

# MAD Assignment



- Explain the key features and advantage of using flutter for mobile app develop.
- (\*) Single codebase for multiple platform  
write down one codebase for both platfo
  - (\*) Hot Reload : Instantly see changes in the app without restarting making develop faster
  - (\*) Fast Performance - Uses the Dart language and a compiled approach
  - (\*) Open Source & sharing community support.

## Advantages

- (\*) Faster Development
- (\*) Cost Effective
- (\*) Reduced Performance

Discuss how the flutter framework differ from traditional approaches and why it has gained popularity in development comm.

Single codebase v/s separate codebase  
Traditional Approach: Developer need to write separate code by Android.

- Rendering Engine vs. Native UI compo
- Traditional Approach : Relies on pl
- Native UI compo which can lead to inconsistencies and performance is

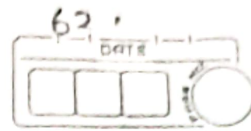
Why Flutter gained popularity?

- (i) faster development with hot Dev
- can install/see UI changes
- (ii) Cross platform efficiently : Business save time and resources by maintain a single codebase
- (iii) Consistent UI Across devices . Si
- flutter does not truly across different or version
- (iv) Improved performance : AOT compilation and direct access.

Q.2.

- (a) Describe the concept of widget Tree
- widget Tree in Flutter.

In Flutter, the widget Tree is the fundamental structure, it is hierarchical arrangement of widget the UI is entirely built Flutter UI is based using widget can be statless or The widget widget tree determines



## Input & Interaction widget.

TextField - Accepts text input from user

Elevation Button - A button with eleva

Ex. column (

children [

TextField (decoration: InputDecoration

elevated Button (

onPressed: () {

print ("button pressed");

} child: Text ("Submit")

)

}

}

/

## Display & Styling

Text: displays text on the screen

Image :- Shows images from n.w

Icon - Display icons

card :- A material design card with rounded

Ex. column (

children: [

Text ("welcome to Flutter", style

TextStyle (fontSize: 24

fontWeight: FontWeight.bold)





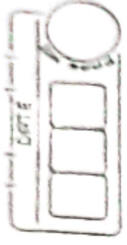
```
Example: class ProfileCard extends {  
    final String name;  
    final String imageUrl;  
    ProfileCard({required this.name, required  
        @override  
        widget build(BuildContext)  
        return Card(  
            child: Column(  
                children: [  
                    Image.network(imageUrl)  
                    SizedBox(height: 10)  
                ]  
            )  
        )  
    }  
}
```

(b) Provide examples of commonly used widget and their roles in creating widget.

1. Structure widget.

These widget acts as the foundation building the UI Material App. The parent widget of Flutter app that provides efficient configuration.

Ex. MaterialApp(  
 home: Scaffold(  
 appBar: AppBar(  
 title: Text('Flutter App')  
 )  
 )  
)



Discusses the important state management in Flutter application.

In Flutter, state refers to data that can change during the lifetime.

- User input
- UI changes
- Network changes
- Animation states

There are two types of states

Ephemeral state, small UI specific that doesn't get affected whole app

App wide states - Data stored across multiple widgets

- Efficient UI updates updates: Flutter UI rebuild whenever state changes

Efficient state management

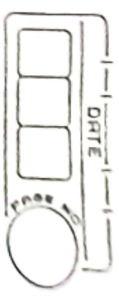
- code Maintainability & scalability

Managing state properly makes the code modular.

compose and contrast the difference

diff state management approaches available in Flutter such as

set, setState, Provider and Binkload



Provider : App wide state

Pros - Lightweight, recommended by  
Flutter efficient

cons: Boilerplate code for nested  
provided

Best use cases: Medium scale apps

Riverpod App: wide state (More scalable)

Pros : Eliminates providers limitation

Cons: Requires learning new concepts  
Best cases: Large apps needing global  
state

Example: Shopping cart.

Q4

Explain the process of integrating  
Firestore with Flutter.

→

Step (i)

Create a Firestore project

- Go to File console

click on "Add Project" and enter  
a project name

- configure google analytics.

Step (ii) Register the Flutter app  
with Firestore

- In Firestore project dashboard

click "Add App" and select

- For Android Enter android

package name.



Step (iv) install firebase dependencies

add firebase dependencies  
firebase core  
firebase auth  
cloud firestore

Step (v) configure firebase for android

Step (vi) Initialize flutter  
void main () async {  
 widget FlutterBinding().initialize();  
 await firebase.initializeApp();  
}

### Benefits of firebase

- (i) Easy to set up and scale  
No need to manage backend infrastructure scales automatically
- (ii) Authentication  
provides email password  
google facebook etc.
- (iii) cloud storage  
secure the file storage.
- (iv) push Notification.