**Vivekanand Education Society’s** 

**Institute of Technology**

**(Affiliated to University of Mumbai, Approved by AICTE & Recognized by Govt. of Maharashtra)**

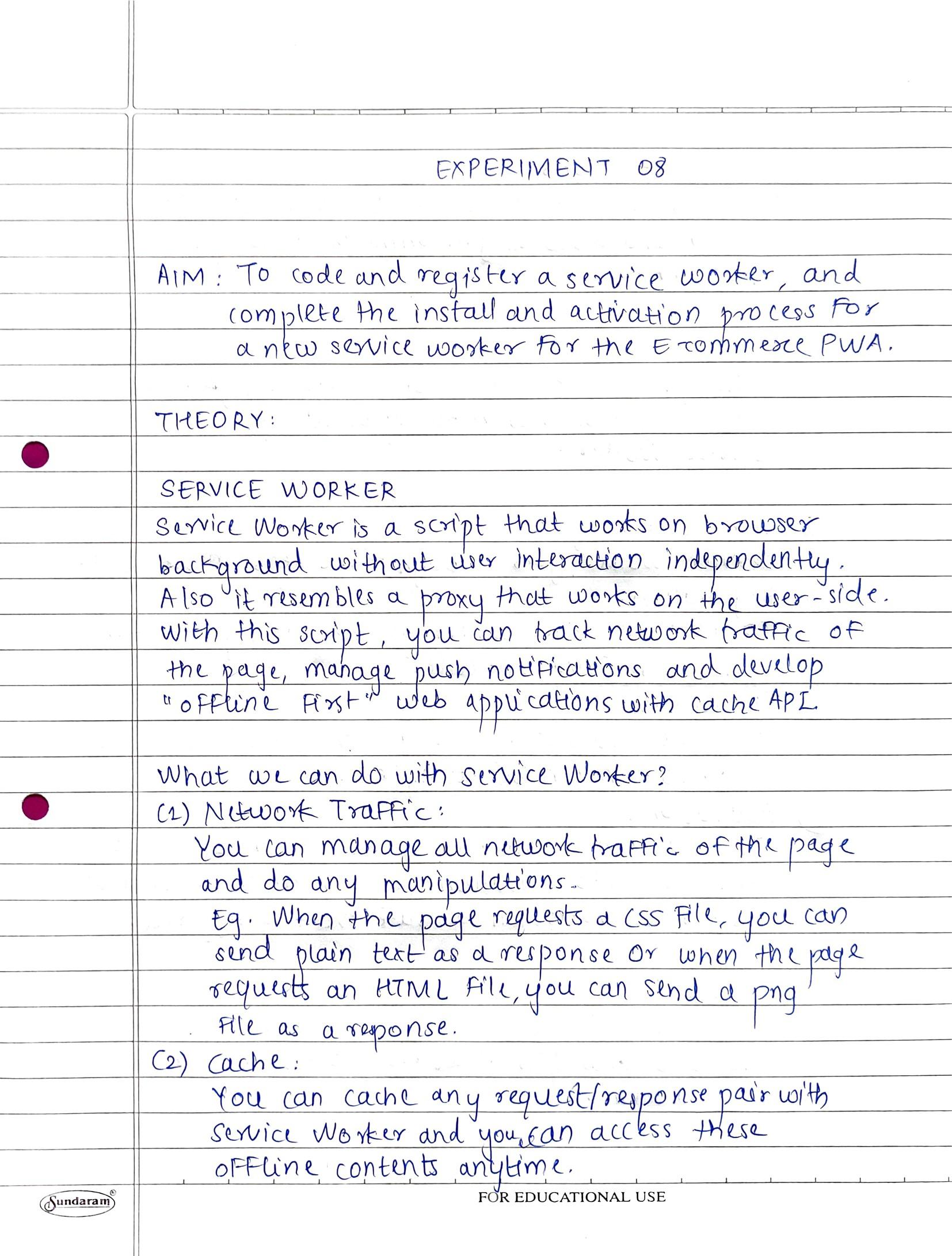
**Department of Information Technology**

