MATERIAL DESIGN

TEAM MEMBERS:

M.Aakash ***20203271401201 (Team Leader)***

A.Aathil ***20203271401202***

S.K.Abinesh ***20203271401203***

P.Akash ***20203271401204***

Under the guidance of

Mrs.P.S.Sujatha,M.Sc,MCA,M.Phil(Phy)..,M.Phil(C.S)..,

Assistant Professor

Table of content

Title

1. INTRODUCTION

Material Design is a design language created by Google in 2014 for their mobile operating system, Android. The goal of Material Design is to create a unified look and feel across all devices, platforms, and screen sizes. Material Design is based on the principles of flat design, but also incorporates elements of skeuomorphic design, which is the use of realistic textures and objects in a digital interface.Material Design is characterized by its use of bold, vibrant colors, high contrast, and the use of depth and shadow to create a sense of hierarchy and organization. It also emphasizes the use of typography, with a focus on legibility and readability.Some of the key features of Material Design include the use of material surfaces, which are responsive and can interact with user input, and the use of animation to provide feedback and enhance the user experience. Material Design also encourages the use of consistent icons, grid-based layouts, and a modular design approach to make it easy to scale and adapt to different devices and screen sizes. Overall, Material Design is a design language that prioritizes usability and accessibility while also providing a visually engaging and cohesive experience across all platforms and devices.

1.1 Overview

Material Design supports team collaboration by providing:

- Material Design is based on flat design principles with elements of skeuomorphic design.
- It emphasizes the use of bold and vibrant colors, high contrast, and typography.
- Material Design prioritizes usability, accessibility, and consistency in user interfaces.

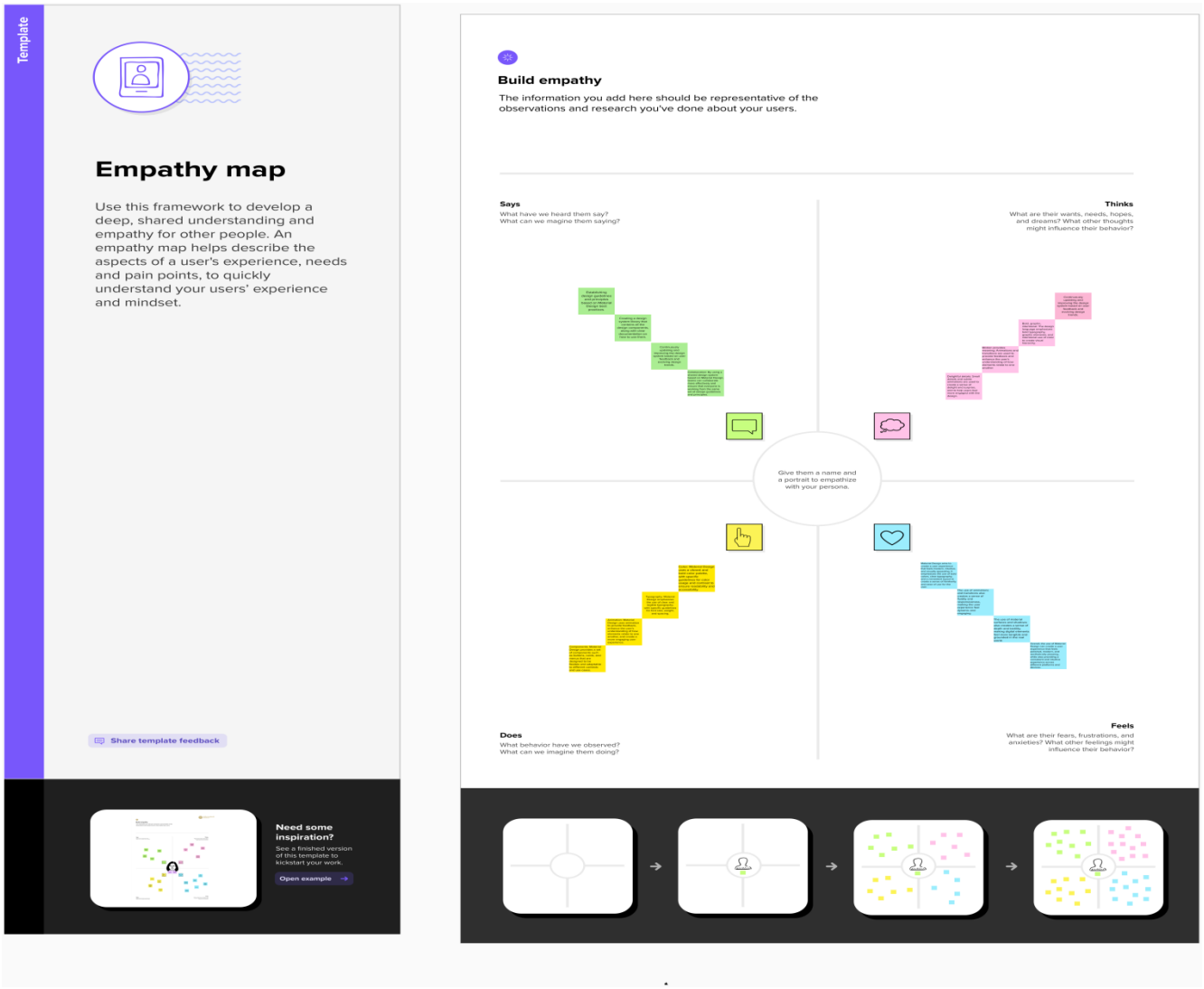
Interactions between Material Design and apps take place using *messages* in the context of particular *spaces*. For example, an app may send some simple text (a type of message) into a specific Material space (a type of Material).

