

## Experiment No: 07

**Aim:** To Converting your MERN website into a Progressive Web App (PWA) that users can "Add to Homescreen" just like a mobile app.

### Steps:

#### Step 1: Create and Add a `manifest.json` file

In your React app (inside the `public/` folder), create a file called `manifest.json` with this content:

```
{
  "name": "Your App Name",
  "short_name": "AppShort",
  "start_url": ".",
  "display": "standalone",
  "background_color": "#ffffff",
  "theme_color": "#000000",
  "description": "A sample PWA for experiment.",
  "icons": [
    {
      "src": "icons/icon-192x192.png",
      "sizes": "192x192",
      "type": "image/png"
    },
    {
      "src": "icons/icon-512x512.png",
      "sizes": "512x512",
      "type": "image/png"
    }
  ]
}
```

\*\*\*\*\*Make sure to place `icon-192x192.png` and `icon-512x512.png` inside `public/icons/`.

#### Step 2: Add Manifest Link in `public/index.html`

Add this inside the `<head>` tag of your `public/index.html`:

```
<link rel="manifest" href="./manifest.json" />
<meta name="theme-color" content="#000000" />
```

### Step 3: Add the Service Worker (Optional if not using CRA)

If you're using **Create React App (CRA)**, it's already PWA-ready with `serviceWorkerRegistration.js`.

If **not** using CRA, create a `serviceworker.js` in `public/`:

```
// public/serviceworker.js
const CACHE_NAME = 'pwa-cache';
const urlsToCache = ['/', '/index.html'];

self.addEventListener('install', event => {
  event.waitUntil(
    caches.open(CACHE_NAME)
      .then(cache => cache.addAll(urlsToCache))
  );
});

self.addEventListener('fetch', event => {
  event.respondWith(
    caches.match(event.request)
      .then(resp => resp || fetch(event.request))
  );
});
```

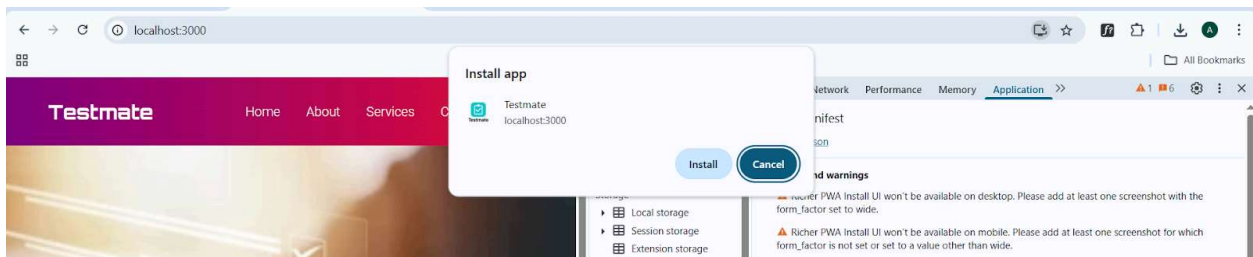
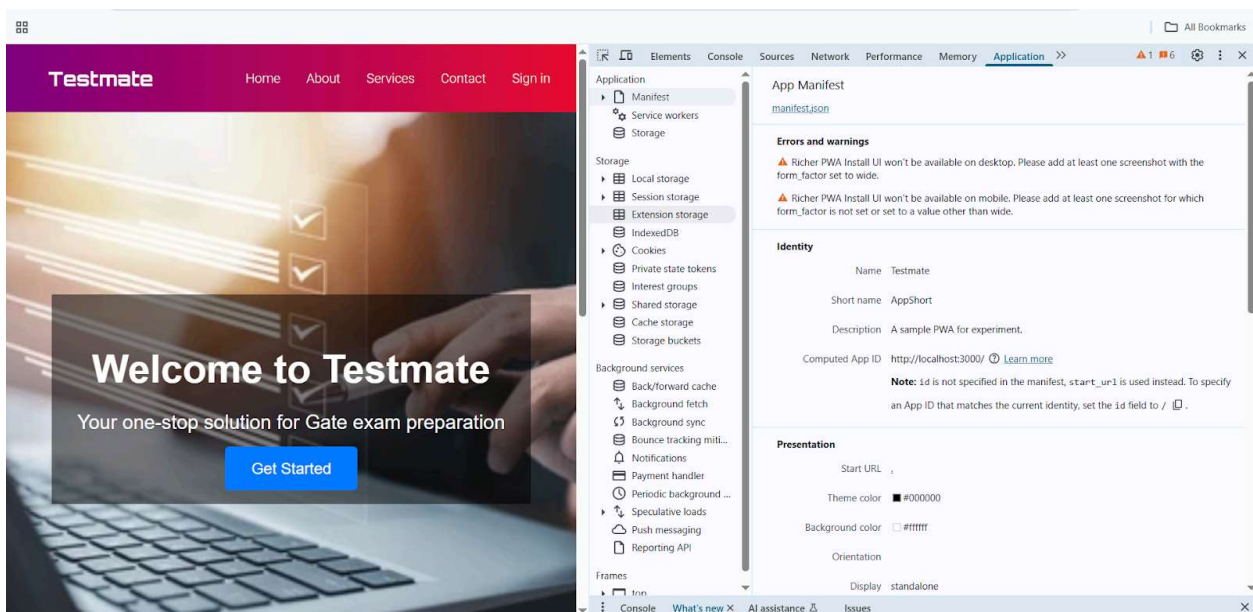
### Step 4: Register Service Worker in React

Inside your main JS file (e.g. `src/index.js`):

```
if ('serviceWorker' in navigator) {
  window.addEventListener('load', () => {
    navigator.serviceWorker.register('/serviceworker.js')
      .then(reg => console.log("Service Worker registered", reg))
      .catch(err => console.log("Service Worker registration failed",
err));
  });
}
```

### Step 5: Test PWA Feature

1. Start your React app locally (or deploy it on Netlify/Vercel).
2. Open in Chrome browser.
3. Open DevTools → Application tab → Manifest.
4. Check if:
  - Manifest is loaded.
  - Icons are valid.
  - Service Worker is registered.
5. You should see an option to “Install App” (from the 3-dot menu or in the URL bar).



in the URL bar).

