

## **RANCANG BANGUN APLIKASI *OFFICIAL STORE* BERBASIS *PROGRESIVE WEB APP* (STUDI KASUS: ONE STOP HYDROPONICS SURABAYA)**

**Naufal Al - Fikri**

D3 Manajemen Informatika, Fakultas Teknik, Universitas Negeri Surabaya

Email : naufal.17050623011@mhs.unesa.ac.id

**Andi Iwan Nurhidayat, S.Kom., M.T.**

Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya

Email : andyl345k5@gmail.com

### **Abstrak**

One Stop Hydroponics Surabaya adalah salah satu toko di kota Surabaya yang menjual dan juga memproduksi berbagai macam kebutuhan untuk pertanian hydroponic. Toko tersebut memproduksi peralatan pertanian hydroponic, melakukan pertanian hydroponic dan juga menjual hasil pertanian hydroponic mereka. Penjualan dan pengenalan produk mereka hanya dilakukan hanya melalui media sosial pada umumnya dan juga secara manual. Seharusnya, dengan kemajuan teknologi yang telah ada sekarang, pengenalan informasi produk dan penjualan suatu produk dapat dilakukan dengan mudah melalui internet terutama melalui sebuah aplikasi. Media yang dipakai juga harus Tangguh terhadap kondisi internet yang buruk karena penyebaran informasi produk ingin menjangkau kemanapun hingga ke tempat-tempat yang memiliki kondisi jaringan internet yang buruk. Namun One Stop Hydroponics Surabaya juga belum memiliki website official store tersendiri untuk melakukan semua hal itu.

Pembuatan aplikasi official store untuk One Stop Hydroponics Surabaya berbasis *Progresive Web App*, yaitu dengan tujuan website tersebut dapat merespon dengan baik saat kondisi sedang offline dan dalam keadaan internet yang kurang baik. Tidak dipungkiri bahwa *Progresive Web App* dapat merespon jauh lebih baik dibandingkan dengan website biasa.

**Kata Kunci:** *Progresive Web App, Official Store, Hydroponics.*

### **Abstract**

One Stop Hydroponics Surabaya is one of the shops in the city of Surabaya that sells and also produces a variety of needs for hydroponic agriculture. The shop produces hydroponic farming equipment, conducts hydroponic farming and also sells their hydroponic agricultural products. The sale and introduction of their products is only done through social media in general and also manually. Supposedly, with advances in technology that has existed now, the introduction of product information and the sale of a product can be done easily through the internet, especially through an application. The media used must also be resilient to poor internet conditions because the dissemination of product information wants to reach everywhere to places that have poor internet network conditions. Even though One Stop Hydroponics Surabaya also doesn't have its own official store website to do all of that.

Creating an official store application for One Stop Hydroponics Surabaya based on a *Progressive Web App*, that is, with the aim that the website can respond properly when conditions are offline and in a poorly internet condition. It is undeniable that the *Progressive Web App* can respond much better than a normal website.

**Keywords:** *Progresive Web App, Official Store, Hydroponics.*

### **PENDAHULUAN**

One Stop Hydroponics Surabaya adalah salah satu toko yang berada di kota Surabaya yang menjual dan juga memproduksi berbagai macam kebutuhan untuk pertanian hidroponik. Toko tersebut memproduksi peralatan pertanian hidroponik, melakukan pertanian menggunakan metode hidroponik dan juga menjual hasil pertanian hidroponik mereka. Segala sesuatu yang mereka lakukan mulai dari pengenalan informasi toko,

informasi produk, penjualan dan pembelian oleh konsumen masih dilakukan dengan cara manual dan juga hanya memanfaatkan media sosial pada umumnya.

Seharusnya, dengan kemajuan teknologi yang telah ada sekarang, pengenalan informasi produk dan penjualan suatu produk dapat dilakukan dengan mudah melalui internet terutama melalui sebuah aplikasi. Media yang dipakai juga harus tangguh terhadap kondisi internet yang buruk karena

penyebaran informasi produk ingin menjangkau kemanapun hingga ke tempat-tempat yang memiliki kondisi jaringan internet yang buruk. Data Kementerian Komunikasi dan Informatika menyebutkan, pengguna internet di Indonesia hingga kini telah mencapai 82 juta orang, dan dengan capaian tersebut Indonesia berada pada peringkat ke-8 di dunia. Dan juga Lebih lanjut, CupoNation Indonesia menyampaikan masyarakat Indonesia lebih gemar berbelanja online melalui *mobile* daripada platform yang lain.

Menurut penelitian oleh Devi Anggraeni yang memiliki judul penelitian “Rancang Bangun Aplikasi E-commerce Berbasis *Progressive Web Apps* Pada Distro Romusha”. Penelitian ini bertujuan untuk mengembangkan suatu aplikasi toko online yang berbasis *Progressive Web Apps* yang tetap dapat berjalan walaupun keadaan jaringan internet kita sangat buruk, dan juga dapat melakukan penyampaian informasi produk kepada pelanggan.

Berdasar permasalahan diatas, penulis mengusulkan ide membuat *progressive web app* yang bisa bekerja pada jaringan internet yang buruk bahkan bisa berjalan secara offline, tampak seperti aplikasi *mobile* dan juga *Progressive Web App* adalah website yang bisa bekerja dalam kondisi offline, menyajikan loading time hampir secara instan, aman, dan juga tangguh terhadap jaringan yang buruk. Dengan adanya *Progressive Web App* seharusnya kecepatan akses website akan semakin tinggi. Untuk One Stop Hydroponics Surabaya yang belum memiliki website *official store* tersendiri untuk melakukan semua hal itu.

Pada website *official store* ini penulis akan menggunakan metode pengembangan perangkat lunak model *waterfall* yang dapat memudahkan penulis dalam mengerjakan aplikasi dan jugal menyelesaikan tugas akhir. Model *waterfall* ini akan dilakukan secara beruntut mulai dari proses analisa studi kasus hingga melakukan pengujian aplikasi. Hal pertama yang dilakukan penulis adalah melakukan analisis studi kasus pada pemilik toko One Stop Hydroponic Surabaya terlebih dahulu, yaitu mulai dari fitur apa saja yang dibutuhkan dan ingin di terapkan pada website, setelah itu penulis melakukan proses perancangan dan desain berdasar hasil analisis pada narasumber yaitu pemilik toko One Stop Hydroponic Surabaya. Setelah tahapan perancangan selesai penulis melakukan proses implementasi rancangan dengan melakukan *coding* untuk pembuatan website *official store* tersebut. Lalu setelah semua proses tersebut selesai penulis akan melakukan tahapan terakhir dari *waterfall* yaitu pengujian aplikasi *Official Store* One Stop Hydroponic Surabaya.

