



รายงาน Assignment 2

Progressive Web Application

รายวิชา 240-420 Web Engineering and Applications

จัดทำโดย

นายจาศี อินทปัญญา 5910110044

นายณัฏฐ์สันต์ จันทระ 5910110095

นายปณณ เรืองจรรูวัฒนา 5910110419

เสนอ

อาจารย์ สุนทร วิฑูรพจน์

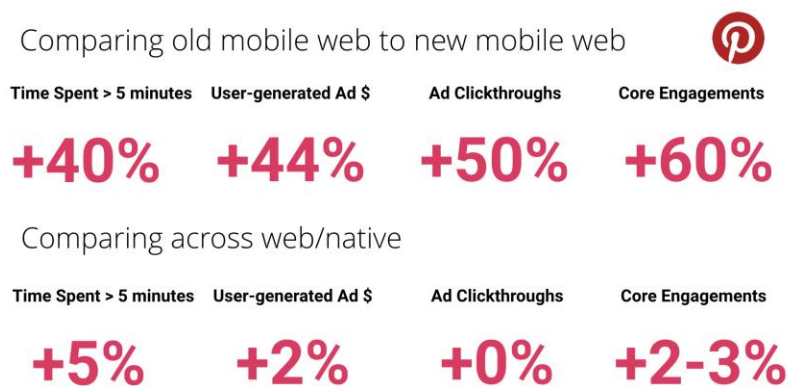
มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่

ผลการศึกษา Progressive Web Application

Progressive Web Application หรือ PWA เป็นเทคนิคหรือวิธีการที่จะทำให้ Web Application ที่พัฒนาสามารถมี UX หรือความสะดวกสบายในการใช้งานเพิ่มมากขึ้นได้ โดยมีเป้าหมายหลัก ๆ 3 ข้อ ได้แก่

1. Reliable หรือความน่าเชื่อถือ โดย PWA จะมีหน่วยการทำงานที่เรียกว่า Service worker ที่เขียนด้วยภาษา JavaScript ทำหน้าที่จัดการและควบคุมข้อมูลที่ร้องขอมา และเก็บข้อมูลมาไว้ในแคช เพื่อลดปัญหาการเชื่อมต่อเน็ตเวิร์คที่ไม่เสถียร ซึ่งแม้ไม่มีอินเทอร์เน็ตแต่สามารถเข้าหน้าเว็บที่เคยร้องขอไว้ได้
2. Fast หรือความรวดเร็ว ซึ่งตัว Service worker เก็บข้อมูลที่จำเป็นแล้ว การดึงข้อมูลเพิ่มเติมจึงทำได้เร็วมากยิ่งขึ้น
3. Engaging หรือความน่าดึงดูด ด้วย PWA สามารถถึงหน้าเว็บเข้ามาอยู่ในหน้าจอเมนูของโทรศัพท์ที่ได้โดยไม่ต้องลง Native Application ทำให้สามารถใช้งานได้เต็มหน้าจอของโทรศัพท์ อีกทั้งยังสามารถมีการทำงานการแจ้งเตือน พร้อมทั้งตกแต่งหน้าตาเพิ่มเติมของ Application ได้อีกด้วย ทำให้เพิ่มความสะดวกสบาย ความน่าดึงดูดใจได้มากขึ้น

โดย Pinterest สามารถทำกำไรให้กับบริษัทด้วยการใช้ PWA กับ Web Application ทำให้ Web Application มีความรวดเร็วยิ่งขึ้น โหลดรูปภาพที่เป็นเนื้อหาหลักของ Pinterest ได้มีประสิทธิภาพมากขึ้น ส่งผลให้มียอดการใช้งานที่เพิ่มมากขึ้น



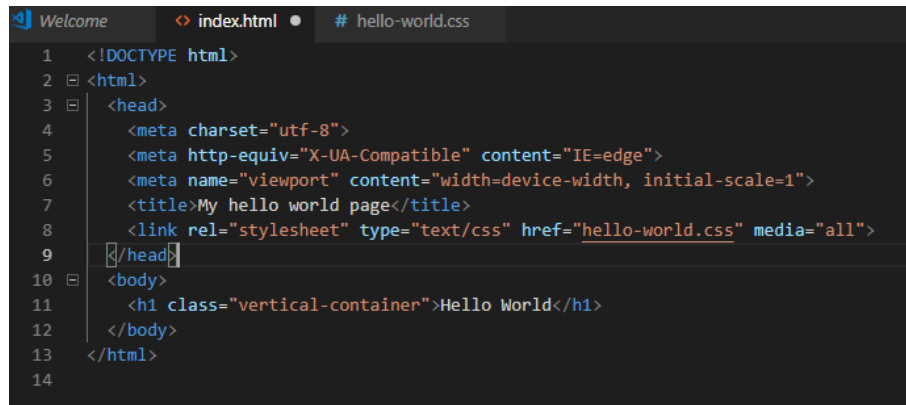
รูปที่ 1 ผลกำไรที่เพิ่มขึ้นจากผลการประมวลผลที่ดีขึ้นของ Pinterest

