Dhruv Maurya D15B/31

MAD Assignment No. 1 A1 DATE: 04

Explain key features and advantages of using flutter for mobile ~~web~~ app development/

① Single code base for multiple platforms
② Hot Reload. Instantly see changes in the app without restarting making developement faster
③ Fast performance: uses Dart language and a compiled approach for smooth high performance app
④ Opensource and Strong community Support Backend by google and a large developer community ensuring continous improvements & resources
⑤ Reduced performance issues: The app runs natively without relying on intermediate bridges like in react native reducing lag

Discuss how the flutter framework differs from traditional approaches & why it has gained popularity in developers community.

① Single codebase vs Separate codebase
Traditional approach relies on platform native component
uses a single dart based code- platform reducing developm

@ Rendering Engine vs UI Compone

Traditional approach: Developers rely UI components which can lead inconsistencies and performance issu

flutter: uses the skia rendering engi draws everything from scratch e a consistent UI across devices

Why Flutter has gained popularity.

1. Faster development with hot reload. Can instantly see UI changes without restarting the app making the itera process much quicker
2. Cross platform inconsistency
   Business save time and resource maintaining a single codebase for m platforms
3. Consistent UI across devices: Since flutt does not rely on native component the UI looks and behaves the same different versions
4. Improved performance:
   AOT compilation and direct acces to GPU rendering ensuring smooth animations and high performance

DATE:

Describe the concept of the widget tree in flutter Explain how widget composition is used to build complex user interfaces.

Widget tree in flutter:
In flutter, the widget tree is the fundamental structure that represents the UI of an application. It is a hierarchical arrangement of widgets where each widget defines a part of the user Interface.
Flutter UI is entirely built using widgets which can be stateless or stateful
The widget tree determines how the UI is rendered and updated when changes occur

Benifits of widgets Composition.

1. Reusability
2. Maintainability
3. Performance

b. Provides Example of Commonly used and their roles in creating a wi tree

⇒

1. Structural widget:
This widget act ast the foundation building the UI
Material App: The root widgets of th App that provides the essential con Scaffold provides a basic layout structur including an app bar, body, floatin action button etc
Container: A versatile widget used for p padding, margin & background custo

Ex: MaterialApp (
home: Scaffold (
appBar: AppBar (title: Text (
body: Container (
padding: EdgeInsets.all
child: Text ("Hello, Flut
));

DATE:

Inputs & Interaction widgets:

Textfeild - Accepts text input from users
Elevation Button - A button with elevation
Gesture Detector - Detects like taps. Swipes and long press

Ex:
```
Column (
  children [
    Textfield ( decoration: Input Deco
    Elevate Button (
      onPressed: (){
      print ("Button pressed") }
      child Text (Button Subm
    ) ; } ]);
```

Display & Styling widgets.
Text - display text on the screen
Image - shows images from assets network or memory
Icon : Display Icons
Card - A material design card with rounde corners and elevation.

```
column (
  children [
    Text ("Welcome");)
    Image. network ("url" ] ]);
```

Q.3)

a. Discuss the important State mgm in flutter applications

→

In flutter State refers to data th change during the lifetime of application. This includes:

User Input
UI changes
Network changes
Animation States

There are two type of States
· Ephermal state : Small UI specific state does not affect the whole App.
· App wide states: Data stared accross wedgets.

Importance of state management:
1) Efficient UI updates
2) Code maintainability.
3) Data consistancy
4) Synchronizatn

b. Compare & contrast the different state mgm Approaches avaliable in flutter such as State, provider & riverpod. Provide scenario where each approach is evitable.