Aplikasi *Official Store* One Stop Hydroponic Surabaya ini yang akan diterapkan menjadi website

berbasis *Progressive Web App* ini diharapkan akan membuat pelanggan akan lebih mudah mendapat informasi terbaru dari toko karena lebih mudah diakses, pelanggan juga dapat melakukan pembelian barang barang hidroponik, baik bibit tanaman, peralatan tanam hidroponik, sayur dan buah hasil tanam hidroponik dan juga pelanggan akan mendapatkan dua jenis tampilan baik tampilan *website* dan juga tampilan *mobile apps* hanya dengan satu aplikasi yang sama.

Maka dari itu, perlu dilakukan perancangan dan pembuatan website *official store* untuk One Stop Hydroponics Surabaya, dengan judul : **“RANCANG BANGUN APLIKASI OFFICIAL STORE BERBASIS PROGRESIVE WEB APP (STUDI KASUS: ONE STOP HYDROPONICS SURABAYA)”**.

## KAJIAN PUSTAKA

### Official Store

Menurut Verisign *Official Store* atau Toko Resmi yang biasanya melakukan penjualan secara online menggunakan website resmi toko dimana penjual atau *owner* melakukan pengelolaan penjualan dan pemasaran produk mereka sendiri tanpa adanya perantara, jadi mereka langsung menjual produknya kepada pembeli langsung secara mandiri agar mereka bisa mendapat keuntungan yang lebih banyak disbanding melakukan penjualan melalui *marketplace*.

### Progressive Web Apps

*Progressive Web App* adalah suatu teknik bagaimana Anda dapat mengakses dengan cepat pengalaman website dan aplikasi menjadi satu tanpa harus menginstall atau memasang aplikasi tersebut. *Progressive Web App* memiliki fitur untuk mengatur *caching*, jaringan, dan notifikasi yang bernama *service worker*.

### Service Worker

*Service worker* merupakan sebuah kode program yang berjalan pada latar belakang browser, dan terpisah dari halaman web, yang memiliki fungsi sebagai pintu gerbang ke berbagai fitur yang tidak memerlukan interaksi dengan pengguna. Sekarang *service worker* telah menyediakan berbagai macam fitur seperti pemberitahuan sinkronisasi dan push pada latar belakang. Di masa yang akan datang, diharapkan *service worker* dapat mendukung hal lainnya seperti sinkronisasi berkala. Fitur inti yang dimaksud adalah kemampuan mencegah dan menangani *request network*, termasuk pengelolaan *cache respons* melalui program. Yang menyebabkan API ini menarik yaitu

karena dapat membuat anda merasakan pengalaman offline, yang memberikan developer kontrol penuh atas pengalaman. *Connectivity independent* - Ditingkatkan dengan *service worker* agar dapat bekerja offline atau ketika koneksi jaringan sedang buruk.

1. **App-like** menggunakan model shell aplikasi untuk menyediakan navigasi dan interaksi gaya aplikasi.
2. **Fresh** selalu memperbarui karena adanya *service worker*.
3. **Safe** Dilayani melalui TLS untuk mencegah aplikasi diintai dan memastikan konten belum dirusak.
4. **Discoverable** Dapat diidentifikasi sebagai "aplikasi" berkat manifest W3C dan ruang lingkup registrasi *service worker* yang memungkinkan mesin pencari untuk menemukannya.
5. **Re-engageable** Membuat keterlibatan ulang mudah melalui fitur seperti pemberitahuan push.
6. **Installable** Memungkinkan pengguna untuk "menyimpan" aplikasi yang mereka anggap paling berguna di layar beranda mereka tanpa kerumitan toko aplikasi.
7. **Linkable** Mudah dibagikan melalui URL dan tidak memerlukan instalasi yang rumit. Keuntungan dari progressive web app bisa dideskripsikan dalam poin-poin berikut :
8. **Progressive** Bisa digunakan untuk semua pengguna terlepas dari pilihan browser karena PWA dibuat dengan peningkatan progresif sebagai core.
9. **Responsive** Pas dengan segala faktor bentuk, desktop, seluler, tablet, atau apa pun yang berikutnya.

### Javascript

Javascript adalah salah satu bahasa script yang sifatnya dinamis yang dapat digunakan untuk membuat halaman HTML statis agar lebih interaktif. Blok kode javascript diawali dan diakhiri dengan tag `<script>`. Tag `<script>` memiliki satu atribut penting yang bernama *language*. Atribut biasanya berupa angka dan berfungsi untuk menyatakan bahasa script yang sedang digunakan. Atribut ini juga diindikasikan browser yang mendukung versi javascript yang digunakan (Sianipar, 2017). Javascript bersifat *opensource* yang artinya dapat digunakan oleh semua orang. Javascript merupakan bahasa script yang sangatlah ringan sehingga script yang dibuat akan langsung dieksekusi

tanpa harus dikompilasi terlebih dahulu (Setiawan, 2017).

### Cascading Style Sheet (CSS)

Cascading Style Sheet (CSS) adalah salah satu kode pemrograman yang digunakan untuk mengatur gaya tampilan atau layout supaya lebih menarik, yang digunakan oleh programmer dan web designer untuk menentukan warna, tata letak font, dan semua aspek yang berhubungan dengan presentasi dokumen di web tersebut. Kode CSS paling banyak dimasukkan ke dalam halaman HTML dengan menambahkan tag `<link>` pada tag `<head>`. (Setiawan, 2017).

### Node.js

Menurut Alex Handy (2011:6) Node.js merupakan sebuah platform perangkat lunak pada sisi server dan aplikasi jaringan. Ditulis dengan menggunakan bahasa pemrograman JavaScript dan dijalankan di semua jenis sistem operasi dengan tidak adanya perubahan pada *sourcecode*. Node.js memiliki *library* server HTTP sendiri, sehingga tidak memerlukan program web server tambahan seperti apache atau nginx untuk menjalankan server web sendiri.

### MYSQL

Perangkat lunak yang biasa digunakan untuk melakukan manajemen database adalah *MYSQL*. *MYSQL* berbasis *open source*, sehingga dapat digunakan secara gratis. Pengolahan data pada *MySQL* menggunakan bahasa *SQL (Structured Query Language)* (Solichin, 2016). *SQL* adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Sadeli, 2014).

