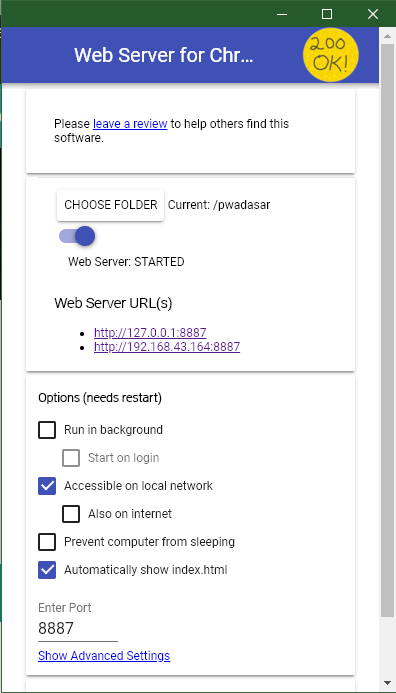
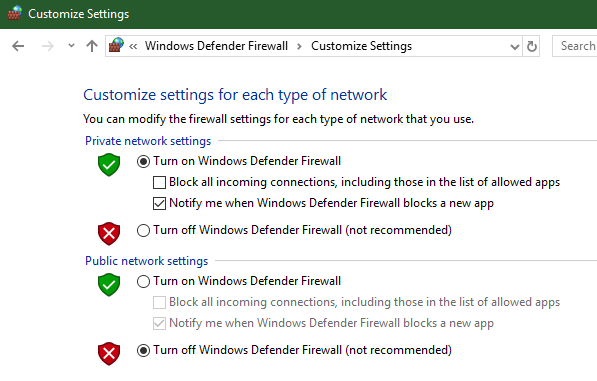
1. Menyalakan dan mengkonfigurasi Web Server for Chrome.

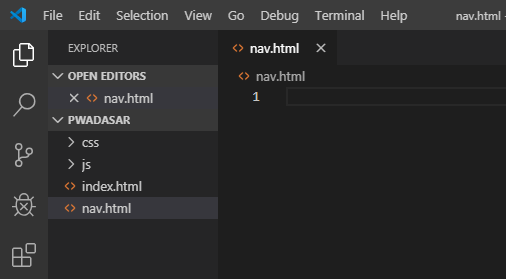


Choose folder lalu pilih folder Latihan yang telah didownload sebelumnya, kemudian START Web Server dan centang Automatically show index.html. Klik web server.

1. Matikan Firewall pada jaringan pubik.



1. Buka link dari web server dan lihat tampilannya.
2. Buat file dengan nama **nav.html**, seperti ditunjukkan pada gambar berikut.



Isikan dengan source code berikut untuk menambahkan daftar menu sidenav dan topnav.

|  |
| --- |
| <li><a class="waves-effect" href="#home"><i class="material-icons left">home</i> Home</a></li>  <li><div class="divider"></div></li>  <li><a class="waves-effect" href="#about"> <i class="material-icons left">supervisor\_account</i> About Us</a></li>  <li><div class="divider"></div></li>  <!--<li><a id="logo-container" class="brand-logo center pulse"><img src="img/icon.png" class="circle" style="height: 23px;" href="#"></a></li>--> |

1. Buat folder “pages” pada folder project tersebut. Lalu, buat file “home.html”dan “about.html. Isi file “home.html” dengan kode berikut.

|  |
| --- |
| <div class="parallax-container" style="background-image: url('img/bg-nav.jpg'); background-attachment: scroll; height: 70%;">    <div class="row container center">      <h3 style="color: azure;">Selamat Datang</h3>      <h5 style="color: azure;">Embrio Class</h5>    </div>  </div>  <div class="section grey lighten-2">    <div class="row container">      <h5 class="header">Progressive Web Application</h5>      <p class="grey-text text-darken-3 lighten-3">Sebuah aplikasi web yang menggunakan beberapa teknologi modern yang membuat user experience seperti menggunakan Native App (Android/iOS).</p>        <div class="divider black"></div>          <p class="center">Ini adalah laman aplikasi web progresif pertama karya “Tuliskan Namamu”.<br>             Semoga paham dengan materi yang diberikan dan dapat bermanfaat untuk kedepannya.<br>             Enjoy the class!~    </div>  </div> |