1.2 Purpose

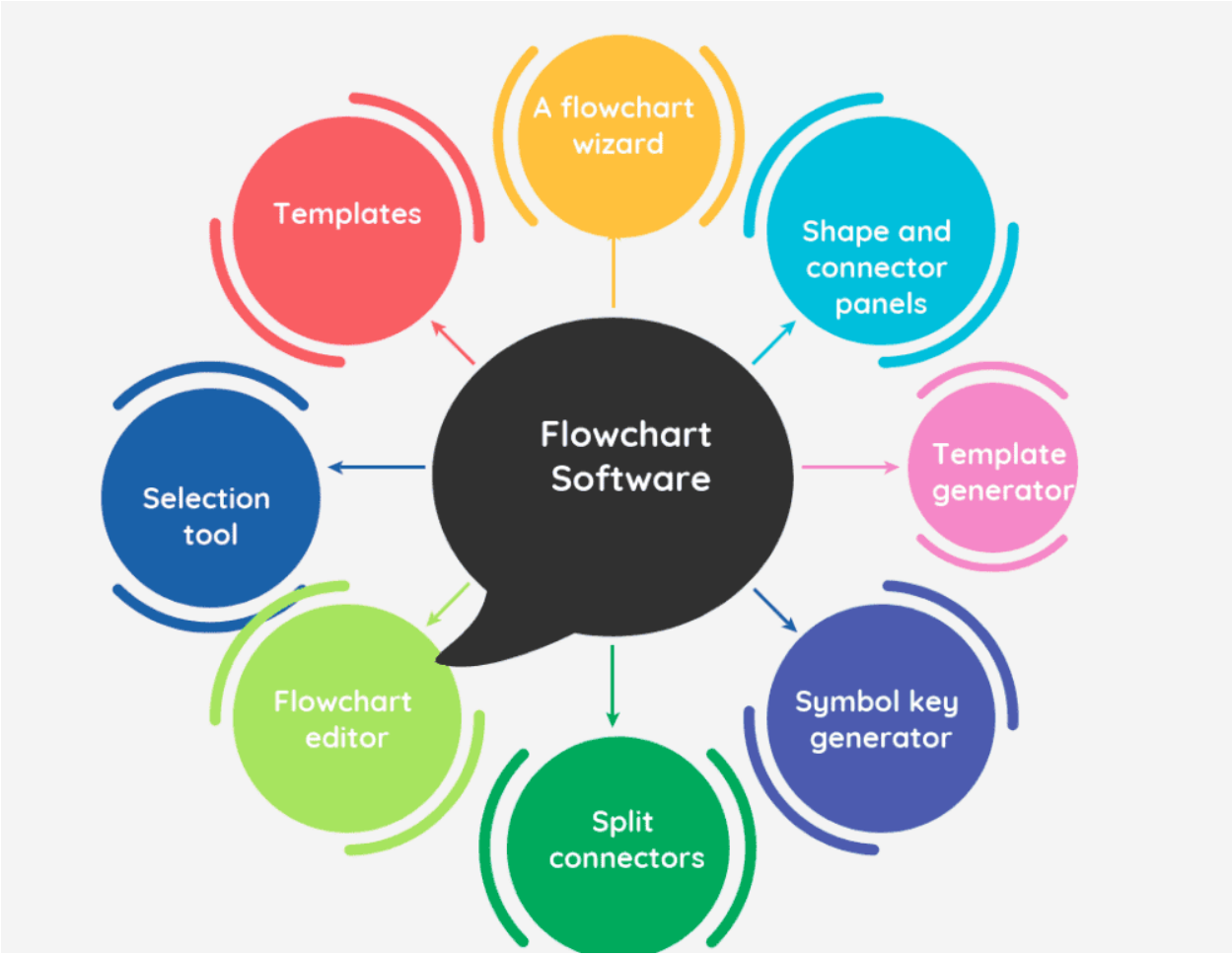
- To provide a unified and consistent look and feel across all devices and platforms.
- To prioritize usability, accessibility, and clarity in digital products and interfaces.
- To create visually engaging and cohesive experiences for users.
- To encourage adaptability to different screen sizes and devices through modular design.
- To provide guidelines and principles for designers to create intuitive and easy-to-use products.

