

- Aim :To write metadata for an eCommerce Progressive Web App (PWA) in a web app manifest file to enable the “Add to Home Screen” feature.
- Theory: A Web App Manifest is a JSON file that contains essential metadata about a Progressive Web App (PWA). This metadata allows browsers to recognize and display the web app appropriately when a user chooses to install it. For the "Add to Home Screen" feature to function, the manifest file must define key details, such as the app name, icons, and display preferences.
 - Key Components of a Web App Manifest:
 - name: The full name of the app, displayed when users add the app to their home screen. It should be clear, descriptive, and concise.
 - short_name: A shorter version of the app's name, used when there isn't enough space (e.g., on the home screen or app launcher).
 - description: A brief explanation of the app's purpose. It gives users an idea of what to expect when they add the app to their home screen.
 - start_url: Defines the entry point for the app. When the app is launched from the home screen, this URL is opened first.
 - display: Specifies how the app should appear when launched. Options include:
 - standalone: App behaves like a native app with no browser chrome.
 - fullscreen: App runs in full-screen mode.
 - minimal-ui: App provides a minimal UI with navigation controls.
 - browser: App behaves like a regular web page in a browser.
 - background_color: The background color that appears during the app's splash screen as it loads.
 - theme_color: Sets the color of the browser's UI elements, such as the address bar, to match the app's branding.
 - orientation: Specifies whether the app should open in portrait or landscape mode, improving the experience on mobile devices.
 - scope: Defines the navigation scope for the PWA. It limits the URLs that are considered within the app's domain.
 - icons: Specifies the various icon images used for the app on the home screen, app launcher, or other areas. Icons should be provided in multiple sizes for different devices and resolutions.
 - Importance for PWAs: The web app manifest is fundamental for delivering a native-like experience on mobile and desktop devices. It gives users a smooth, app-like feel while ensuring that the app is visually consistent across different platforms. For eCommerce PWAs, this is crucial to enhance user experience, especially in terms of engagement and accessibility.
- Code :


```
{
  "short_name": "FilmFiesta",
  "name": "Film Fiesta - Movie App",
```