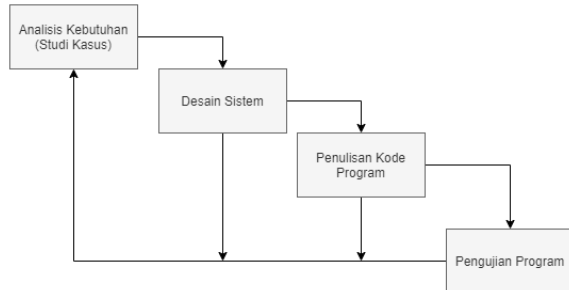
### METODE

#### Metode Pengembangan Perangkat Lunak

Metode yang digunakan oleh penulis untuk pembuatan aplikasi *Official Store* One Stop Hydroponic Surabaya adalah metode model *waterfall* (air terjun). Model *Squential Linier* atau yang biasa disebut dengan model pengembangan *waterfall* adalah sebuah model metode pengembangan perangkat lunak yang paling lama ada, dan paling banyak digunakan. Model ini merupakan model yang dikerjakan sesuai hirarki atau tahapan atau beruntut, yang dimulai dengan Analisis, Desain, Kode, Pengujian. Model *waterfall* ini dimulai dengan proses analisis kebutuhan



data pada studi kasus, data tersebut digunakan dalam perencanaan dan desain aplikasi yang akan dibuat, setelah itu akan melakukan pengkodean untuk menciptakan sebuah aplikasi website yang berbasis *progressive web apps*.



Gambar 1 Hirarki dalam Model Waterfall

Pada gambar Gambar 1 merupakan hirarki atau tahapan yang ada pada model *waterfall*. Semua tahapan itu akan dikerjakan oleh penulis hingga aplikasi selesai.

## METODE REKAYASA

### 1. Metode Pengumpulan Data

Pengumpulan data yang dilakukan penulis dalam melakukan penelitian ini dilakukan dalam beberapa metode, seperti sebagai berikut.

- Observasi  
Kegiatan kunjungan pada toko One Stop Hydroponic Surabaya untuk mengetahui bagaimana proses penjualan produk kepada para pelanggan.
- Wawancara  
Kegiatan yang dilakukan penulis untuk mengetahui kebutuhan apa saja yang diinginkan oleh pemilik toko.
- Studi Dokumentasi Data  
Kegiatan pengumpulan data yang diperlukan untuk perancangan dan desain aplikasi, seperti data produk produk yang disediakan dan data kategori produk.

### 2. Gambaran Umum Sistem

Pelayanan penjualan dan pemasaran produk dari toko saat ini masih dilakukan secara konvensional dan juga hanya mengandalkan media sosial saja. Penjualan seperti ini memiliki kekurangan karena pemilik toko melayani pelanggan secara langsung dan juga penjualan melalui media sosial terkadang kurang bisa dipercaya pelanggan dan sering tenggelam oleh kiriman pengguna media sosial yang lain.

### 3. Analisis Sistem yang Diusulkan

Pada tahap ini penulis melakukan rekomendasi fitur sesuai dengan kebutuhan yang diinginkan oleh pemilik toko One Stop Hydroponic Surabaya. Berikut fitur-fitur dari aplikasi yang direkomendasikan.

- A. Website sebagai toko online *Official Store* yang akan menampilkan informasi terbaru produk, iklan produk, promosi produk, list produk, gambar produk, deskripsi, harga produk.
- B. Fitur pencarian produk berdasarkan nama produk yang tertera.
- C. Fitur filter produk yang akan difilter berdasarkan kategori produk.
- D. Fitur informasi tentang produk terbaru dan juga fitur promosi produk.
- E. Fitur keranjang belanja untuk proses transaksi penjualan produk.
- F. Fitur pendaftaran akun untuk pelanggan baru.

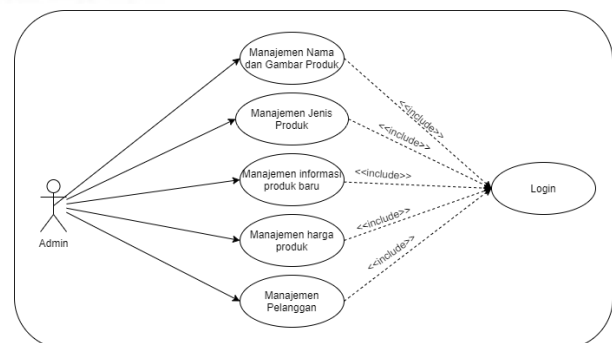
## DESAIN SISTEM

Perancangan aplikasi pada penelitian ini menggunakan diagram perancangan *Unified Modelling Language (UML)*.

### 1. Use Case Diagram

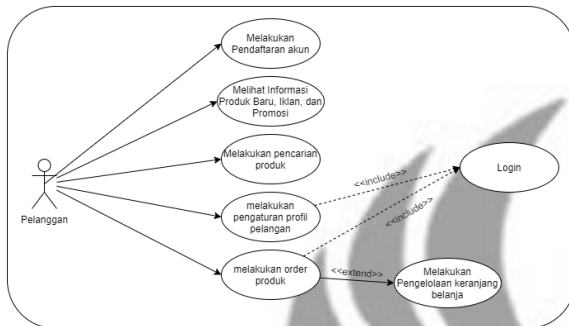
*Use Case Diagram* adalah gambaran interaksi pengguna sistem terhadap sistem dengan menjalankan fungsi-fungsi yang dapat diterima pada sistem ini.

- A. Use Case Diagram Admin, diagram ini menggambarkan interaksi yang terjadi antara admin pada aplikasi yang dirancang. Aktor admin dapat melakukan pengelolaan seluruh data yang ada di aplikasi dengan login admin terlebih dahulu.



Gambar 2 Use Case Diagram Admin

- B. Use Case Diagram Pelanggan, diagram ini menggambarkan interaksi yang terjadi antara pelanggan pada aplikasi yang dirancang. Aktor pelanggan dapat melakukan Tindakan seperti melihat informasi, mencari produk, melakukan order produk, melihat promosi iklan, dan melakukan pengaturan profil pelanggan, dan seluruhnya dilakukan dengan login pelanggan terlebih dahulu.

