

1. INTRODUCTION

1.1 Overview

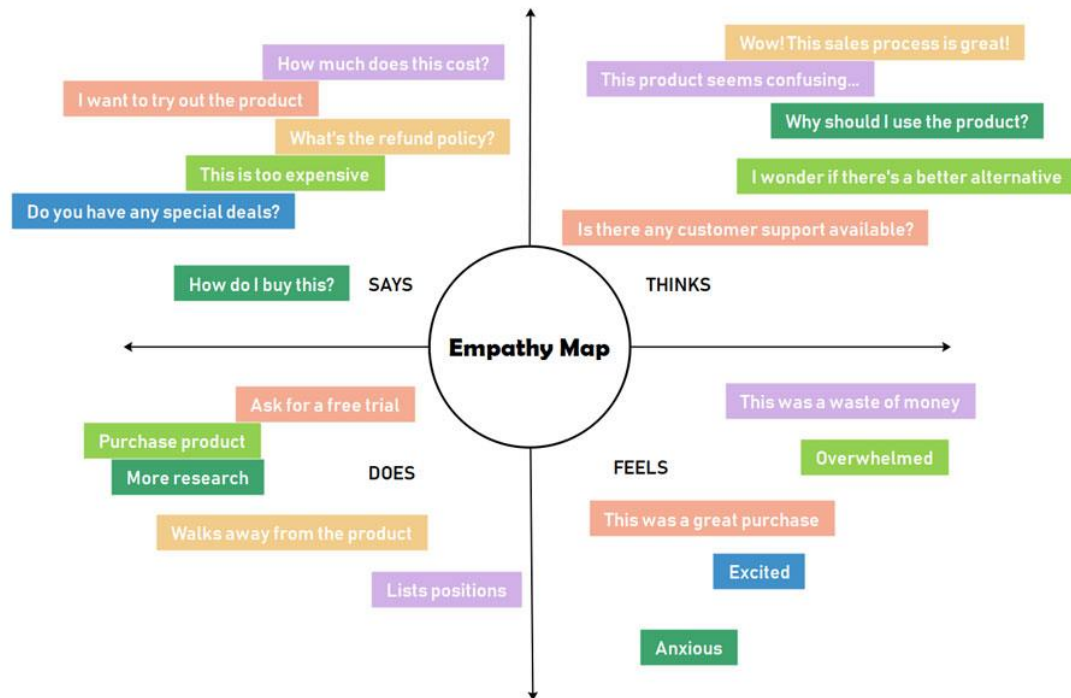
The Adaptive Mail app is a sample project that illustrates how to use the Android Compose UI toolkit to create a conversational user interface. It's designed to look and feel like a messaging app, enabling users to send and receive messages, as well as view a history of previous messages. The app demonstrates how to use Compose UI toolkit's essential features, such as data management and user interaction, and highlights some of the key features of Compose UI toolkit. The app provides an example of how to build a messaging app with a modern, dynamic UI using Compose UI toolkit.

The Adaptive Mail app is a sample project that showcases how to use the Compose UI toolkit to build a messaging app with a conversational UI. The app uses various Compose UI widgets such as Text, Row, Column, and Box to create a messaging interface. It also demonstrates how to use state management in Compose to keep track of the user's messages and the conversation history.

1.2 Purpose

The purpose of the Adaptive Mail app is to demonstrate how to use the Android Compose UI toolkit to create a conversational UI. It serves as a sample project to help developers learn how to use Compose UI toolkit to build modern and dynamic user interfaces for Android apps. The app provides an example of how to use various Compose UI widgets, state management, animations, and transitions to create a messaging app with a conversational UI. By studying the code of the Adaptive Mail app, developers can learn how to apply these techniques to their own apps and create more engaging and user-friendly interfaces.

2. Problem Definition & Design Thinking



3.RESULT

The output of the Adaptive Mail app is a sample messaging app with a conversational UI built using the Android Compose UI toolkit. When the app is launched, users are presented with a messaging interface that allows them to send and receive messages. The app's UI is dynamic and responsive, with smooth animations and transitions between screens. The messaging interface includes a text input field and a send button, allowing users to compose and send messages to another user. The app also displays the conversation history, allowing users to view their previous conversations. The app showcases some of the key features of the Compose UI toolkit, including its flexible layout system, powerful set of widgets, and reactive programming model. By studying the app's code, developers can learn how to use these features to create their own messaging apps or apply them to other types of apps.