ผลการทดลอง Progressive Web Application

การสร้างเว็บพื้นฐาน และการ hosting ผ่าน firebase service

1. สร้างโครงสร้างเว็บพื้นฐาน

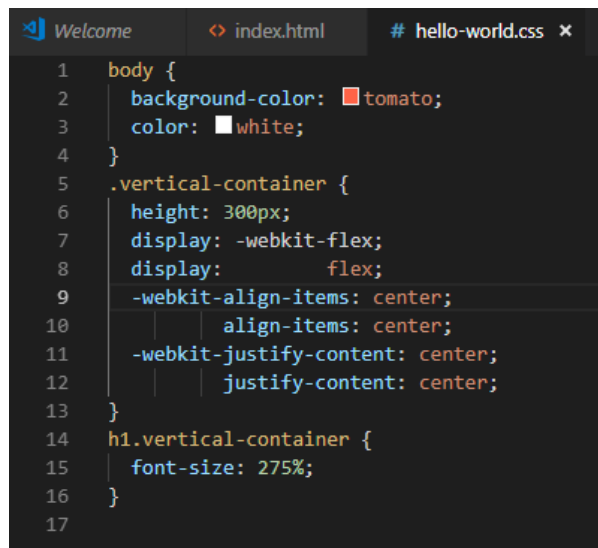
Index.html



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <title>My hello world page</title>
8   <link rel="stylesheet" type="text/css" href="hello-world.css" media="all">
9 </head>
10 <body>
11   <h1 class="vertical-container">Hello World</h1>
12 </body>
13 </html>
14
```

รูปที่ 2 หน้าเว็บ index.html เริ่มต้น

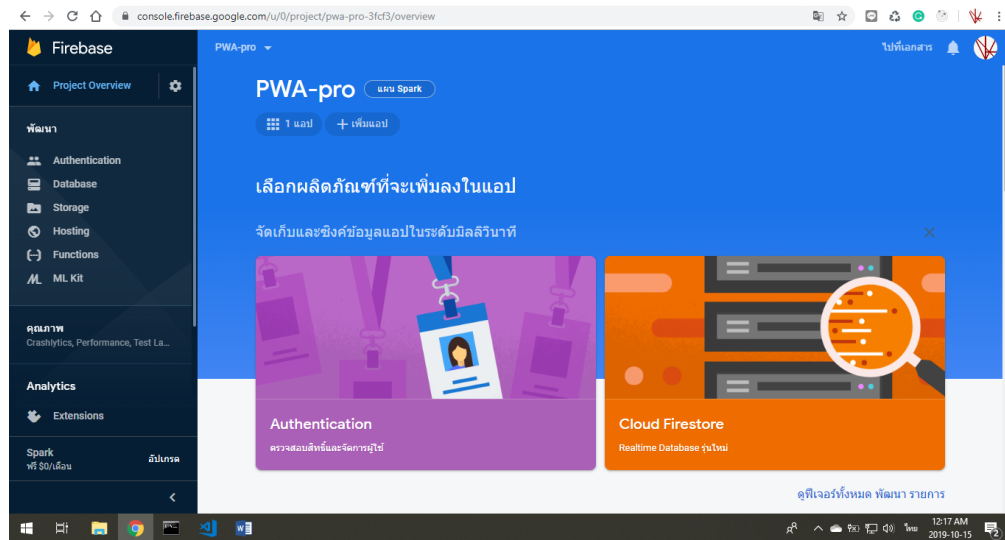
Hello-world.css