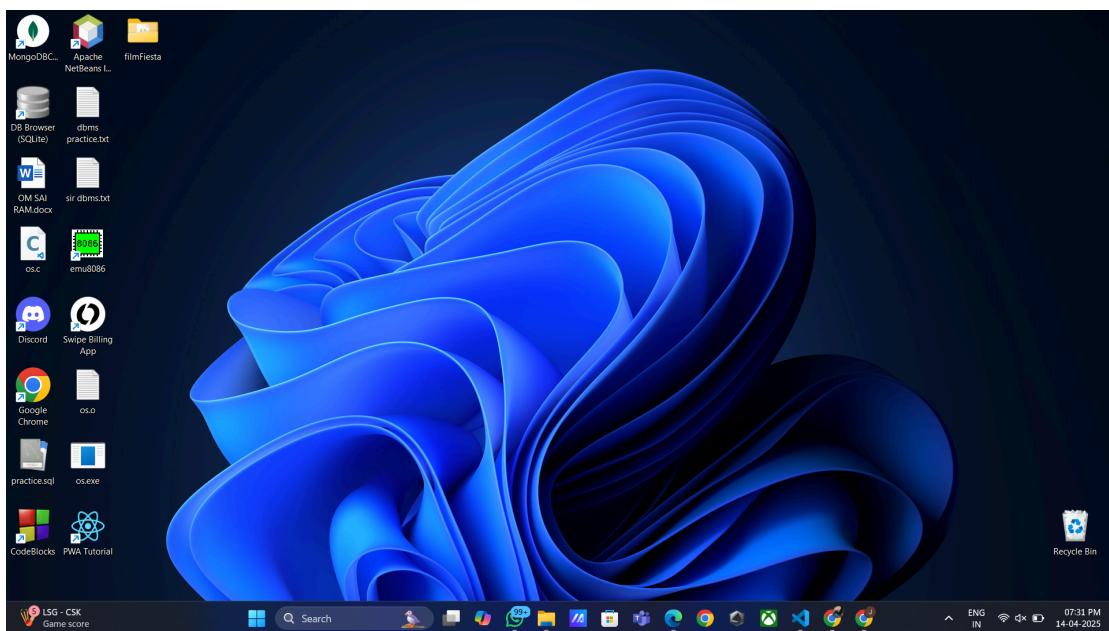
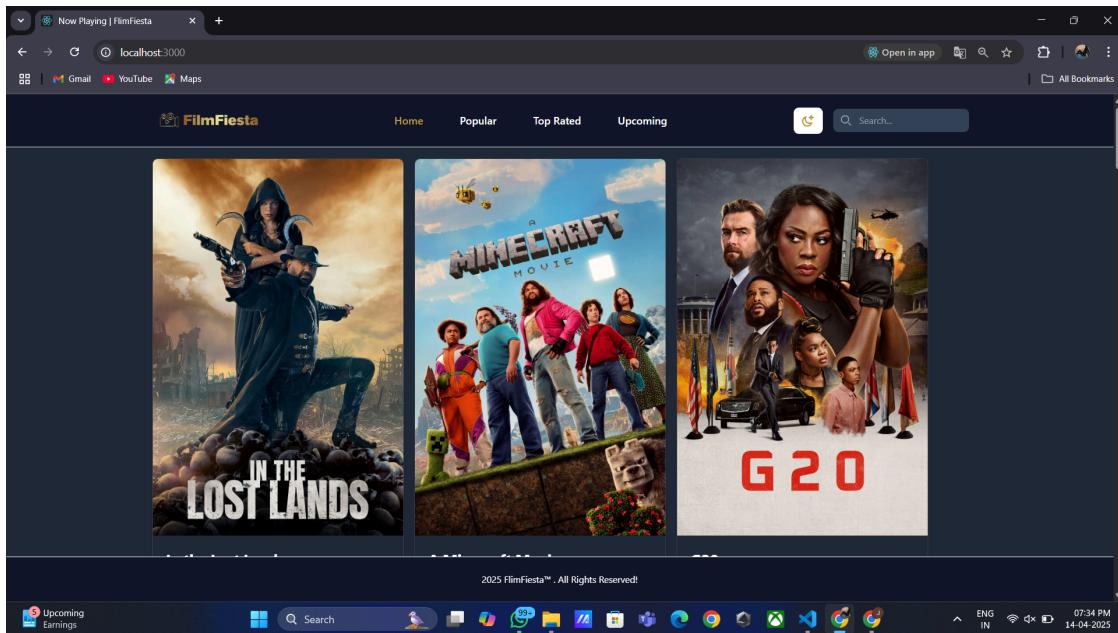
```

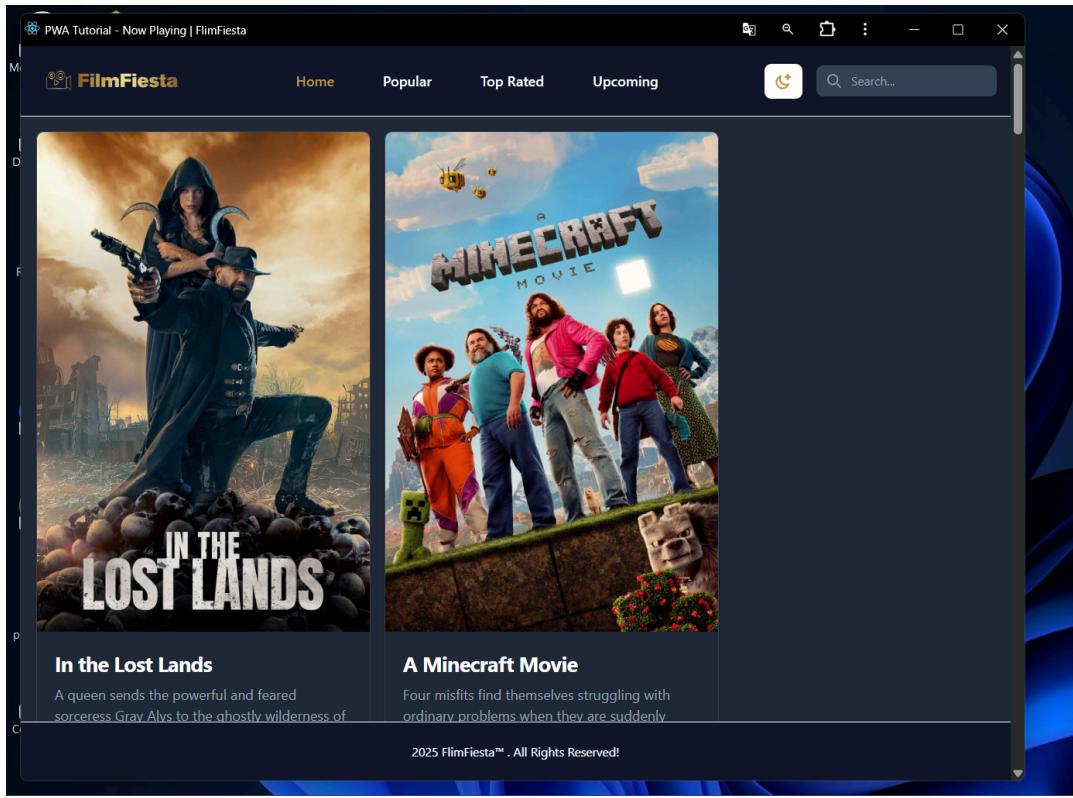
"icons": [
  {
    "src": "logo192.png",
    "type": "image/png",
    "sizes": "192x192"
  },
  {
    "src": "logo512.png",
    "type": "image/png",
    "sizes": "512x512"
  }
],
"start_url": ".",
"display": "standalone",
"theme_color": "#000000",
"background_color": "#ffffff"
}

```

- Output :

The screenshot shows the Service Worker Manager interface with the 'Manifest' tab selected. The left sidebar lists various service worker components: Application (Manifest, Service workers, Storage), Storage (Local storage, Session storage, Extension storage, IndexedDB, Cookies, Private state tokens, Interest groups, Shared storage, Cache storage, Storage buckets), and Background services (Back/forward cache, Background fetch, Background sync, Bounce tracking mitigation, Notifications, Payment handler, Periodic background tasks). The main panel is divided into sections: Identity (Name: Film Fiesta - Movie App, Short name: FilmFiesta, Description, Computed App ID: http://localhost:3000/), Presentation (Start URL: ., Theme color: #000000, Background color: #ffffff, Orientation: landscape, Display: standalone), and Protocol Handlers (Note: id is not specified in the manifest, start_url is used instead. To specify an App ID that matches the current identity, set the id field to /). A note at the bottom states: ⚠ Define protocol handlers in the manifest to register your app as a handler for custom protocols when your app is installed. Need help? Read URL protocol handler registration for PWAs.





- Conclusion : Writing the metadata in the Web App Manifest file is vital for enabling the “Add to Home Screen” feature in an eCommerce PWA. By ensuring that essential details like the app's name, icons, start URL, and display settings are correctly defined, users can enjoy a seamless experience when adding the app to their home screen. This integration is key for increasing user retention and engagement, as it allows eCommerce platforms to deliver an immersive, fast, and convenient browsing experience.