2.PROBLEM DEFINITION & DESIGN THINKING

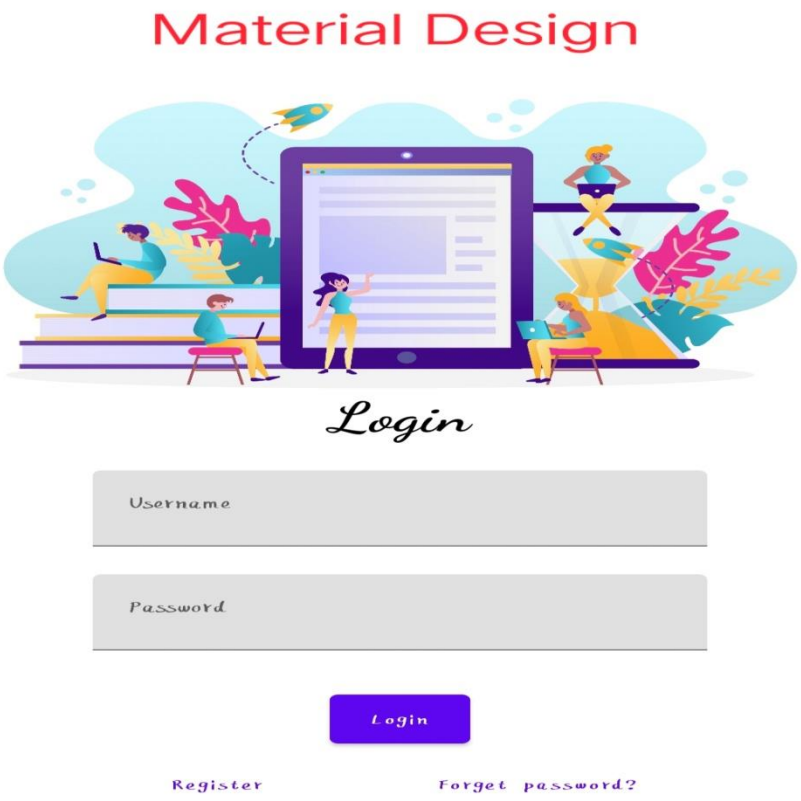
2.1 Empathy map:

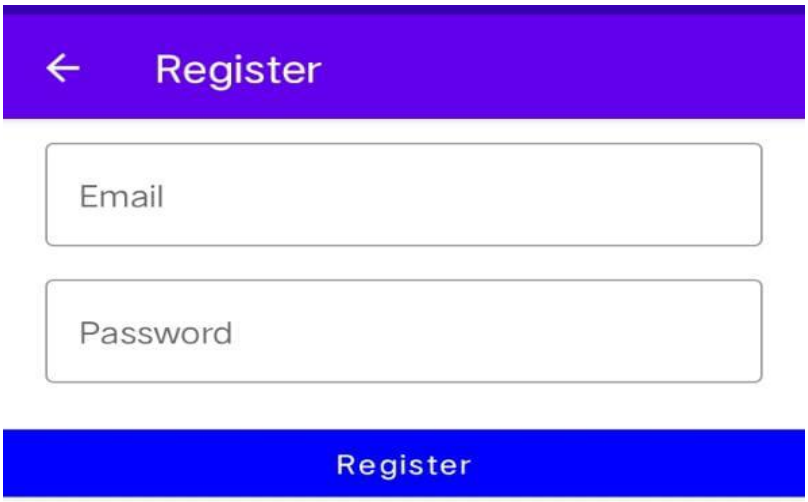


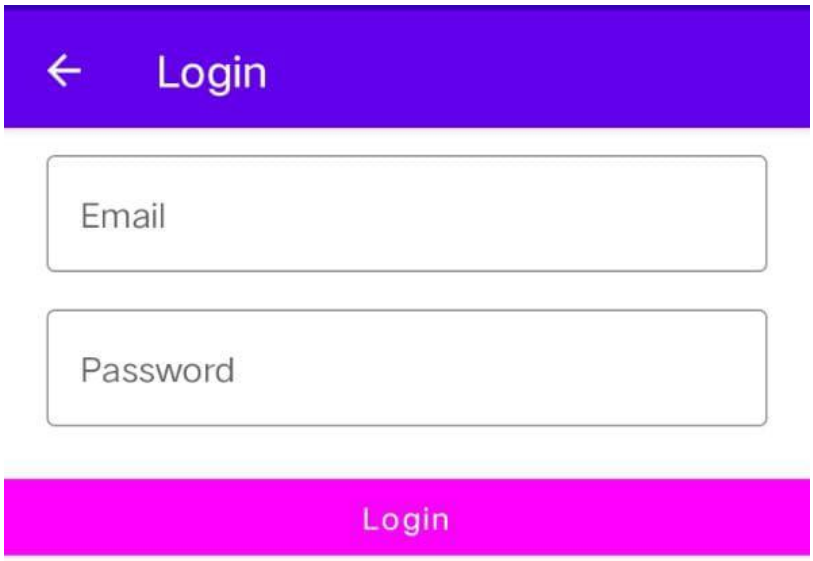
2.2Ideation & Brainstorming Map:



3. RESULT


Material Design
Login
Username
Password
Login
Register
Forget password?


Register
Email
Password
Register


Login
Email
Password
Login

4.ADVANTAGES & DISADVANTAGES

Advantages

- **Consistency**

  Material Design provides a consistent design language across different platforms, making it easier for users to navigate and use different apps and websites.

- **Flexibility**

  Material Design offers a wide range of design components and guidelines, making it easy for designers to create unique and visually appealing designs.

- **User-friendly**

  Material Design emphasizes user-centric design principles, making it easy for users to understand and interact with the design.

- **Cross-platform compatibility**

  Material Design can be implemented on multiple platforms, such as web, Android, and iOS, making it a versatile design language.

- **Accessibility**

  Material Design has built-in accessibility features, such as high contrast mode and larger text options, making it more accessible for users with disabilities.

Disadvantages

- **Limitations**

  While Material Design offers a wide range of design components and guidelines, it may limit the creativity of designers who want to create unique and unconventional designs.

- **Learning curve**

  Since Material Design has specific design guidelines and principles, it may take some time for designers to learn and implement them effectively.

- **Similarity**

  Since Material Design is a widely used design language, many apps and websites may look similar, making it hard for them to stand out.

- **Complexity**

  Material Design can be complex, especially when implementing more advanced features, which may require a higher level of technical expertise.

- **Resource-intensive**

  Implementing Material Design can be resource-intensive,especially for smaller development teams or those with limited resources.