1. Buat file dengan nama **service-worker.js** di dalam folder project. Biarkan kosong terlebih dahulu. Kemudian buka kembali file index.html dan tambahkan kode berikut di bagian paling bawah sebelum tag tutup </body>.

|  |
| --- |
| <script>      // REGISTER SERVICE WORKER      if ('serviceWorker' in navigator) {        window.addEventListener('load', function() {          navigator.serviceWorker.register('/service-worker.js')          .then(function() {            console.log('Pendaftaran ServiceWorker berhasil');          })          .catch(function(){            console.log('Pendaftaran ServiceWorker gagal');          });        })      } else {        console.log("ServiceWorker belum didukung browser ini");      }  </script> |

Kemudian mendaftarkan file service-worker.js menggunakan method navigator.serviceWorker.register(). Buka Dev Tools pada Google chrome (Ctrl + Shift + i) saat membuka page latihan. Jika berhasil maka tampilan pada Service Workers akan seperti ini.

1. Untuk menyimpan asset – asset file ke cache, sekarang kita akan menulis kode service worker pada file service-worker.js dengan menambahkan kode berikut.

|  |
| --- |
| const CACHE\_NAME = 'firstpwa';  var urlsToCache = [      '/',      '/nav.html',      '/index.html',      '/pages/home.html',      '/pages/about.html',      '/css/materialize.min.css',      '/js/materialize.min.js',      '/js/script.js',      '/img/parallax.png',      '/img/icon.png'  ];  self.addEventListener('install', function(event){      event.waitUntil(          caches.open(CACHE\_NAME)          .then(function(cache) {              return cache.addAll(urlsToCache);          })      );  }); |

Bila kode berjalan dengan baik, maka seharusnya asset - asset tadi akan terdaftar pada cache browser. Kamu dapat mengecek cache yang sudah terdaftar pada **DevTools** tab **Application** bagian **Cache Storage**. Klik tanda segitiga kebalik di sebelah kiri menu **Cache Storage** untuk melihat daftar cache. Sampai sini baru menyimpan asset – asset file ke cache.

1. Tambahkan kode berikut pada file service-worker.js agar halaman menggunakan asset yang sudah disimpan di cache:

|  |
| --- |
| self.addEventListener('fetch', function(event) {      event.respondWith(          caches.match(event.request, {cacheName:CACHE\_NAME})          .then(function(response) {              if(response){                  console.log("ServiceWorker: Gunakan aset dari cache: ", response.url);                  return response;              }              console.log("ServiceWorker: Memuat aset dari server: ", event.request.url);              return fetch(event.request);          })      );  }); |

Tambahkan kode berikut di bagian akhir file service-worker.js:

|  |
| --- |
| self.addEventListener("activate", function(event) {    event.waitUntil(      caches.keys().then(function(cacheNames) {        return Promise.all(          cacheNames.map(function(cacheName) {            if (cacheName != CACHE\_NAME) {              console.log("ServiceWorker: cache " + cacheName + " dihapus");              return caches.delete(cacheName);            }          })        );      })    );  }); |

1. Menambahkan aplikasi ke home screen. Agar sebuah PWA bisa dipasang ke *home screen.* Sebuah web app manifest didaftarkan melalui elemen link yang disimpan di dalam tag head di setiap halaman HTML seperti saat kita akan menggunakan CSS eksternal. Ubah kode index.html dengan memasukkan kode berikut di dalam elemen <head>:

|  |
| --- |
| <meta name="theme-color" content="#00897B"/>  <link rel="manifest" href="/manifest.json"> |

