MAD & PWA Lab Journal

Experiment No.	Assignment-2
Assignment 2 Questions	1. Define Progressive Web App (PWA) and explain its significance in modern web development.Discuss the key characteristics that differentiate PWAs from traditional mobile apps 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. 3. Describe the lifecycle of Service Workers, including registration, installation, and activation phases. 4. Explain the use of IndexedDB in the Service Worker for data storage.
Roll No.	37
Name	Nihal Nagdev
Class	D15B
Subject	MAD & PWA Lab
Lab Outcome	LO4:Understand various PWA frameworks and their requirements LO5: Design and Develop a responsive User Interface by applying PWA Design techniques LO6:Develop and Analyze PWA Features and deploy it over app hosting solutions
Grade:	

Nihal Nagdev
DIS B
37

MPL Assignment 2

04/05

1) Define progressive web app and explain its significance in modern web development. Discuss the key ~~significance in mo~~ characterstics that differentiate PWA from traditional mobile app.

A progressive web app - is a web application that delivers a native app like experience using modern web technology. It works across devices, supports offline access, and can be instaled ~~on~~ without an app store.

Significance in Modern web development.

- Cross-Platform : Runs on any device with a web browser.
- Online ~~web~~ Support - Uses service workers for caching.
- Fast Performance. Loads quickly even on slow networks.
- Engagement. supports push notifications and installation.
- Lower costs: A single codebase reduces development effort.

15.

Feature	PWA	Traditional Mobile App
Installation	Via Browser	App store
offline Support	Via Caching	Via explicit design.
push notifications	Yes	Yes
Platform Dependency	works on all browsers	Platform specific.
updates	Automatic	Requires user updates.

2] Define responsive web design and explain its importance in the context of PWA. Compare responsive, fluid ~~and~~ and adaptive web design approaches.

Responsive web design ensures a website adapts

Sundaram
FOR EDUCATIONAL USE

dre dynamically to different ~~er~~ screen sizes using flexible layouts and media queries.

Importance in PWAs.

- Consistent user experience, Optimizes for all screen sizes.
- Better Accessibility: Enhances ~~usi~~ usability ~~acc~~ across devices.
- Single Codebase - Reduces Development effort.
- SEO benefits - Google Favors website ~~friendly~~ friendly design.

Comparison of web Design Approaches

Feature	Responsive	Fluid	Adaptive
definition	Adjusts layout dynamically	Uses % based width	uses fixed breakpoints.
flexibility	high	high	limited
Development Effort	Moderate	low	high.

Sundaram
FOR EDUCATIONAL USE

Best Use Case	PWAs, modern sites	simple layouts	Specific Device layouts.

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

A service worker is a background script that enhances PWAs by enabling offline caching, push notifications and background sync.

Lifetime Phases.

1] Registration
- Registered ~~wi~~ via navigator.service.worker.register().
- Browser downloads and installs the scripts.

2] Installation
- Triggers the install event
- Caches static assets for offline use

3] Activation
- Triggers the activate event.

Sundaram
FOR EDUCATIONAL USE

· Cleans up ~~old~~ old caches ~~and~~ and takes control of open pages.

4] Idle to fetch handling
· Intercepts network requests and serves cached data.

5] Update to termination.
· Deletes new versions and replaces old workers.
· The browser terminates inactive service.

~~of~~

4] Explain the use of Indexed DB in service worker for data storages.

Indexed DB is a browser based ~~My~~ NoSQL database for storing structured ~~data~~ data patiently.

Benefits in service workers.

· Offline storage: saves data when the user is offline.
· Efficient Data syncing: updates local data when online.
· Non Blocking operations: Uses asynchronous API to prevent delays.

Ex.

```
let db;
const req = indexed DB. open("PWA", 1);

req. onupgradeneeded = function (event){

  let db = event. target. result;
  db. create Object Store("articles", { keyPath:
    "id" });
};

function storeArticle(article) {

  let transaction = db. transaction ("articles",
                                    "read write");

  let store = transactions. objectstore("articles");
  store.put (article);
}
```