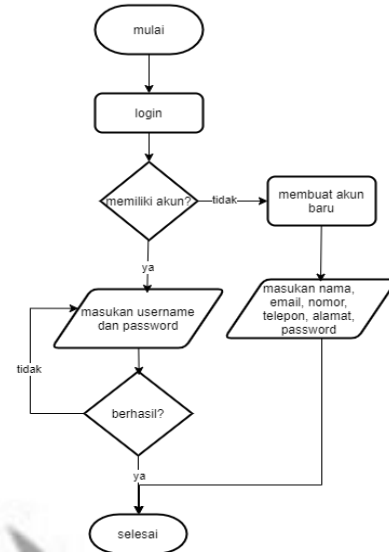


Gambar 3 Use Case Diagram Pelanggan

## 2. Flowchart Diagram

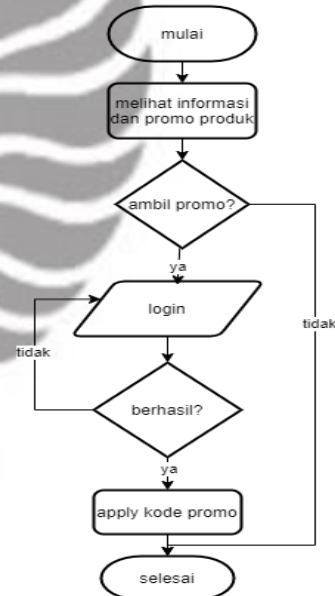
Diagram alir atau yang biasa kita sebut sebagai *flowchart* merupakan diagram yang memiliki arus yang menggambarkan alur penyelesaian sebuah masalah, dengan simbol grafis yang menyatakan aliran algoritma dengan bentuk kotak beserta urutannya dan dihubungkan dengan menggunakan panah pada tiap langkahnya, berikut ini adalah desain *flowchart* diagram sistem.

- a. Flowchart Diagram Login dan Registrasi pelanggan. Pada diagram ini terlihat bagaimana pelanggan dapat melakukan login untuk melakukan order, dan melakukan pendaftaran jika belum memiliki akun.



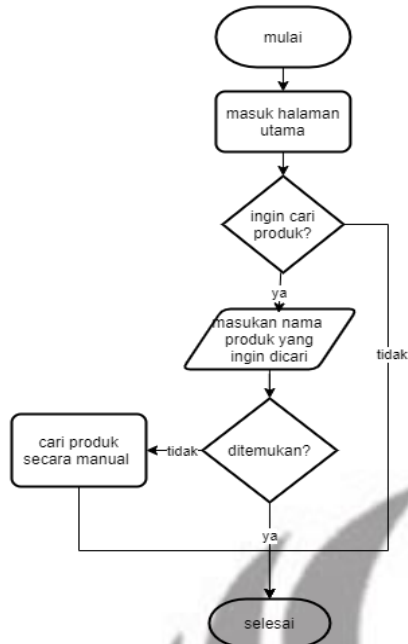
Gambar 4 Flowchart diagram login dan registrasi

- b. Flowchart Diagram Informasi Produk dan Promosi, pada diagram ini terlihat bagaimana pelanggan dapat melakukan klaim kode promo pada aplikasi.



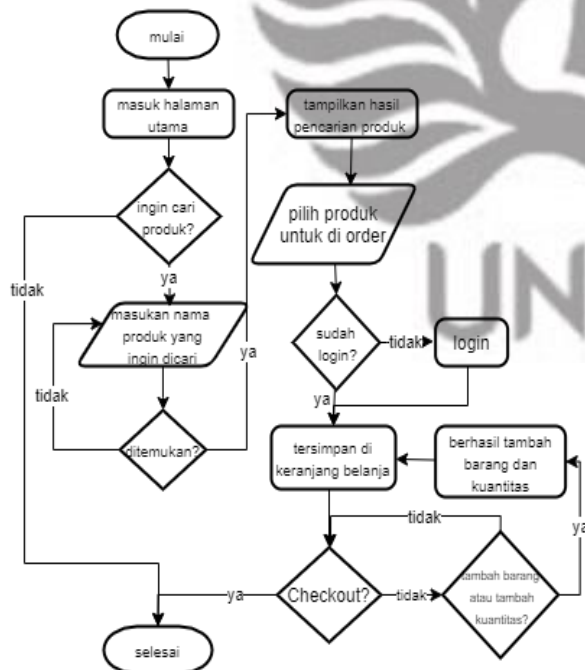
Gambar 5 flowchart diagram informasi produk dan promosi

- c. Flowchart diagram pencarian produk pelanggan. Pada diagram ini terlihat bagaimana pelanggan akan mencari produk hingga mendapatkan produk yang dicari atau dibutuhkan pada halaman aplikasi. Berikut alur yang dilakukan.



Gambar 6 flowchart diagram pencarian produk pelanggan

- d. Flowchart diagram order produk pelanggan, pada diagram ini terlihat bagaimana pelanggan Ketika melakukan order pada aplikasi. Berikut alur yang dilakukan.



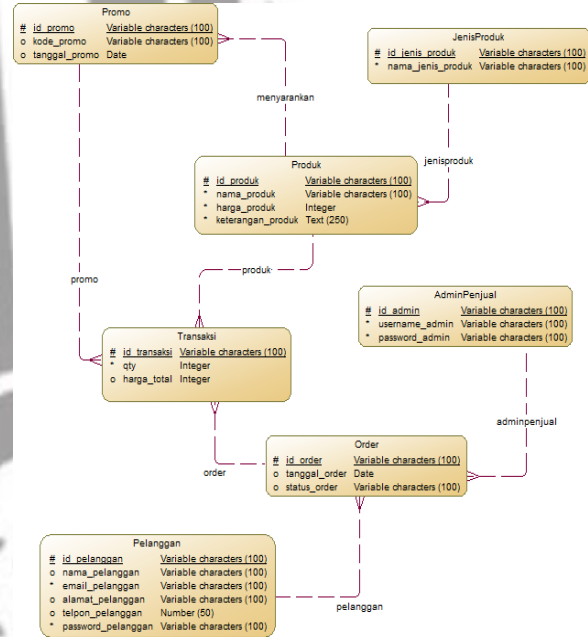
Gambar 7 Flowchart diagram order produk pelanggan

## HASIL DAN PEMBAHASAN

Hasil dari Tugas Akhir ini adalah sebuah rancang bangun Aplikasi *Official Store* berbasis *Progressive Web Apps* yang bertujuan agar mempermudah pemilik toko dalam proses publikasi informasi dan pemasaran produk dan juga mempermudah pelanggan dalam melakukan transaksi penjualan dan informasi terbaru dari toko dengan aman pada *Official Store* (Toko Resmi) toko One Stop Hydroponic Surabaya.