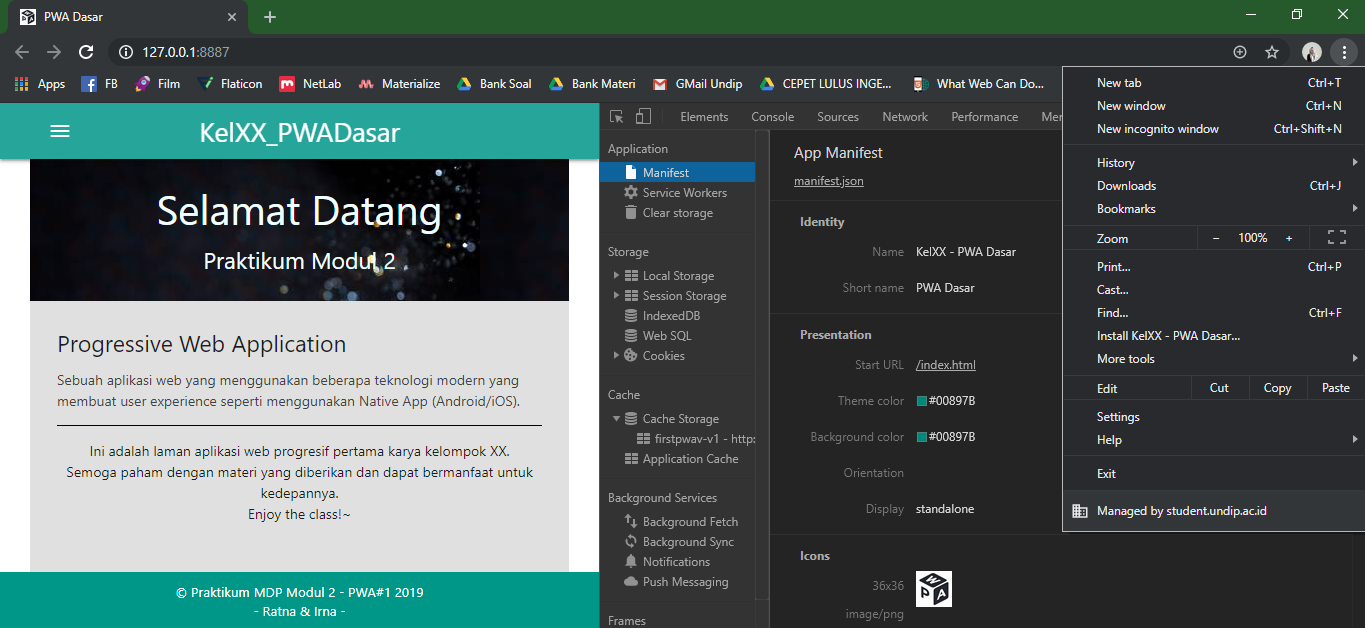
1. Selanjutnya, buat sebuah file bernama manifest.json di folder project berdampingan dengan index.html dan isikan dengan konten berikut ini:

|  |
| --- |
| {      "name": "Embrio Class - PWA Dasar",      "short\_name": "PWA Dasar",      "description": "Aplikasi PWA Dasar Embrio Class 2019",      "start\_url": "/index.html",      "display": "standalone",      "background\_color": "#00897B",      "theme\_color": "#00897B",      "icons": [{                  "src": "img/icon/Icon-36.png",                  "sizes": "36x36",                  "type": "image/png",                  "density": "0.75"                }, {                  "src": "img/icon/Icon-48.png",                  "sizes": "48x48",                  "type": "image/png",                  "density": "1.0"                }, {                  "src": "img/icon/Icon-72.png",                  "sizes": "72x72",                  "type": "image/png",                  "density": "1.5"                }, {                  "src": "img/icon/Icon-96.png",                  "sizes": "96x96",                  "type": "image/png",                  "density": "2.0"                }, {                  "src": "img/icon/Icon-144.png",                  "sizes": "144x144",                  "type": "image/png",                  "density": "3.0"                }, {                  "src": "img/icon/Icon-192.png",                  "sizes": "192x192",                  "type": "image/png",                  "density": "4.0"                }      ]  } |

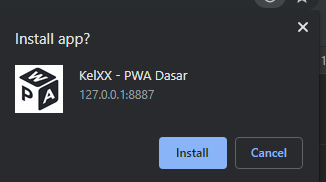
1. Terakhir simpan file icon di dalam folder project dan beri nama icon.png. Jangan lupa untuk menghapus tanda komentar yang ada di nav.html agar icon muncul di halaman kerja kalian.

|  |
| --- |
| <li><a id="logo-container" class="brand-logo center pulse"><img src=" img/icon.png" class="circle" style="height: 23px;" href="#"></a></li> |

1. Untuk menguji web app manifest telah berhasil di daftarkan, silahkan buka kembali aplikasi PWA di Chrome, lalu buka tab **Application** di jendela **Chrome Dev Tools** dan pilih menu **Manifest** di sebelah kiri. Periksa bahwa icon dan element lainnya telah terdefinisikan di dalam file manifest.json.
2. Terakhir, install aplikasi dengan cara menekan tombol titik tiga dipojok kanan atas, seperti yang ditunjukkan pada gambar dibawah ini. Lalu, klik “Install ....”.