5. APPLICATION

Material design is a design system developed by Google that is used to create cohesive and visually appealing interfaces for software applications. To create a material design application, you should follow the guidelines set out in the material design specification, which includes a set of principles, guidelines, and components that are used to create a consistent and intuitive user interface.

To create a material design application, you should start by understanding the key principles of material design, which include the use of bold colors, clear typography, and simple, intuitive interactions. You should also use the material design components, which include buttons, cards, lists, menus, and navigation bars, to create a cohesive and consistent interface.

In terms of tools and technologies, there are a variety of frameworks and libraries that you can use to create a material design application, such as Google's Material Design Components library, Bootstrap Material Design, and Angular Material. These frameworks provide pre-built components and styles that adhere to the material design guidelines, making it easier to create a visually appealing and functional interface.

Ultimately, creating a material design application involves a combination of design and development skills, as well as an understanding of the principles and guidelines set out in the material design specification. With the right tools and approach, however, you can create an application that is both visually appealing and user-friendly.

6. CONCLUSION

Material Design is a design language developed by Google in 2014 that aims to create a consistent user experience across all devices and platforms. It is based on the principles of paper and ink, and uses bold typography, vibrant colors, and subtle shadows and animations to create a modern, clean, and intuitive interface.

Over the years, Material Design has become widely adopted by designers and developers across the globe, thanks to its easy-to-use design tools, extensive documentation, and open-source nature. It has also evolved with time, adding new components, features, and best practices to stay relevant in the fast-changing world of digital design.

In conclusion, Material Design is a powerful design language that can help you create beautiful, functional, and user-friendly interfaces for your digital products. Whether you're designing a mobile app, a website, or a desktop application, Material Design can provide you with the tools and guidance you need to create a seamless and enjoyable user experience.

7. FUTURE SCOPE

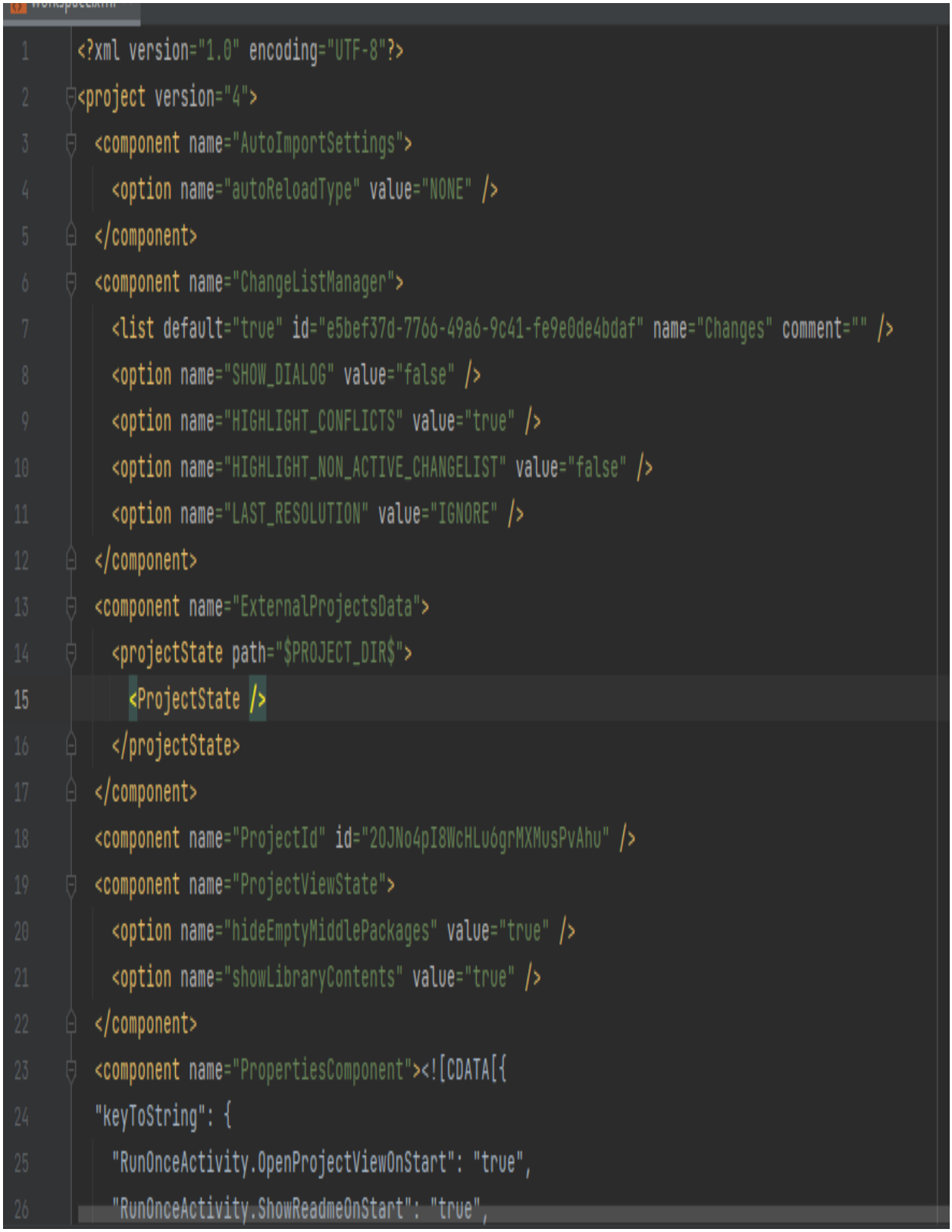
Future scopes in material design include designing for augmented reality and virtual reality, wearables, voice interfaces, accessibility, and cross-platform design. These emerging technologies and devices present exciting opportunities for material design to create consistent and intuitive user experiences.

