Understanding App development ecosystems

Android vs iOS

- Android: Java, Kotlin
- iOS: Swift
- Cross-platform frameworks: Flutter, React Native, Xamarin

Why Flutter is ideal for beginners:

- Single codebase for Android + iOS + Web + Desktop.
- Fast development (Hot Reload).
- Strong UI capabilities (Material + Cupertino).

Data types & variables

- int, double, String, bool, List, Map
- var, final, const

Functions

- name, parameters and return types

Class

A **class** is a blueprint (template) used to create objects. It defines **properties (data)** and **methods (actions)**. Think of **a class as a design** and **an object as a product built from that design**.

Example

```
class Car {
  String brand;
  int speed;

  Car(this.brand, this.speed);
}
```

Explanation

- brand and speed are properties/variables
- The constructor *Car(this.brand, this.speed)* creates an object with values

Creating an object

```
var myCar = Car("Toyota", 120);

print(myCar.brand); // Toyota
print(myCar.speed); // 120
```

Methods

A method is a function inside a class. It describes actions the object can perform.

Example

```
class Calculator {

  int add(int a, int b) {
    return a + b;
  }

static int sub(){
  return 5;
}

}

var calc = Calculator();
```

```
print(calc.add(3, 5)); // 8
print(Calculator().sub()); // 5
```

Method types

i). **Instance methods** - require an object

ii). **Static methods** - belong to the class itself (no object needed)

Enum

Enum stands for enumeration, a list of **predefined constant values**. Useful for values that don't change and should be limited.

Examples

User roles: admin, student, teacher

Days of week: Monday – Sunday

Gender: male, female

App navigation states

Dart example

```
enum UserRole {
  admin,
  teacher,
  student
}
```

Using enum

```
UserRole role = UserRole.teacher;
if (role == UserRole.teacher) {
  print("Welcome teacher!");
}
```

Development Environment Setup

Tools

- **Flutter SDK**
- **Android Studio** (for SDK + emulator)
- **VS Code** (preferred for beginners)
- Dart & Flutter plugins

Task 01

Build and run first flutter app.

Widgets

A **widget** is a building block of the UI. Widgets describe *what* you want to show, not *how* to draw it.

Text("Hello")

This widget tells Flutter: Display the word “Hello”, Flutter handles drawing on screen.

Widgets **compose** into a tree: Small widgets → bigger widgets → full screens → full apps

This is called the **widget tree**.

Some of the widgets categories

i). Structural Widgets (Screen organization)

These provide structure, layout, and scaffolding.

MaterialApp	Wraps the entire app with Material design, themes, navigation
CupertinoApp	Wraps the app with iOS design
Scaffold	Full page layout: app bar, floating button, body
AppBar	Top navigation bar
Drawer	Side navigation panel
BottomNavigationBar	Tab navigation at bottom
TabBar / TabBarView	Top navigation tabs
SafeArea	Avoids system UI overlaps (notch, status bar)
Navigator	Screens routing (push/pop screens)

ii). Stateful vs Stateless widgets

This is the most important conceptual difference in Flutter.

Stateless Widget	**Stateful Widget**
UI does NOT change after build	UI CAN change over time
No internal variable updates	Has internal state
Good for static content	Can re-render when variables change

iii). Layout widgets (Arrangement)

Layout widgets don’t show anything, they **organize** other widgets.

A. Single child layout

Center	Centers child
Padding	Adds spacing around child

Container	Styling, size, borders, background
Align	Positioned within parent
SizedBox	Fixed width/height or spacers
AspectRatio	Enforces width–height ratio

B. Multi child layout

Row	Horizontal arrangement
Column	Vertical arrangement
Stack	Overlapping children
Wrap	Auto-line wrapping when row/column is full
GridView	Grid layout (picture gallery)
ListView	Scrolling list
Expanded	Takes remaining space in row/column
Flexible	Shares available space flexibly

iv). Visible / Display / Presentational UI Widgets

These are the widgets users see and interact with:

Text	Displays string text
Icon	Shows predefined icons
Image.asset	Local images
Image.network	Online images
CircleAvatar	Profile photos
Card	Material-style container for content
Divider	Thin line to separate content
Chip	Small info element (tags/labels)
Tooltip	Show hint text on hover/long press
RichText	Styled multi-span text

v). Buttons & Interactions

ElevatedButton	Main action button (with shadow)
TextButton	Flat text button
OutlinedButton	Button with border
IconButton	Icon as button
FloatingActionButton	Round floating action button
GestureDetector	Custom gesture detection (tap, swipe, drag)
InkWell	Touch ripple effect on tap

vi). Scrollable Widgets

SingleChildScrollView	Scroll when content is large
ListView	Vertical/horizontal scroll list
GridView	Grid scroll list
PageView	Swipe between full screens

CustomScrollView	Advanced scrolling layouts
SliverAppBar	Collapsing / floating app bars
SliverList / SliverGrid	Sliver list/grid building blocks

vii). Input & Form Widgets

TextField	Input text
TextFormField	TextField with validation
Form	Wraps multiple inputs, handles validation
Checkbox	True/False selection
Switch	Toggle on/off
Radio	Choose one from a group
DropdownButton	Selection list menu
Slider	Select numeric value from range
DatePicker	Choose date
TimePicker	Choose time