```
1 body {
2   background-color: tomato;
3   color: white;
4 }
5 .vertical-container {
6   height: 300px;
7   display: -webkit-flex;
8   display: flex;
9   -webkit-align-items: center;
10   align-items: center;
11   -webkit-justify-content: center;
12   justify-content: center;
13 }
14 h1.vertical-container {
15   font-size: 275%;
16 }
17
```

รูปที่ 3 css ของหน้าเว็บ index.html

- จากนั้นใช้บริการ hosting จาก google firebase hosting service โดยไปที่เว็บไซต์หลักของ firebase และสร้างโปรเจกต์



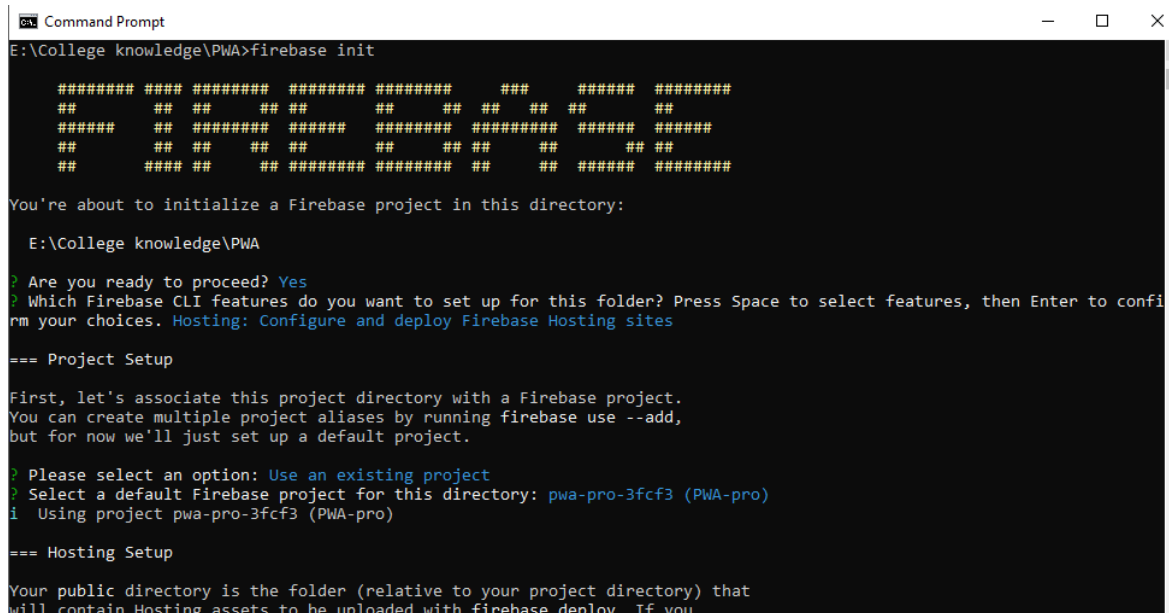
รูปที่ 4 หน้าเว็บของบริการ firebase

- นำ code configuration ของ firebase มาใช้ในไฟล์ html ดังกล่าวเพื่อให้ในการ hosting ซึ่งมีวิธีการคือสร้าง tag script ขึ้นมาแล้วเขียนโปรแกรมภาษา JavaScript ที่มี configuration code ของ firebase อยู่

```
8      <link rel="stylesheet" type="text/css" href="hello-world.css" media="all">
9    </head>
10    <script src="https://www.gstatic.com/firebasejs/7.2.0/firebase-app.js"></script>
11    <script>
12      var firebaseConfig = {
13        apiKey: "----",
14        authDomain: "pwa-pro-3fcf3.firebaseio.com",
15        databaseURL: "https://pwa-pro-3fcf3.firebaseio.com",
16        projectId: "pwa-pro-3fcf3",
17        storageBucket: "pwa-pro-3fcf3.appspot.com",
18        messagingSenderId: "574184343847",
19        appId: "1:574184343847:web:cfebd20ce5aa6ec7540102"
20      };
21      // Initialize Firebase
22      firebase.initializeApp(firebaseConfig);
23    </script>
24
25    <body>
26      <h1 class="vertical-container">Hello World</h1>
27    </body>
28  </html>
29
```

รูปที่ 5 เพิ่ม config code ลงใน script ของ index.html

4. ทำ directory ของเราให้กลายเป็น firebase project ด้วยคำสั่ง firebase init โดยเลือกตัวเลือกบริการ hosting



```
E:\College knowledge\PWA>firebase init

##### 
## 
##### 
## 
##### 
## 
##### 
## 
##### 
## 
##### 

You're about to initialize a Firebase project in this directory:

E:\College knowledge\PWA

> Are you ready to proceed? Yes
> Which Firebase CLI features do you want to set up for this folder? Press Space to select features, then Enter to confirm your choices. Hosting: Configure and deploy Firebase Hosting sites

=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

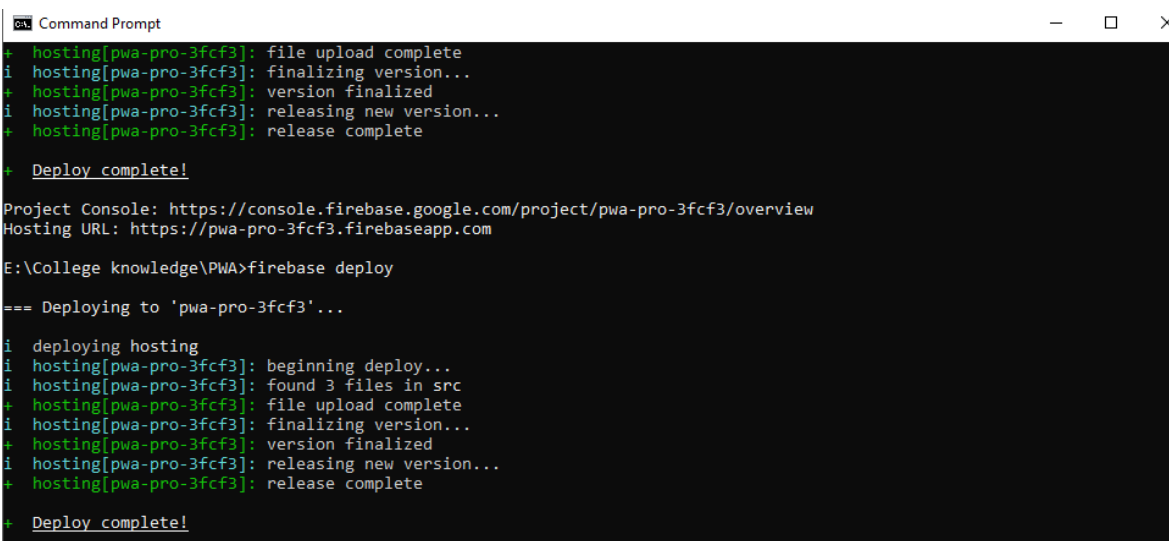
> Please select an option: Use an existing project
> Select a default Firebase project for this directory: pwa-pro-3fcf3 (PWA-pro)
i Using project pwa-pro-3fcf3 (PWA-pro)

=== Hosting Setup

Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
```

รูปที่ 6 ผลการเขียนคำสั่ง firebase init

5. นำโค้ดใน directory ไป deploy บนบริการ hosting ของ firebase ด้วยคำสั่ง firebase deploy



```
+ hosting[pwa-pro-3fcf3]: file upload complete
i hosting[pwa-pro-3fcf3]: finalizing version...
+ hosting[pwa-pro-3fcf3]: version finalized
i hosting[pwa-pro-3fcf3]: releasing new version...
+ hosting[pwa-pro-3fcf3]: release complete

+ Deploy complete!

Project Console: https://console.firebase.google.com/project/pwa-pro-3fcf3/overview
Hosting URL: https://pwa-pro-3fcf3.firebaseio.com

E:\College knowledge\PWA>firebase deploy

=== Deploying to 'pwa-pro-3fcf3'...

i deploying hosting
i hosting[pwa-pro-3fcf3]: beginning deploy...
i hosting[pwa-pro-3fcf3]: found 3 files in src
+ hosting[pwa-pro-3fcf3]: file upload complete
i hosting[pwa-pro-3fcf3]: finalizing version...
+ hosting[pwa-pro-3fcf3]: version finalized
i hosting[pwa-pro-3fcf3]: releasing new version...
+ hosting[pwa-pro-3fcf3]: release complete

+ Deploy complete!
```

รูปที่ 7 ผลการ deploy ขึ้น server ของ firebase

การพัฒนาเว็บไซต์เพื่อรองรับการทำ Progressive Web Application

- กำหนด register service worker ใน index.html

