Q2.a. The three types of mobile application development. And their pros and cons.

Ans.

- Native Apps

- Web apps

- Hybrid apps

===== Native==========

they are developed for specific devices, in a platform-specific programming language – meaning that an app developed for iOS won't exist in the Google Play Store or work on an Android phone.

native apps can take full advantage of the device features, such as the camera, the contact list, GPS, and more. They run directly from the platform they are built for, and they offer a genuine native experience.

Advantages….

- they are fast and responsive because they are built for that specific platform.

- they have the best performance

- they are distributed in app stores

- they are more interactive, intuitive and run much smoother in terms of user input and output.

- internet connection is not required. Although, it depends on the functionality.

Disadvantages…

- experienced developers are need bcos it uses difficult languages.

- More expensive

- not a better option for simple apps

====Web apps==========

Web apps are simply websites that 'feel' like an app because of their interactivity and functionality. They are served through the internet, they run in browsers and cannot be downloaded to a mobile device like native apps can. Web apps lie on the opposite side of the spectrum from native apps, and this is clear through their upsides and downfalls.

Advantages…

- Easy to build

- easy to maintain

- not expensive

- build one app for all platforms.

Disadvantages...

- needs a browser to run

- it is slower as compared to native apps

- cannot leverage device utilities

- not interactive and intuitive like native apps.

- no icon on mobile desktop.

===Hybrid apps==========

Hybrid mobile apps sit exactly between native apps and web apps. They have a feel of a native app because they are downloadable from an app store and live on your home screen, but they rely on rendering in a browser that's embedded inside the app. Hybrid apps are a brilliant syndication of the best of both worlds. They leverage the native benefits while allowing for the same app to be available both through iOS App Store and Google Play Store.

Advantages.....

- easier to build as compared to native app due to the web technology

- no browser needed as opposed for a web app

- cheaper than native apps

- one app for all platforms using a technology like cordova

- have access to device inernal APIs

Disadvantages...

- slower than native apps

- more expensive than standard web app

- less interactive like native apps

- customization will take you away from hybrid to native

a. what are the pros and cons of using cross platform framework?

Ans.

Cross platform frameworks are used to develop codebase to be used for different platforms in mobile app development. Forinstance, cross platform can allow a codebase to be used for windows, android, IOS platform etc.. Some cross platform frameworks are;

Xamarin, Appcelerator, NativeScript, RubyMotion, QT

Pros of cross-platform mobile app development —

this is the interesting part...read carefully.

= Speed

Cross-platform mobile apps are faster in terms of development as compared to native iOS and Android apps. One codebase can be used on multiple platforms instead of creating codebase for each platform.

= Less Development Cost

The development of the cross-platform mobile app is cost-effective & of course a smart way to develop your app. It saves your time and money.

=Simplicity

You can upgrade the app at any moment and fix the bugs without any problem. With technologies like PhoneGap and Appcelerator, the mobile app developer can easily deploy changes in cross-platform solution.

=Consistent Design and Easy for Marketing

Cross-platform mobile apps are consistent in design and well navigated. It helps the mobile app developers to design different apps in different platforms and helps in proper sync between two different mobile apps. Though it fulfills the demands of specific sets of people, it also makes marketing easier and creates better opportunities.

=Significant Reach

The cross-platform mobile app can reach global users with all types of smartphones or tablets, whether they are operated on Android, iOS, windows or any other operating system. It seamlessly runs on all operating systems without any glitch.

Cons of cross-platform mobile app development

=User Experience

Developing an app for different OS and screen layout is a bit challenging for mobile app developers. It's always important for an app to run seamlessly in major operating systems. Major successful apps are built with native iOS or Android OS to create a better engagement platform with more users.

=Integration challenges

Integrating cross-platform app to local setting and engaging third-party cloud service is difficult. This is the biggest disadvantage. HTML5 hybrid apps depend on callback-style programming to communicate

with native plugins, which makes the code unnecessarily complicated. Also, for some tasks, this might lead to impractically slow solutions.

=Platform limitations

Each platform has a unique set of features and functionality to make the native app more creative. In cross-platform apps, these functionalities and tools are not available while developing an app for multiple platforms. Native cross-platform SDKs is not mature yet and its GUI needs to be coded each time for platform-specific look and feel.