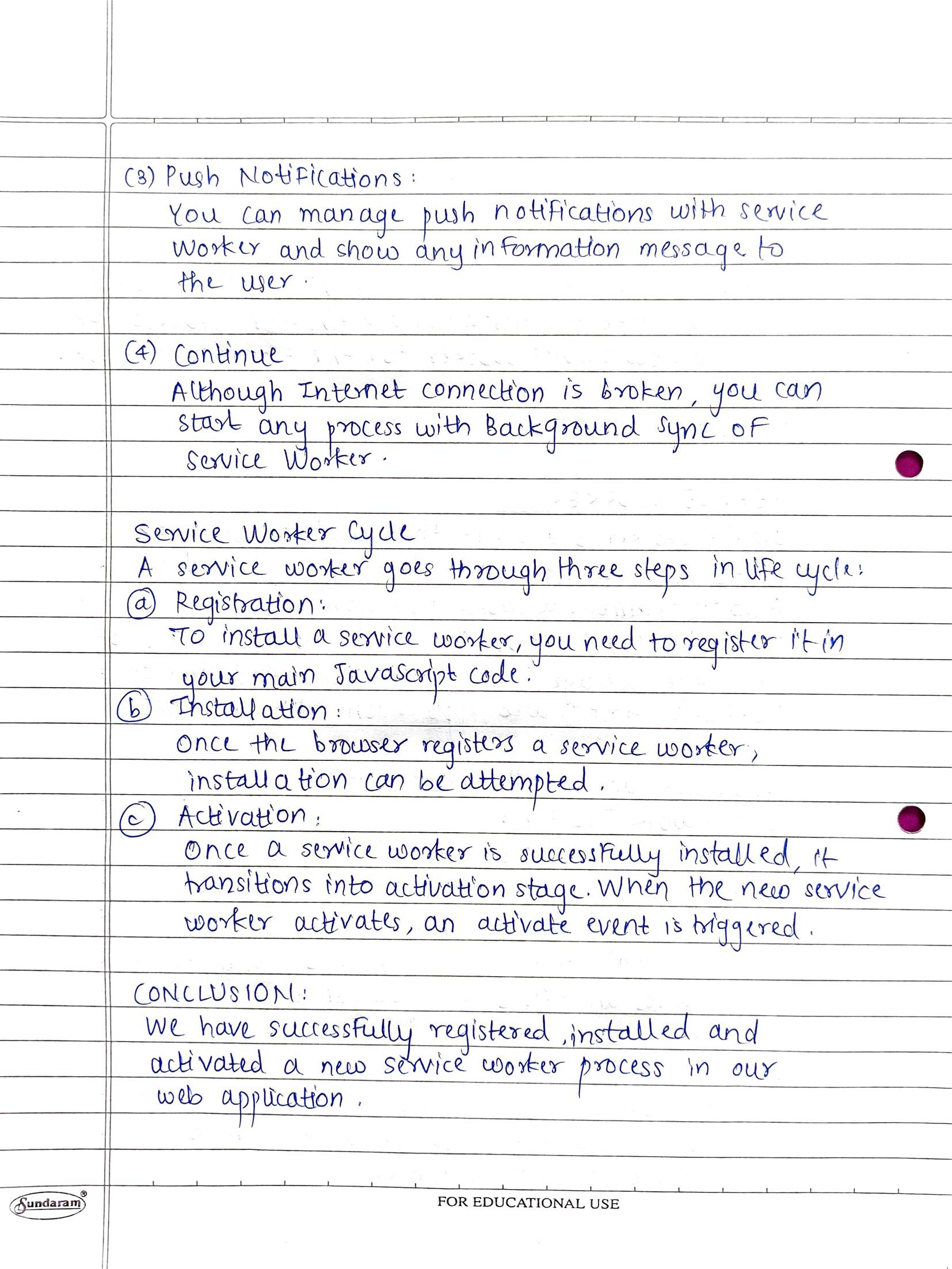
A.Y. 22-23

MAD & PWA Lab

Journal

| Experiment No. | 08 |
| --- | --- |
| Experiment  Title. | To code and register a service worker, and complete the install and activation process for a new service worker for the E-commerce PWA |
| Roll No. | 53 |
| Name | SAKSHI SANTOSH PATIL |
| Class | D15B |
| Subject | MAD & PWA Lab |
| Lab Outcome | LO5: Design and Develop a responsive User Interface by applying PWA Design techniques |
| Grade: |  |

****

****

**CODE:**

**Layout.html**

<script>

if ("serviceWorker" in navigator) {

navigator.serviceWorker

.register("{{url\_for('static', filename='js/serviceWorker.js')}}")

.then(function (registration) {

console.log(

"Service Worker registered successfully with scope: ",

registration.scope

);

})

.catch(function (error) {

console.log(

"Service Worker registration failed with error: ",

error

);

});

}

</script>

**Serviceworker.js**

self.addEventListener('install', function(event) {

console.log('Service Worker installing.');