```
24 <script>
25   if('serviceWorker' in navigator) {
26     navigator.serviceWorker.register('/sw.js')
27       .then(function() {
28         console.log('Service Worker Registered');
29       });
30   }
31 </script>
```

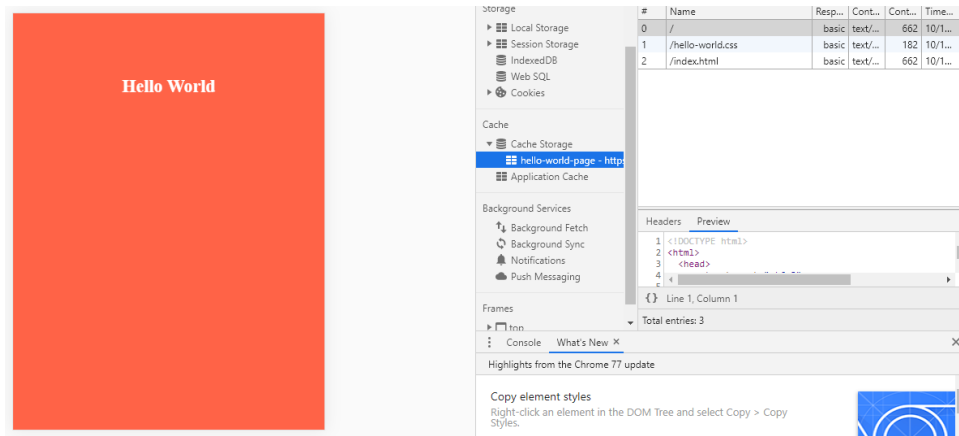
รูปที่ 8 service worker ใน index.html

- สร้างไฟล์ sw.js เพื่อเป็น resource cache

```
< index.html  firebase.json  JS sw.js  x  # hello-world.css
1  var cacheName = 'hello-world-page';
2  var filesToCache = [
3    '/',
4    '/index.html',
5    '/hello-world.css'
6  ];
7  self.addEventListener('install', function(e) {
8    console.log('[ServiceWorker] Install');
9    e.waitUntil(
10     caches.open(cacheName).then(function(cache) {
11       console.log('[ServiceWorker] Caching app shell');
12       return cache.addAll(filesToCache);
13     })
14   );
15 });
16 self.addEventListener('activate', event => {
17   event.waitUntil(self.clients.claim());
18 });
19 self.addEventListener('fetch', event => {
20   event.respondWith(
21     caches.match(event.request, {ignoreSearch:true}).then(response => {
22       return response || fetch(event.request);
23     })
24   );
25 });
26
```

รูปที่ 9 ผลการเขียนโปรแกรมไฟล์ sw.js

จะเห็นว่า sw.js ทำหน้าที่ทำให้ browser เก็บหน้าเว็บไว้ใน cache ของ browser



รูปที่ 10 ผลการทดสอบหน้าเว็บจากการใส่ sw.js

8. สร้างไอคอนสำหรับการเพิ่มเข้าไปในหน้าต่างโทรศัพท์ โดยการสร้าง manifest.json พร้อมกับใส่เข้าไปใน index.html

