



Project Report

Innovative mobile application, Weather Wear: Where wardrobe meets the weather conditions.

Submitted by: Rhythm Sharma

Student Id: 10614346

Applied research project.

MSc in Information Systems and Computing

Dublin Business School

Under Supervision of: Agatha Mattos

January 2024

Declaration

I hereby declare that this mobile application, which I have submitted to Dublin Business School for the academic year 2022-2023 for the award of MSc. in Information Systems with Computing is the result of my own investigations and findings.

I confirm that this work has not been previously submitted for any degree, diploma, or other qualifications at this or any other institution. Where I have consulted the published work of others, it is always clearly attributed, and where I have quoted from the work of others, the source is always given. All other contributions are acknowledged. This work complies with the guidelines for academic integrity and plagiarism of the institution to which it is submitted.

Signed: Rhythm Sharma

Student ID: 10614346

Date: 07-01-2024

Acknowledgment

I would like to extend my sincere thanks to my dissertation supervisor **Agatha Mattos**, for her constant support, insightful guidance, and encouragement throughout my research journey. Her valuable feedback has been crucial in enhancing the quality and depth of this study. My gratitude also goes to the distinguished faculty at Dublin Business School, whose continuous support and sharing of knowledge, technologies, and tools were fundamental to the completion of my dissertation. Finally, my heartfelt appreciation is for my family and friends, whose steadfast belief and encouragement have been a source of inspiration. This dissertation stands as a testament to their support and motivation.

Abstract

The "Weather Wear" application represents a groundbreaking fusion of real-time weather forecasting and digital wardrobe management. This innovative app utilises cutting-edge technologies such as Expo, React JS, and Firebase, alongside a complimentary Weather API, to deliver a user-centric and intuitive experience. Key features include an organised digital wardrobe, tailored clothing recommendations based on current weather conditions, and an easy-to-use UI. The report describes the development process, technological infrastructure, and the app's unique features. It focuses on the app's cross-platform interoperability and smart algorithm for outfit recommendation. Future improvements, such as geolocation capabilities, configurable user profiles, and a social networking component, are being discussed. The application uniquely addresses the everyday challenge of selecting suitable attire in fluctuating weather scenarios, exemplifying a creative merger of technology and daily life conveniences.

Table of Contents

Declaration	1
Acknowledgment	2
Abstract	3
Introduction	7
Problem Statement	7
Why did we get this idea to start a project?	7
Key Aspects of this application:	7
Development Stages and Technological Framework	8
Limitations	9
Structure of the report	9
Literature Review	10
Weather based recommendations:	10
1.1 Clothing Recommendation- Paper I	10
1.2 A Study on the Clothes Recommendation- Paper II	10
Digital Wardrobe Applications:	11
2.1 Whering	11
2.2 Capsule Wardrobe Apps	11
Methodology	13
3.1 Flow Chart-	13
3.1.2 User Entry Point	14
3.1.3 Home/Weather Screen	15
3.1.4 Wardrobe Screen	18
3.1.4.1 Clothing Categories and Addition	19
3.1.4.2 Database Interaction	22
3.1.5 Profile Screen	23
3.1.6 Password Reset	25
3.2 Implementation	28
3.2.1 EXPO-	28
Use of Expo in Weather Wear-	29
3.2.2 React:	32
React Native components:	32
Hooks	32
Promises	33
3.2.3 Firebase:	35
Firebase Firestore	35
Firebase Authentication	36
3.2.4 Prototype:	37
Findings & Discussions	42
How do our clothes affect our health?	42
1.1 Material Choices and Health Impacts:	42
1.2 The Role of Fit and Comfort:	42
1.3 Indirect Health Effects and Protection:	42
1.4 Impact on Workplace Safety and Sleep Quality:	42
Weather forecasts play an important role.	43
What to wear?	43
Key Features	45

Platform Independent application-	45
Advantages of This Approach:	46
Using Weather API-	46
Wardrobe Management-	46
Decision Making Algorithm-	46
Future Work	48
Editable Wardrobe Items	48
Geo Location	48
User Profile	48
Laundry	49
Clothes with RFID or QR Code to scan the suitable temperature.	49
Social element	49
Essential clothing items required.	49
Subscription plans for more items space	50
More categories	50
Conclusion	51
References-	53
Additional Links	55

Table of Figures

Figure 1 Flow Chart	13
Figure 2 Login Screen	14
Figure 3 Sign up interface.	15
Figure 4 Fetching the data from API and Database	16
Figure 5 Outfit suggestion in Dublin	17
Figure 6 Outfit Suggestion for Noida	17
Figure 7 Outfit suggestion for Dubai	18
Figure 8 Wardrobe Screen with categories	19
Figure 9 Add Items in each category	20
Figure 10 Users can enter their preferred temperature for each item.	20
Figure 11 When the item is added it shows this alert.	21
Figure 12 The Items show in the category list.	21
Figure 13 Added New item to the database with Waterproof check.	22
Figure 14 Latest item in the database.	22
Figure 15 Profile Screen	23
Figure 16 Added profile image for Test Two user.	24
Figure 17 Forgot password link sent alert.	25
Figure 18 Firebase authentication template for Password reset mail.	26
Figure 19 Reset password link sent to user's Gmail.	26
Figure 20 Expo development server	29
Figure 21 Error log in expo go while developing weather wear.	30
Figure 22 eas.json code screenshot	31
Figure 23 Build Dashboard From Expo	31
Figure 24 Cloud Firestore database collection	35
Figure 25 Password hashing	37
Figure 26 Weather wear on paper	37
Figure 27 Storage details of the application in the android device.	38
Figure 28 Outfit suggestion	39
Figure 29 Prototype Design plan	39
Figure 30 Decision-making Algo Planning.	40
Figure 31 Items Stored in Tops	41
Figure 32 Experience map of the user	43
Figure 33 Future categories	50

Introduction

The Weather Wear app is a unique combination of weather information coupled with clothing recommendations, aimed to help users make informed decisions about their everyday outfit based on the weather. In an era where convenience and accuracy are crucial, this app stands out by offering real-time, location-specific weather updates as well as personalised outfit recommendations. The Weather Wear app introduces a groundbreaking concept: it equips users with essential information to dress appropriately, blending comfort, safety, and style, regardless of the weather. This feature is particularly vital in regions with unpredictable weather patterns, where choosing the right attire can significantly affect an individual's well-being. The app's smart algorithm accounts for various weather parameters such as temperature, humidity, and precipitation, offering not just fashionable but also sensible clothing choices. It streamlines the morning routine, allowing users to efficiently plan their attire without the usual hassle.

The app's real-time adaptability to weather changes means users are always prepared, come sunshine or rain. Weather Wear is more than just an app; it's a comprehensive assistant for daily life, seamlessly integrating into the user's routine. It embodies the perfect blend of technology and practicality, enhancing both functionality and aesthetics. Users can step out confidently, knowing they are well-dressed for any weather scenario, effortlessly merging comfort, safety, and style. This app simplifies decision-making by combining weather forecasts and wardrobe choices into one platform, saving valuable time. It addresses a real-life challenge, helping individuals focus on more important aspects of their lives without the worry of weather uncertainties. Weather Wear is not just a utility tool; it's an essential part of daily living, aiding users in navigating their day with assurance and style.

Problem Statement

Why did we get this idea to start a project?

As an immigrant, my experience with the Weather Wear app began with personal challenges. It was difficult to adjust to a new country's weather, which differed from my own. I found myself often checking the daily forecast to decide on suitable items of clothing, which was necessary given the frequent and abrupt weather changes. This experience revealed a large gap in available resources, sparking the concept for the Weather Wear app. I envisioned an app that would deliver not only weather updates but also useful wardrobe suggestions. The application was created to ease daily clothing decisions and help people like me move comfortably to unusual weather patterns by combining convenience with essential functionality.

Key Aspects of this application:

In this application we have three main aspects to work on. They are going to make it one of a kind. Although we have researched on the topic there are many applications for weather and have digital wardrobes and smart closets available. But we don't have any applications for digital wardrobe with weather integration.

First and the main aspect of this project is focused on developing a wardrobe section to upload images from actual user's wardrobe collections. In which we have four categories to segregate the clothes uploaded by users to the Firestore database.

Second aspect of this application is to supply current weather conditions. By integrating a free Weather API into the Weather Wear app could significantly enhance its functionality and user experience, offering real-time, location-specific, and right clothing recommendations based on current weather conditions. This attention to detail is complemented with an intuitive and user-friendly layout that allows users of all ages and technological skill levels to simply explore and benefit from the app. Personalization features go a step further, allowing the app to adapt outfit recommendations and weather notifications to each user's unique choices and needs. This tailored approach guarantees that the app stays relevant and valuable in its users' daily life, whether they are planning their work wardrobe, getting ready for a relaxing outing, or prepping for outdoor adventures.

The third and crucial aspect of this application involves using the advanced capabilities of Firebase's Firestore, a NoSQL database, for data management. This powerful database service will underpin the functionality of the app by efficiently handling and storing the large volumes of user-uploaded wardrobe images and corresponding metadata. Firestore's NoSQL structure is ideal for this purpose due to its scalability, speed, and flexibility in managing unstructured data. It also ensures real-time syncing and high availability, which are critical for providing users with a seamless and responsive experience. Moreover, Firestore's robust security features will protect users' data, ensuring privacy and safety in data handling. Integrating this technology is pivotal for the application's performance, enabling it to handle complex data operations with ease and reliability.

Development Stages and Technological Framework

The development of the Weather Wear application is structured into four critical stages: research, design, prototype, and evaluation. Our primary focus during the design stage is to create an application that is not only visually appealing but also highly user-friendly. To achieve this, we utilised a combination of Expo, React JS, and Firebase.

Expo provides a convenient platform for developing React Native applications, enabling us to efficiently manage the frontend interface and interactions (6). React JS, known for its robust and dynamic capabilities in building user interfaces, is essential for the app's responsive and intuitive design. Firebase, with its comprehensive backend services including Firestore for database management and Authentication for user security, supports the app's data handling and storage requirements.

The final product of these developmental phases and technological integrations is a platform-agnostic, interactive application prototype. This prototype exemplifies the harmonious fusion of smooth design and practical functionality. It stands as proof of our dedication to creating an innovative and useful tool for daily weather-related clothing decisions. Furthermore, this

prototype undergoes thorough testing for user interaction, design effectiveness, and technical robustness.

Limitations

- **Connectivity Requirements:**

Being an online application, it requires a stable internet connection for real-time weather updates and database access, limiting its use in areas with poor connectivity.

- **Hardware and Software Compatibility:**

The app needs to be compatible with a wide range of devices and operating systems, which can be challenging to maintain.

- **Adaptation to Diverse Fashion Styles:**

The app may struggle to cater to the vast array of personal fashion preferences and cultural clothing styles, which can limit its applicability across different user demographics.

Structure of the report

- **Introduction:** This section presents the Weather Wear app, emphasising its goals and significance in the areas of lifestyle improvement and technical growth.
- **Literature Review:** An in-depth review of current research on digital wardrobe solutions, weather prediction technologies, and their use in everyday life. This chapter looks at current trends and highlights important challenges and opportunities.
- **Methodology and Implementation:** This chapter describes the strategic approach and development of the Weather Wear app, covering design methodologies, the technological framework (React, Expo, Firebase), user interface design, weather API integration, and Firestore database configuration along with the firebase authentication.
- **Findings and Discussion:** An analysis of significant finding from the research phase, later in the development and testing phases, as well as a discussion of their outcomes.
- **Key Features:** An in-depth look at the Weather Wear app's unique features and how they help users.
- **Future Work:** Investigate potential areas for app development and enhancement in future editions.
- **Conclusion:** Summarises the project's successes, assesses the app's influence on users, and makes recommendations for future study and development.

Literature Review

The goal of this literature is to improve understanding of the research area, with a particular emphasis on two key components, the concept of a digital wardrobe and the implementation of weather-based recommendations. By analysing various applications, the evaluation decides how each individually addresses the integration of technology with personal fashion, as well as their success in delivering outfit choices based on both individual style preferences and external factors such as weather.

