CSA 7107T
Mobile Application Development

Complete Study Notes

HandNotes
by
Kamal Kishor

1. Introduction to Mobile Application Development

What is Mobile Application Development?

Process of designing, developing, testing, and deploying applications for mobile devices such as smartphones and tablets.

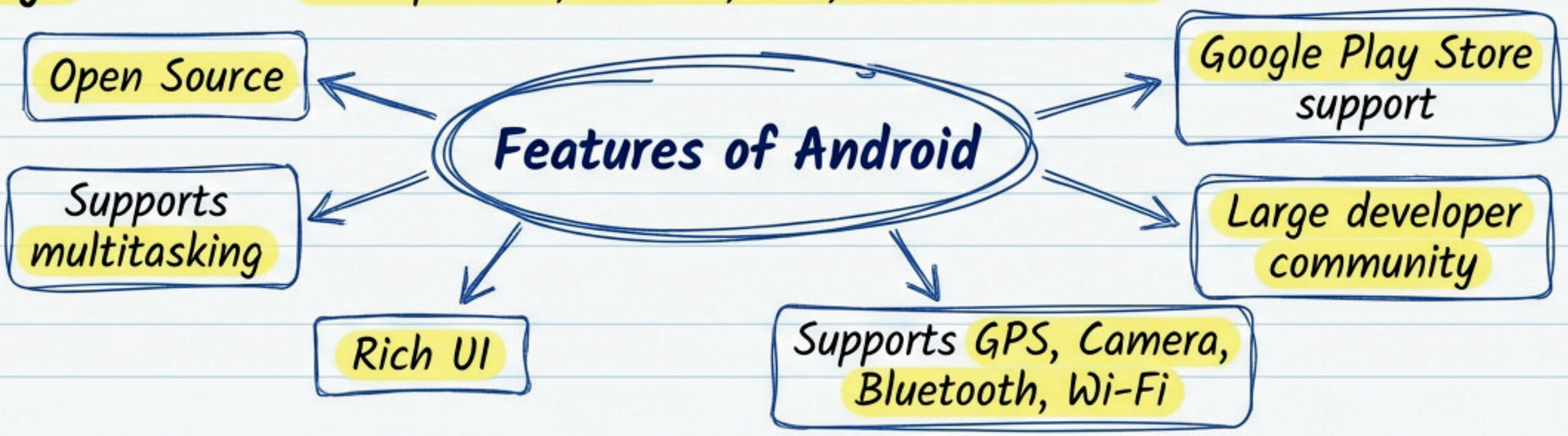
- Android
- iOS
- Windows (limited)

In BCA, we mainly focus on **Android Application Development.**

---

2. Android Overview

What is Android? **Open-source**, **Linux-based** mobile operating system developed by **Google**. Used for smartphones, tablets, TVs, and wearables.



3. Android Architecture

5. Applications
Built-in apps (Phone, SMS, Browser) & User-installed apps.

4. Application Framework
Provides APIs for developers: Activity Manager, Window Manager, Content Providers, Resource Manager, Notification Manager.

3. Android Runtime
ART (Android Runtime). Core Java libraries.
Executes .dex files.

2. Native Libraries
Written in C/C++. Includes: SQLite (database), WebKit (browser engine), Media framework, OpenGL (graphics).

1. Linux Kernel
Base layer. Handles: Process management, Memory management, Device drivers. Provides security and hardware abstraction.

4. Android SDK & Tools

Android SDK (Software Development Kit)

A set of tools required to develop Android applications.

Includes: SDK Tools, Platform Tools, Build Tools, Emulator, APIs.

Android Studio

Official IDE for Android development.

- Code editor
- Layout designer
- Debugging tools
- Emulator
- Gradle build system

5. Android Virtual Device (AVD) & Emulator

AVD Definition:

A configuration that defines:

- Device type
- Screen size
- Android version
- Hardware properties.

Emulator Definition:

Virtual Android device.
Used to test apps without real device.
Simulates hardware features.

