Assignment No.: 7  
Title: Add to Home Screen Feature using Progressive Web App (PWA)  
Name: Harshit Raheja  
Class: D15B  
Roll Number: 45

Aim: To implement “Add to Home Screen” feature in a web application using PWA concepts like manifest file and service worker.

Theory: A Progressive Web App (PWA) enables a web application to be installable on a user’s device similar to native apps. This is achieved by using a manifest file to describe app metadata and a service worker to enable caching and offline capabilities. When correctly configured, browsers prompt users to “Add to Home Screen,” giving a native-app-like experience.

Code:  
// index.html  
<!DOCTYPE html>

<html>

<head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <title>Phone Validator App</title>

    <meta name="apple-mobile-web-app-status-bar" content="#2c3e50">

    <meta name="theme-color" content="#2c3e50">

    <link rel="manifest" href="manifest.json">

    <link rel="icon" href="favicon.ico" type="image/x-icon">

</head>

<body>

    <h1>Phone Number Validator</h1>

    <p>Enter phone number with country code (e.g., +14152007986)</p>

    <input type="text" id="phoneInput" placeholder="Enter phone number" />

    <br />

    <button onclick="validatePhone()">Validate</button>

    <div id="result"></div>

    <script>

        function httpGetAsync(url, callback) {

            const xmlHttp = new XMLHttpRequest();

            xmlHttp.onreadystatechange = function () {

                if (xmlHttp.readyState === 4 && xmlHttp.status === 200)

                    callback(xmlHttp.responseText);

            }

            xmlHttp.open("GET", url, true);

            xmlHttp.send(null);

        }

        function validatePhone() {

            const phone = document.getElementById("phoneInput").value.trim();

            if (!phone) {

                document.getElementById("result").textContent = "Please enter a phone number.";

                return;

            }

            const url = `https://phonevalidation.abstractapi.com/v1/?api\_key=5be9903a0b564a78a7f26f4886b0a005&phone=${encodeURIComponent(phone)}`;

            httpGetAsync(url, function (response) {

                const data = JSON.parse(response);

                if (data.valid) {

                    document.getElementById("result").textContent = `✅ Valid Number (${data.country.name}) - Carrier: ${data.carrier}`;

                } else {

                    document.getElementById("result").textContent = "❌ Invalid phone number.";

                }

            });

        }

        window.addEventListener('load', () => {

            if ('serviceWorker' in navigator) {

                navigator.serviceWorker.register('serviceworker.js').catch(err => console.error('SW registration failed:', err));

            }

        });

    </script>

</body>

</html>  
//manifest.json  
{

    "name": "Phone Validator App",

    "short\_name": "PhoneCheck",

    "start\_url": "index.html",

    "display": "standalone",

    "background\_color": "#f4f6f8",

    "theme\_color": "#2c3e50",

    "scope": ".",

    "description": "Check if a phone number is valid using Abstract API.",

    "icons": [

        {

            "src": "images/phone-icon-192.png",

            "sizes": "192x192",

            "type": "image/png"

        },

        {

            "src": "images/phone-icon-512.png",

            "sizes": "512x512",

            "type": "image/png"

        }

    ],

    "screenshots": [

        {

            "src": "images/screenshot-desktop.png",

            "sizes": "1920x878",

            "type": "image/png",

            "form\_factor": "wide"

        },

        {

            "src": "images/screenshot-mobile.png",

            "sizes": "431x768",

            "type": "image/png",

            "form\_factor": "narrow"

        }

    ]

}

// serviceworker.js  
var staticCacheName = "phone-validator-pwa-v1";

self.addEventListener("install", function (e) {

    e.waitUntil(

        caches.open(staticCacheName).then(function (cache) {

            return cache.addAll([

                "/",

                "/index.html",

                "/manifest.json",

                "/serviceworker.js",

                "/images/phone-icon-192.png",

                "/images/phone-icon-512.png"

            ]);

        })

    );

});

self.addEventListener("fetch", function (event) {

    event.respondWith(

        caches.match(event.request).then(function (response) {

            return response || fetch(event.request);

        })

    );

});

Screenshots:

A screenshot of a computer

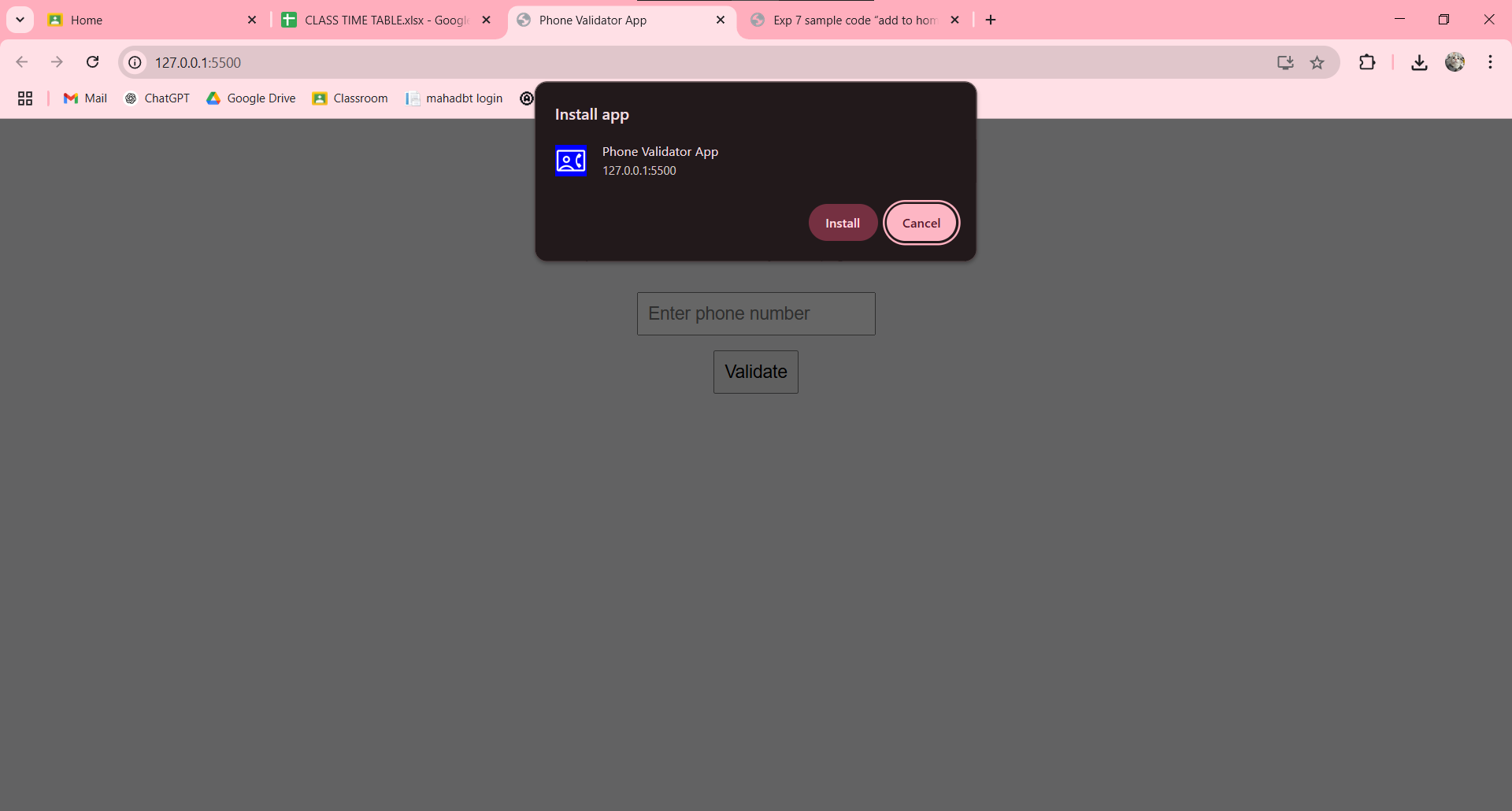
AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.



A screenshot of a computer

AI-generated content may be incorrect.

Conclusion: The PWA was successfully configured to support “Add to Home Screen” functionality. Manifest, service worker, and necessary meta tags were implemented to give the web app an installable, native-like experience.