Weather based recommendations:

1.1 Clothing Recommendation- Paper I

The paper "Weather-to-Garment: Weather-oriented Clothing Recommendation" by Liu et al. (2017) presents a system that recommends clothing from a user's personal collection based on weather conditions. This innovative system addresses the complex challenge of aligning clothing attributes with varying weather categories. Using a scoring function and a comprehensive dataset, it intelligently suggests the most appropriate or complementary items to match the user's chosen garment. This research is significant in the fields of computer vision and fashion technology, highlighting the fusion of practical needs with sophisticated algorithmic solutions (2).

1.2 A Study on the Clothes Recommendation- Paper II

The paper "A Study on the Clothes Recommendation for a Cold Region of Japan" offers an in-depth exploration into the relationship between clothing insulation, measured as Clo value, and actual clothing choices of individuals in Iwate, Japan, during colder seasons. The study employs surveys to collect detailed data on the types of clothes worn in winter and autumn. It primarily looks to establish a tangible link between the theoretical Clo value and the real-world clothing preferences of people in this cold region.

Surprisingly, the findings suggest that residents in Iwate tend to wear clothing with a lower insulation value (Clo value) than what might be expected given the colder temperatures. This preference for lighter clothing in a relatively cold environment highlights a significant gap in existing clothing recommendation systems, which often rely solely on temperature and Clo values without considering regional or cultural clothing practices. This research is pivotal in refining clothing recommendation systems, particularly in applications like Weather Wear, which could benefit from incorporating these insights. By understanding the actual clothing habits and preferences in various climates, such systems can provide more accurate and contextually relevant advice. This study underscores the importance of regional differences in clothing choices and the need for recommendation systems to adapt to these nuances, ensuring users receive advice that is not only technically sound but also practically applicable and culturally sensitive. (3)

Digital Wardrobe Applications:

2.1 Whering

In the digital wardrobe management landscape, the Whering app stands out for its comprehensive approach to personal style curation and wardrobe organisation. Users are encouraged to digitise their wardrobe by uploading photos or importing images, which becomes the foundation for a suite of features that promote efficient and creative wardrobe utilisation (4). From creating unlimited outfits and lookbooks to planning ensembles in advance, Whering facilitates a user-friendly fashion experience. It further innovates with a 'wardrobe shuffle' feature to help users break from routine styles and discover new combinations within their existing collection. The application also offers personalised outfit suggestions, wear tracking for insights into usage patterns, and tools for garment care. Additionally, it promotes sustainable fashion habits through circular shopping options and provides an inspiration hub with Wishlist & Moodboards for future wardrobe planning (4). These capabilities collectively foster a proactive and sustainable relationship between users and their clothing, emphasising longevity and personal expression without relying on weather-based predictions.

2.2 Capsule Wardrobe Apps

From panaprium's Article, the digital wardrobe management space is diverse, with apps like Save Your Wardrobe promoting sustainability through personal styling and wardrobe digitization, encouraging users to reconnect with their closet. (5) Smart Closet from Cairo extends this by allowing the import and editing of clothes, easing daily outfit management (5). Stylebook from Maryland offers over 90 tools for wardrobe curation and outfit planning, emphasising recycling and inclusivity (5). Pureple, based in Washington, learns user style to suggest outfits, while Lookscope from New York supports closet management and style enhancement. (5) Latitude in New York offers a community-driven platform for outfit inspiration and supports the circular economy. Closet Love from Canada simplifies daily wardrobe decisions by digitising clothes and planning outfits. (5) ShopLook provides an outfit maker for creative expression, Combyne from Germany allows sharing of looks for instant feedback, and 30 Wears encourages wearing each garment multiple times to reduce consumption (5). Wardrobe from New York offers a rental service for luxury fashion, and The Nu Wardrobe from Ireland enables fashion sharing to reduce waste (5).

To end, the literature study goes into digital wardrobe apps and platforms, with an emphasis on outfit style and garment management. Two primary topics run throughout the review: the notion of a digital wardrobe and the deployment of weather-based apparel recommendations. It investigates numerous applications, evaluating their success in combining technology with personal fashion, as well as their capacity to provide outfit options impacted by individual style preferences and external circumstances such as weather.

Key studies include "Weather-to-Garment: Weather-oriented Clothing Recommendation" by Liu et al. (2017), which presents a system that recommends clothing based on weather conditions, combining computer vision and fashion technology. Another significant paper is "A Study on the Clothes Recommendation for a Cold Region of Japan", which explores the

relationship between clothing insulation and actual clothing choices in Iwate, Japan. This study reveals a preference for lighter clothing despite colder temperatures, indicating a gap in existing recommendation systems that often neglect regional and cultural clothing practices.

The review also highlights digital wardrobe applications Whering (4), which allows users to digitise their wardrobe and offers features for outfit creation, wear tracking, and sustainable fashion practices. Other notable apps include Save Your Wardrobe, Smart Closet, Stylebook, Pureple, Lookscope, Latitude, Closet Love, ShopLook, Combyne, 30 Wears, Wardrobe, and The Nu Wardrobe, each contributing uniquely to wardrobe management and personal styling. These range from promoting sustainability, offering outfit planning tools, supporting the circular economy, to facilitating fashion sharing and reducing waste. This diverse array of apps demonstrates the growing intersection of technology and fashion, with a focus on user-centric design and sustainability.

Methodology

3.1 Flow Chart-

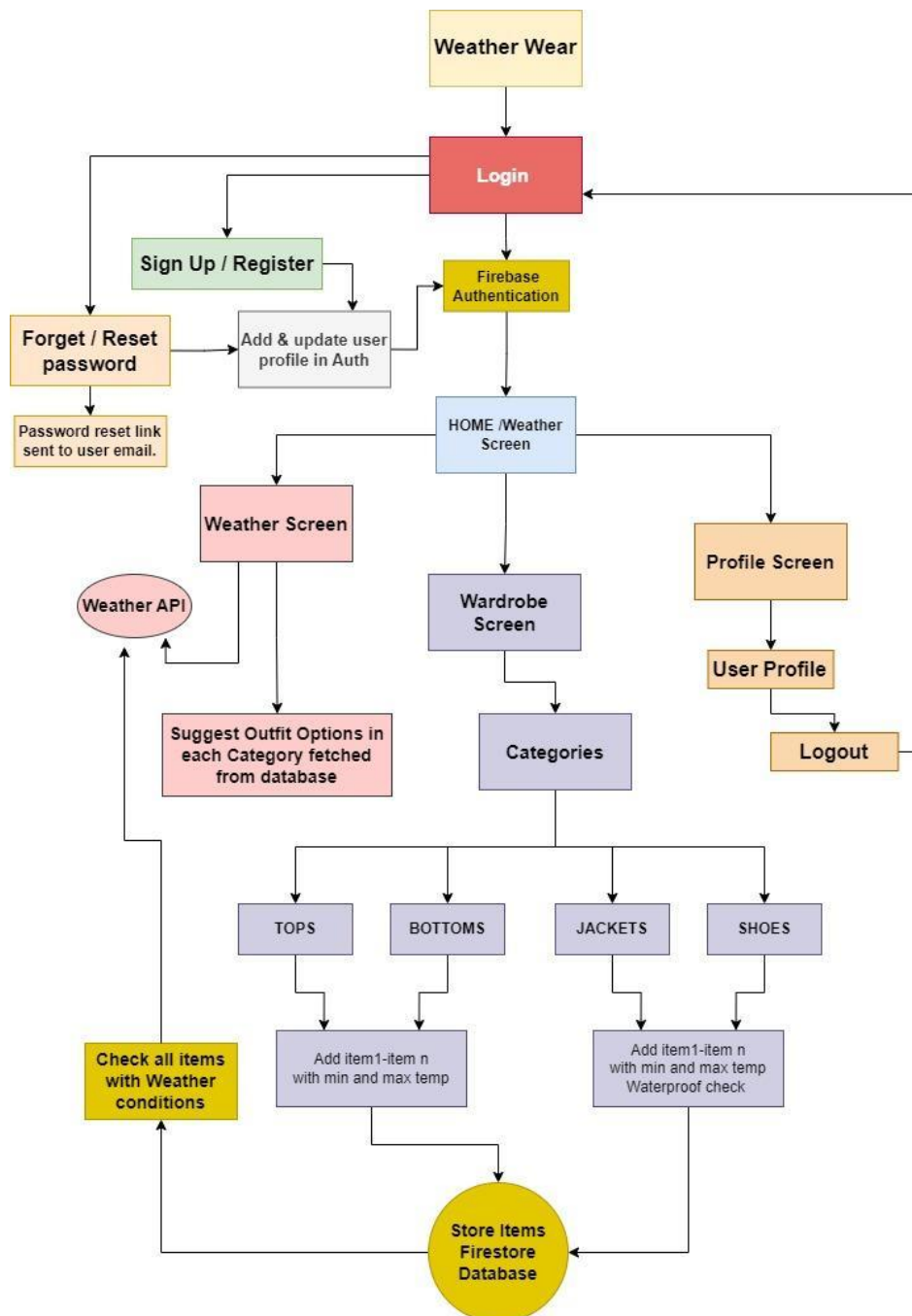


Figure 1 Flow Chart

This flowchart shows a straightforward user interface designed for ease of use, with a focus on personalising the wardrobe management experience based on weather conditions. It outlines the user journey and functionality of the Weather Wear application:

3.1.2 User Entry Point

Users start by either signing up/registering or logging in. For new users, the sign-up process will involve creating a profile that is added to Firebase Authentication. Firebase Authentication offers backend services, simple SDKs, and pre-built UI libraries for authenticating users in your app. It allows for authentication by passwords, phone numbers, and popular federated identity providers such as Google, Facebook, and Twitter.

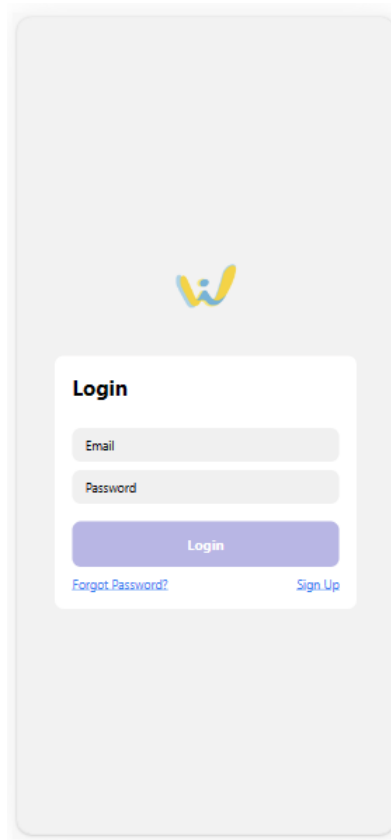


Figure 2 Login Screen

A mobile app mockup for a sign-up screen. At the top, there is a back arrow and the text 'SignUp'. Below this, there are four input fields: the first contains 'test2@gmail.com', the second contains 'Test Tiwo', the third contains seven asterisks, and the fourth contains seven asterisks with a password strength icon on the right. Below the input fields is a purple 'Sign Up' button. At the bottom, there is a link that says 'Go Back to Login'.

Figure 3 Sign up interface.

3.1.3 Home/Weather Screen

After login, users arrive at the home screen, which presents the current weather information obtained via Weather API. As we just added one item for suitable weather conditions. Therefore, it is showing only from one category.

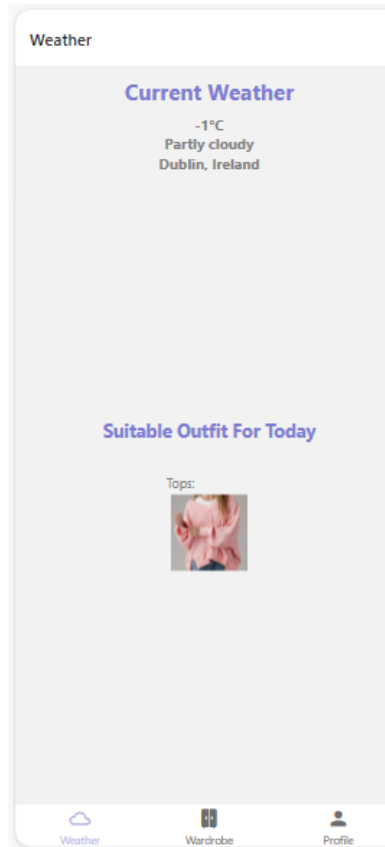


Figure 4 Fetching the data from API and Database

We have Integrated a free Weather API into the Weather Wear app. That significantly enhances its functionality and user experience, offering real-time, location-specific, and right clothing recommendations based on current weather conditions.

