

**Aim:** To write meta data of your Ecommerce PWA in a Web app manifest file to enable “add to homescreen feature”.

### **Theory:**

A Progressive Web App (PWA) is a web application that combines the best features of both web and native apps. It offers a reliable, fast, and engaging user experience by allowing users to install the app directly on their device's home screen without visiting an app store.

### **Web App Manifest**

The Web App Manifest is a JSON file that contains metadata about the application, such as its name, icons, theme color, display mode, and start URL. This file enables browsers to recognize the app as installable, making it possible to add the app to the home screen.

### **Key Properties in the Manifest**

1. **name:** The full name of the application displayed when the app is launched.
2. **short\_name:** A shorter version of the app name used on the home screen.
3. **start\_url:** The entry point of the app when launched from the home screen.
4. **display:** Defines how the app appears (e.g., standalone makes it look like a native app).
5. **background\_color:** Sets the background color of the splash screen when the app is launched.
6. **theme\_color:** Defines the color of the browser's address bar.
7. **icons:** Specifies the icons used for the app on the home screen in different sizes and formats.
8. **scope:** Defines the range of URLs the app controls.

### **Service Worker and Caching**

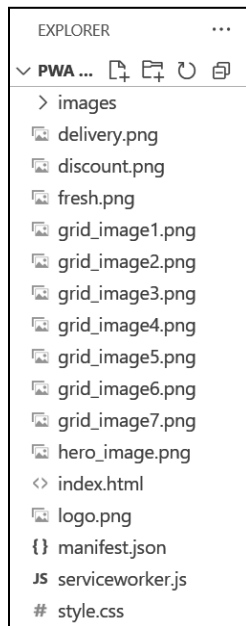
A Service Worker is a script that runs in the background and handles caching, allowing the app to work offline.

- During installation, the service worker caches essential files.
- When the app is fetched, it first checks the cache for faster loading.

### **Implementation in the Code**

In this experiment:

- We created a manifest.json file containing metadata like the app name, icons, colors, and display settings.
- The index.html file links to the manifest file, enabling the "Add to Homescreen" feature.
- We registered a Service Worker (serviceworker.js) to cache the necessary files, ensuring offline functionality.
- This makes the web app feel more like a native app, providing a smooth and reliable experience for users.

**Folder Structure:****Code:****index.html**

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"
  />
    <meta name="theme-color" content="#007BFF">
    <title>Restaurant Website</title>
    <link rel="stylesheet" href="style.css" />
    <link rel="manifest" href="manifest.json">
    <script
      src="https://kit.fontawesome.com/7a4b62b0a4.js"
      crossorigin="anonymous"
    ></script>
  </head>
  <body>
    <nav>
      <div class="navigation container">
        <div class="logo_container">
          
        </div>
        <div class="bar_icon">
          <i class="fas fa-bars"></i>
        </div>
      </div>
    </nav>
  </body>
</html>
```

```
</nav>
<!-- Hero Section -->
<div class="container">
  <div class="hero">
    <div class="hero_image">
      
    </div>
    <div class="hero_content">
      <div class="tag">50% Off on All Products</div>
      <h1>Enjoy Your Delicious Food</h1>
      <p>
        Lorem ipsum dolor sit amet, consectetur adipisicing elit.
        Voluptates, iste corporis tempore necessitatibus inventore ex?
      </p>
      <button class="explore_btn">Explore Now</button>
    </div>
  </div>
</div>
<section class="features">
  <div class="feature">
    
    <div class="feature_content">
      <h3>Discount Voucher</h3>
      Lorem ipsum dolor sit, amet consectetur adipisicing elit.
    </div>
  </div>
  <div class="feature">
    
    <div class="feature_content">
      <h3>Fresh Healthy Food</h3>
      Lorem ipsum dolor sit, amet consectetur adipisicing elit.
    </div>
  </div>
  <div class="feature">
    
    <div class="feature_content">
      <h3>Fast Home Delivery</h3>
      Lorem ipsum dolor sit, amet consectetur adipisicing elit.
    </div>
  </div>
</section>
<div class="divider"></div>
<div class="menu">
  <div class="tag">Our Menu</div>
  <h2>Explore Our Menu</h2>
```