### Conceptual Data Model (CDM)

Dengan menggunakan *Conceptual Data Model* kita dapat membuat tampilan data yang terstruktur yang diperlukan untuk mendukung proses bisnis, dan mengukur kinerja terkait. Pada gambar berikut merupakan rancangan basis data yang digunakan pada Aplikasi *Official Store* One Stop Hydroponic Surabaya berbasis *Progressive Web Apps*.



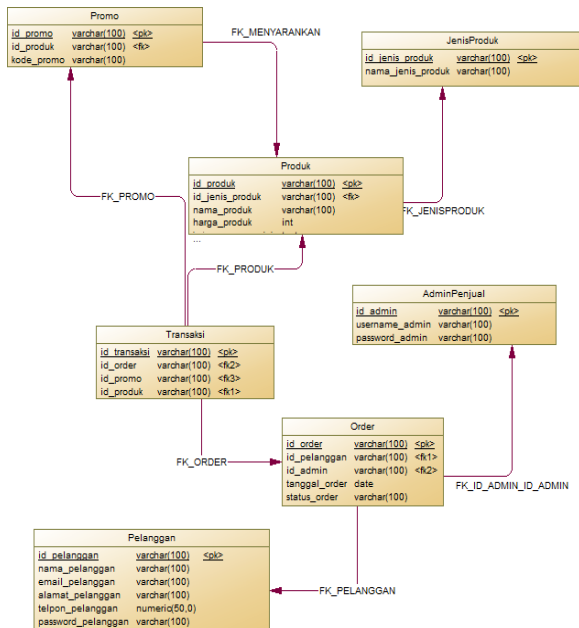
Gambar 8 CDM (Conceptual Data Model)

Pada Gambar 8 terdapat hubungan antar data dalam basis data, terdapat tujuh entitas yaitu Pelanggan, AdminPenjual, Promo, Order, Transaksi, Produk, dan Jenis produk.

### Physical Data Model (PDM)

*Physical Data Model* adalah gambaran yang lebih detail dari sebuah basis data secara fisik. *Physical Data Model* mempresentasikan bagaimana model tersebut akan dibangun dalam database, *Physical Data Model* memperlihatkan seluruh struktur tabel,

termasuk kolom nama, kolom tipe data, kolom batasan, *primary key*, *foreign key*, dan hubungan antar tabel, berikut adalah gambar *Conceptual Data Model* (CDM) yang telah diubah menjadi *Physical Data Model* (PMD).



Gambar 9 PDM (Physical Data Model)

*Physical Data Model* (PDM) digunakan sebagai penggambaran yang lebih detail tentang basis data secara fisik. Pada Gambar 9 menunjukkan struktur penyimpanan data yang sebenarnya.

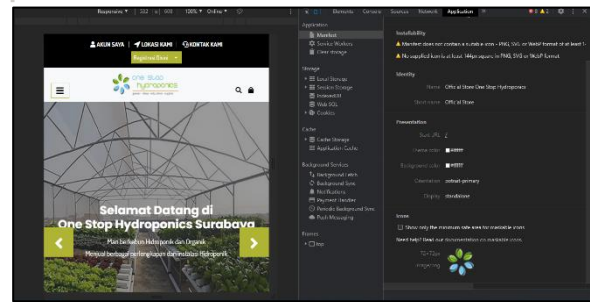
### Web App Manifest

*Web App Manifest* adalah sebuah file *JavaScript Object Notation* (JSON) sederhana yang dapat membuat *developer* memiliki kemampuan agar dapat mengontrol bagaimana aplikasi terlihat oleh *end user* (pengguna), misalnya pada lapisan halaman beranda pada *smartphone* kita, mengarahkan pengguna kepada apa yang dapat diluncurkan dan menentukan tampilan saat aplikasi telah diluncurkan. *Web App Manifest* dapat memberikan kemampuan penyimpanan situs langsung ke lapisan halaman beranda perangkat (*smartphone*), dengan cara.

- Website akan memiliki ikon aplikasi dan nama aplikasi yang membedakan dari website yang lain.
- Website akan dapat menampilkan sesuatu kepada pengguna seperti layaknya aplikasi *native* selagi *resource* website sedang dipulihkan dari *cache*.

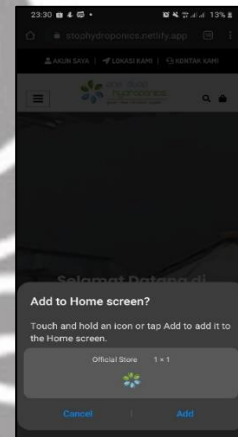
Dengan terpasangnya *web app manifest* kita bisa melakukan set ikon untuk digunakan di browser dan

hal ini akan membuat pengalaman pengguna yang lebih baik.



Gambar 10 Tampilan manifest.json yang bekerja pada website

Pada gambar 10 menunjukkan tampilan bahwa *manifest.json* bekerja pada web dan dapat diketahui melalui *developer tools* yang ada pada *browser google chrome*.



Gambar 11

Pada Gambar 11 menunjukan bahwa website memiliki ikon dan juga kemampuan untuk ditambahkan ke dalam layar beranda ponsel dengan muncul notifikasi untuk ditambahkan ke dalam layar beranda ponsel,



Gambar 12



dan pada Gambar 12 menunjukkan hasil setelah website berhasil ditambahkan ke layar beranda ponsel, seperti yang sudah diatur pada manifest.json sebelumnya.

