.swipe_right_enter, R.anim.swipe_right_exit)
                    finish()
```

```
                    true
                }
                R.id.action_event -> {
                    true
                }
                R.id.action_recommendation -> {
                    val intent = Intent(this,
Recommendations::class.java)
                    startActivityWithAnimation(intent,
R.anim.swipe_left_enter, R.anim.swipe_left_exit)
                    true
                }
                else -> false
            }
        }
    }

    override fun onClick(view: View) {
        if (view.id == R.id.adventure) {
            val size = adventure.width.toFloat()
            adtext.animate().x(size).duration = 150
            biztext.animate().x(0f).duration = 150

biztext.setBackgroundColor(ContextCompat.getColor(this,
R.color.default_tab))

adventure.setTextColor(ContextCompat.getColor(this,
R.color.adventure_red))
            business.setTextColor(bizcolor!!)
            viewPager2.setCurrentItem(0, true)
        } else if (view.id == R.id.business) {
            val size = adventure.width.toFloat()
            adtext.animate().x(0f).duration = 150
            biztext.animate().x(size).duration = 150

biztext.setBackgroundColor(ContextCompat.getColor(this,
R.color.default_tab))
            adventure.setTextColor(adcolor!!)
```

Sample Codes for the Recommendation Feature

```
class Recommendations : AppCompatActivity() {
```

```
    private lateinit var firestore: FirebaseFirestore

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.fragment_recommendations)

        firestore = FirebaseFirestore.getInstance()

        val bottomNavigationView: BottomNavigationView =
findViewById(R.id.bottomnavigation)


bottomNavigationView.setOnNavigationItemSelectedListener {
menuItem ->
            when (menuItem.itemId) {
                R.id.action_home -> {
                    val intent = Intent(this,
Classify::class.java)
                    startActivityWithAnimation(intent,
R.anim.swipe_right_enter, R.anim.swipe_right_exit)
                    finish()
                    true
                }
                R.id.action_event -> {
                    val intent = Intent(this,
Event::class.java)
                    startActivityWithAnimation(intent,
R.anim.swipe_right_enter, R.anim.swipe_right_exit)
                    true
                }
                R.id.action_recommendation -> {
                    true
                }
                else -> false
            }
        }
        val recyclerView: RecyclerView =
findViewById(R.id.recoList_RecyclerView)
        val layoutManager = LinearLayoutManager(this)
        recyclerView.layoutManager = layoutManager

        fetchSavedImages { querySnapshot ->
```

```
            displaySavedImages(querySnapshot)
        }
    }

    private fun startActivityWithAnimation(intent:
Intent, enterAnim: Int, exitAnim: Int) {
        startActivity(intent)
        overridePendingTransition(enterAnim, exitAnim)
    }

    private fun fetchSavedImages(onSuccess: (QuerySnapshot)
-> Unit) {
        firestore.collection("Recommendation Saved Images")
            .get()
            .addOnSuccessListener { documents ->
                onSuccess(documents)
            }
            .addOnFailureListener {
            }
    }

    private fun displaySavedImages(querySnapshot:
QuerySnapshot) {
        val recyclerView: RecyclerView =
findViewById(R.id.recoList_RecyclerView)

        val adapter = RecommendationAdapter(this,
querySnapshot)
        recyclerView.adapter = adapter

        val layoutManager = LinearLayoutManager(this,
LinearLayoutManager.HORIZONTAL, false)
        recyclerView.layoutManager = layoutManager

        val spacingInPixels =
resources.getDimensionPixelSize(R.dimen.spacing_between_ite
ms_linear)
```

Sample Codes for Recommendation Results

```
class ResultsRecommendation : AppCompatActivity() {
```

```
    private lateinit var selectedEvent: String
    private lateinit var selectedGender: String
    private lateinit var selectedType: String
    private lateinit var selectedColor: String
    private lateinit var selectedPattern: String
    private lateinit var storage: FirebaseStorage
    private lateinit var firestore: FirebaseFirestore

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.result_recommendation)

        FirebaseApp.initializeApp(this)

        selectedEvent = intent.getStringExtra("event") ?:
""
        selectedGender = intent.getStringExtra("gender") ?:
""
        selectedType = intent.getStringExtra("type") ?: ""
        selectedColor = intent.getStringExtra("color") ?:
""
        selectedPattern = intent.getStringExtra("pattern")
?: ""

        storage = FirebaseStorage.getInstance()
        firestore = FirebaseFirestore.getInstance()

        val recyclerView: RecyclerView =
findViewById(R.id.recyclerView)
        val layoutManager = GridLayoutManager(this, 2)
        recyclerView.layoutManager = layoutManager

        val textTags: TextView =
findViewById(R.id.textTags)

        queryRecommendations(selectedEvent, selectedGender,
selectedType, selectedColor, selectedPattern) { documents -
>
            displayRecommendations(documents)

            if (!documents.isEmpty) {
                val firstDocument = documents.documents[0]
```