```
<div class="grid">
  <div class="item1">
    
  </div>
  <div class="item2">
    
  </div>
  <div class="item3">
    
  </div>
  <div class="item4">
    
  </div>
  <div class="item5">
    
  </div>
  <div class="item6">
    
  </div>
</div>
```

```
</div>
<div class="item7">
  
</div>
</div>
</div>
</div>
<div>
  <div class="footer_container container">
    <div class="footer_logo">
      
    </div>
    <div class="link_lists">
      <h3>Main Links</h3>
      <ul>
        <li>Order Tracking</li>
        <li>New Order</li>
        <li>Contact Us</li>
        <li>News & Blogs</li>
      </ul>
    </div>
    <div class="link_lists">
      <h3>Support</h3>
      <ul>
        <li>About Us</li>
        <li>Privacy Policy</li>
        <li>Terms & Conditons</li>
      </ul>
    </div>
    <div class="news_letter">
      <h3>Support</h3>
      <input type="email" placeholder="Enter your Email..." />
      <h3>Follow Us</h3>
      <div class="icon_container">
        <div class="icon">
          <i class="fa fa-facebook"></i>
        </div>
        <div class="icon">
          <i class="fa fa-twitter" aria-hidden="true"></i>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

    <div class="icon">
      <i class="fa fa-instagram" aria-hidden="true"></i>
    </div>
    <div class="icon">
      <i class="fa fa-youtube" aria-hidden="true"></i>
    </div>
  </div>
</div>
</footer>
<script>
  if ('serviceWorker' in navigator) {
    navigator.serviceWorker.register('serviceworker.js')
      .then(() => console.log("Service Worker Registered"))
      .catch(error => console.log("Service Worker Registration
Failed", error));
  }
</script>
</body>
</html>

```

## manifest.json