Outfit Suggestions is based on the data fetched from the database, the app suggests outfit options categorised into tops, bottoms, jackets, and shoes, proper for the current weather conditions. We tested the application for different locations and as per the data provided by the user in the wardrobe. The algorithm suggests the outfit options from available items. If a user wants to get more options, they should upload all the items to the application from their physical wardrobe to the virtual wardrobe in weather wear.

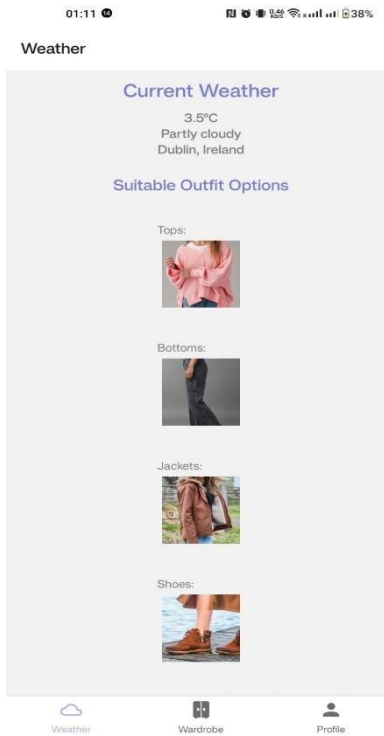


Figure 5 Outfit suggestion in Dublin

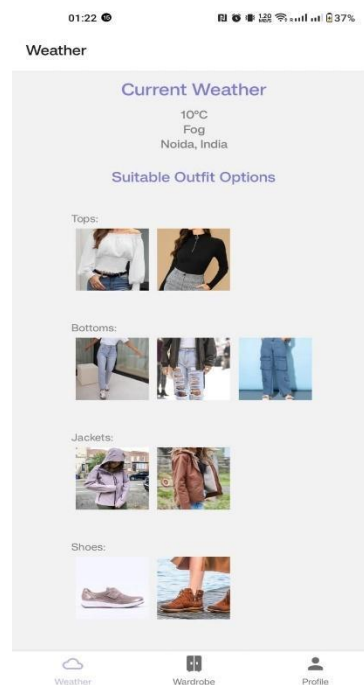


Figure 6 Outfit Suggestion for Noida

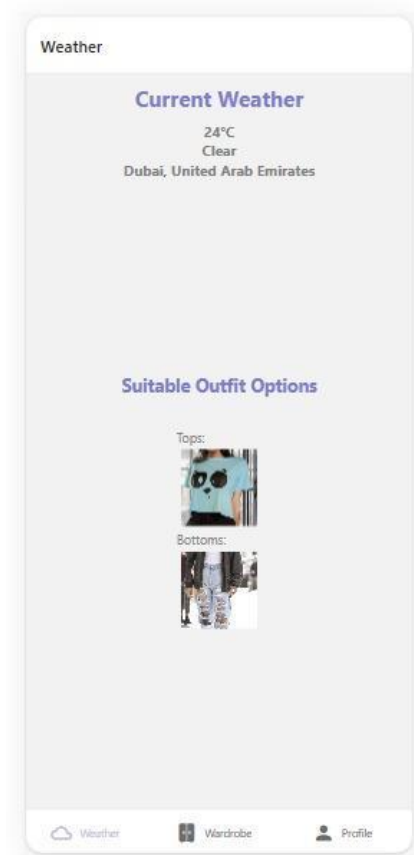


Figure 7 Outfit suggestion for Dubai

From figure 5 to figure 7 we have three separate locations for a user who has a few number items in the wardrobe. The decision-making algorithm is suggesting the different outfits according to the weather conditions in each location. The items in the wardrobe are added by the user's preference of suitable temperature and comfort.

3.1.4 Wardrobe Screen

From the Home screen, users can navigate to the Wardrobe screen where they can manage their clothing items. The active screen colour changes when the user changes the screen. So, in the next figure 8 we have an active wardrobe screen.

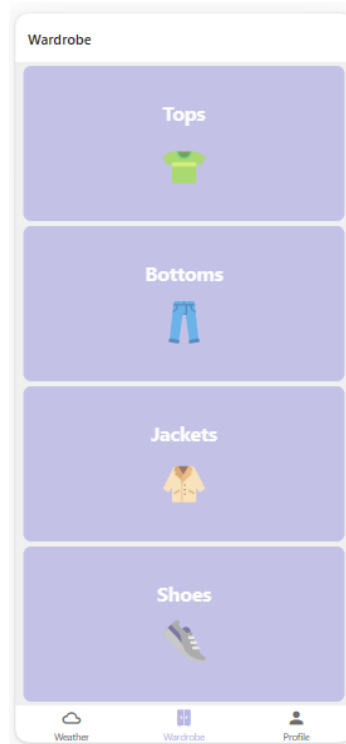


Figure 8 Wardrobe Screen with categories

3.1.4.1 Clothing Categories and Addition

Users can check their wardrobe items divided into categories and add items specifying the minimum and maximum temperature range for each. For jackets, there is an added waterproof checkbox to indicate if the item is suitable for rain.

Figure 9, illustrates the process of adding items to various categories in the app, showcasing the user interface for wardrobe organisation and item categorization.

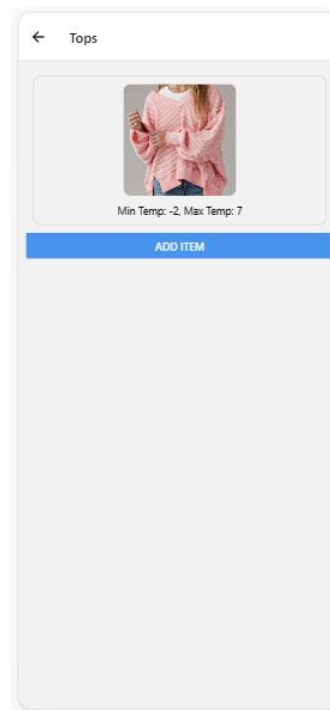


Figure 9 Add Items in each category

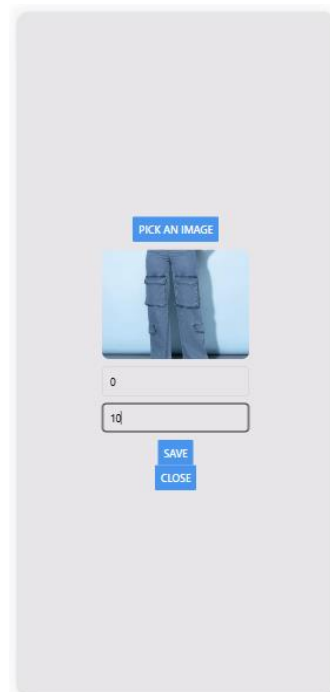


Figure 10 Users can enter their preferred temperature for each item.

Figure 10, shows the feature allowing users to set their preferred temperature range for each clothing item, enhancing personalised outfit recommendations.

In figure 11, Displayed here is a notification alert that appears when a user successfully adds an item to their digital wardrobe, confirming the action will close the modal and refresh the for fetching the updated item.

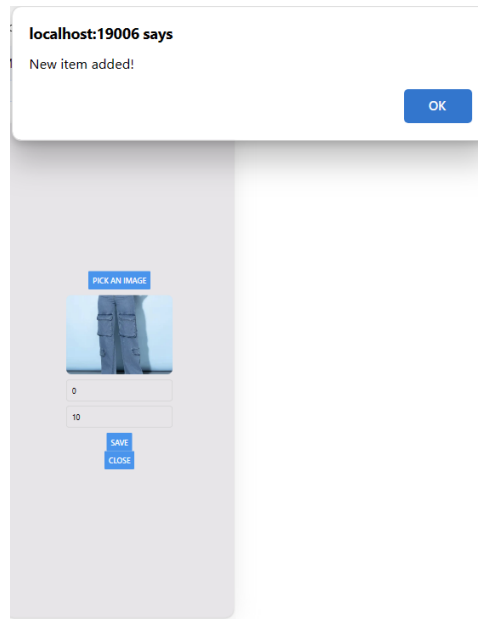


Figure 11 When the item is added it shows this alert.

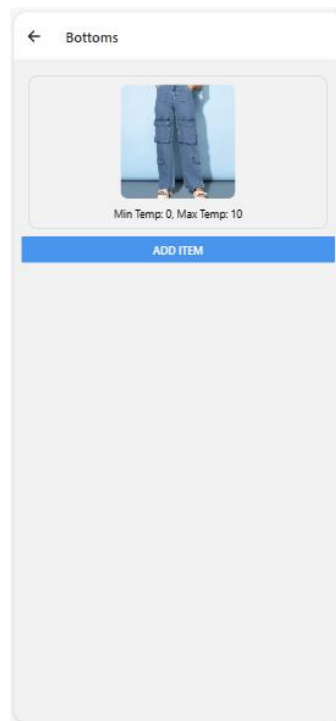


Figure 12 The Items show in the category list.

In figure 12, it presents how added items are displayed within the category list, offering a clear and organised view of the user's digital wardrobe.

This figure 13 shows the interface after adding a new item to the database, featuring the option to mark it as waterproof, which aids in providing more tailored outfit recommendations based on weather conditions.

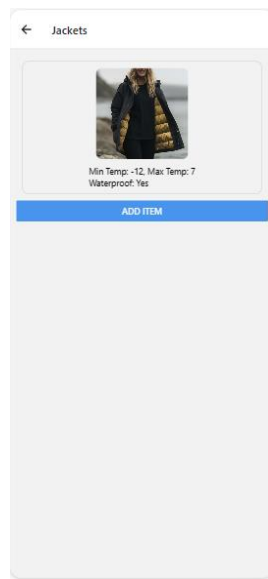


Figure 13 Added New item to the database with Waterproof check.

3.1.4.2 Database Interaction

All the clothing items and their details are stored in the Firestore Database, which allows for persistent, real-time data storage and retrieval.

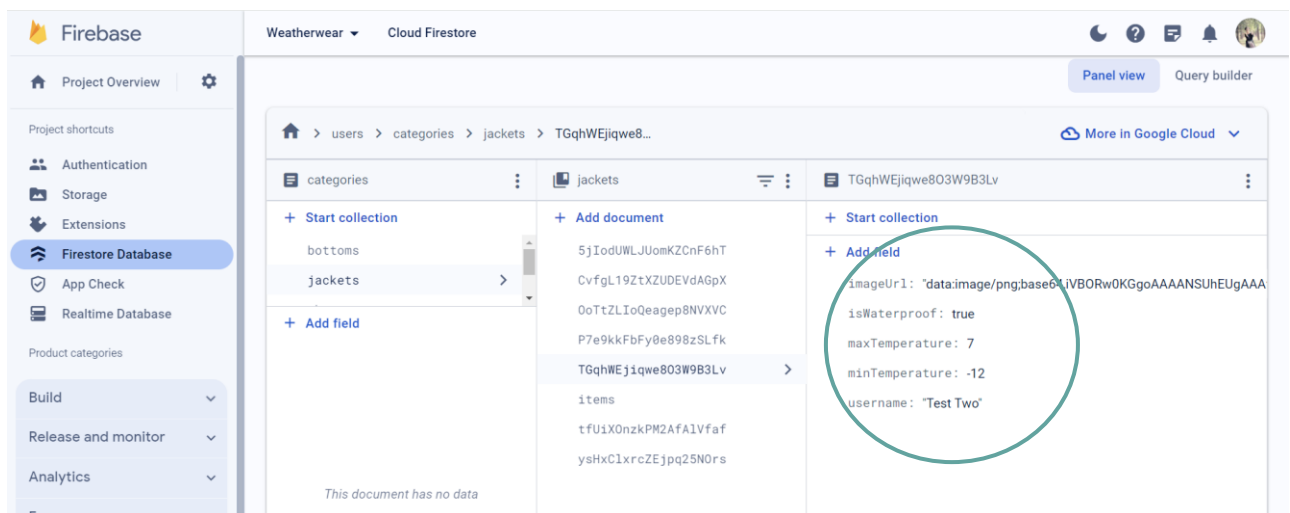


Figure 14 Latest item in the database.

3.1.5 Profile Screen

Users can also access their profile screen where they can view the profile.