- Augmented reality and virtual reality: With the increasing popularity of augmented reality (AR) and virtual reality (VR), material design can play a significant role in designing intuitive and seamless user experiences. Material design can help create consistent and familiar interactions across different AR/VR devices and applications.

- Cross-platform design: With the increasing number of devices and platforms, material design can help create consistent and familiar interfaces across different devices and platforms. Material design principles can ensure that the user experience is consistent across different devices and platforms, making it easier for users to navigate and interact with different applications.

8.APPENDIX

A.Source Code



```
<?xml version="1.0" encoding="UTF-8"?>
<project version="4">
  <component name="AutoImportSettings">
    <option name="autoReloadType" value="NONE" />
  </component>
  <component name="ChangeListManager">
    <list default="true" id="e5bef37d-7766-49a6-9c41-fe9e0de4bdaf" name="Changes" comment="" />
    <option name="SHOW_DIALOG" value="false" />
    <option name="HIGHLIGHT_CONFLICTS" value="true" />
    <option name="HIGHLIGHT_NON_ACTIVE_CHANGELIST" value="false" />
    <option name="LAST_RESOLUTION" value="IGNORE" />
  </component>
  <component name="ExternalProjectsData">
    <projectState path="$PROJECT_DIR$">
      <ProjectState />
    </projectState>
  </component>
  <component name="ProjectId" id="20JNo4pI8WcHLu6grMXMusPvAhu" />
  <component name="ProjectViewState">
    <option name="hideEmptyMiddlePackages" value="true" />
    <option name="showLibraryContents" value="true" />
  </component>
  <component name="PropertiesComponent"><![CDATA[{
  "keyToString": {
    "RunOnceActivity.OpenProjectViewOnStart": "true",
    "RunOnceActivity.ShowReadmeOnStart": "true",
```

```
    "RunOnceActivity.cidr.known.project.marker": "true",
    "cidr.known.project.marker": "true",
    "last_opened_file_path": "C:/android app/Owl-M-A-Material-Design-Study-App-main/build.gradle"
  }
}]]></component>
  <component name="RunManager">
    <configuration name="app" type="AndroidRunConfigurationType" factoryName="Android App">
      <module name="Owl_Application.app.main" />
      <option name="DEPLOY" value="true" />
      <option name="DEPLOY_APK_FROM_BUNDLE" value="false" />
      <option name="DEPLOY_AS_INSTANT" value="false" />
      <option name="ARTIFACT_NAME" value="" />
      <option name="PM_INSTALL_OPTIONS" value="" />
      <option name="ALL_USERS" value="false" />
      <option name="ALWAYS_INSTALL_WITH_PM" value="false" />
      <option name="CLEAR_APP_STORAGE" value="false" />
      <option name="DYNAMIC_FEATURES_DISABLED_LIST" value="" />
      <option name="ACTIVITY_EXTRA_FLAGS" value="" />
      <option name="MODE" value="default_activity" />
      <option name="CLEAR_LOGCAT" value="false" />
      <option name="SHOW_LOGCAT_AUTOMATICALLY" value="false" />
