Name:- Harsh Pramod Padyal D15B 40

Page No.
Date

MPL Assign 01

Q.1) a) Explain the key features and advantages of using Flutter for mobile app development

Ans.

Key Features of Flutter:-

1. Single codebase.
2. Fast performance
3. Hot Reload.
4. Rich UI components.
5. Native-like Experience
6. Cross-platform support
7. Open-source.

Advantages of Using Flutter:-

1. Saves Time & Effort.
2. High Speed Development.
3. Cost-Effective.
4. Attractive UI
5. Good Performance.
6. Easy Integration.

b) Discuss how the Flutter framework differs from traditional approaches and why it has gained popularity.

Ans How Flutter Differs from Traditional Approaches:-

1. Single Code base.
2. Hot Reload - Traditional apps require full restart after changes, But flutter updates instantly.
3. UI Rendering - Traditional apps require full restart after changes, but flutter updates instantly.

4. Performance:- Flutter compiles directly to native machine code, making it faster than frameworks that use a bridge (eg. React Native)

5. Customization:- Traditional UI design depends on platform-specific components, but Flutter provides fully customizable widgets.

Why Flutter is Popular Among Developers:

1. Fast Development - Hot Reload and Single codebase save time.

2. Cross-Platform support - works on mobile, web and desktop.

3. Beautiful UI - Rich, customizable widgets for modern designs.

4. High Performance - Runs smoothly without a bridge like React Native.

SDS Page No.
Date / /

Q. 2) a) Describe the concept of the widget tree in Flutter. Explain how widget composition is used to build complex user interfaces.

Ans.

Concept of widget Tree in Flutter :-

In Flutter, everything is a widget. widgets are arranged in a tree structure, called the widgets tree. This tree represents the UI of the app, where parent widgets contain child widgets.

For example, a Scaffold widget can have a column widget, which contains Text and Button widgets. Changes in widgets update the tree dynamically.

For eg :-

1. A ListView can contain multiple Card widgets.

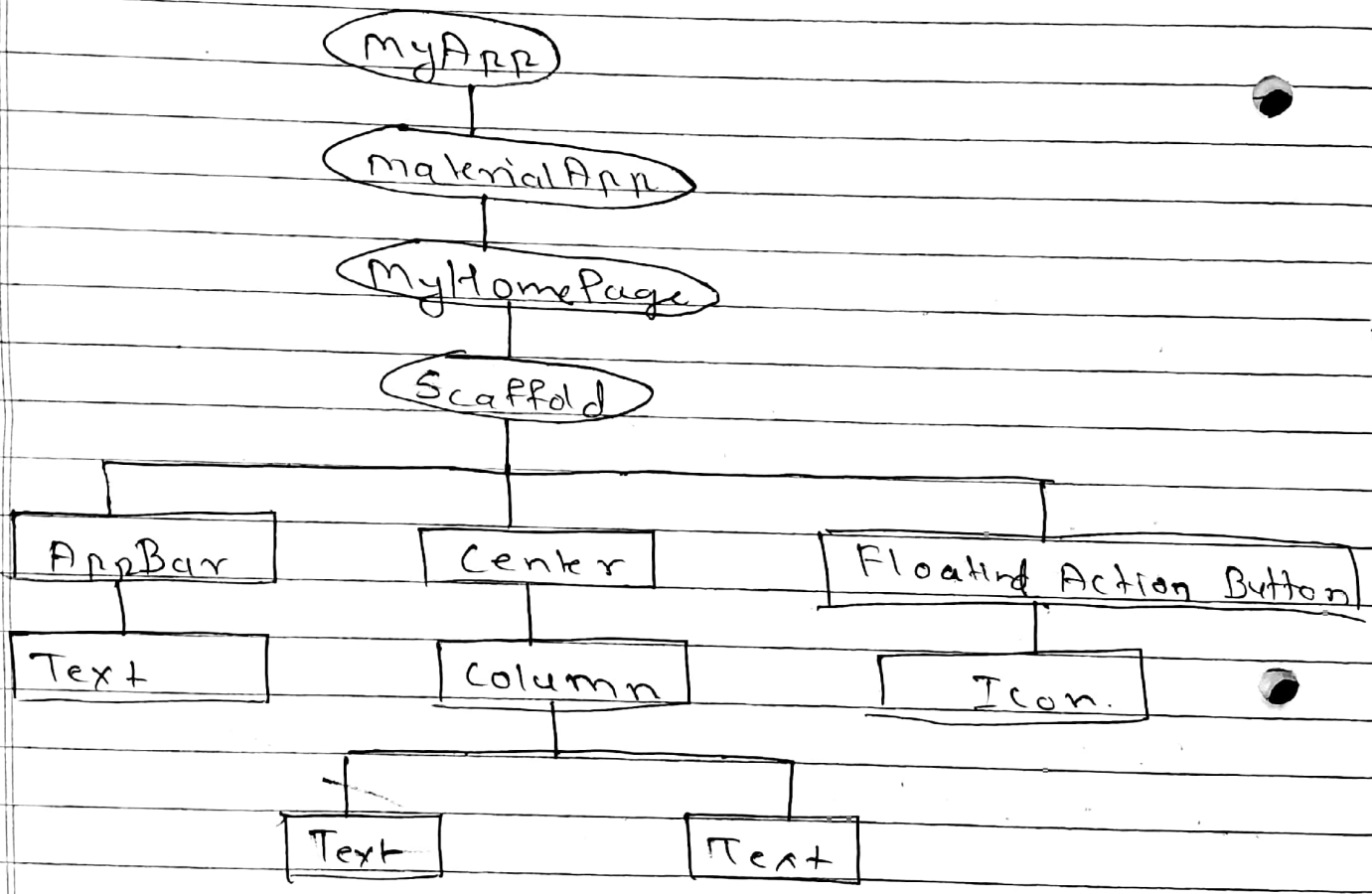
2. A column can hold Texts, Images, and Buttons.

