* User Interface Design for Mobile Application

- A mobile user interface (mobile UI) is a graphical and usually touch-sensitive display on a mobile device, such as [smartphone] or [tablet] that allows the user to interact with the device's apps, features, content and functions
- The mobile user interface design is what you see, when you use an application.

It includes:-

- The layout of the information, commands and content in an app.
- Click points must be usable for touch-based selection with a finger.
- Maximize the content window size. On small screen, the UI should not unnecessarily dominate screen size.

* Types of User Interfaces for Mobile Apps:-

Here are different types of mobile UI patterns that you must know.

(1) **Splash Screen**

The first screen you see on the launch of any app that shows the name of the app is the splash screen. It is used as a way to conceal the processing time and help the user know that the app is being prepared in the back.

(2) **Onboarding Screen**

These screens help first-time users understand the app's main features, which tell the app's next steps, essential elements, and functionality and make the user journey less complicated.

(3) **Home Screen**

It is the main screen of the entire app. All other screens are linked to home page as it highlights the functional menu and features.

(4) **Login and Profile Screens**

- The Profile screen, holds all your information and progress and must provide ~~to~~ customization to the fullest.
- Logic Screen must clear all unnecessary distractions and make the experience quick, interactive and smooth.

5 App Screens

- App Screens are an important part of any app.
- They are the main screens that user interact with.

Mobile-app Design Examples

Whatsapp, Airbnb, Venmo, Starbucks etc.

* Managing Application data.

- There are various data storage options available on the android platform.
- When you develop Android applications, you have a number of options in terms of how you store and manage your data.
- Internal Storage

You can store data items to the internal storage on your user's device. The advantage to this is that it does not rely on any external media.

but the disadvantage is that some devices have ~~extra mdi~~ extremely limited amounts of storage space available.

FileOutputStream datafileOutput = openFileOutput("datafile", Context.MODE_PRIVATE);

- **SD Cards**

Many Android users are dependent on external media storage such as SD cards due to insufficient internal storage.

SD cards offers your apps an increased amount of storage space, but they impose additional considerations.

- **Databases**

- If your app uses more complex data items, you can create a SQLite database to store them using the relational model.
- When you create a SQLite database for your app, you can access it from any point within your java code, but it will not be accessible from anywhere else.