```

index.html 0001-468585655.png manifest.json x
1 {
2   "name": "Hello World PWA",
3   "short_name": "Hi",
4   "icons": [
5     {
6       "src": "icons/icon-128x128.png",
7       "sizes": "128x128",
8       "type": "image/png"
9     }, {
10    "src": "icons/icon-144x144.png",
11    "sizes": "144x144",
12    "type": "image/png"
13  }, {
14    "src": "icons/icon-152x152.png",
15    "sizes": "152x152",
16    "type": "image/png"
17  }, {
18    "src": "icons/icon-192x192.png",
19    "sizes": "192x192",
20    "type": "image/png"
21  }, {
22    "src": "icons/icon-512x512.png",
23    "sizes": "512x512",
24    "type": "image/png"
25  }
26 ],
27 "start_url": "/index.html",
28 "gcm_sender_id": "103953800507",
29 "display": "standalone",
30 "background_color": "#FF9800",
31 "theme_color": "#FF9800"
32 }

```

รูปที่ 11 ผลการเขียนโปรแกรมไฟล์ manifest.json

ซึ่งใน icons ให้ใส่เส้นทางของรูปภาพขนาดต่าง ๆ เข้าไป

9. เพิ่ม feature notification โดยการใส่ script ใน index.html และสร้าง ไฟล์ javascript

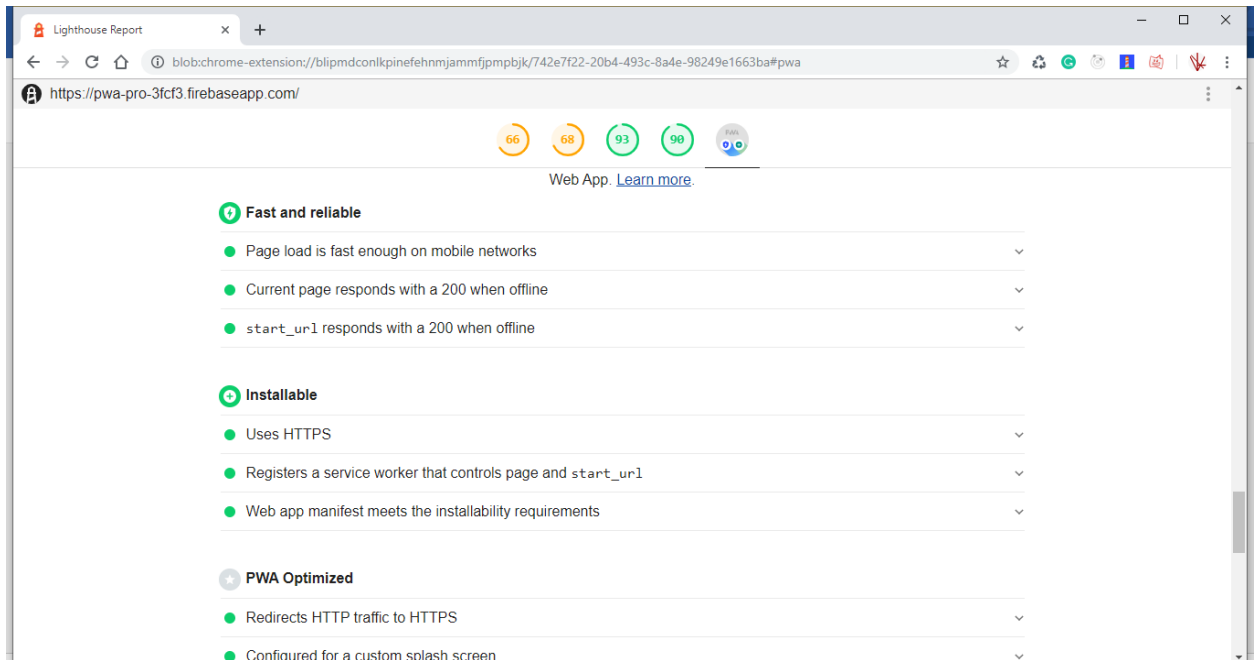
```
<script>
  const messaging = firebase.messaging();
  messaging.requestPermission()
    .then(function() {
      console.log('Notification permission granted.');
      return messaging.getToken();
    })
    .then(function(token) {
      console.log(token);
    })
    .catch(function(err) {
      console.log('Unable to get permission to notify.', err);
    })
</script>
```

รูปที่ 12 ผลการเขียน script การทำ notification ใน index.html

```
index.html JS firebase-messaging-sw.js 0001-468585655.png {} manifest.json JS sw.js
1 importScripts('https://www.gstatic.com/firebasejs/4.4.0/firebase-app.js');
2 importScripts('https://www.gstatic.com/firebasejs/4.4.0/firebase-messaging.js');
3
4 var firebaseConfig = {
5   apiKey: "--",
6   authDomain: "pwa-pro-3fcf3.firebaseio.com",
7   databaseURL: "https://pwa-pro-3fcf3.firebaseio.com",
8   projectId: "pwa-pro-3fcf3",
9   storageBucket: "pwa-pro-3fcf3.appspot.com",
10  messagingSenderId: "574184343847",
11  appId: "1:574184343847:web:cfebd20ce5aa6ec7540102"
12 };
13 firebase.initializeApp(firebaseConfig);
14
15 const messaging = firebase.messaging();
16 messaging.setBackgroundMessageHandler(function(payload) {
17   const title = 'Hello World';
18   const options = {
19     body: payload.data.body
20   };
21   return self.registration.showNotification(title, options);
22 });
```

รูปที่ 13 ผลการเขียนโปรแกรมไฟล์ firebase-message-sw.js

10. ใช้ lighthouse extension ตรวจสอบประสิทธิภาพของเว็บ



รูปที่ 14 ผลการทดสอบ PWA ของหน้าเว็บด้วย lighthouse

Website: <https://pwa-pro-3fcf3.firebaseio.com/>

GitHub: <https://github.com/ming2540/PWA-test>