

ASSIGNMENT - 2

Q.1 A progressive Web App (PWA) is a type of, web application that leverages modern web technologies to deliver a native app-like experience to users across different platforms & devices. PWAs are designed to combine the best features of web & mobile apps, providing users with fast, reliable & engaging experiences while also offering developers a more efficient & flexible approach to building deploying applications.

* Key characteristics of PWAs -

① Progressive enhancement :-

PWAs are built with progressive enhancement in mind, meaning they can work for any user regardless of the browser or device they are using. They utilize features like responsive design, service workers & graceful degradation to ensure that users get a consistent experience regardless of their device or network conditions.

② Responsive Design :-

PWAs are designed to adapt to different screen size & orientations, providing a seamless user experience across desktops, tablets & Smartphones.

③ Connectivity Independence :-

PWAs can work even when in low or no network connectivity environments by utilizing service workers to cache content & enable offline

access. This ensures that users can still access the app & its content even when they are offline or on a slow network.

In comparison to traditional mobile apps, PWAs offer several advantages:-

① Cross platform compatibility :-

Unlike native mobile apps, which require separate development efforts for different platforms (IOS, Android). PWAs are built using web technologies & can run on any platform with a modern web browser.

② No Appstore approval process :-

PWAs can be deployed directly on the web without going through a lengthy app store approval process. This allows developers to push updates & new features more quickly & easily.

③ Lower development Costs :-

Since PWAs can be built using web technologies developers can leverage their existing skills & tools reducing the development costs associated with learning platform-specific languages & framework.

Q.2. → Responsive web design is an approach to web design that aims to create web pages that respond to the user's behaviour & environment based on screen size, platform & orientation. This is achieved through the use of flexible grids & layouts using CSS media queries. The goal is to provide an optimal viewing

experience across a wide range of devices from desktop computers to smartphones & tablets without the need for separate mobile or desktop version of a website.

① Responsive web design uses CSS media queries to dynamically adjust the layout & content of a website based on the characteristics of the device or viewport size.

It offers a fluid & flexible web design that can adapt to different screen sizes & orientations.

② Fluid web designs-

Fluid web design, also known as liquid layout, involves designing a website layout using percentages for widths other than fixed pixel values.

③ Adaptive web designs-

Adaptive web design involves creating multiple fixed layout designs targeted at specific device size or breakpoints.

In summary, responsive web design offers a flexible & adaptive approach to creating websites that can seamlessly adjust to various screen sizes & orientation, making it well suited for PWAs.

Qs → The lifecycle of service workers involves several distinct phases, including registration, installation & activation. Here's a breakdown.

① Registration :-

The first step in the lifecycle of a service worker is registration. Registration occurs in the main Javascript file of a web application using the navigator service worker register() method. During registration the web browser attempts to download & parse the service worker script specified in the registration call.

② Installation :-

Once a service worker is successfully registered it enters the installation phase. During installation, the browser installs the service worker script & catches any static assets specified in the service workers install event. The install event is triggered when the service worker script is first installed or when there are changes to the service worker file.

③ Activation :-

After the service worker script is successfully installed it enters the activation phase. During activation, the browser activates the new service worker & removes any previous versions of the service workers. The activate event is triggered when the service worker becomes active & can start controlling clients.

Q4 → Indexed DB is a powerful client side storage mechanism available in modern web browsers. It provides a way for web applications, including those using service workers, to store large amounts of structured data persistently.

① Persistent Storage :-

Indexed DB offers persistent storage, meaning the data stored in the database remains available even after the webpage is closed or the browser is restarted. This makes it suitable for catching data offline, a common requirement for service workers.