

PWA ASSIGNMENT 2

Q.1. A progressive web app is a type of web application that utilizes modern web applications to deliver an app-like experience to users. PWA's are designed to be responsive, fast and engaging and provide a seamless user experience across devices.

• Significance in Modern web Development :

1. Cross Platform compatibility : PWA's can run on any device with a web browser, regardless of the operating system.

2. Offline functionality : One of the key features of PWAs is their ability to work offline or with limited connectivity.

3. Free installation : Unlike traditional apps, PWA don't require installation from an app store. Users can simply access the PWA through web browser and add it to home screen.

• Key characteristics differentiating PWAs from traditional mobile apps.

1. Development Approach : PWAs are typically built using web technologies like HTML, CSS and JS while traditional apps are built using platform specific languages (eg. Swift, Kotlin)

2. Distribution: PWAs distributed via the web and can be easily shared via URL, whereas traditional apps are distributed through app stores and require approval from the respective platform's store.

3. Installation: PWA's are accessed through web browser and can be added to the home screen while traditional apps are downloaded and installed from app stores.

Overall, PWAs offer a compelling alternative to traditional mobile apps, providing a modern, cost-effective and accessible solution for delivering app-like experience on the web.

Q.2. Responsive web design is an approach to web design that aims to create web pages that automatically adjust and adapt their layout and content to fit various screen sizes and devices, ensuring a consistent user experience across desktops, tablets and smartphones.

Importance of PWA:

1. Accessibility: Responsive design makes PWAs accessible to a wider audience, as users can access the app from any device without encountering layout issues.



2. **Consistency**: By maintaining consistent design and layout across devices, responsive web design helps reinforce the branding.
3. **Cross Device compatibility**: Since PWA's aim to provide a seamless user experience across different devices responsive web design plays a crucial role in ensuring that PWA looks and functions well.

Responsive web design	Fluid web design	Adaptive web design
1. Uses flexible grids and layouts that adjust based on screen size.	utilizes fluid grids & layouts that smoothly expand or contract.	creates a multiple versions of the site optimized for specific devices.
2. Relies on CSS media queries to adapt styling based on screen size.	may use media queries but emphasizes fluidity of elements.	uses server side detection to serve diff layouts and versions.
3. One codebase that adapts to various devices.	Emphasizes % based layouts & fluid images.	Requires separate versions of the site for diff devices.
4. Offers consistent user experience across devices.	Provides continuous user experience.	Tailors user experience for specific devices.

Q.3. The lifecycle of service workers involves several phases below is a detailed explanation of each phase.

1. Registration

- The first step using a service worker is register it within the web application.
- Typically done by including a registration script in the main JS file of the application.
- The process is initiated using 'navigate serviceworker register' '()' method.

2. Installation

- Once service worker is registered, the browser begins the installation process.
- During installation the browser downloads the service worker file specified during registration.
- On successful installation the service worker moves on the activation phase.

3. Activation

- After service worker is successfully installed, it goes through activation phase.
- During activation, new version of service worker become active and takes control of certain events and  $url$  requests within the scope of registration.



Q.4.

Indexed DB is low level API for client side storage of significant amounts of structured data, including files and blobs. It is useful for web applications that need to store large amount of data locally such as PWA's.

Here's how Indexed DB can be used in a service worker for data storage:

1. Caching Assets: One of the primary use cases of Indexed DB in a service worker is caching assets such as HTML, CSS, JS files, images and other resources. This allows application to function offline.
2. Dynamic Data storage: Indexed DB can also be used to store dynamic data generated by web application.
3. Background synchronization: service workers can periodically sync data with a server in the background even when the application is not been used.
4. Performance optimization: By storing frequently accessed data in Indexed DB service workers can improve performance of applications.