6. Android Application Components

1. Activity

Represents a single screen.
Entry point for user interaction.
Example: Login screen

2. Service

Runs in background.
No UI.
Example: Music player

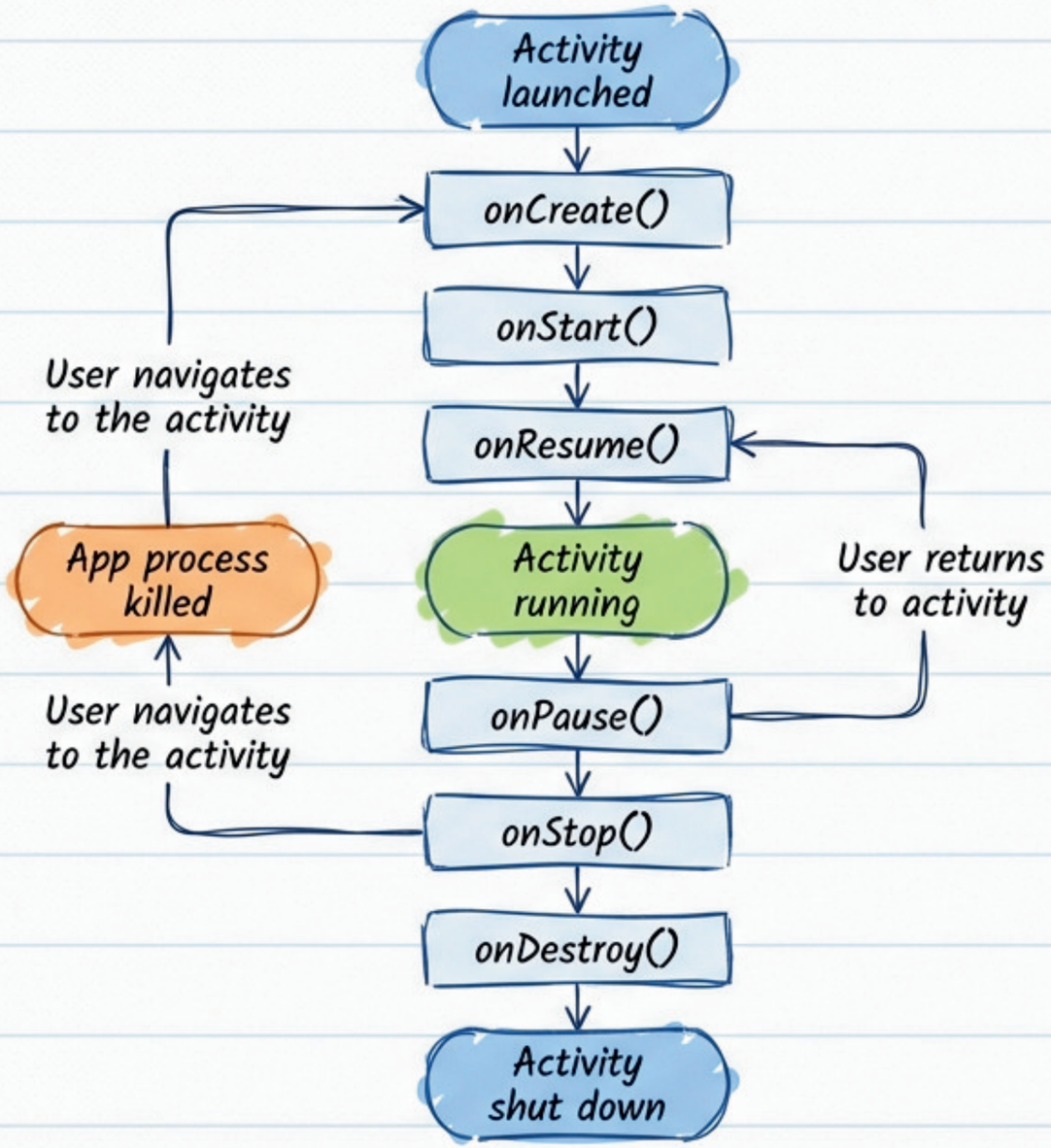
3. Broadcast Receiver

Responds to system-wide messages.
Example: Battery low, SMS received

4. Content Provider

Manages shared data. Used to access contacts, media, etc.

7. Android Application Life Cycle



1. **onCreate()** – Activity created
2. **onStart()** – Activity visible
3. **onResume()** – User interacting
4. **onPause()** – Activity partially hidden
5. **onStop()** – Activity not visible
6. **onDestroy()** – Activity destroyed

Important!
Understanding lifecycle helps:
- Save data
- Release resources
- Improve performance.

8. User Interface (UI) in Android

Views (Basic UI components)

- TextView
- EditText
- Button
- ImageView
- CheckBox
- RadioButton

ViewGroup (Layouts)

Container for views.

- LinearLayout
- RelativeLayout
- ConstraintLayout
- FrameLayout
- TableLayout

9. Layout Optimization

- ☑ Use ConstraintLayout for better performance
- ☑ Avoid deep layout nesting
- ☑ Use wrap_content and match_parent
- ☑ Use reusable layouts

10. Drawable Views

Drawables are graphical resources.