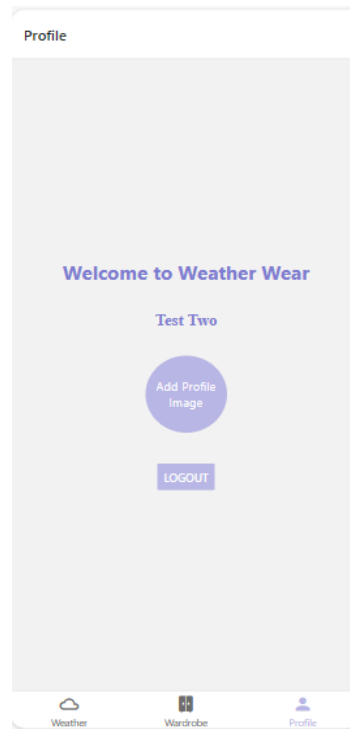


Figure 15 Profile Screen

In the current app prototype, the profile screen is very simple, displaying only the user's username and allowing for profile image upload. However, in future versions, we intend to considerably improve the profile page. This will have more features in future versions such as the option to update personal information, link social network accounts, change the profile image, change the username, and select different themes, among others. These modifications are intended to create a more personalised and engaging user experience.

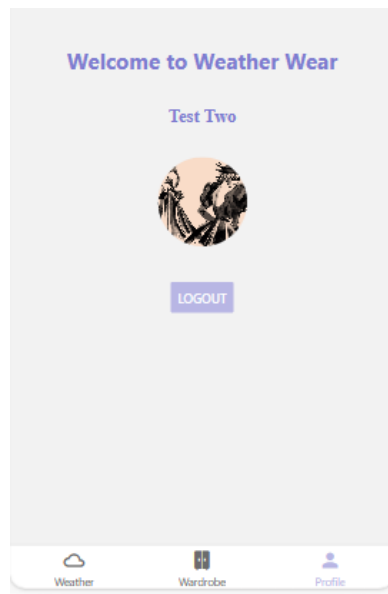


Figure 16 Added profile image for Test Two user.

This image shows the profile of the 'Test Two' user with a newly added profile picture, illustrating how users can personalise their accounts by uploading images in the current version of the app.

Lastly, we have a logout button on the profile screen. When users are finished, they can log out of the application. Which will redirect them to the login page.

3.1.6 Password Reset

In case the user forgets the password, they can reset it by clicking the forget password on the login page.

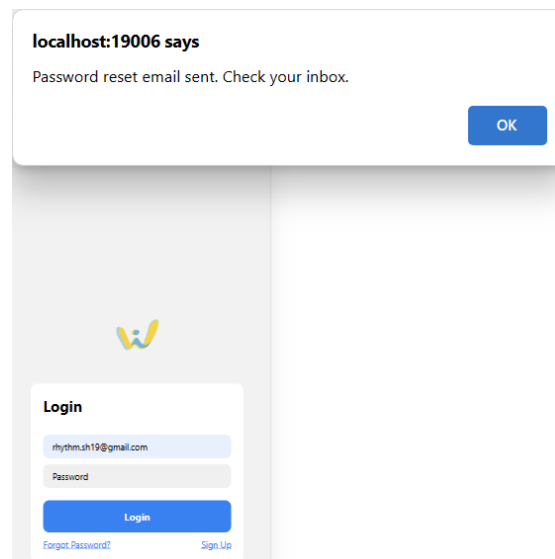


Figure 17 Forgot password link sent alert.

For password reset functionality, our application utilises Firebase Authentication's standard template. This template, as shown in the attached screenshot figure 18, is an essential part of `u s e r a c c o u n t a d m i n i s t r a t i o n .`

It can be tailored to meet unique requirements or branding rules, guaranteeing a smooth user experience while retaining the app's visual consistency and functionality.

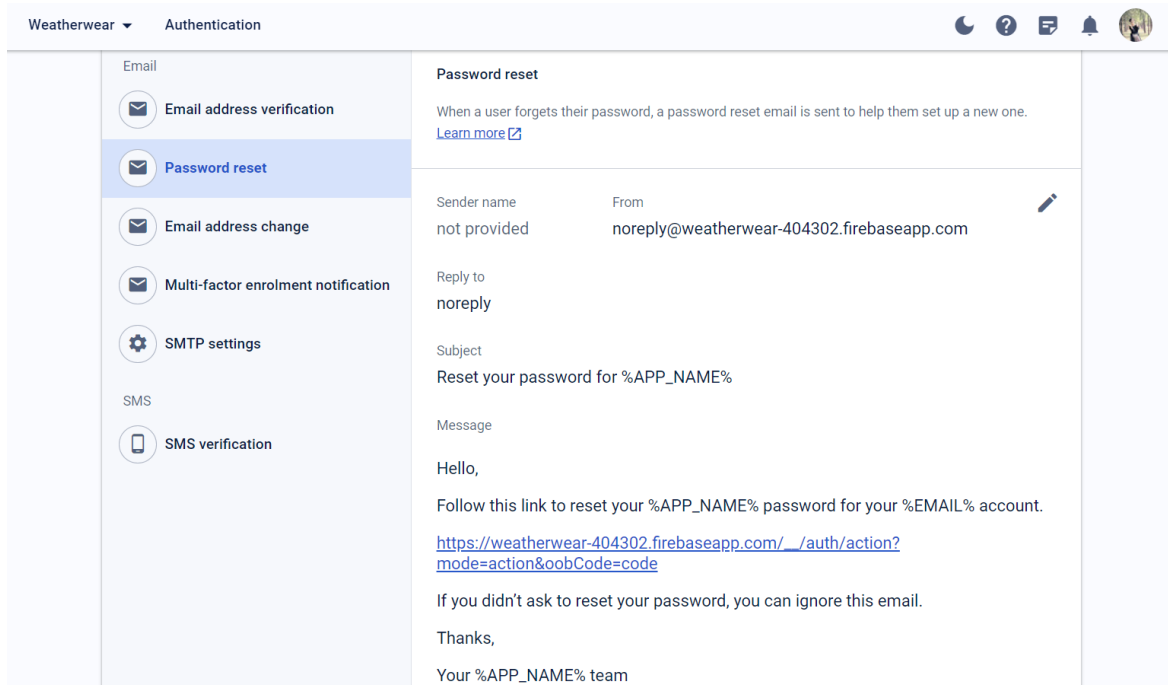


Figure 18 Firebase authentication template for Password reset mail.

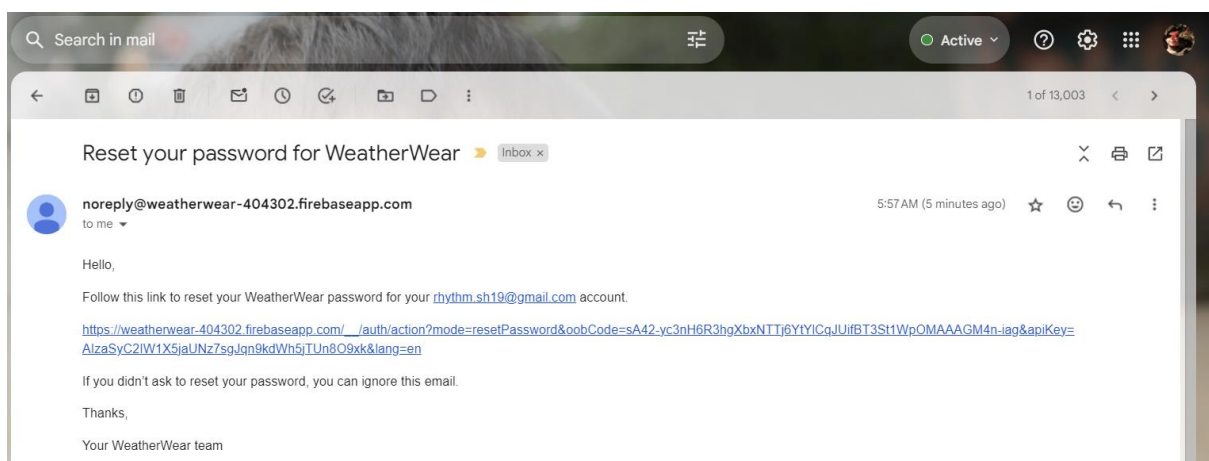


Figure 19 Reset password link sent to user's Gmail.

Figure 19 is the screenshot of the reset password link that has been sent to a user's Gmail account, illustrating the process of secure password recovery facilitated by the app's integration with Firebase Authentication.

3.2 Implementation

Implementing Expo, React, and Firebase in an application development project is a diverse process, with each adding unique advantages and functions to the application. We have taken help from this video as a tutorial on how to build a simple authentication flow with React Native, Firebase, and Expo. The link for the video is here (12). The video covers the following topics:

- Setting up the project with Expo and React Navigation
- Creating the login and home screens
- Using Firebase Authentication to sign in and sign out users.
- Displaying the current user's email
- Handling errors
- The video also includes a demo of the app in action.

The video is well-paced and easy to follow, and the instructor does a good job of explaining the concepts (12). This video is highly recommended to anyone who is interested in learning more about React Native and Firebase.

3.2.1 EXPO-

Expo is an open-source framework for developing universal native apps for Android, iOS, and web using JavaScript and React. It streamlines the development process by giving a set of tools and services for creating, deploying, and swiftly iterating on iOS and Android apps using the same JavaScript/TypeScript codebase. Expo allows developers to access mobile devices' native APIs without writing any native code, making it easier to implement features such as camera access, push notifications, and location services. In this case we have chosen Expo because it is an open-source development environment (6). Due to the expo, we didn't need the android studio.



Figure 20 Expo development server

In figure 20, we have the scripts:

- start: Launches the app with Expo.
- android: Specifically launches the app for Android devices.
- ios: Starts the app for iOS devices.
- web: Launches the application on web platforms.

Use of Expo in Weather Wear-

While developing the Weather Wear app, we found that using Expo provided several important benefits that fit our project goals and needs (6). Here's how we leverage Expo in our Weather Wear app:

- **Efficient Development Cycle:** With Expo, our development and prototyping phases become faster and more efficient. This is important to us as it allows rapid iteration based on user feedback and testing.
- **Cross-Platform Capability:** One of the biggest benefits for us is being able to develop a single codebase that works seamlessly on iOS and Android. This not only saves development time but also ensures greater accessibility to our applications.
- **Enhanced User Interface:** The availability of rich libraries and UI components in Expo has allowed us to create an attractive and user-friendly interface. This is especially important for effectively displaying weather forecasts and clothing recommendations.
- **Simplified Updates:** The ability to broadcast live updates is a game changer for us. This enables faster deployment of new features or critical updates, improving our application's response to changes in weather data.
- **Robust Testing and Debugging:** Expo's tools were invaluable in testing and debugging our application. This ensures that we can maintain a stable and reliable user experience,

which is important for user satisfaction. In figure 21, we have an error log for the application.

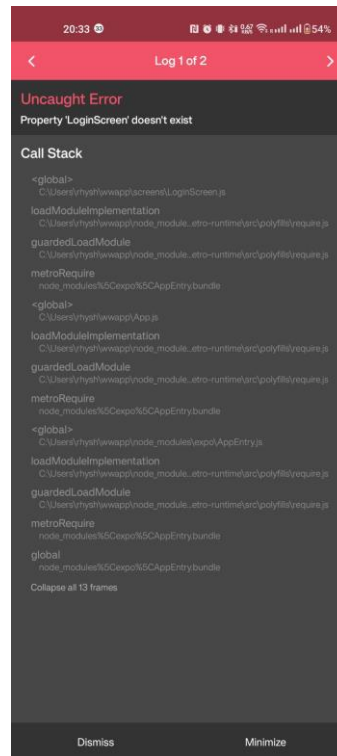


Figure 21 Error log in expo go while developing weather wear.