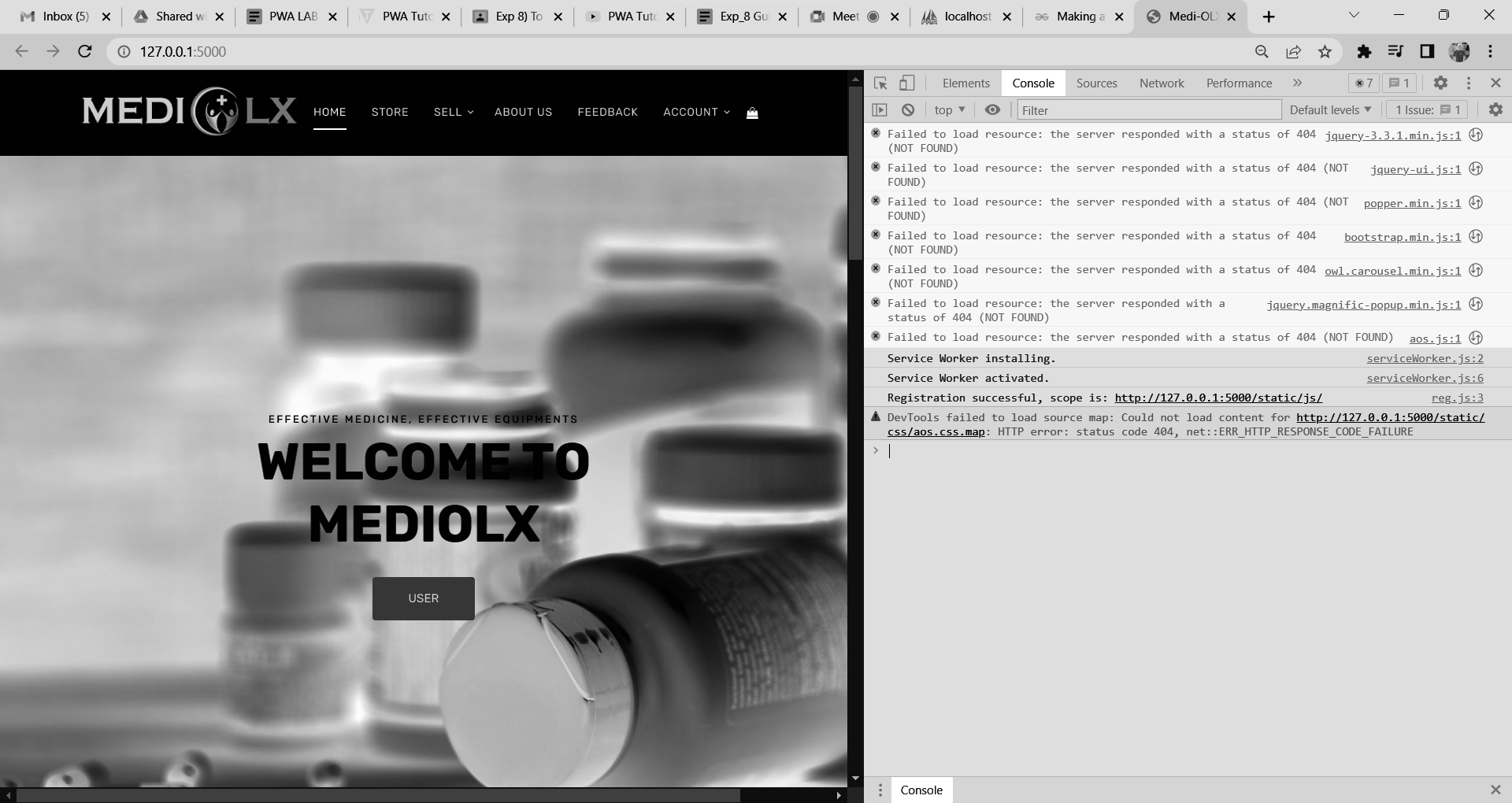
});