→

Set state :

DATE:

Senerios for Each approach

- use StState when managing simple UI elements within a single widget
- use provider wne changing state across multiple widget

Explain the process of integrating firebase with flutter application discuss the benefits of using firebase a a backend solution

1. Create a firebase project from Console
2. Register the flutter app with firebase Enter the andwid package name and download sevices
3. Install Firebase dependinces
4. configure firebase for android & Ios
5. Initiate firebase in flutter

Benifits of using firebase:

(1) Easy to setup & scale

(2) Authentication
Provides email / password, etc A

(3) Cloud storage : Stores Assets on cl

(4) Push Notification : Sends real t
Notification

b. Highlight the firebase services commo
used in flutter development & provide a
overview of how data synchronization is
achieved

⇒ (1) Firebase Authentication
(2) cloud Firebase
(3) Realtime Database
(4) Firebase Cloud Messaging
(5) Firebase Analysis
(6) Firebase Hosting

Data Synchronization in firebase
Firebase ensures real time data synchroni
across multiple devices and platforms usi
firebase & real time Database

DATE:

Cloud Firestore Sync Mechanism:
Uses real time listners to update UI instantly when data changes
Ex: Firebase Firestore.instance.collection("users").
Snapshots().listen((Snapshot) {
for(var doc in snapshot.docs) {
print(doc['name']); }});

Real time database Sync Mechanism
Uses Persistent websockets connections for live updates.
Ex Database reference ref = FirebaseDatabase.instance.
ref("messages");
ref.onValue.listen((event) {
print(event.snapshot.value);

Offline Data Sync

Firestore caches data locally & syncs changes when the device is online

Cloud functions for automated updates

Automates backend logic like triggers when data changes

Dhruv Maurya DISB/31

MPL Assignment - 2 DATE: 03/03/24

1. Define progressive web App (PWA) and Explain its significance in modern web development. Discuss the key characteristics that differentiate PWAs from traditional mobile apps.

- Significance of PWA in Modern web development
- Cross platform compatibility
- Offline Support
- Fast performance
- No app store required
- Lower development cost

Key difference between PWA & traditional mobile Apps

Feature	PWA	Traditional Mobile App
Installation	Direct from brower	Download from app store
Internet required	Work offline with Caching	Usually requires Internet
Performance	Fast with service workers	Faster but needs installation
Updates	Automatic no app store approval	Manual updates needed
Development cost	Lower	Higher

Q.2) Define responsiveness web design and explain in the context of progressive web apps. Contrast responsiveness fluid and adaptive web design approaches

→ Responsive web design (RWD) is technique that progress adjust automatically to different screen and devices

- But Importance of responsive design is PWA
- Better User experience. PWA work smoothly device
- SEO prefer.. google ranks responsive like's

Comparison of web design approaches

Approach	How it works	Pros
Responsive	Use flexible grids and CSS media queries to adjust layout.	Work on all devices (complex design)
Fluid	Use percent based width instead of fixed pixels to elements resize smoothly	works well on different screen sizes easy to implement
Adaptive	Uses fixed layouts that change at specific breakpoints	Optimized for known screen sizes required design for screen size

DATE:

Key difference
Responsive adapts dynamically to all screens

3) Describe the life cycle of service worker including to registration installation and activation phase.
Lifecycle of service workers
A service worker is a script that runs in the background and helps a web app work offline, load faster and send push notifications.
1. Registration phase
The browser registers the service worker using JavaScript code Eg :-
if ('service worker in navigator') {
Navigator.serviceWorker.register ('/sw.js')
then (() => console.log ('service worker registered)?
Catch (error => Console.log (Registration failed error)

4) Installation phase
The service worker downloads necessary (HTML, CSS, JS) and stores them in cache.
Code Ex.
self.addEventListener ('install', event => {
event.waitUntil (
caches.open (app.cache).then (cache =>
returns cache.addAll (['/index.html',
'style.css']);
});
});

This ensures the app loads even without the inte

3. Activation phase
- The old Service worker is replaced with the
- Unused cache files from the previous versio

decided final step fetch x Sync

Q-4) Explain the use of Indexed DB in the Syste

→ Working of data Storage.

Use of Indexed DB in Service Worker for dat
Storage indexed DB is browser database th
stores large amounts of structed data lik
Retrieving data efficiently.

Why use Indexed DB in Service workers?
1. Offline Support - Store data when offline and s
latest
2. Efficient storage - Stores data in structu
like user setting, cart items or form inputs

Opening the database
let db
let request = indexed DB. Open ('My Database')
request.onsuccess = function (event) {
db = event.target.result
}

Creating a store & adding data
request.onupgradeneeded = function (event) {
let db = event target result
let store = db.createObject store (Users,
{key path, id})
store.add ({id: 1, name: 'John', age: 25})