Gambar 3.13 App Manifest PWA



Gambar 3.14 Install PWA

1. Setelah diinstall, akan muncul ikonnya pada Desktop.



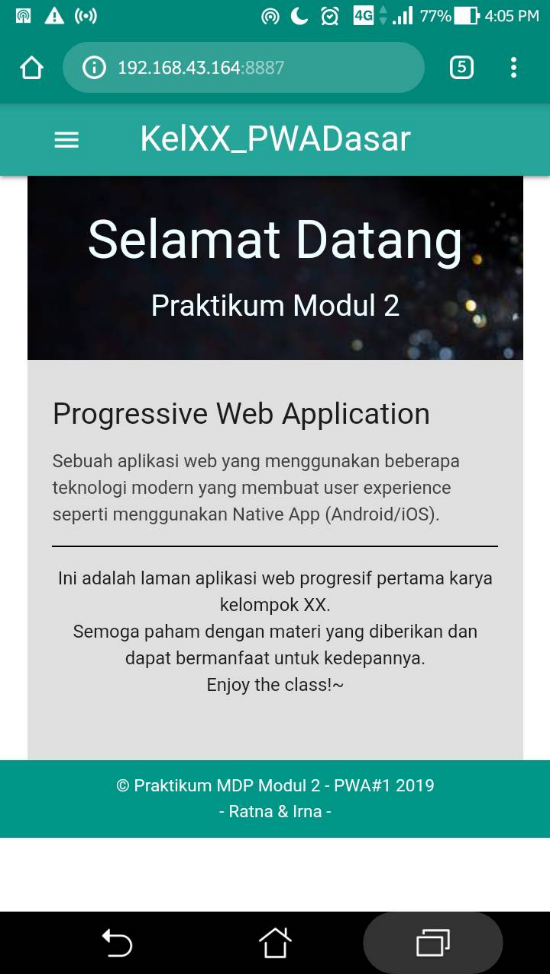
Gambar 3.15 Ikon PWA Desktop

1. Buka aplikasinya dari Desktop, maka akan seperti gambar berikut ini karena tipe tampilannya standalone, maka akan tidak akan terlihat url nya seperti pada saat dibuka melalui browser.