Login Page :



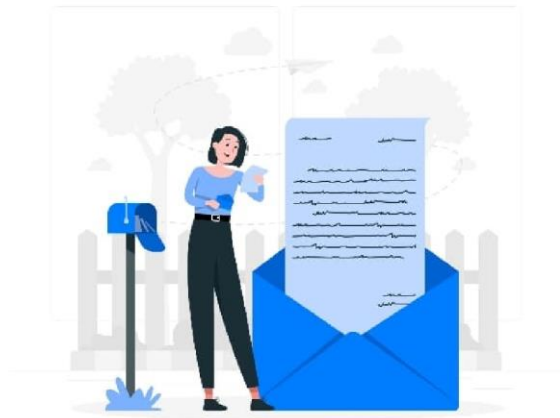
Login

Login

[Sign up](#)

[Forget password?](#)

Register Page :



Register

Register

Have an account? [Log in](#)

Main Page:

Home Screen



Send Email

View Emails

View Mail Page:

View Mails

Receiver_Mail: kavya78@gmail.com

Subject: Android

Body: This is an Adaptive Email app

Receiver_Mail: shirishbokka7@gmail.com

Subject: Order

Body: your courier has arrived

4. ADVANTAGES & DISADVANTAGES

4.1 Advantages

1. Declarative approach: Compose UI toolkit uses a declarative approach to build UI, which allows developers to write UI code in a more intuitive and efficient way. This makes it easier to create dynamic and responsive UI, as developers can focus on describing what the UI should look like rather than how it should be built.
2. Less boilerplate code: Compose UI toolkit reduces the amount of boilerplate code required to build UI, making the development process faster and more efficient. This is achieved through the use of composable functions, which can be reused throughout an app to create complex UI elements.
3. Faster development time: Compose UI toolkit allows developers to create UI faster than with traditional Android UI development frameworks. This is because Compose UI toolkit provides a more modern and efficient way of building UI, with features such as automatic view recycling, reduced layout hierarchy, and faster rendering.
4. Improved performance: Compose UI toolkit is designed to provide improved performance compared to traditional Android UI development frameworks. This is achieved through features such as optimized view recycling, efficient rendering, and reduced memory usage.

4.2 Disadvantages

1. Learning curve: Compose UI toolkit is a relatively new technology, and developers may need to invest time and effort to learn how to use it effectively.
2. Limited third-party support: Since Compose UI toolkit is a new technology, there is limited third-party support for it. This means that developers may need to build custom components or widgets to meet their specific requirements.
3. Compatibility issues: Compose UI toolkit is only compatible with Android devices running Android 5.0 (Lollipop) or later. This means that developers may need to support multiple UI frameworks for backward compatibility.

5.APPLICATION

The Adaptive Mail app built with the Android Compose UI toolkit can be used as a sample project to learn how to use Compose UI toolkit to create a messaging app with a conversational UI. Developers can use the app's code as a starting point to create their own messaging apps or to learn how to use Compose UI toolkit to build modern and dynamic UI for their Android apps.

The messaging app built with the Compose UI toolkit can be used in a variety of applications, including social media apps, customer service chatbots, and messaging platforms for businesses or organizations. The app can also be customized to meet the specific requirements of different applications, such as adding features for file sharing, video calls, or group messaging.

Overall, the Adaptive Mail app built with the Android Compose UI toolkit can be used as a reference project to learn how to build messaging apps or as a starting point to create customized messaging apps for different applications.

6.CONCLUSION

the Adaptive Mail app is a sample project that demonstrates how to use the Android Compose UI toolkit to build a messaging app with a conversational UI. The app showcases some of the key features of the Compose UI toolkit, data management, and user interactions.

The Adaptive Mail app has several advantages, including a declarative approach to UI development, less boilerplate code, and improved performance. However, there are also some potential disadvantages, such as a learning curve, limited third-party support, and compatibility issues.

Developers can use the Adaptive Mail app as a reference project to learn how to use Compose UI toolkit to build messaging apps or as a starting point to create customized messaging apps for different applications. To create an Android messaging app with the Compose UI toolkit, developers would need to follow the standard Android app development process, with a focus on using Compose UI toolkit to create the messaging UI and implement messaging functionality.

7. FUTURE SCOPE

The Android Compose UI toolkit is a relatively new technology and is still evolving. Therefore, there is a significant future scope for the Adaptive Mail app and other messaging apps built with Compose UI toolkit.

Here are some potential areas for future development and improvement:

1. Improved performance: As Compose UI toolkit is still a new technology, there is room for improvement in terms of performance. Future updates to Compose UI toolkit may address this issue and provide even better performance than what is currently available.
2. Third-party support: The Compose UI toolkit ecosystem is still developing, and there is potential for more third-party libraries and frameworks to be developed to support Compose UI toolkit. This could make it easier for developers to create more complex messaging apps with additional features and functionality.
3. Integration with other technologies: There is scope for integration of Compose UI toolkit with other technologies, such as machine learning and natural language processing, to provide more advanced and intelligent messaging capabilities.
4. Cross-platform development: As Compose UI toolkit is specific to Android, there is potential for future development to support cross-platform development across multiple platforms, such as iOS and web.

Overall, the future scope for the Adaptive Mail app and other messaging apps built with Compose UI toolkit is promising, with potential for improved performance, increased third-party support, integration with other technologies, and cross-platform development.

8. APPENDIX