```
{
  "name": "Restaurant Website",
  "short_name": "cooking",
  "start_url": "index.html",
  "display": "standalone",
  "background_color": "#5900b3",
  "theme_color": "black",
  "scope": ".",
  "description": "Cooking, also known as cookery or professionally as the
culinary arts, is the art, science and craft of using heat to make food
more palatable, digestible, nutritious, or safe.",
  "icons": [
    {
      "src": "images/app.png",
      "sizes": "192x192",
      "type": "image/png"
    },
    {
      "src": "images/big.png",
      "sizes": "512x512",
      "type": "image/png"
    }
  ]
}
```

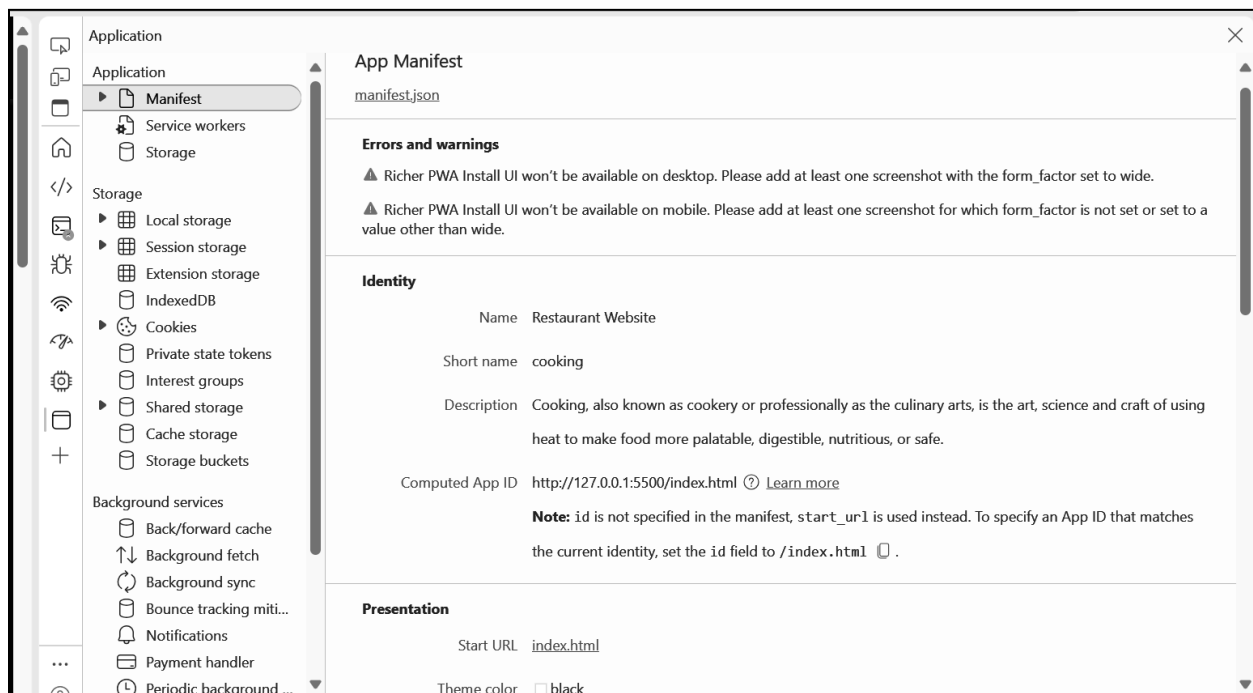
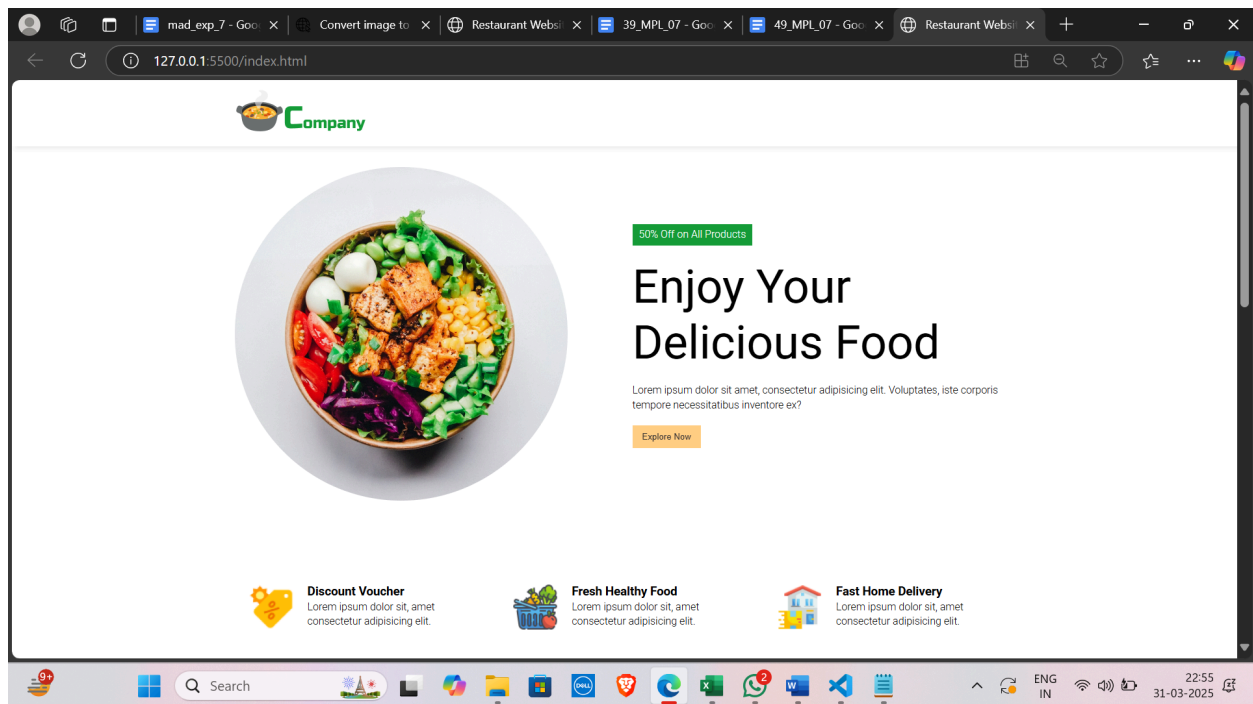
```
]
}
```

### serviceworker.js

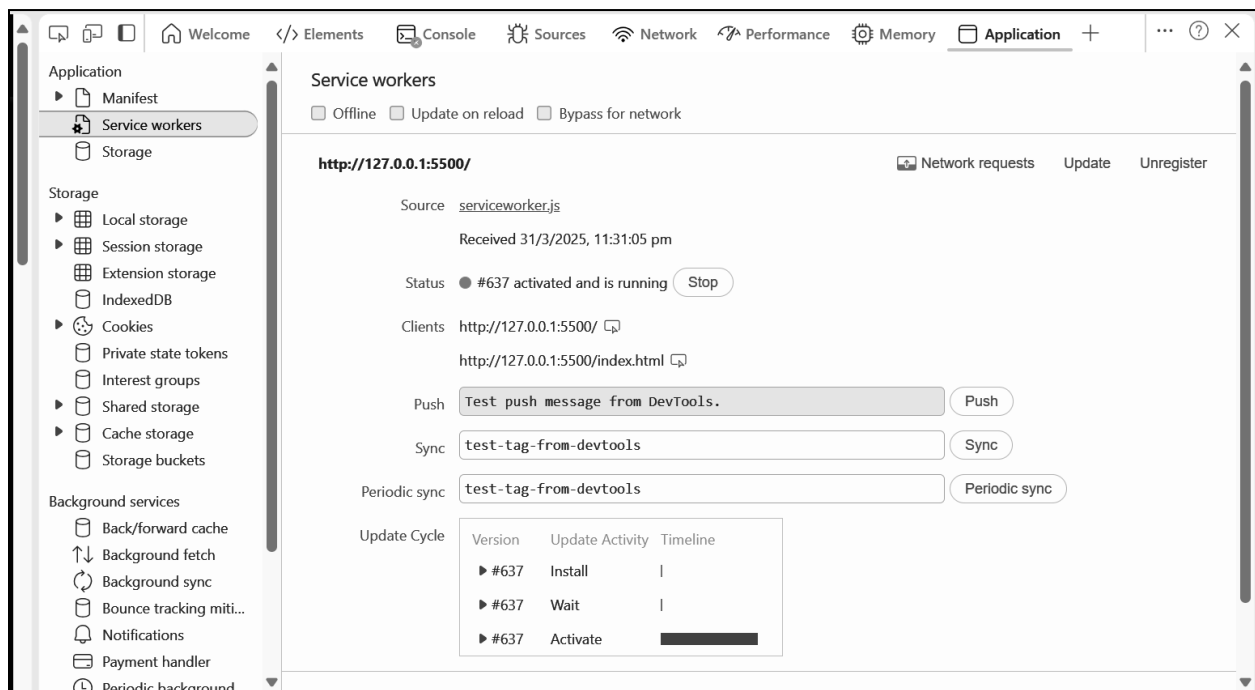
```
const CACHE_NAME = "cooking-cache-v1";
const FILES_TO_CACHE = [
  "index.html",
  "style.css",
  "manifest.json",
  "images/app.png",
  "images/big.png"
];