```
                val event =
firstDocument.getString("event") ?: ""
                val type = firstDocument.getString("type")
?: ""
                val color =
firstDocument.getString("color") ?: ""
                val pattern =
firstDocument.getString("pattern") ?: ""

                val tagsText = buildString {
                    if (event.isNotEmpty())
append("$event  ")
                    if (type.isNotEmpty()) append("$type
")
                    if (color.isNotEmpty())
append("$color  ")
                    if (pattern.isNotEmpty())
append("$pattern ")
                }
                textTags.text = tagsText
            }
        }
```

Sample Codes for Build Gradle

```
plugins {
    id("com.android.application")
    id("org.jetbrains.kotlin.android")
    id("com.google.gms.google-services")
    kotlin("kapt")
}

android {
    namespace = "software_engineering.project.ombre"
    compileSdk = 34

    defaultConfig {
        applicationId =
"software_engineering.project.ombre"
        minSdk = 26
        targetSdk = 31
        versionCode = 2
```

```
        versionName = "2"

        testInstrumentationRunner =
"androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            isMinifyEnabled = false
            proguardFiles(
                getDefaultProguardFile("proguard-android-
optimize.txt"),
                "proguard-rules.pro"
            )
        }
    }
    compileOptions {
        sourceCompatibility = JavaVersion.VERSION_17
        targetCompatibility = JavaVersion.VERSION_17
    }
    kotlinOptions {
        jvmTarget = "17"
    }
    buildFeatures {
        mlModelBinding = true
        viewBinding = true
    }
}

dependencies {
    implementation("androidx.core:core-ktx:1.12.0")
    implementation("androidx.appcompat:appcompat:1.6.1")

implementation("com.google.android.material:material:1.10.0
")

implementation("androidx.constraintlayout:constraintlayout:
2.1.4")
    implementation("org.tensorflow:tensorflow-lite-
support:0.3.1")
    implementation("org.tensorflow:tensorflow-lite-
metadata:0.1.0")
```

```
    implementation("com.google.firebase:firebase-firestore-
ktx:24.9.1")
    implementation("com.google.firebase:firebase-
storage-ktx:20.3.0")
 implementation("com.google.firebase:firebase-
inappmessaging-ktx:20.4.0")
    testImplementation("junit:junit:4.13.2")

androidTestImplementation("androidx.test.ext:junit:1.1.5")

androidTestImplementation("androidx.test.espresso:espresso-
core:3.5.1")
    testImplementation ("org.mockito:mockito-core:3.10.0")

    implementation(platform("com.google.firebase:firebase-
bom:32.2.2"))
    implementation
("com.google.android.material:material:1.10.0")
    implementation
("com.github.bumptech.glide:glide:4.12.0")
    implementation ("com.firebaseui:firebase-ui-
storage:8.0.0")
    implementation ("com.google.firebase:firebase-
storage:20.3.0")
    implementation ("com.google.code.gson:gson:2.9.0")
    implementation ("com.firebaseui:firebase-ui-
firestore:8.0.0")
    implementation ("com.firebaseui:firebase-ui-
database:8.0.0")
    implementation ("com.squareup.picasso:picasso:2.71828")
    kapt ("com.github.bumptech.glide:compiler:4.12.0")
}
```

Sample Codes for the Model Results

```
@Suppress("DEPRECATION")
class ModelResults : AppCompatActivity() {

    private lateinit var binding:
ActivityModelResultsBinding
    private lateinit var imageUrl: String

    private lateinit var resView: TextView
    private lateinit var imageView: ImageView
    private var resultImageUri: Uri? = null

    private val TAG = "ModelResults"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding =
ActivityModelResultsBinding.inflate(layoutInflater)
        val view = binding.root
        setContentView(view)

        resView = view.findViewById(R.id.resView)
        imageView = view.findViewById(R.id.resultImage)

        imageUrl = intent.getStringExtra("imageUri") ?: ""
        val resultText =
intent.getStringExtra("resultText") ?: ""

        binding.resView.text = resultText

        loadFirebaseImage(imageUrl)

        binding.save.setOnClickListener {
            saveResults(imageUrl, resultText)
        }

        binding.tryreco.setOnClickListener {
            navigateToChooseEventGenderActivity()
        }
        val bottomNavigationView: BottomNavigationView =
findViewById(R.id.bottomnavigation)
```

```
bottomNavigationView.setOnNavigationItemSelectedListener {
menuItem ->
            when (menuItem.itemId) {
                R.id.action_home -> {
                    val intent = Intent(this,
Classify::class.java)
                    startActivity(intent)
                    true
                }
                R.id.action_event -> {
                    val intent = Intent(this,
Event::class.java)
                    startActivity(intent)
                    true
                }
                R.id.action_recommendation -> {
                    val intent = Intent(this,
Recommendations::class.java)
                    startActivity(intent)
                    true
                }
                else -> false
            }
```