Types: Bitmap, Shape, Vector, Layer List

Used for: Backgrounds, Buttons, Icons

11. Android UI Controls

Button, Spinner, ListView, RecyclerView, ProgressBar, SeekBar.

12. Event Handling & Listeners

Event Handling

Responding to user actions like: Click, Touch, Long press.

Listeners

- OnClickListener
- OnTouchListener
- OnItemSelectedListener

13. Adapters

Adapters act as a bridge between data and UI.

Types: ArrayAdapter, SimpleAdapter, CustomAdapter.

Used in: ListView, Spinner, GridView.

14. Menus in Android

Used to provide actions to users.

Types of Menus:

- 1. Options Menu
- 2. Context Menu
- 3. Popup Menu

15. Dialogs & Toasts

Dialogs

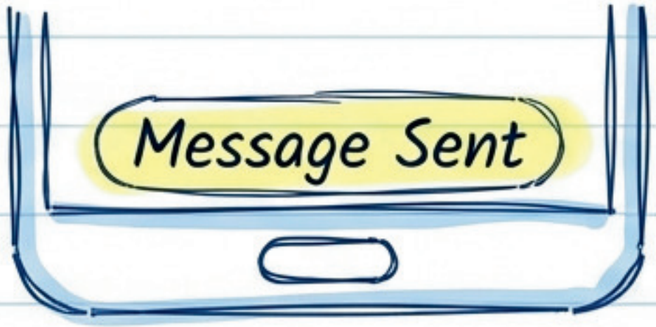
Used to show messages or ask input.

Types: AlertDialog, Custom Dialog.

Toast

Small popup message.

Short duration.

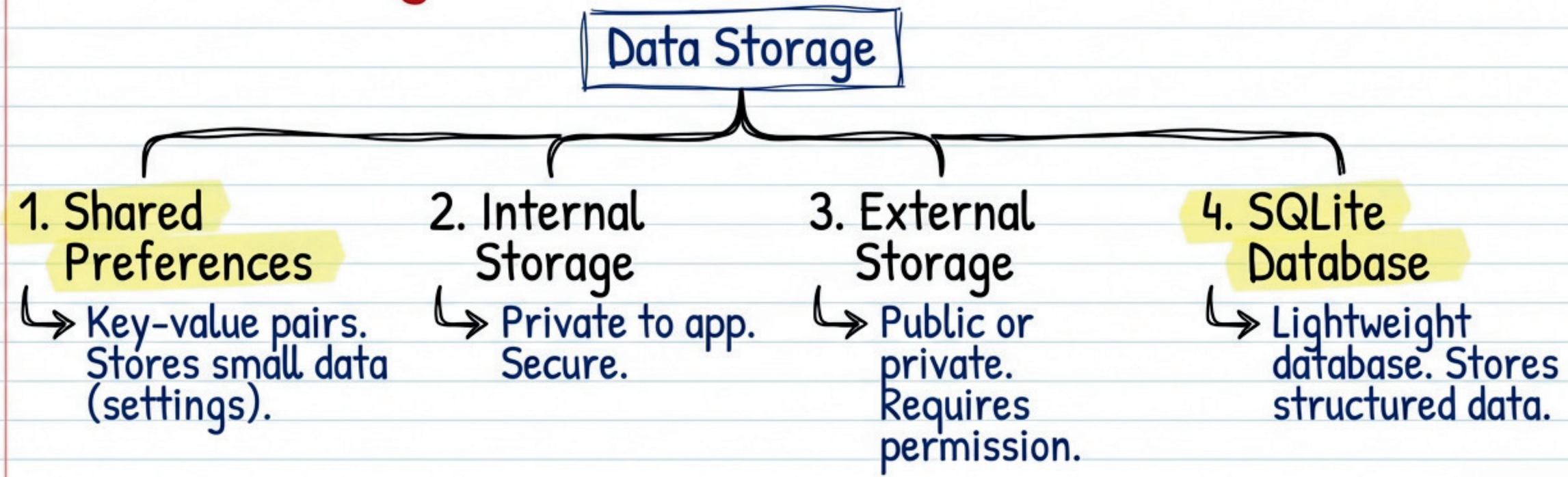


16. Notifications

- Used to alert users
- Shown in notification bar
- Supports actions, icons, sounds