- Community and Resources: Access to Expo's vast community and documentation has been a huge support in our development process, especially when troubleshooting complex issues (6).
- Integration with external services: We integrate various third-party APIs for real-time weather data, and Expo has made this process easier and more manageable.
- Building apk for android: EAS (Expo Application Services) is a tool provided by Expo. When developing Android apps with EAS Build, our default file format is the Android App Bundle (AAB/.aab), which is ideally suited for distribution through the Google Play Store (7). However, it's important to note that AABs cannot be directly installed on Android devices or emulators. For direct installations, such as testing on a device, we need to create an Android Package (APK/.apk) file instead. This distinction ensures that we utilise the appropriate file format for each specific deployment scenario. We have a screenshot in figure 22, we need to update the eas.json file with the same code to build an apk which we can download from the dashboard (7). We have a figure 23 for dashboard preview of the available download in the build artifact.

eas.json

Copy

```
{
  "build": {
    "preview": {
      "android": {
        "buildType": "apk"
      }
    },
    "preview2": {
      "android": {
        "gradleCommand": ":app:assembleRelease"
      }
    },
    "preview3": {
      "developmentClient": true
    },
    "preview4": {
      "distribution": "internal"
    },
    "production": {}
  }
}
```

Now you can run your build with the following command:

Terminal

Copy

```
- eas build -p android --profile preview
```

Figure 22 eas.json code screenshot

Build artifact		APK					<div><div>Download</div><div></div></div>	
Status	Start time	Wait time	Queue time ?	Build time	Total time	Availability ?		
<div><div></div>Finished</div>	Jan 6, 2024 12:57 AM	None	49s	10m 41s	11m 31s	28 days		

Figure 23 Build Dashboard From Expo

In short, adopting Expo (6) for our Weather Wear app has significantly streamlined our development process, allowing us to focus on delivering a high-quality, feature-rich app that meets our users' needs in a timely and effective manner. Completes the prototype precisely.

3.2.2 React:

React, a JavaScript library developed by Facebook, is used for building user interfaces, especially for single-page applications. It helps in creating interactive UIs, efficiently updating and rendering components when data changes. React's component-based architecture enables developers to build reusable UI components, manage state effectively, and create large-scale, dynamic web applications with ease (15).

React Native components:

In weather wear, we are using React Native for implementing the applications logic. Here is a brief of the components we have used in development.

View: This is a core container component that allows for layout using flexbox, style, touch handling, and accessibility controls. It's used for wrapping and styling UI components.

Text: A component that displays text. It is used to display text associated with photos (such as temperature ranges in our app).

Button: Provides a basic button component for tasks like adding an item, selecting an image, saving, and dismissing the modal.

TextInput: Allows users to enter text. In your app, it is used to enter the lowest and maximum temperatures.

Modal: A component that displays material above an enclosing view. It creates a modal panel for adding new items.

Image: Displays images. This code is used to display the selected picture and the item images in the list.

ScrollView: It is a container that allows content to be scrolled. It covers the main view, allowing you to browse through the list of items and the form in the modal.

StyleSheet: Styles the components using CSS.

Hooks

useState: Using this hook to declare state variables. In our code, it is used for:

- *data*: to hold the list of objects retrieved from the Firestore.
- *modalVisible*: allows you to control the modal's visibility.
- *pickedImage*: to save the URI of the chosen picture.
- *minTemperature and maxTemperature*: are used to hold temperature inputs.
- Used to create state variables *wardrobeData*, *outfitSuggestions*, and *loading*.
- *wardrobeData* and *outfitSuggestions* store arrays of clothing items for different categories (tops, bottoms, jackets, shoes).
- *loading* is a boolean that tracks whether the component is in the process of fetching data.

useEffect: Used to perform side effects in the component. It is called once every component setup to retrieve items from the Firestore using the *fetchItems* method. In *outfit.js* *useEffect*

hook is also used for generating outfit suggestions based on the current temperature and description props, as well as the fetched wardrobeData. It runs whenever these dependencies change.

These components and hooks (14) efficiently integrate UI elements, state management, and external database queries to deliver a dynamic and responsive user experience by recommending clothes based on weather.

Promises

In our Weather Wear application, we extensively use Promises, particularly through the `fetch()` function and the `async/await` syntax, to streamline handling asynchronous operations in React.js. This approach is vital for effectively managing API requests and data loading processes (13).

One key implementation of Promises in our app is when we retrieve weather data and wardrobe suggestions. Here's how we've applied these concepts:

API Requests with Async/Await:

When our app needs to fetch weather data or user wardrobe information, it makes API requests to external services. These operations are inherently asynchronous – we request data from a server and must wait for a response (13).

To handle this elegantly, we use `async/await`. By marking a function as `async`, we can then use the `await` keyword within it to pause execution until a Promise (like an API response) is resolved.

This means the app waits for the necessary data before proceeding, ensuring that any subsequent code that depends on this data only runs after the data has been successfully retrieved.

Preventing Issues with Asynchronous Code:

Without proper handling, asynchronous operations can lead to issues such as trying to render data that hasn't arrived yet, resulting in errors or incomplete information being displayed.

By using Promises with `async/await`, we avoid such pitfalls. It allows our app to handle tasks like updating the UI with new weather data or suggesting outfits only after the relevant data is available (13).

Application in Weather Wear:

In Weather Wear, when a user opens the app, it needs to fetch the current weather conditions based on their location and then suggest appropriate outfits from their digital wardrobe.

We implement an asynchronous function to first fetch the weather data. Once we have this data, we proceed to filter the wardrobe based on the weather conditions. This sequential process is crucial for providing accurate and useful outfit suggestions.

Additionally, during these operations, we might show a loading indicator (like a spinner), enhancing the user experience by informing them that data is being fetched. Once the data is loaded, the app updates the UI with the relevant information or suggestions.

To summarise, the usage of Promises, specifically `async/await`, is a key component of our Weather Wear app's functionality, allowing for fast and error-free asynchronous operations. This method guarantees that our users enjoy a seamless and responsive experience, with correct data shown at the appropriate times.

3.2.3 Firebase:

Google created Firebase to help developers create mobile and online applications (11). It offers developers several tools and services to help them create high-quality apps, expand their user base, and make money. Firebase provides features like analytics, databases, messaging, and crash reporting to help developers move rapidly and focus on their consumers. Firebase's Firestore database is a versatile and scalable database for mobile, web, and server development (11).

Firebase Firestore

Firestore is a NoSQL document database designed for automated scalability, excellent performance, and easy application development. While the Firestore interface has many features with traditional databases, being a NoSQL database, it varies in how it depicts connections between data items. In figure 24, shown the firestore's dashboard for weather wear application and the collection of items in the database.

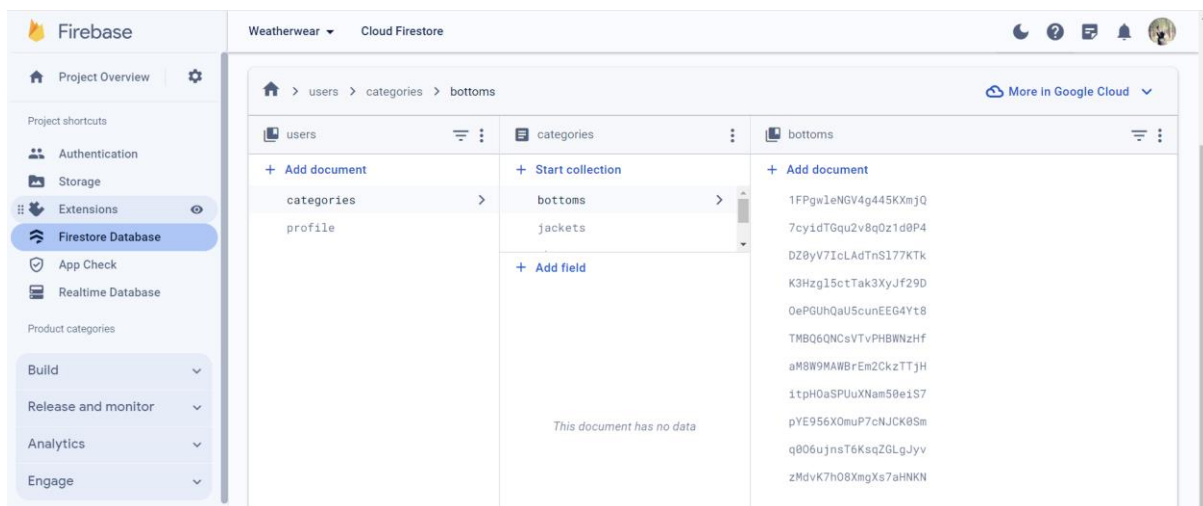


Figure 24 Cloud Firestore database collection

Use of Firestore in app:

collection, addDoc, getDocs, query, where: These functions from Firebase Firestore are used to interact with our Firestore database. They are used to define the collection reference, add new documents, retrieve documents, and set up queries that are used to fetch items specific to a user.

Why have we selected Firestore?

For our Weather Wear software, we selected Firestore, a NoSQL document database, due to its automated scalability, excellent performance, and ease of application development. Firestore's structure is ideal for our purposes, thanks to its rapid data processing and real-time update

capabilities. Unlike typical relational databases, Firestore organises data into documents and collections, which is ideal for our app's dynamic and adaptable data requirements, such as storing a variety of user wardrobe items and weather data. This NoSQL method enables us to represent data in a more intuitive and scalable manner, allowing for faster reads and writes, which are critical for presenting our consumers with real-time outfit recommendations based on the latest weather conditions. Furthermore, Firestore's smooth interaction with other Firebase services improves our app's functionality and offers a strong backend infrastructure, which is critical for ensuring a smooth and responsive user experience.

Firestore Authentication

Firestore Authentication is a robust and reliable authentication solution that makes it easy for developers to add user sign-in and sign-up to their applications (11). It offers many features:

- Email and password sign-in: Users can create an account and sign in with their email address and password.
- Social sign-in: Users can sign in with their existing accounts from Google, Facebook, Twitter, GitHub, and more.
- Real-time user data: Firestore Authentication provides real-time data about the user, such as their ID, email address, and name. This information can be used to personalise the user's experience and provide relevant content.
- Security: Firestore Authentication is built on Google's security infrastructure, which is known for its reliability and scalability.

Use of Authentication in Weather wear:

Firestore Authentication, these parameters determine how user passwords are processed before they're stored. We have used password hashing algo which is shown in next figure 25, that algorithm is provided by firestore authentication itself. This typically involves hashing the password, which is a way of encoding it so that the original password isn't stored in plain text in the database.

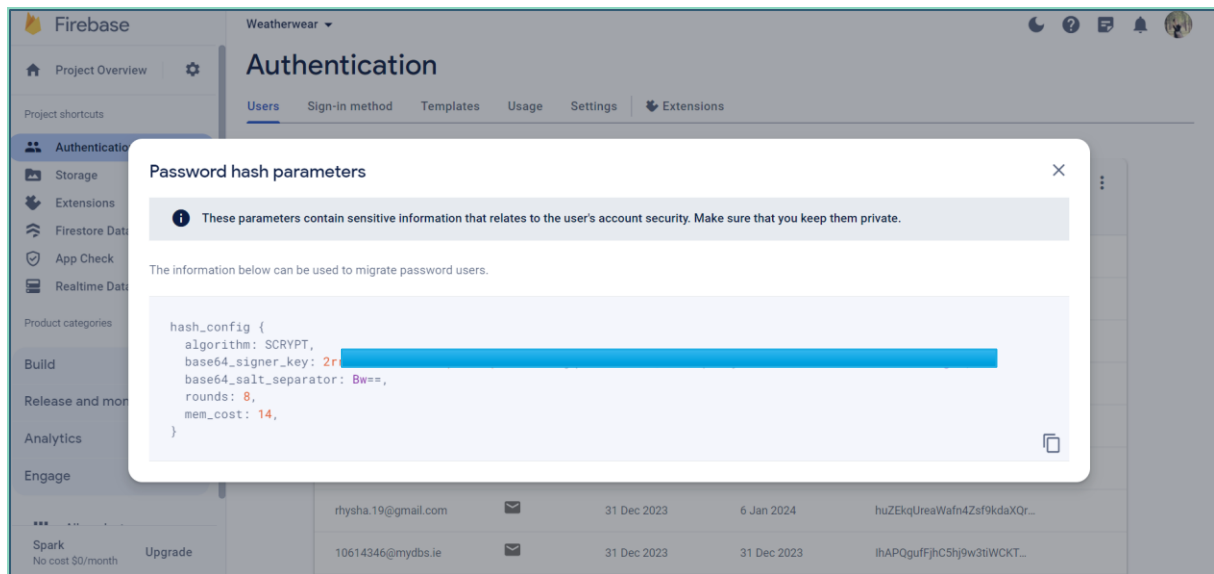


Figure 25 Password hashing

Why do we need Authentication?

