Mayur Jaiswal
D15B
24

03/03

MPL assignment 2

Q.1) Define progressive webapp (PWA) and explain its significance in modern web development. Discuss the key characteristics PWAs from traditional mobile apps

→ A progressive webapp (PWA) is a type of web application that works like a mobile app but runs in a browser 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

Difference in PWA and traditional apps

PWA	Traditional App
- Direct from browser	- Download from app store
- work offline with caching	- usually requires internet
- Fast with service worker	- Fast but need installation
- automatic updates	- manual updates
- lower development cost	- higher development cost

Q.2) Define responsive web design and explain its importance in the context of progressive web apps. Compare and construct responsive, fluid and adaptive web design approaches

→ Definition of responsive web design: 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 mobile on mobile, tablets and desktops without needing separate version of website

Importance of responsive design in PWAs

(1) better user experience
(2) faster load time
(3) SEO benefits
(4) cost effective

Comparison:

approach	how it works	pros	cons
responsive	flexible grids and CSS media queries	works on all devices	can be complex
fluid	% based width instead of fixed pixels	works well on different screen size	less control

Key difference:

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

Q3) Describe the lifecycle of services workers, including registration, installation and activation phases

→ lifecycle of services workers: A services 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 three main phases

(1) Registration phase: The browser register the service workers using JS

example: if ('service worker' in navigator) {
navigator.serviceWorker.register("/sw.js").
.then(() => console.log("Service Registered"))
.catch(error => console.log(error));
}

(2) Installation phase: (i) The service worker downloads necessary files (HTML, CSS) and stores them in cache.
(2) if successful, it moves to the activation phase

example: self.addEventListener("install", event => {
event.waitUntil(
caches.open('app-cache').then(cache => {
return cache.addAll(['/', '/index.html']);
});
});

(3) activation phase: The old service worker is replaced with new one
Cache files from the previous versions are deleted
final step: Fetch and clear.

Once activated, the services workers intercept network request, serve cache files and typed data.

Q4) Explain the use of indexed DB in the services workers for data storage.

→ Use of indexed DB in services workers for Data storage: IndexedDB is a browser database that stores large amount of data like Json object It helps PWA work offline by saving and retrieving data efficiency.
Why use IndexedDB in service workers?
(1) offline support, (2) efficient storage, (3) faster access

(4) persistent data.
how services workes use IndexedDB?

* Opening the database:

```
let db;
let request = IndexedDB.open('My-database', 1);
request.onsuccess = function (event) {
    db = event.target.result;
};
```

Creating a store and adding data

```
request.onupgradeneeded = function (event) {
    let db = event.target.result;
    let store = db.createObjectStore('users');
    store.add ({id: 1});
};
```

fetching data in service worker.

```
let transaction = db.transaction('users', 'readonly');
let store = transaction.ObjectStore("user");
let getUser = Store.get(1);
getUser.onSuccess = function () {
    console.log(getUser.result);
};
```