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

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

**Screenshot:**





New tab Ctrl+T
 New window Ctrl+N
 New InPrivate window Ctrl+Shift+N

Zoom — 67% + ↗

Favorites Ctrl+Shift+O
 Collections Ctrl+Shift+Y
 History Ctrl+H
 Shopping
 Downloads Ctrl+J
 Apps
 Extensions
 Browser essentials

Delete browsing data Ctrl+Shift+Delete
 Print Ctrl+P
 Split screen
 Screenshot Ctrl+Shift+S
 Find on page Ctrl+F

### Install Restaurant Website app

Publisher: 127.0.0.1:5500

Use this site often? Install the app which:

- Opens in a focused window
- Has quick access options like pin to taskbar
- Syncs across multiple devices

Install

Not now

Restaurant Website

## Enjoy Your Delicious Food

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Voluptates, iste corporis tempore necessitatibus inventore ex?

Explore Now

**Discount Voucher**  
 Lorem ipsum dolor sit amet, consectetur adipiscing elit.

**Fresh Healthy Food**  
 Lorem ipsum dolor sit amet, consectetur adipiscing elit.

**Fast Home Delivery**  
 Lorem ipsum dolor sit amet, consectetur adipiscing elit.

23:10

31-03-2025

## Conclusion

In this experiment, we successfully implemented the "Add to Homescreen" feature by creating a web app manifest file and registering a service worker for caching. During the process, we faced minor issues with the service worker registration due to incorrect file paths, which we resolved by properly specifying the cache files and verifying the manifest link in the HTML.