As the developers of the Weather wear app, we value the security and privacy of our customers' data. By integrating Firebase Authentication, we aim to provide a seamless and secure sign-in experience for our users. This solution not only protects their personal information, but it also improves the entire user experience by allowing for customised features and settings. With Firebase Authentication, we can enable multiple sign-in methods, manage users, and maintain high security requirements without the need to construct our own authentication system. This allows us to concentrate on developing Weather Wear's core features, ensuring that our customers have consistent and secure access to the most recent weather updates and wardrobe suggestions based on their preferences and local climatic circumstances.

3.2.4 Prototype:

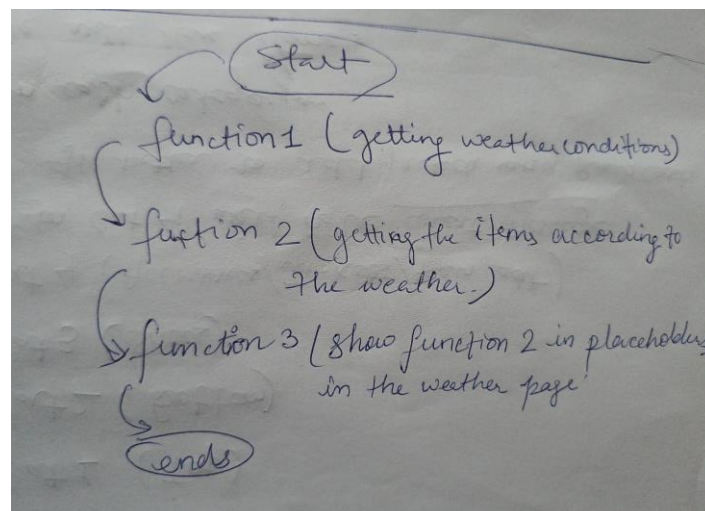


Figure 26 Weather wear on paper

In figure 26, we have noted the basic functionalities on the piece of paper. It is the journey of weather wear application from a paper to android device. In the next figure 27, we have the application storage details on the device.

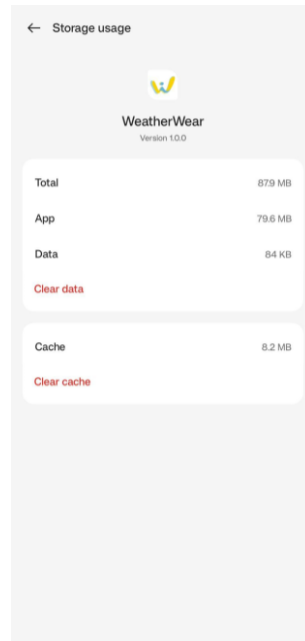


Figure 27 Storage details of the application in the android device.

We have also added the figure 28, for the current outfit suggestions from the weather wear prototype for the test user. To access the test user profile, we can login via email: test1@gmail.com and the password is Password.

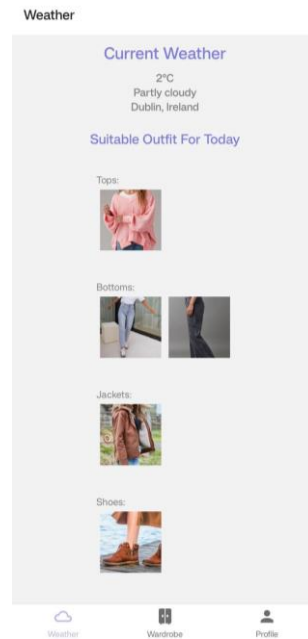


Figure 28 Outfit suggestion

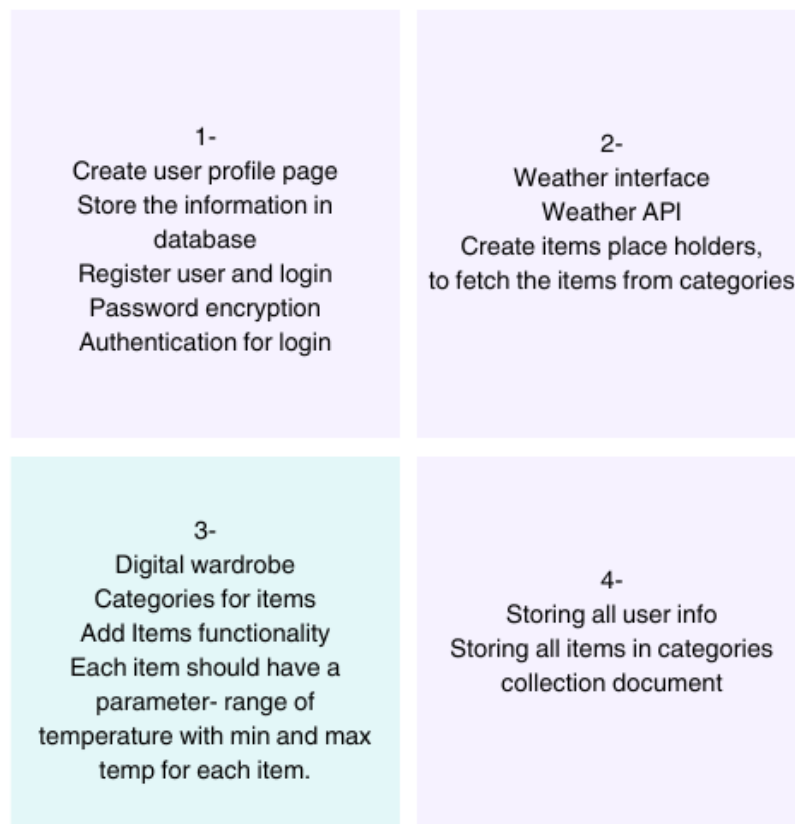


Figure 29 Prototype Design plan

In figure 29, we planned 4 phases for the prototype to initiate the app development. After getting all dependencies set for the development we worked on the first phase, where we created the profile page of this prototype with minimum functionalities. Just to get a user on

the app. Where anyone can sign up in the application and the firebase's authentication is authorising the user to enter into the application. We have developed an authentication flow with Firebase and React Native. We have created a simple login screen with two input fields for email and password, and two buttons for logging in and sign up. We have also created a profile screen that displays the username of the logged-in user.

Furthermore, we added the weather interface to integrate the weather API in the application. Where we are currently getting the weather temperatures along with the weather description. After completing the wardrobe page requirements, we created the outfit component to fetch the clothing items from the user's database to give the suggestion on outfit.

In the wardrobe interface, we divided the categories into four for this prototype. In upcoming versions, we can add more categories like accessories with a parameter of winter, summer, or rainy days. Well currently we have tops, bottoms, jackets, and shoes. For storing the items from user interface to database we have created a collection in firestore database where each category has its path to store items with given temperature values by the user and we are also using the username to save with the items. We can use the uid from the authentication but currently we are not using it. By adding the username property, we are only fetching the data if the username matches with the current username. Without this property the application was fetching all the data posted by any user and that was a security issue. While testing this functionality we came up with this difficulty. To solve this bug, we have used the "username" property in the code.

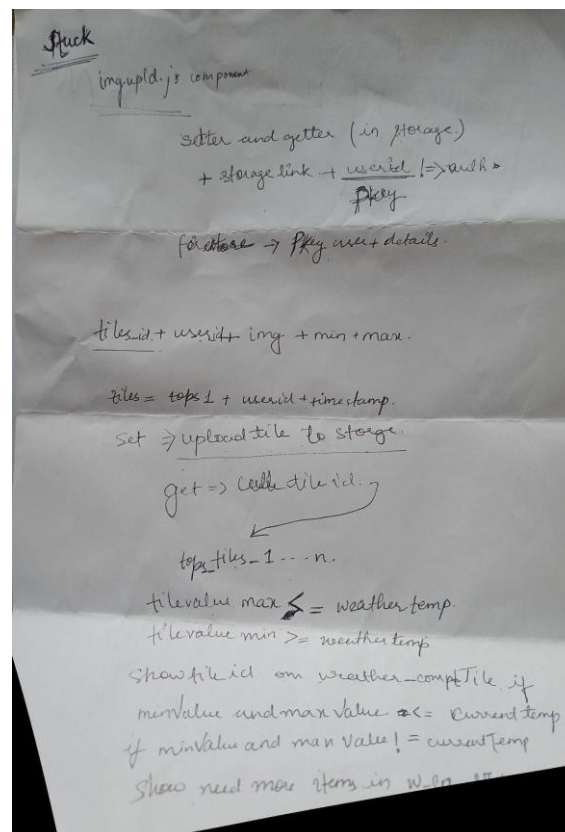


Figure 30 Decision-making Algo Planning.

In figure 30, we have a handwritten logical solution for the difficulties in wardrobe functionalities. When we got stuck in the development and looking for the solution. In unusual places we find them. We started to think of each possible way of making the wardrobe more user friendly along with the application requirements.

In figure 31, we have the wardrobe items from the ‘tops’ category in the test user profile. For testing purposes, we have added these items with a variation of temperature range.

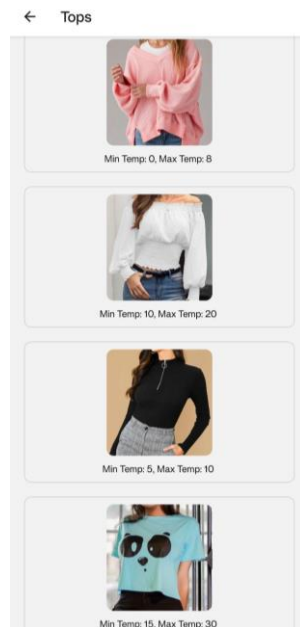


Figure 31 Items Stored in Tops

The outfit recommendation process in Weather Wear is governed by a set of decision-making rules, where user wardrobe data is meticulously compared with real-time weather API data. This comparison provides appropriate wardrobe selections based on current weather conditions and user preferences. In the future, upgrades are going to enable customers to choose between a single clothing recommendation or multiple options, providing additional freedom and personalization. Continuous testing and user input drive the app's progress, encouraging our commitment to continue refining and expanding its features, ensuring it stays a cutting-edge, user-centric solution for wardrobe management and weather adaption.

Findings & Discussions

How do our clothes affect our health?

Our analysis explores the influence of clothing on health and well-being, based on insights from the Mirage News article "Wardrobe Wellness: How Your Clothes Affect Your Health" (1). Recognizing clothing as a vital part of daily life, we examine its role beyond just aesthetics, focusing on its substantial effects on health, comfort, and general well-being (1).

1.1 Material Choices and Health Impacts:

The article (1) highlights the importance of selecting the right materials for healthy clothing. Natural materials like cotton, linen, wool, and silk are healthier alternatives to synthetic materials. Cotton is praised for its breathability, aiding air circulation and reducing moisture buildup. Linen, made from the flax plant, has natural antibacterial properties and temperature regulation. Wool is recognized for its thermoregulating properties, making it suitable for both warm and freezing weather, while silk, known for its luxury, is hypoallergenic and gentle on the skin. In contrast, synthetic materials, though durable, often trap heat and moisture and may have harmful chemicals, leading to discomfort and skin issues.

1.2 The Role of Fit and Comfort:

The fit and comfort of clothing are crucial for health. Ill-fitting clothes can lead to discomfort, restricted movement, and health issues like circulation problems, skin chafing, and respiratory difficulties. Clothes that are too loose pose risks of physical accidents. The article emphasises that comfort should not be sacrificed for style, as clothing that is both stylish and comfortable can enhance self-confidence and contribute to mental well-being.

1.3 Indirect Health Effects and Protection:

Clothing choices also indirectly affect health. For instance, dark-coloured clothes absorb more heat, making them less suitable for hot weather, while light-coloured clothes reflect heat. Additionally, clothing can offer protection against environmental factors like sunburn, insect bites, and harmful UV rays. The importance of clothing in temperature regulation, preventing skin conditions, influencing mental well-being, exercise performance, and contributing to musculoskeletal health is also highlighted (1).

1.4 Impact on Workplace Safety and Sleep Quality:

In certain professions, clothing is vital for safety, with fire-resistant materials and high-visibility clothing preventing injuries (1). Furthermore, what one wears to bed can affect sleep quality, where loose, breathable nightwear promotes better sleep. These aspects show the multifaceted role of clothing in daily life and health.

Weather forecasts play an important role.

