

MPL Assignment - 2

DATE: 06/03/25

NAME : HITESH TANHANI

CLASS:- D15B

ROLL NO: 57

SUBJECT : MPL

ASSIGNMENT - 2

Define progressive Web App (PWA) and explain its significance in Modern web development discuss the key characteristics that differentiate PWAs from traditional mobile apps

A progressive Web App (PWA) is a type of web application that works like a mobile app but runs in a browser. It can be installed on a device, works offline and provides a fast and smooth user experience.

### Significance of PWA in Modern Web Development

1. Cross-Platform compatibility
2. Offline support
3. Fast performance
4. No App store required
5. Lower Development Cost.

### Key Differences Between PWA and Traditional mobile Apps:

Feature	PWA	Traditional Mobile App
Installation	Direct from browser	Download from App Store
Internet	Works offline	Usually requires internet
Performance	Fast with service workers	Faster but needs installation

updates	Automatic, no app store approval	Manual updates needed
Development cost	Lower (one codebase for all)	Higher (separate apps for each platform)

Q.2 Define responsive web design and explain its importance in the context of Progressive Web Apps. compare and contrast responsive fluid and adaptive web design approaches.

→ Responsive Web Design (RWD) is a technique that makes web pages adjust automatically to different screen size and devices. It ensures a good user experience on mobiles, tablets and desktops without needing separate versions of website.

### Importance of Responsive Design in PWAs

1. Better User Experience
2. Faster Load Time
3. SEO Benefits
4. Cost Effective

## Key Differences

- Responsive adapts dynamically to all screens
- Fluid resizes smoothly but may not be fully optimized
- Adaptive loads different layouts based on device type.

3 Describe the lifecycle of service workers including registration, installation and activation phases.

### → Lifecycle

A service worker is a script that runs in the background and helps a web app work offline, load faster, and send push notifications. Its lifecycle has 3 main phases ..

#### 1. Registration Phase

- The browser registers the service worker using Javascript.

Code Example:

```
if ('serviceWorker' in navigator) {  
    navigator.serviceWorker.register('/sw.js')  
        .then(() => console.log('Service Worker  
Registered'))  
        .catch(error => console.log('Registration  
Failed:', error))
```

This tells the browser to install and activate the Service Worker.

## 2. Installation Phase

- The Service Worker downloads necessary files (HTML, CSS, JS) and stores them in cache
- If successful, it moves to the activation phase

Code Example :

```
self.addEventListener('install', event => {
  event.waitUntil(
    caches.open('app-cache').then(cache => {
      return cache.addAll(['/index.htm',
        '/style.css']);
    })
  );
});
```

This ensures the app loads even without the internet

### 3. Activation Phase

- The Old Service Worker is replaced with new One
- Unused cache files the previous version are deleted.

code Example:

```

self.addEventListener('activate', event => {
  event.waitUntil(
    caches.keys().then(keys => {
      return Promise.all(keys.map(key => {
        if (key !== 'app-cache') {
          return caches.delete(key);
        }
      }));
    })
  );
});

```

- The Service Worker is now fully active and control network requests.

### Final Step: Fetch & Sync

Once activated, the Service Worker intercepts network requests, serves cached files, and syncs data when the internet is available.

The lifecycle makes PWAs faster, more reliable, and capable of working offline.

DATE: 4

4 Explain the use of Indexed DB in the Service Worker for Data Storage

Indexed DB is a client-side NoSQL database used in Service Workers for storing structured data, enabling offline access, caching and background synchronization in web application.

Why Use IndexedDB in Service Workers?

- Offline storage: Saves API responses and users data for offline use.
- Persistent storage: Retains data even after page reloads.
- Data synchronization: Updates stored data when internet is available.
- Asynchronous & Non Blocking:

IndexedDB runs in the background, prevent performance issues.