---

17. Data Storage in Android



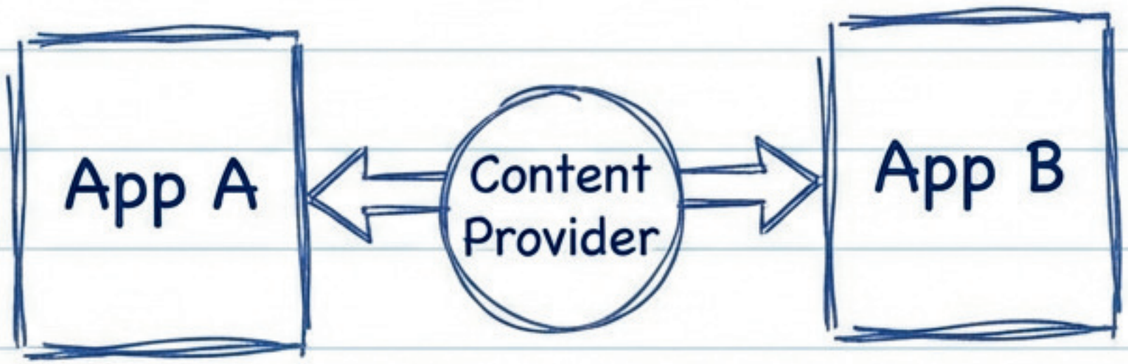
18. Using SQLite

- Embedded database
- No server required
- Supports CRUD operations
- Uses SQL queries

19. Content Providers

Share data between applications.
Standard interface.

- Contacts Provider
- Media Store



20. Networking APIs

Used to fetch data from internet.

- HTTP requests, REST APIs, JSON/XML parsing.

Libraries

- HttpURLConnection, Volley, Retrofit.

21. RESTful Web Services

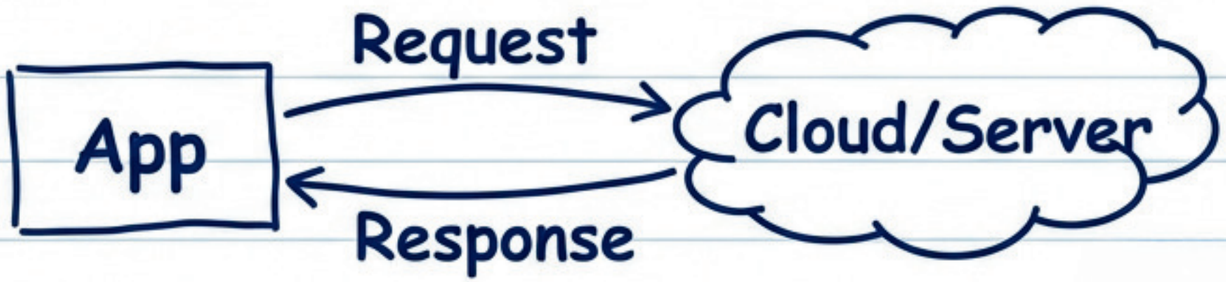
REST = Representational State Transfer

HTTP methods:

- GET
- POST
- PUT
- DELETE

Data format:

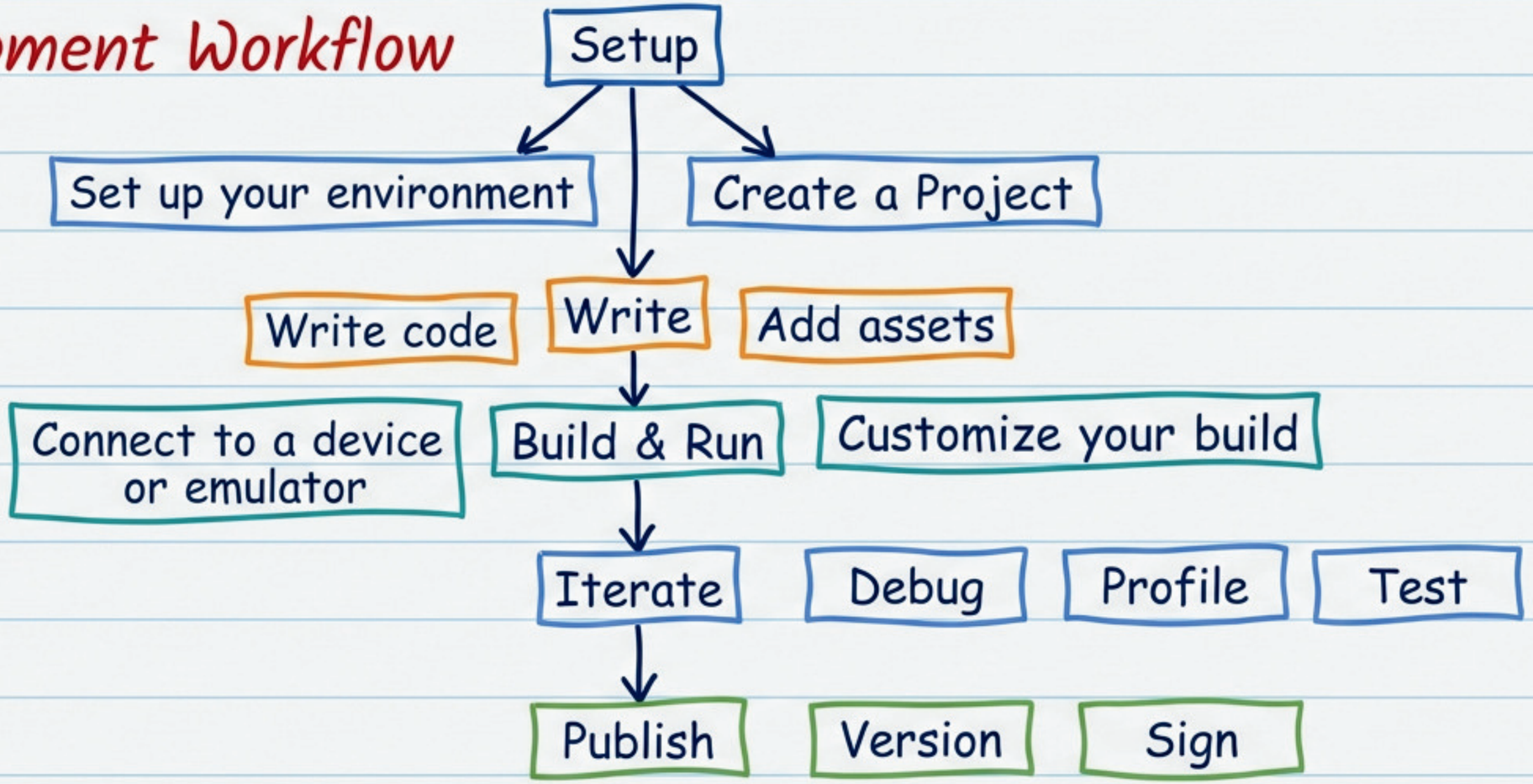
- JSON (mostly)
- XML



22. Google Play Services

Provides APIs for: Maps, Location, Authentication, Cloud Messaging, Analytics.

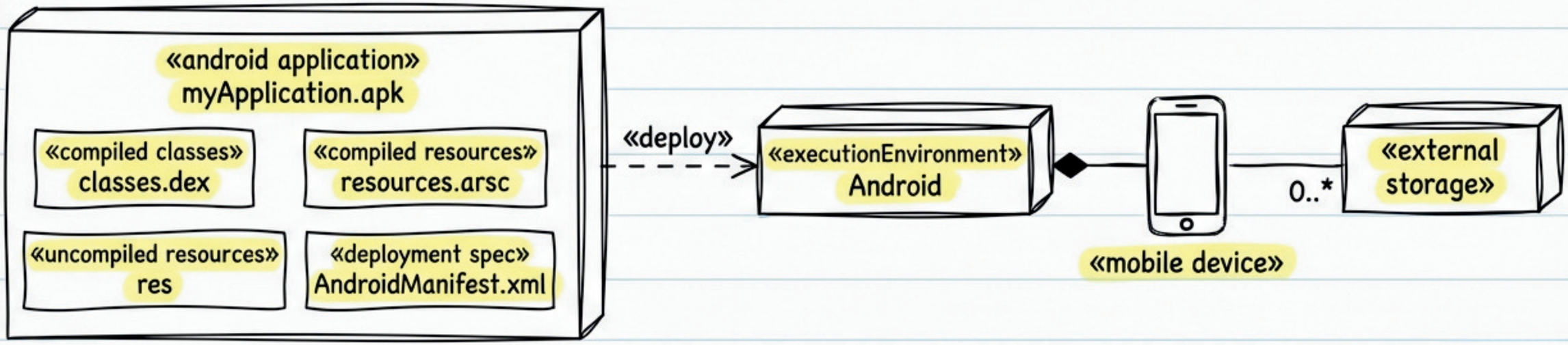
Development Workflow



23. Publishing Android Application

1. Generate signed APK
2. Create Google Play Developer account
3. Upload app
4. Add description, screenshots
5. Submit for review

Android Application Deployment Diagram



24. Testing Android Applications

Types of Testing:

- Unit Testing, UI Testing, Instrumentation Testing.

Tools:

- Android Emulator, Real devices, Logcat.

Conclusion

- Mobile Application Development enables developers to:
  - Create user-friendly mobile apps
  - Use modern tools and APIs
  - Integrate internet and databases
  - Publish apps globally

Good Luck!