The findings from the National Center for Atmospheric Research (NCAR) underscore the significant role that weather forecasts play in the daily lives of Americans. With over 90% of adult Americans checking the weather more than three times daily, it's clear that weather information is not just a casual interest but a crucial part of daily decision-making (National Center for Atmospheric Research, 2009) (9). This frequent engagement with weather forecasts, particularly for guiding clothing choices, highlights a widespread need for exact and accessible weather information. It suggests that people are actively seeking insights into weather patterns not just for curiosity but for practical reasons, like planning their attire and activities for the day. This behaviour reflects an awareness of the impact of weather on comfort, safety, and overall well-being, and underscores the value of applications like the Weather Wear app, which can provide targeted, useful information in this regard.

What to wear?

In this article (8) we have read about a virtual wardrobe, where a designer has multiple layout ideas for her pocket closet. In "A Virtual Wardrobe — My First UX/UI Project," Sara Binsaeed details her journey as a UXDI student creating a mobile app for fashion and wardrobe management. Tasked with addressing user needs in fashion, Sara's research identified key challenges faced by users like Ian, who needed an efficient way to manage and recall his outfit combinations. The app solution, developed through user-centric design processes including experience mapping and iterative testing, offers closet organisation, outfit creation, and saving capabilities directly from a mobile device. Its final design boasts an elegant interface with thoughtful features like integrated navigation and quick-add options, highlighting the significance of user-focused design in fashion tech. Future enhancements include personalised categorization, calendar integration, and outfit sharing, underscoring the app's potential in simplifying wardrobe management (8).

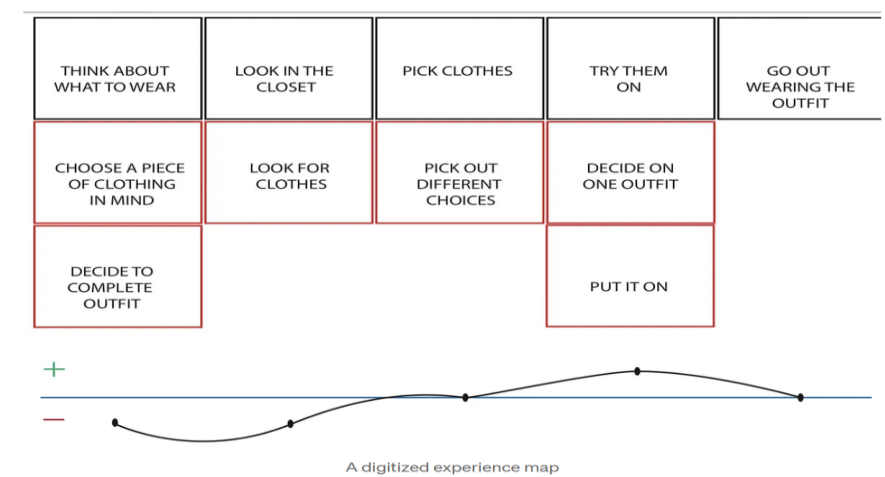


Figure 32 Experience map of the user

Source- Binsaeed, S. (8)(2019) Image in 'A Virtual Wardrobe — My First UX/UI Project'. Medium.

It is a visual representation of the end-to-end experience that a user goes through in order to decide what to wear.

Finally, our findings demonstrate a multidimensional link between clothes, health, and technology. The Mirage News article "Wardrobe Wellness: How Your Clothes Affect Your Health" (1) emphasises the importance of clothing for both physical and mental health. It emphasises the benefits of natural materials such as cotton, linen, wool, and silk over synthetic equivalents in terms of breathability, temperature regulation, and skin friendliness. The necessity of finding the correct fit is also emphasised, since it not only reduces health problems but also improves mental well-being through self-esteem. Clothing also acts as a protective barrier against environmental elements, which is critical in maintaining safety at work and good sleep.

Simultaneously, the National Center for Atmospheric Research puts light on the critical significance of weather forecasts in daily American life (9), with most adults relying on weather updates for practical decisions such as wardrobe selection. This conduct emphasises the need for accurate and easily accessible weather information. Meanwhile, Sara Binsaeed's work on a virtual wardrobe smartphone app represents a significant advancement in fashion technology. This program, aimed at easing wardrobe management, integrates components such as closet organisation, outfit design, and schedule integration to meet the user's fashion needs. It exemplifies the ever-changing convergence of technology and fashion, which works for user-centric design and the desire of individualised fashion solutions.

Key Features

Platform Independent application-

As it can run on android, iOS and web. As we have used Expo to develop this application. While the application prototype was in development phase, we have tested the application on web via local host, also on the android phone and iOS iPad by using expo go application. It lets you see the output of the code, when we run the server, it generates a QR code which we can scan in the Expo Go app (6).

Cross-Platform Compatibility: Our application is designed to run on Android, iOS, and web platforms. This is a significant advantage as it broadens our potential user base. By developing with Expo, we can write a single codebase that works across these different platforms, which is more efficient than writing separate code for each platform.

Use of Expo: Expo is an open-source platform for making universal native apps for Android, iOS, and the web with JavaScript and React. It simplifies the development process by providing a set of tools and services which are beneficial for building, deploying, and quickly iterating on iOS, Android, and web apps from the same codebase (6).

Development and Testing Process: Testing on Web via Localhost: While our application was in the prototype phase, we tested it on the web. This likely involved running the application on a local development server, viewable in a web browser. This step is crucial for first debugging and fast development cycles (12).

Testing on Android and iOS Devices: We also tested the app on actual devices - an Android phone and an iOS iPad. Testing on real devices is essential for understanding the user experience and catching issues that may not appear in web-based testing.

Using Expo Go for Testing: Expo Go is an app available on Android and iOS that facilitates the testing of our application. When we run our server, it generates a QR code. Scanning this QR code with the Expo Go app loads our application directly onto the device (6). This approach allows for immediate testing on mobile platforms without the need for a full deployment.

Real-Time Updates: One of the significant advantages of using Expo and Expo Go is the ability to see the output of our code in real time. When we change a line of code in our development environment, these changes can be immediately reflected in the app running on our web browser or on our mobile device through Expo Go (6). This feature is known as "hot reloading" or "live reloading," and it dramatically speeds up the development process by allowing immediate feedback and iteration.

Advantages of This Approach:

Efficiency in Development: By using a single codebase for all platforms and leveraging real-time updates, we can develop and iterate our app more quickly.

Ease of Testing: The ability to test on different platforms without needing to deploy the app each time saves time and resources.

Immediate Feedback: Real-time reflection of code changes helps in quickly identifying and fixing issues or experimenting with new features.

Using Weather API-

Weather API ensures that the application gets real-time or near real-time weather data. This is crucial for providing correct clothing recommendations. For instance, if there's a sudden change in weather – like unexpected rain or a heatwave – the app can immediately update its suggestions. Using a free API can be cost-effective, especially for a startup or a new app. It reduces the overhead costs while still providing essential functionality.

Wardrobe Management-

The Wardrobe Management feature in our Weather Wear application exemplifies a modern, user-centric approach to fashion and convenience. By enabling users to digitally add and organise their clothing items within the app, coupled with each item's minimum and maximum temperature data stored in the Firestore database, we provide tailored outfit suggestions based on real-time weather conditions. The addition of a waterproof checkbox for jackets and shoes enhances functionality for rainy days. This level of customization, particularly in a prototype phase without restrictions on the number of items, underscores our commitment to user comfort and the adaptability of the app. As users input their preferences and requirements once, they enjoy a seamless, personalised experience in outfit selection in the available options, effortlessly matching their wardrobe to the day's weather. This innovative approach not only simplifies daily routines but also showcases the potential of integrating digital tools with personal lifestyle needs.

Decision Making Algorithm-

Our application employs a sophisticated algorithmic system that seamlessly integrates real-time weather information with individual user preferences, resulting in tailored outfit recommendations. This advanced algorithm considers several crucial factors, including temperature, precipitation, and the user's personal wardrobe choices. By doing so, it guarantees that each outfit suggestion is not only suitable for the prevailing weather conditions but also aligns perfectly with the user's comfort preferences. This personalised approach ensures that our users are always well-prepared and fashionably attired, regardless of the weather's unpredictability.

The core functionality of the outfit.js component centres around the generation of outfit recommendations. It relies on a carefully crafted algorithm that analyses the current weather conditions and the contents of the user's wardrobe. This algorithm follows a systematic decision-making process to curate outfits that are both weather-appropriate and in harmony with the user's fashion preferences.

The decision-making process begins with fetching the user's wardrobe data from our database. This data encompasses a variety of clothing items, categorised into tops, bottoms, jackets, and shoes. It forms the foundation for outfit selection. Next, the algorithm takes the current temperature into consideration. It filters through the user's wardrobe to identify clothing items that are suitable for the prevailing temperature range. This ensures that the outfits suggested are comfortable and appropriate for the weather conditions at hand. Additionally, the algorithm considers the weather description, such as "Rain" or "Snow." Based on this information, it determines whether to include waterproof jackets and shoes in the outfit recommendations. This thoughtful consideration ensures that users are equipped with the right attire to tackle adverse weather conditions.

Once the outfit suggestions have been meticulously generated, they are presented to the user within the app's interface. Users can explore these suggestions, conveniently view images of the recommended clothing items, and make informed decisions about their attire for the day. Our application's innovative algorithm combines weather data and user preferences to offer personalised outfit suggestions. This algorithmic approach, encapsulated within the outfit.js component, ensures that users receive wardrobe recommendations that are not only weather-conscious but also reflect their individual style for all weather conditions.

To summarise the key features of our application, developed using Expo, stands out for its platform independence and cross-platform compatibility. This enables it to function flawlessly across Android, iOS, and web platforms. Using Expo's development environment, we swiftly tested our app across a variety of devices and web browsers, assuring a stable and usable experience. One of the most notable aspects is the integration of a real-time Weather API, which provides precise weather data required for our core operation - customised outfit suggestions. Our app's Wardrobe Management function enables users to save images of their clothes, including characteristics such as temperature compatibility, which improves their variety of outfit choices. Furthermore, our app's decision-making system consistently blends real-time weather conditions with the user's wardrobe choices, ensuring that each recommendation is weather suitable. This entire approach, from development to functionality, demonstrates our dedication to providing a dynamic and user-centric solution.

Future Work

Editable Wardrobe Items

In further attempts to develop the weather wear app, we are exploring the addition of a "Edit Wardrobe Items" function to our future work plan. This feature would provide customers more control over their wardrobe within the app, allowing them to adjust the temperature suitability settings for clothing items and delete items as needed. Users will benefit from a more personalised and adaptive wardrobe management experience, ensuring that outfit suggestions keep up with their changing preferences and weather circumstances. This tool, which includes a user-friendly interface, visual organisation aids, and backup methods for added peace of mind, seeks to expedite wardrobe management, and improve the overall user experience.

Users will be able to simply review their wardrobe items, alter temperature suitability settings, and remove items they no longer need. This versatility allows customers to keep a wardrobe that is not only weather-appropriate, but also reflects their current style and lifestyle preferences. The "Edit Wardrobe Items" function is consistent with our dedication to user-centric design, providing a streamlined and efficient wardrobe management experience while ensuring that users' outfit suggestions always reflect their specific needs and preferences.

Geo Location

With Geolocation, our app can accurately determine the user's current location, allowing us to provide hyper-localised weather forecasts. This means users will receive precise weather information tailored to their exact whereabouts, including temperature, precipitation, wind conditions, and more. Additionally, we can offer location-specific clothing recommendations, considering not only the weather but also local climate variations. Geolocation also enables features like automatic weather updates as users move from one place to another, ensuring they stay prepared for changing weather conditions wherever they go. Overall, Geolocation integration empowers our weather wear app to deliver real-time, location-specific weather data and outfit recommendations, making it an indispensable tool for staying comfortable and stylish in any location and weather scenario.

User Profile

The enhancement of the user profile experience in our weather wear app focuses on providing users with a broad range of customizable options and features. This includes basic account settings adjustments like changing the display name or password, alongside more personalised options such as adding biographical details like location and a brief bio. Privacy settings empower users to manage their data visibility and security, while notification preferences allow them to tailor how they receive updates from the app.

Social media integration improves the user experience by allowing for quick sharing and connection, while a dark mode toggle accommodates diverse visual preferences and can minimise eye strain in low-light circumstances. A dedicated support and help centre are essential for giving users help and information, resulting in a seamless and delightful app experience.