      <option name="INSPECTION_WITHOUT_ACTIVITY_RESTART" value="false" />
      <option name="TARGET_SELECTION_MODE" value="DEVICE_AND_SNAPSHOT_COMBO_BOX" />
      <option name="SELECTED_CLOUD_MATRIX_CONFIGURATION_ID" value="-1" />
      <option name="SELECTED_CLOUD_MATRIX_PROJECT_ID" value="" />
```

```
<option name="SELECTED_CLOUD_MATRIX_PROJECT_ID" value="" />
<option name="DEBUGGER_TYPE" value="Auto" />
<Auto>
  <option name="USE_JAVA_AWARE_DEBUGGER" value="false" />
  <option name="SHOW_STATIC_VARS" value="true" />
  <option name="WORKING_DIR" value="" />
  <option name="TARGET_LOGGING_CHANNELS" value="lldb process:gdb-remote packets" />
  <option name="SHOW_OPTIMIZED_WARNING" value="true" />
</Auto>
<Hybrid>
  <option name="USE_JAVA_AWARE_DEBUGGER" value="false" />
  <option name="SHOW_STATIC_VARS" value="true" />
  <option name="WORKING_DIR" value="" />
  <option name="TARGET_LOGGING_CHANNELS" value="lldb process:gdb-remote packets" />
  <option name="SHOW_OPTIMIZED_WARNING" value="true" />
</Hybrid>
<Java />
<Native>
  <option name="USE_JAVA_AWARE_DEBUGGER" value="false" />
  <option name="SHOW_STATIC_VARS" value="true" />
  <option name="WORKING_DIR" value="" />
  <option name="TARGET_LOGGING_CHANNELS" value="lldb process:gdb-remote packets" />
  <option name="SHOW_OPTIMIZED_WARNING" value="true" />
</Native>
```

```
    <Profilers>
      <option name="ADVANCED_PROFILING_ENABLED" value="false" />
      <option name="STARTUP_PROFILING_ENABLED" value="false" />
      <option name="STARTUP_CPU_PROFILING_ENABLED" value="false" />
      <option name="STARTUP_CPU_PROFILING_CONFIGURATION_NAME" value="Java/Kotlin Method Sample (legacy)" />
      <option name="STARTUP_NATIVE_MEMORY_PROFILING_ENABLED" value="false" />
      <option name="NATIVE_MEMORY_SAMPLE_RATE_BYTES" value="2048" />
    </Profilers>
    <option name="DEEP_LINK" value="" />
    <option name="ACTIVITY_CLASS" value="" />
    <option name="SEARCH_ACTIVITY_IN_GLOBAL_SCOPE" value="false" />
    <option name="SKIP_ACTIVITY_VALIDATION" value="false" />
    <method v="2">
      <option name="Android.Gradle.BeforeRunTask" enabled="true" />
    </method>
  </configuration>
</component>
<component name="SpellCheckerSettings" RuntimeDictionaries="0" Folders="0" CustomDictionaries="0" DefaultDictionary="application-level" UseSingleDictiona
<component name="TaskManager">
  <task active="true" id="Default" summary="Default task">
    <changelist id="e5bef37d-7766-49a6-9c41-fe9e0de4bdaf" name="Changes" comment="" />
    <created>1681274899552</created>
    <option name="number" value="Default" />
    <option name="presentableId" value="Default" />
    <updated>1681274899552</updated>
```

```
      <created>1681274899552</created>
      <option name="number" value="Default" />
      <option name="presentableId" value="Default" />
      <updated>1681274899552</updated>
    </task>
    <servers />
  </component>
</project>
```