### Service Worker

*Service Worker* adalah sesuatu yang berjalan di latar belakang *browser* pengguna dan merupakan salah satu jenis web worker yang ada. *Service Worker* sendiri mulanya adalah sebuah file javascript yang berjalan pada *thread* yang berbeda dengan *thread browser* utama, menangani permintaan jaringan, *caching*, mengembalikan sumber daya aplikasi, dari *cache* sebelumnya. Beberapa asset web pun bisa disimpan sebagai *local cache* pada *service worker* sehingga dengan keadaan internet yang kurang baik pun, pengguna masih dapat mendapatkan merasakan pengalaman pengguna aplikasi yang baik. Berikut merupakan kode program dari *service worker*, seperti yang terlihat pada Kode Program 1.

```
if(!self.define){
  const e=e>{"require"!==e&&(e+="js");

  let r=Promise.resolve();
  return i[e]||(r=new Promise(async r=>{
    if("document"in self){
      const
      i=document.createElement("script");
      i.src=e,document.head.appendChild(i),
      i.onload=r}else importScripts(e,r())
    },

    r.then(())=>{
      if(!i[e])throw new Error(`Module ${e}
      didn't register its module`);

      return i[e]}},r=(r,i)=>{
        Promise.all(r.map(e)).then(e=>
        i(l===e.length?e[0]:e)),
        i={require:Promise.resolve(r)};

        self.define=(r,s,t)=>[i[r]|| (i[r]=Promise.resolve
        e()).then(())=>let i={};
        const o={uri:location.origin+r.slice(1)};
        return
        Promise.all(s.map(r=>{switch(r){case"exports":re
        turn i;
          case"module":return o;

          default:return e(r)}})).then(e=>{const
        r=t(...e);
          return i.default||(i.default=r),i)
        })
      }

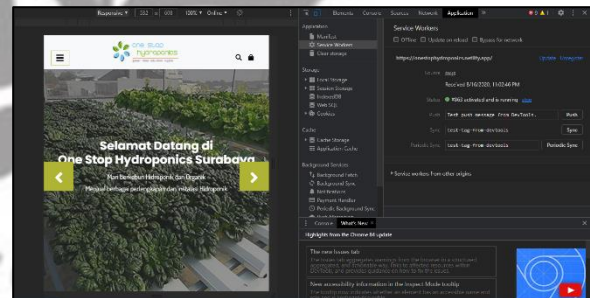
      define("./sw.js",["./workbox-
      59ddc918"],(function(e){
        "use
        strict";self.addEventListener("message",
        e=>{
          e.data&&"SKIP_WAITING"===e.data.type&&se
          lf.skipWaiting()
        }
      ),
      e.precacheAndRoute([
        {url:"about.html",
          revision:"4db4663722f65df6eb439eebb80ee7
          de"},{url:"cart.html",
```

```
revision:"c0672ef1eb2d5bd357049020894a56
c0"},{url:"checkout.html",
  revision:"ce2a67d8e129a51ef27f17ed10dc09
f2"},{url:"contact-us.html",
  revision:"e2c3d6d1e25cec91350b205309e41d
8d"},{url:"gallery.html",
  revision:"e8982ef4717d84fefe01128716cc32
50"},{url:"index.html",
  revision:"bfb4da3e0f5f95967215fcb88f9919
0d"},{url:"manifest.json",
  revision:"6f02617c1dfdd726e4151e6332a1b1
3e"},{url:"my-account.html",
  revision:"0362eb3d7f40e17a8854a493abe7f0
8b"},{url:"shop-detail.html",
  revision:"df8f2ae8424bb5ebc28d2360ec1773
ae"},{url:"shop.html",
  revision:"31538048e9d5a0a72e96bbe68e6c84
83"},{url:"wishlist.html",
  revision:"2427ed70a416c43fbf5d9fdca127fb
20"}],{}),

  e.registerRoute(/\.?(png|jpg|jpeg|gif|b
mp|webp|svg|ico)$/ ,
  new e.NetworkFirst({
    cacheName:"images",
    plugins:[new
e.ExpirationPlugin({
  maxEntries:10,purgeOnQuotaError:!0
})
})
}),"GET")
});
```

Kode Program 1 Kode Program Service Worker

Setelah selesai membuat *service worker* dan berhasil memasang pada website yang telah dibuat, penulis melakukan pengecekan pada website melalui *browser* google chrome yang melalui *developer tools*.



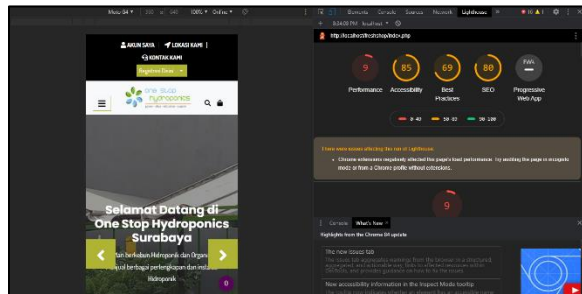
Gambar 13 Tampilan Service Worker bekerja pada website

Pada Gambar 13 terlihat ikon berwarna hijau dan bertuliskan "activated and is running", yaitu menandakan bahwa *service worker* sudah berhasil bekerja dan aktif pada website tersebut.

Dilakukannya pengujian yaitu bertujuan agar dapat mengukur tingkat keberhasilan dan keefisienan aplikasi yang telah dibangun. Metode pengujian yang digunakan untuk mengukur nilai aplikasi yaitu menggunakan *lighthouse* yang sudah tersedia didalam *developer tools* pada *browser* google chrome. Dari hasil pengujian yang didapatkan oleh *Lighthouse* kita bisa menilai dan menyimpulkan apakah web apps kita bangun sudah cukup optimal atau belum optimal, jika



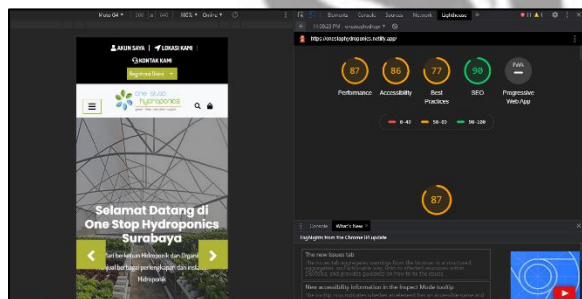
salah satu poin penilaian web belum optimal, maka nanti akan terlihat tanda penilaian berwarna merah. Berikut adalah hasil dari pengujian dengan *Lighthouse* sebelum menjadi PWA.



Gambar 14 Hasil pengujian sebelum menjadi PWA

Pada gambar 14 menunjukkan bahwa hasil pengujian pertama sebelum menjadi pwa dapat dibilang website masih belum optimal karena pada *performance* bahkan menunjukkan warna merah.

Setelah ditambahkan *Service Worker* dan *App Manifest* pada aplikasi website, dan dilakukan pengujian lagi, maka hasil pengujian menggunakan *lighthouse* menjadi seperti gambar berikut.



Gambar 15 Hasil pengujian sesudah menjadi PWA

Pada Gambar 15 menunjukan bahwa setelah aplikasi ditambahkan *Service Worker* dan *App Manifest* dan website menjadi PWA, hasil pengujian menjadi cukup optimal dan tidak ada tanda merah. Dan pengguna bisa merasakan pengalaman yang berbeda dengan penggunaan website lainnya, karena pengguna tetap bisa mengakses walaupun dengan keadaan jaringan internet yang sedang buruk.