b) Provide examples of commonly used widgets and their roles in creating a widget tree.

Ans. 1. Scaffold
2. Appbar
3. Text
4. Image
5. container
6. Row.

7. column
8. ListView
9. Elevated Button
10. TextField
11. Card
12. Stack.

Tree :-



Q-3)a) Discuss the importance of state management in Flutter application

Ans Importance of state Management in Flutter Applications:-

State management is important because it controls how app stores, updates, and displays data when the user interacts with it.

Why state Management is needed?

1. Keeps UI updated
2. Improves Performance
3. manage Complex Data
4. Ensures Smooth User Experience.

Types of State in Flutter:-

1. Local State.
2. Global state.

b) Compares and contrast the different state management

Ans

Approach	How it works	when to use
SetState	updates UI by calling setState() in a StatefulWidget	Best for small Apps or managing state within a single widget. Example, Toggling a button color.
Provider	Uses InheritedWidget to share state across widgets efficiently.	Suitable for medium-sized apps where data needs to be shared between multiple widgets Example, managing user authentication.
Riverpod	An improved version of Provider with better performance and simpler Syntax	Best for large apps that need complex state management with dependency injection. Example, Handling API data and app-wide themes.

SDS Page No.
Date / /

Q.4) a) Explain the process of integrating Firebase with a Flutter application. Discuss the benefits of using Firebase as a backend solution

Ans

Process of Integrating Firebase with a Flutter Application:-

1. Create a Firebase Project -
   Go to [Firebase Console](https://console.firebase.google.com/), create a new project

2. Add Firebase to Flutter App -
   Register the app (Android / iOS) and download the google-services.json (Android) or GoogleService-Info.plist (iOS).

3. Install Firebase Packages -

4. Initialize Firebase - Import Firebase in 'main.dart' and call 'Firebase.initializeApp()'.

5. Use Firebase Services - Implement authentication, database, or cloud functions as needed.

SDS Page No.
Date / /

Q.4) b) Highlight the Firebase Services commonly used in Flutter development and provide a brief overview of how data synchronization is achieved.

Ans.

Common Firebase Services Used in Flutter Development:-

1. Firebase Authentication - Provides user sign-in methods (Google, Email, Facebook, etc.)

2. Cloud Firestore - A NoSQL database

3. Firebase Realtime Database - Stores and updates data instantly across all connected database.

4. Firebase Cloud Storage - Usage for storing and retrieving files like images and videos.

5. Firebase Hosting - Deploys web apps with fast and secure hosting.

SDS Page No.
Date / /

How Data synchronization is Achieved:-

1. Real-time Updates - Firestore and Realtime Database sync data across devices instantly.

2. Listeners and streams - widgets listen for changes and update the UI automatically.

3. offline support - Firebase caches data, allowing app to work offline and sync when online.

This ensures fast, smooth, and automatic data updates in Flutter apps.