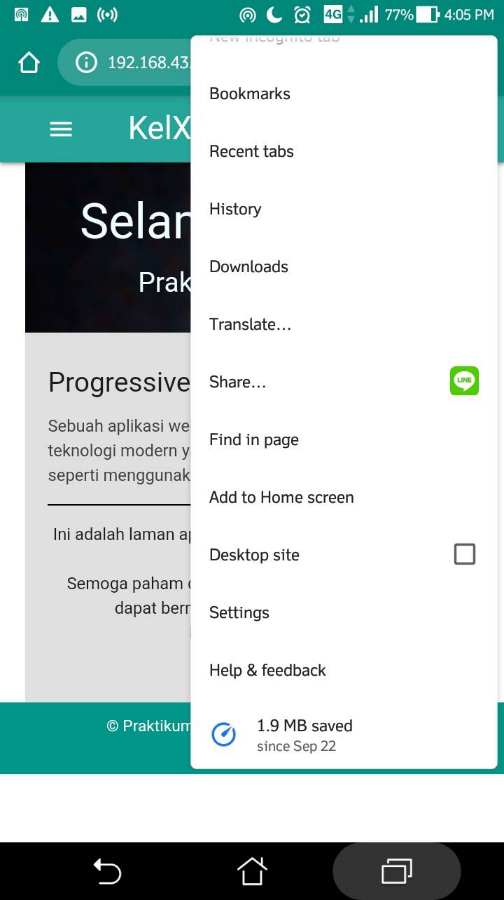


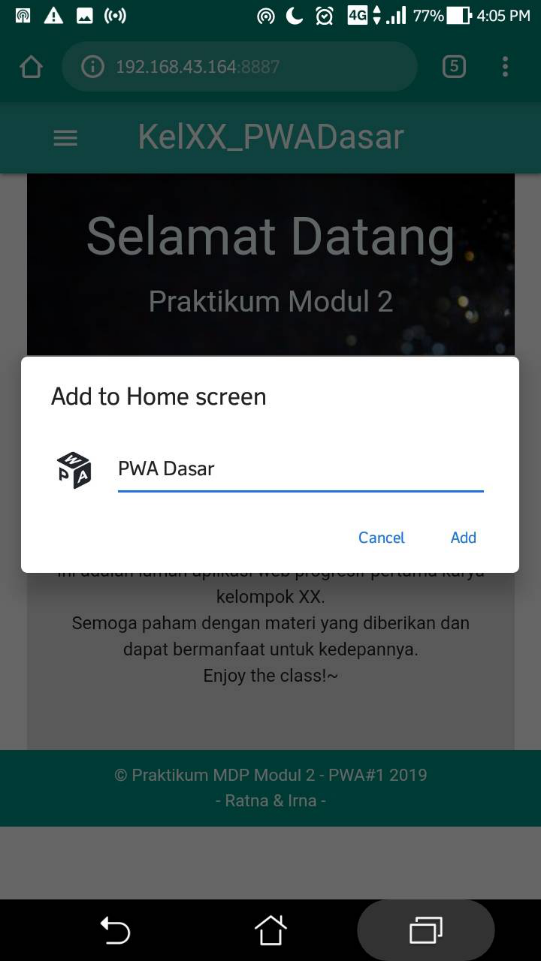
Gambar 3.16 Tampilan PWA pada Desktop

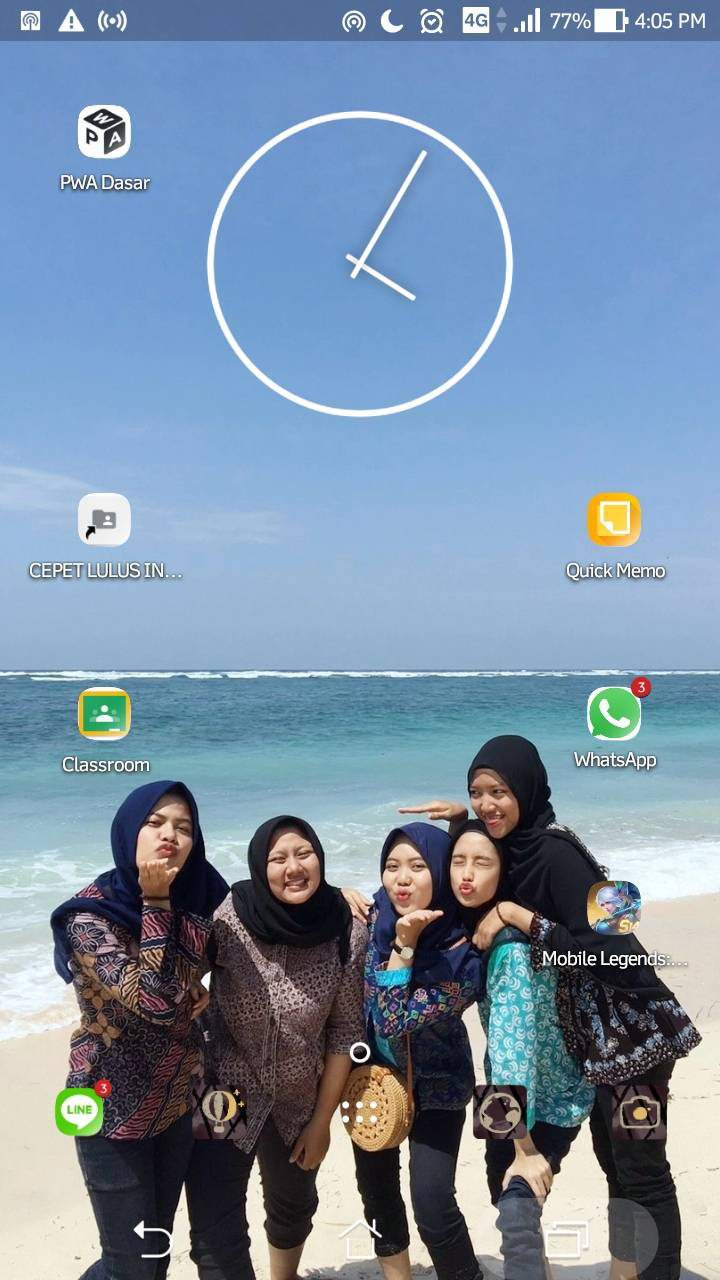
1. Install dan buka PWA nya melalui Mobile dengan cara satu kan jaringan internet pada Desktop dan Mobile.
2. Kemudian, ketik URL web servernya yang tadi ditambah dengan port pada Web Server for Chrome.



1. Lalu, install PWA nya di Mobile.







1. Buka PWA nya melalui Mobile.