### Berjalan Pada Jaringan Internet yang Buruk

Salah satu kelebihan yang dimiliki PWA yaitu aplikasi ini tetap dapat digunakan pada saat jaringan internet buruk bahkan dalam keadaan *offline*, tetapi dengan *term of condition* atau syarat kondisi yang harus dipenuhi terlebih dahulu, seperti *service worker* yang telah aktif dan berjalan sebelumnya, karena pada tahap ini aplikasi akan memanfaatkan

*service worker* sebagai tempat penyimpanan data-data yang pernah masuk kedalam perangkat sebelumnya. Sehingga, Ketika kita sebagai pengguna membuka halaman pada saat keadaan jaringan kurang baik atau bahkan *offline*, *service worker* dapat menampilkan halaman berdasarkan data terakhir yang telah diakses oleh kita sebagai pengguna, data tersebut disimpan oleh *service worker* dalam bentuk *cache*. Maka jika jaringan internet kurang baik atau keadaan sedang *offline* maka kita akan tetap dapat mengakses aplikasi web seperti dibawah ini.

Pada aplikasi ini memiliki beberapa fitur utama seperti halaman beranda, halaman tentang kami, halaman belanja, halaman galeri, dan halaman kontak kami. Untuk mengetahui aplikasi dapat berjalan ketika internet buruk atau *offline* maka penulis melakukan uji dengan mengakses beberapa fitur utama pada aplikasi terlebih dahulu seperti beranda, tentang kami, dan kontak kami, setelah itu penulis keluar aplikasi dan mematikan internet, setelah itu kembali mengakses aplikasi.



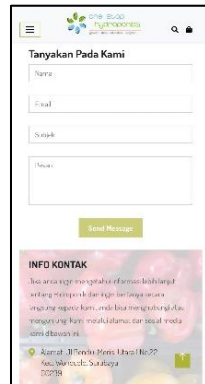
Gambar 16

Pada Gambar 16 adalah tampilan dari halaman tentang kami ketika diakses di dalam ponsel pada keadaan *offline* tetapi sudah pernah dilakukan akses sebelumnya dan tanpa membersihkan *cache* aplikasi.



Gambar 17

Pada Gambar 17 merupakan tampilan dari halaman beranda, ponsel pada keadaan *offline* tetapi sudah pernah dilakukan akses sebelumnya dan tanpa membersihkan cache aplikasi.



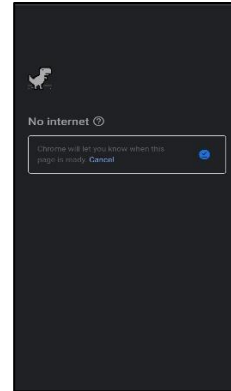
Gambar 18

Dan gambar 18 merupakan tampilan halaman kontak kami. Ketiga halaman itu tetap dapat diakses ketika keadaan offline karena *service worker* melakukan *load* ulang *cache* yang ada sebelumnya, sehingga halaman tersebut dapat diakses.



Gambar 19

Sedangkan jika penulis mengakses halaman belanja hasil nya menunjukan seperti pada Gambar 19, karena sebelumnya pada saat *online* penulis tidak mengakses terlebih dahulu halaman belanja tesebut, jadi *service worker* tidak menyimpan *cache* sehingga tidak dapat melakukan *load* kembali.



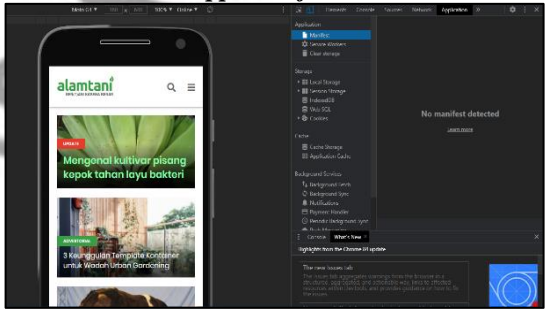
Gambar 20

sedangkan jika sebuah website *Non-PWA* maka akan menunjukan langsung menunjukan tampilan seperti Gambar 20 dan tidak bisa menampilkan apapun sampai kembali *online*. Akan tetapi jika pengguna melakukan pembersihan *cache* dengan menggunakan aplikasi *cleaner* pada ponsel, dan *cache* ikut terhapus maka aplikasi tidak dapat menampilkan halaman dari data terakhir yang pernah diakses. Yang berarti, pengguna harus online terlebih dahulu agar bisa mengakses aplikasi.

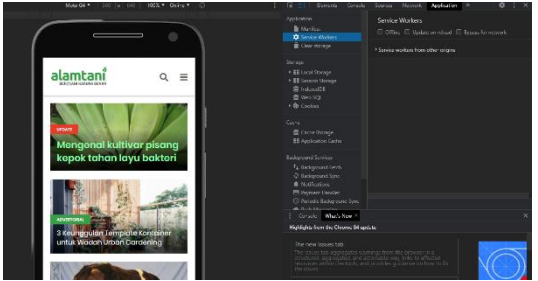

### Perbedaan PWA dan Website Non PWA


Pada tahap ini penulis melakukan perbandingan dengan website non pwa, yaitu website sejenis yang berfokus pada bidang pertanian yaitu website *alamtani.com*, dengan hasil pengujian pada Tabel 1 sebagai berikut.

Tabel 1 Tabel Perbandingan PWA dan Website Non PWA

PWA	Non PWA
<b>1. Perbedaan App Manifest</b> 	
<p>Terdapat <i>App Manifest</i> yang terdeteksi pada <i>developer tools browser chrome</i> untuk mengatur tampilan ikon, nama aplikasi, warna, <i>splash screen</i>.</p>	<p><i>App Manifest</i> tidak ada dan tidak akan terdeteksi pada <i>developer tools browser chrome</i>, seperti pada Gambar 21</p>

Gambar 21

<p><b>2. Perbedaan <i>Service Worker</i></b></p>  <p style="text-align: center;">Gambar 22</p>	
Memiliki <i>Service Worker</i> untuk menyimpan cache website yang bisa dilihat pada <i>developer tools browser chrome</i> .	<i>Service Worker</i> tidak akan terdeteksi pada <i>developer tools browser chrome</i> , seperti pada Gambar 22.
<p><b>3. Perbedaan Ikon Aplikasi</b></p>  <p style="text-align: center;">Gambar 23</p>	
Ketika ditambahkan kelayar ponsel atau di <i>install</i> , aplikasi akan memiliki gambar <i>Ikon Aplikasi</i> .	Ketika ditambahkan ke layar, tetapi website tidak akan memiliki gambar ikon aplikasi seperti pada Gambar 23.
<p><b>4. Perbedaan <i>Splash Screen</i></b></p>  <p style="text-align: center;">Gambar 24</p>	
Ketika di akses akan menampilkan tampilan	Ketika diakses akan langsung menuju ke browser tanpa melalui

<p><i>Splash Screen</i>, seperti pada Gambar 24.</p>	<p>halaman splash screen terlebih dahulu, karena tidak memiliki <i>splashscreen</i>.</p>
<p><b>5. Perbedaan Tampilan Akses</b></p>  <p style="text-align: center;">Gambar 25</p>	
Ketika di akses akan masuk ke dalam bentuk aplikasi PWA.	Ketika diakses tidak akan berbentuk seperti aplikasi, hanya seperti website yang diakses pada <i>browser</i> seperti biasa, seperti yang terlihat pada Gambar 25.