These proposed improvements aim to supply a complete and interesting user profile experience while also improving user control and interaction within the weather wear app. By paying close attention to user needs and preferences, the app transforms from a wardrobe management tool to a platform for personal expression and community participation.

Laundry

This feature will allow users to manage their wardrobe more effectively by marking clothes that are currently in the laundry. Integrated within the item cards, users can simply tap a laundry icon to temporarily disable an item from appearing in outfit suggestions. This function is particularly useful for items that are dirty or unavailable, providing a convenient alternative to deleting them from the digital wardrobe. Users can easily reactivate these items by clicking on the laundry icon again, restoring them to active status. This addition aims to make wardrobe management within the app more dynamic and reflective of the user's real-life clothing availability.

Clothes with RFID or QR Code to scan the suitable temperature.

Adding RFID or QR codes to clothes can make weather apps even better. This means we'd put these special codes on clothes when we make them. These codes hold useful info, like what temperatures it's good for (both hot and cold), and even things like what it's made of or how to take care of it. This cool idea would make it easier for people to get personalised outfit suggestions based on the weather. It'd also help them keep track of their clothes more easily.

Social element

We intend to include a social component in the Weather Wear app, allowing users to add friends or family using their identities. This feature will allow users to share their wardrobe selections with their social circle, which is especially handy for coordinating attire for events that need colour-coordinated or theme-based dressing. Users can present their available wardrobe alternatives for specific days, allowing for collective decision-making. The social element not only improves the app's functionality, but it also taps into our core drive for social connection, which could increase the app's popularity and user engagement, like other successful social applications.

Essential clothing items required.

The upcoming travel functionality for our weather clothing app is likely to transform the way customers prepare for their journeys. It all starts with consumers entering crucial information, such as their preferred vacation location and the dates of their trip's start and end. This valuable information serves as the foundation for a variety of user-centric functionalities. The app then takes centre stage, generating exact and location-specific weather forecasts for the duration of the user's travel. With this real-time meteorological data, the app provides individualised packing recommendations, advising users on the types of clothing and accessories they should bring to match the expected weather conditions at their location.

The application moves on to do an in-depth evaluation of the user's current wardrobe. It carefully assesses if the user's current wardrobe assortment is appropriate for the impending

excursion and detects any gaps. If the user's wardrobe lacks clothing appropriate for the weather conditions at the journey destination, the app steps in to provide realistic options. It ensures that they are ready to face Mother Nature's unpredictability with confidence and style. This feature makes trip planning easier, improves travel preparedness, and elevates the whole travel experience by reducing the headaches of last-minute packing and buying. This feature allows users to begin on their trips completely equipped and attractively dressed, regardless of the weather conditions.

Subscription plans for more items space

In the future development roadmap for the weather wear app, one intriguing path to pursue is the introduction of subscription plans that provide customers with additional item storage capacity. This feature would allow us to give customers more flexibility and ease when managing their wardrobe, especially as they increase more clothing items over time. Premium plans provide more storage space, allowing users to catalogue a wider range of clothing for different weather conditions and occasions. This subscription model not only ensures our app's long-term survival, but also enables users to simply curate their wardrobe, resulting in a more rewarding and comprehensive weather wear experience.

More categories

In future updates, we plan to enhance the wardrobe section of our app by introducing additional categories. This expansion will include specific sections for accessories, tailored for different seasons such as winter and summer. This update aims to provide a more comprehensive and organised wardrobe management experience, allowing users to categorise and access their accessories more effectively based on the season. Like in the figure 33, we have a visual representation of the accessories category.

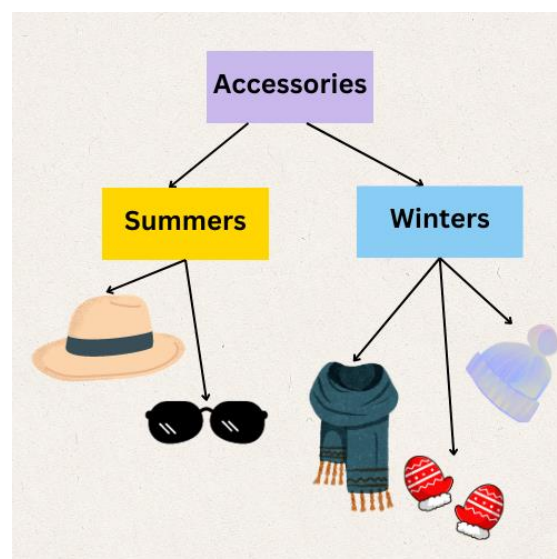


Figure 33 Future categories

Conclusion

The Weather Wear app marks a significant breakthrough in the digital wardrobe management sector, particularly in the realm of weather-based clothing recommendations. Its unique proposition lies in its ability to fuse real-time meteorological data with a personalised wardrobe management system. This synergy provides users with outfit suggestions that are not just fashionable but also align seamlessly with the prevailing weather conditions. In a world where convenience and bespoke solutions are increasingly sought after, this app offers a blend of both, addressing a real-world problem with a smart, tech-driven solution.

Throughout the development of this project, several challenges were encountered and tackled. One of the primary challenges was the technological integration necessary for the app's functionality. Utilising cutting-edge technologies such as Expo, React JS, Firebase, and Weather API, the app was crafted to be not only highly functional but also user-friendly, catering to a diverse audience irrespective of their tech-savviness. These technologies were meticulously chosen and integrated to ensure that the app runs smoothly across different platforms, thereby enhancing accessibility and user engagement.

However, in the journey of technological innovation, certain limitations and avenues for future enhancement have been identified. The dependency on internet connectivity is one such constraint, potentially limiting the app's use in areas with poor network coverage. Additionally, ensuring hardware compatibility across a wide range of devices poses a continuous challenge, necessitating regular updates and optimizations. Catering to the vast diversity in fashion preferences and cultural clothing styles is another area that calls for ongoing refinement to make the app universally applicable.

Looking ahead the app has several chances for growth and enhancement. The addition of geolocation services might increase the app's usability by offering more specific weather data and outfit recommendations. Another possible feature that might make the app more involved is social sharing, which allows users to discuss their style preferences or coordinate outfits with friends and family. Enhancing user profile customisation will provide a more personalised experience by adapting the app's features to individual tastes and demands.

The Weather Wear app exemplifies how technology can significantly enrich daily life. By streamlining the decision-making process for daily attire and adeptly adapting to the ever-changing dynamics of weather and personal style preferences, the app not only conserves time but also elevates the user experience. Its ability to offer practical, weather-appropriate, and stylish clothing suggestions caters to the modern individual's desire for efficiency and personalization in daily routines.

Moreover, the app's design and functionality reflect a deep understanding of its users' lifestyles. It acknowledges the busy schedules of contemporary life, where every minute saved is valuable, and the increasing desire for personalised services that cater to individual tastes and needs. By addressing these aspects, the Weather Wear app not only serves as a utility tool but also as a fashion advisor, merging the realms of technology, lifestyle, and fashion in a seamless and intuitive manner.

The app's potential to evolve into a more comprehensive lifestyle assistant is vast. Future iterations could include features like smart wardrobe management, where the app suggests decluttering unused items or recommends purchases based on missing essentials in the user's wardrobe. Integration with online shopping platforms to suggest purchases directly from the app could also enhance user convenience. Moreover, leveraging machine learning algorithms could enable the app to learn from user preferences over time, offering increasingly tailored suggestions.

To summarise, the Weather Wear app is an example of innovation at the crossroads of technology and fashion. It's a product that goes beyond usefulness, offering customers a seamless combination of functionality and design. As technology evolves and becomes a vital part of everyday life, the Weather Wear app serves as a potential blueprint for future advancements, set to revolutionise how we engage with technology in our daily routines. Its continual progress and possible expansions set the way for a future in which technology not only simplifies work, but also enhances our lifestyle choices and personal expressions.

References

- (1) Mirage News. (2023) 'Wardrobe Wellness: How Your Clothes Affect Your Health'. [online] Available at: <https://www.miragenews.com/wardrobe-wellness-how-your-clothes-affect-your-1009124/> [Accessed 14 Dec. 2023].
- (2) Liu, Y., Gao, Y., Feng, S. and Li, Z. (2017) 'Weather-to-Garment: Weather-oriented Clothing Recommendation', *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*. Available at: <https://ieeexplore.ieee.org/document/8019476> (Accessed: 20 December 2023).
- (3) Y. Nakano, D. Nishioka, Y. Saito and Y. Murayama, "A Study on the Clothes Recommendation for a Cold Region of Japan," 2016 49th Hawaii International Conference on System Sciences (HICSS), Koloa, HI, USA, 2016, pp. 436-443, doi: 10.1109/HICSS.2016.61.
- (4) Whering.co.uk. (2023) How to use Whering. [online] Available at: <https://whering.co.uk/how-to> [Accessed 26 Dec. 2023].
- (5) Panaprium. (2024). 12 Best FREE Capsule Wardrobe Apps In 2024. [online] Available at: <https://www.panaprium.com/blogs/i/capsule-wardrobe-apps> [Accessed 1 Jan. 2024].
- (6) Expo, n.d. Create Your First App. Available at: <https://docs.expo.dev/tutorial/create-your-first-app/> [Accessed 11 Oct. 2023].
- (7) Expo (2023). *Build APKs for Android Emulators and devices*. [online] Expo Documentation. Available at: <https://docs.expo.dev/build-reference/apk/> [Accessed 2 Jan. 2024].
- (8) Binsaeed, S. (2019). *A virtual closet — UXDI project 1*. [online] Medium. Available at: <https://medium.com/@sarabinsaeed/a-virtual-closet-uxdi-project-1-c3dfeafe70c> [Accessed 1 Jan. 2024].
- (9) NCAR (2009). *300 BILLION WEATHER FORECASTS USED BY AMERICANS ANNUALLY, SURVEY FINDS*. [online] Available at: <https://opensky.ucar.edu/islandora/object/archives:2377> [Accessed 1 Jan. 2024].
- (10) K. -H. Liu, H. -L. Chuang and T. -J. Liu, "Clothing Recommendation Based on Deep Learning," 2022 IEEE International Conference on Consumer Electronics - Taiwan, Taipei, Taiwan, 2022, pp. 281-282, doi: 10.1109/ICCE-Taiwan55306.2022.9869006.
- (11) Google, F. (2024). *Get data with Cloud Firestore*. [online] Firebase. Available at: <https://firebase.google.com/docs/firestore/query-data/get-data> [Accessed 5 Jan. 2024].
- (12) Matt, M. with (2021). *React Native Authentication with Firebase and Expo in 27 minutes*. [online] www.youtube.com. Available at: <https://www.youtube.com/watch?v=ql4J6SpLXZA> [Accessed 17 Dec. 2023].

(13)Mittal, P. (2022). *What are Promises in React.js?* [online] www.tutorialspoint.com. Available at: <https://www.tutorialspoint.com/what-are-promises-in-react-js> [Accessed 19 Nov. 2023].

(14)RyanTurner, R. (2019). *Introducing Hooks*. [online] react.dev. Available at: <https://reactnative.dev/blog/2019/03/12/releasing-react-native-059> [Accessed 20 Nov. 2023].

(15)Sharma, N. (2023). *Top React Native UI Component libraries trending in 2023*. [online] www.kellton.com. Available at: <https://www.kellton.com/kellton-tech-blog/10-best-react-native-ui-libraries-dominating-mobile-app-development> [Accessed 7 Nov. 2023].

Additional Links

Git hub link- <https://github.com/RhythmSharma10614346/wwearapp>

Weather API- <https://www.weatherapi.com/>

APK- <https://expo.dev/accounts/rhysha.19/projects/weatherwear/builds/119a15f4-5aad-4886-9c73-6a5ee9454e80>

Designed Images- <https://www.canva.com/>

Logo Designed (figma)- [https://www.figma.com/file/OWz2DhCDLCKUXIVzkOcei9/Expo-App-Icon-%26-Splash-\(Community\)?type=design&node-id=1-1357&mode=design](https://www.figma.com/file/OWz2DhCDLCKUXIVzkOcei9/Expo-App-Icon-%26-Splash-(Community)?type=design&node-id=1-1357&mode=design)

Sequence diagram (draw.io)- <https://app.diagrams.net/>