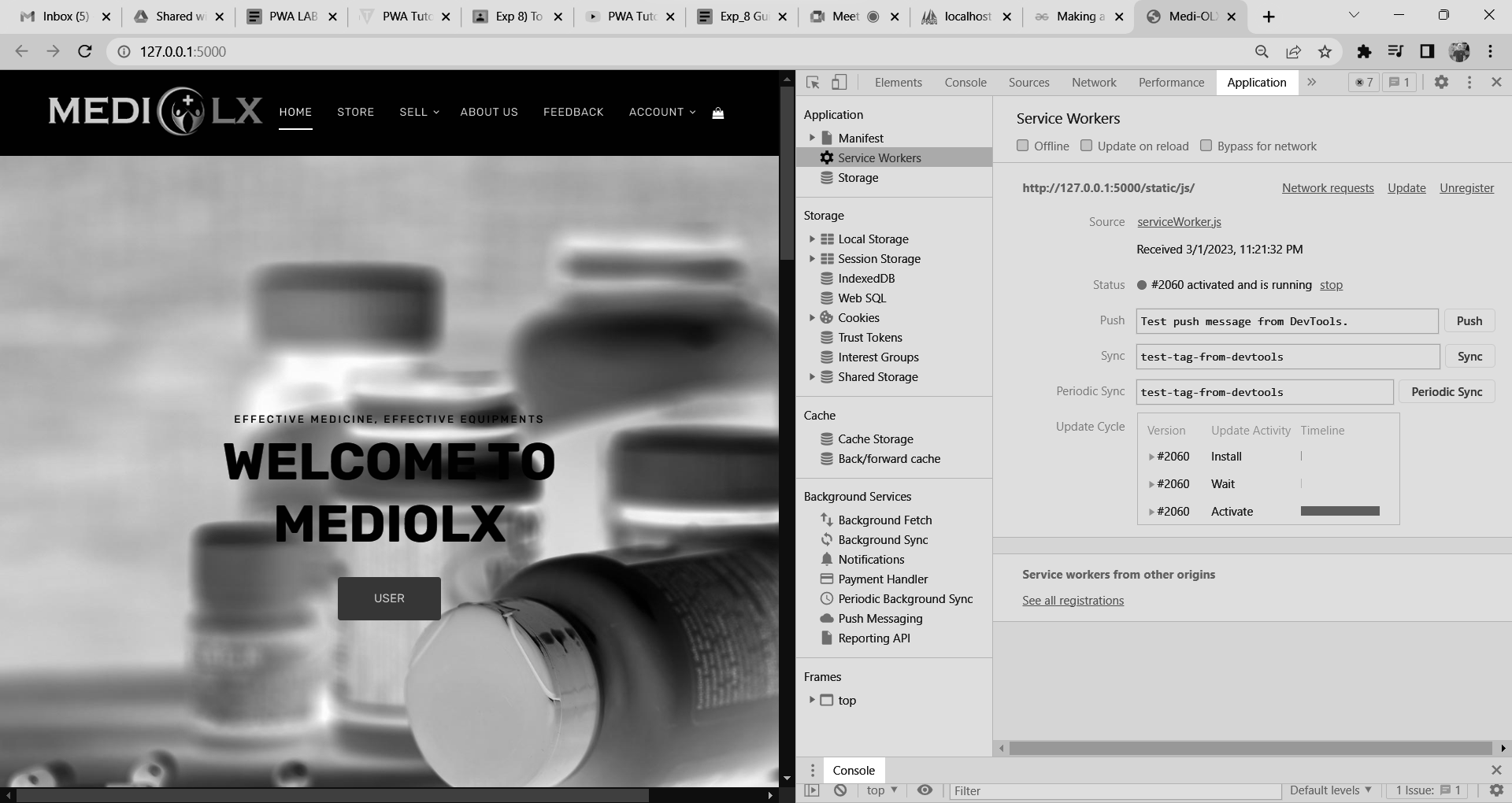
self.addEventListener('activate', function(event) {

console.log('Service Worker activated.');

});

**OUTPUT:**

****

****

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

event.waitUntil(preLoad());

});

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

event.respondWith(

checkResponse(event.request).catch(function () {

console.log("Fetch from cache successful!");

return returnFromCache(event.request);

})

);

console.log("Fetch successful!");

event.waitUntil(addToCache(event.request));

});

self.addEventListener("sync", (event) => {

if (event.tag === "syncMessage") {

console.log("Sync successful!");

}

});

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

if (event && event.data) {

var data = event.data.json();

if (data.method == "pushMessage") {

console.log("Push notification sent");

event.waitUntil(

self.registration.showNotification("PWA Tutorial", {

body: data.message,

})

);

}

}

});

var filesToCache = [

'/',

'/index.html',

];

var preLoad = function () {

return caches.open("offline").then(function (cache) {

// caching index and important routes

return cache.addAll(filesToCache);

});

};

var checkResponse = function (request) {

return new Promise(function (fulfill, reject) {

fetch(request).then(function (response) {

if (response.status !== 404) {

fulfill(response);

} else {

reject();

}

}, reject);

});

};

var addToCache = function (request) {

return caches.open("offline").then(function (cache) {

return fetch(request).then(function (response) {

return cache.put(request, response);

});

});

};

var returnFromCache = function (request) {

return caches.open("offline").then(function (cache) {

return cache.match(request).then(function (matching) {

if (!matching || matching.status == 404) {

return cache.match("offline.html");

} else {

return matching;

}

});

});

};