### Halaman Beranda Website Official Store

Pada halaman beranda ini penulis akan menunjukan perbedaan tampilan beranda pada saat menggunakan tampilan web *browser* normal dan pada saat diakses pada saat tampilan *web app* pada layar ponsel.



Gambar 26 Tampilan halaman beranda

Pada Gambar 26 menunjukan tampilan halaman beranda pada saat menggunakan *browser* normal, pada halaman ini menampilkan fitur-fitur seperti, beranda, tentang kami, belanja, galeri, dan kontak kami.



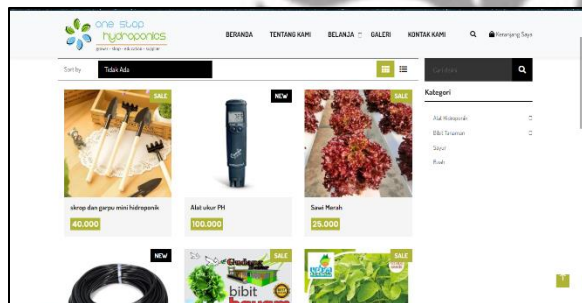


Gambar 27

Pada Gambar 27 menunjukkan tampilan halaman beranda pada saat menggunakan ponsel dan dalam bentuk PWA.

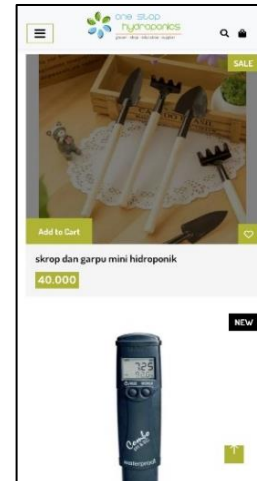
### Halaman Penjualan Website Official Store

Pada halaman penjualan ini penulis juga akan menunjukan perbedaan tampilan pada saat menggunakan tampilan web *browser* normal, dan juga pada saat menggunakan tampilan *web app* pada layar ponsel.



Gambar 28 tampilan halaman penjualan

Pada Gambar 28, tampilan halaman ini terdapat juga beberapa kategori produk yang dijual oleh toko One Stop Hydroponics Surabaya, seperti peralatan hidroponik, bibit tanaman, sayur segar, dan juga buah segar hasil produksi dari pertanian hidroponik.



Gambar 29

Pada Gambar 29 menunjukkan tampilan halaman penjualan pada saat menggunakan ponsel dan dalam bentuk PWA.

### PENUTUP Kesimpulan

Berdasarkan hasil penelitian perancangan dan pengujian aplikasi yang telah dilakukan pada Rancang Bangun Aplikasi Official Store berbasis *Progressive Web App* ini maka dapat penulis disimpulkan bahwa sebagai berikut :

1. Aplikasi website official store dapat menerapkan prinsip dan metode **PWA (Progressive Web Apps)** itu sendiri, yaitu seperti memiliki **Service Worker** yang dapat membuat website berjalan pada saat internet sedang buruk dengan melakukan *load data cache* sebelumnya, tidak memunculkan pesan *network error*, dan juga membuat **Manifest** aplikasi web sehingga menghilangkan tahapan proses download aplikasi normal pada umumnya, sehingga langsung bisa dipasang pada layar beranda ponsel dan juga website PWA ini memiliki **Icon** aplikasi dan **Splashscreen** sehingga aplikasi web official store ini memberikan pengalaman pengguna seperti menggunakan aplikasi *mobile native* pada pengguna.
2. Dengan adanya aplikasi *official store* berbasis *Progressive Web Apps* ini dapat membantu toko One Stop Hydroponics Surabaya dalam penyampaian informasi produk terbaru dari toko serta dapat membantu dalam penjualan toko.



## Saran

1. Aplikasi ini masih perlu dikembangkan dan perlu ditambahkan beberapa fitur lagi yang dapat lebih memudahkan penjual dan juga pelanggan.
2. Perancangan pada aplikasi ini masih sangat sederhana, sehingga perlu untuk melakukan penelitian dan pengembangan lebih lanjut agar aplikasi ini menjadi lebih baik dan lebih efisien.

## DAFTAR PUSTAKA

- D Wirdasai (2009), "Teknologi E-Commerce Dalam Proses Bisnis." *Jurnal SAINTIKOM*, vol.7, pp. 324-335.
- Lurensius Adi (2017), "Platform E-Learning untuk Pembelajaran Pemrograman Web Menggunakan Konsep Progressive Web Apps." *JURNAL TEKNIK ITS Vol.6 no.2*, pp.A579 – A583, 2017.
- Indrajani, (2015). Database Design (Case Study All in One). Jakarta: PT. Elex Media Komputindo.
- M. Shalahuddin dan Rosa A.S, (2014). Buku karangan Rosa A.S dan M. Shalahuddin "UML (*Unified Modelling Language*)".
- Huda, M. & Bunafit Komputer, (2010). Membuat Aplikasi Database dengan Java, MySQL, dan NetBeans. 1st ed. Jakarta: Elex Media Komputindo.
- Alex Handy (2011). "Node.js pushes JavaScript to the server-side"
- Sheppard, D. 2017. *Beginning Progressive Web App Development: Creating a Native App Experience on the Web*. hlm 3.
- Osmani, A. 2015. *Getting Started with Progressive Web Apps*.  
URI=<https://developers.google.com/web/updates/2015/12/getting-started-pwa>.
- IdCloudHouse. 2017. Mengenal Apa itu Progressive Web App.URI=<https://idcloudhost.com/mengenal-apa-itu-progressive-web-app-pwa/>.
- GoogleDeveloper. 2018. Lighthouse.  
URI=<https://developers.google.com/web/tools/lighthouse/>.
- Alan B. Sternecker (2003)*Critical Incident Management*