=No more Flexibility

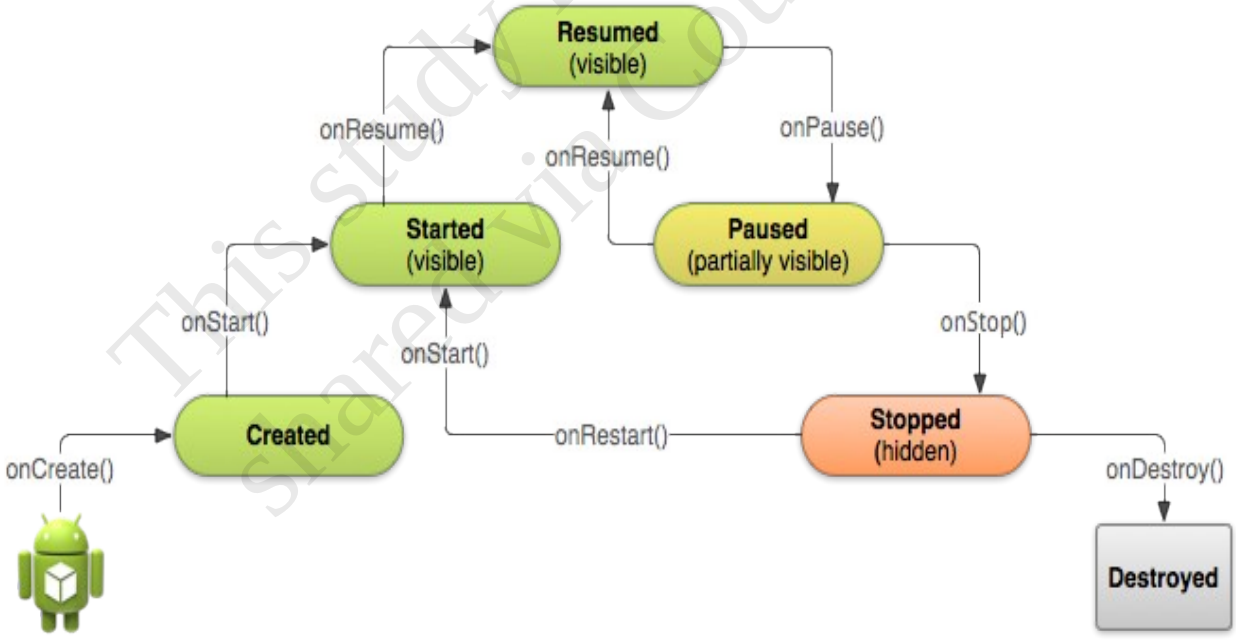
A medium level phone doesn't have high-level hardware configuration to read HTML 5 animation and show the exact GUI and the browser components also vary from platform to platform. So it's relatively painful to support old operating systems. To upgrade the app with latest features needs lots of development work for app developer after upgrading the latest version of SDK.

=Conclusion:

Cross-platform mobile app or Hybrid mobile apps offer good performance in a cost-effective way but native apps give 100% performance and highly productive in nature.

Q3. A. explain the android activity life cycle with the aid of a diagram.

Ans.



Method	Description

onCreate	called when activity is first created.
onStart	called when activity is becoming visible to the user.
onResume	called when activity will start interacting with the user.
onPause	called when activity is not visible to the user.
onStop	called when activity is no longer visible to the user.
onRestart	called after your activity is stopped, prior to start.
onDestroy	called before the activity is destroyed.

Q4. A. Discuss any five android application common structure.

Ans.

- **Views** such as lists, grids, text boxes, buttons, and even an embeddable web browser

- An **Activity Manager** that manages the life cycle of applications and provides a common navigation backstack

- **Activity** is the presentation layer of your app: there will be one per screen, and the Views provide the UI to the activity

- **Data storage** provide data for your apps, and can be shared between apps – database, file, and shared preferences (hash map) used by group of applications

- A **Notification Manager** that enables all apps to display custom alerts in the status bar

Q5. A. identify any four goals of hci

Ans.

The goals of HCI are to produce usable and safe systems, as well as functional systems. In order o produce computer systems with good usability, developers must attempt to:

•understand the factors that determine how people use technology

•develop tools and techniques to enable building suitable systems

•achieve efficient, effective, and safe interaction

•put people first

Q6. A. things to consider when designing for mobile apps.

Ans.

— Responsiveness

Responsiveness has become an integral part of the front end development. You can not avoid going responsive. Whatever your application is it should be fit to view an any size of the screen.

When you say mobile devices, you are not just referring to smartphones. The screen size varies from smartphone to other devices like devices like tablets that people use to access the web on the go. And then there are two different screen modes in which the devices can be viewed — landscape and portrait mode. So the design should consider these variations in the dimensions of screens.

Also, the size of the screens of devices keeps on increasing as newer models of the same device come in the market. At the same time, innovative screens like the new iPhone X and the infinity display on the Samsung Galaxy brings new challenges and possibilities to the UX and UI of mobile applications.

— Keep it Simple

Mobile devices are short in real estate. So knowing what is important is the key. What are the contents to be shown and what to hide? This ensures easy navigation and easy and uncomplicated user experience.

Simplicity is not mean minimalist design. The design should be such that the user should be able to do their tasks in as fewer steps as possible. Also, make it a point to keep only one call to action on one screen.

— Avoid Excessive Scrolling

Excessive scrolling can be really annoying. The design should be such that the user doesn't have to scroll through very long to get to the content that he needs. Instead, create navigation that can lead the user to the specific points he needs to get.

— Know the Latest Trends

Keeping up with the latest trends is essential for creating the most effective UX and UI. To know what is gaining popularity among the user helps learn and draw inspiration to create a better, effective design.

— Finger-Friendly Designs

With mobile devices, we are using the touch of our fingers to navigate through. There are different hand gesture that we use to do specific tasks like swipe to get to the next onboarding screen. Similarly, we have gestures like touch, touch-and-hold, double tap, pinch and zoom that allows us to operate the mobile devices. The user should be able to figure these gestures without any difficulty in a mobile application.

Also, people hold their phone in different ways. Sometimes they hold it with one hand, sometimes with two, and with one hand holding the phone. Most of the time people use their phone with one hand like when scrolling or calling someone. But when texting users need both hands. So the design must be able to facilitate the user's experience by keeping these in mind.

— Give feedbacks

Applications may sometimes take time to load. But a user may not be patient enough to keep waiting for the application to load if there is no feedback. The user may also think that application is broken and refrain from using it. So it is always important to give feedback to the user not just when loading the site but after every time a particular task is completed. Line when an order has been successfully placed. The user should be getting a message that task he did was completed otherwise leaving him in a state of confusion.

Conclusion

Designing the UI/UX for mobile devices are different from that of the desktop. There are constraints of the screen size when placing the elements of the website. The websites used in mobile devices are by touch, so that has to be taken into consideration. Also, the design has to planned keeping in mind the responsiveness of the application. But when the application works well on mobile devices, it increases the accessibility to the users and thus the number of users to your business.