

LAPORAN PROYEK

KOM 205/ Basis Data

Orplant

Dipersiapkan oleh:

Ananda Thalia (G74170048)
Irsyad M. Andre (G74170077)

Departemen Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Pertanian Bogor

2015

Departemen Ilmu Komputer Fakultas Matematika dan IPA Institut Pertanian Bogor	Nomor Dokumen		Halaman
	<i>BASDAT – P5</i>		<i>1/12</i>
	<i>Revisi</i>		<i>Tgl: 21/06/20</i>

Daftar Isi

1 Ringkasan	2
2 Pendahuluan	2
2.1 Latar Belakang	2
2.2 Rumusan Masalah	2
2.3 Tujuan	3
2.4 Batasan Masalah	3
2.5 Solusi Singkat	3
3 Skema Basis Data	3
3.1 Entity-Relational Diagram	3
3.2 Diagram Skematik	3
4 Implementasi	3
4.1 Proses Implementasi	3
4.2 Hasil Implementasi	3
5 Pembagian Kerja dalam Kelompok	3
6 Lampiran	4
6.1 Log Activity Anggota Kelompok	4
6.2 Dokumentasi Studi Lapangan	4
6.3 <Lain-Lain>	4

1 Ringkasan

Berawal dari keresahan komunitas tanaman hias yang belum mempunyai wadah sebagai platform khusus jual-beli dan saling bertukar informasi mengenai tanaman hias. Kami menghadirkan Orplant. Orplant adalah platform tanaman hias yang menjadi wadah para pencinta tanaman hias untuk melakukan transaksi jual-beli melalui fitur Store dan berbagi informasi mengenai tanaman hias melalui fitur Artikel.

2 Pendahuluan

2.1 Latar Belakang

Masyarakat masih merasa kesulitan dalam mencari informasi mengenai tanaman hias seperti cara merawat tanaman secara tepat dan mencari lokasi untuk membelinya. Komunitas tanaman hias juga belum memiliki wadah khusus untuk melakukan transaksi, selama ini mereka hanya bertukar informasi melalui media sosial.

Era globalisasi memudahkan manusia untuk mengakses informasi melalui internet. Berbagai macam informasi dapat diperoleh, salah satunya mengenai tanaman hias. Tanaman hias memiliki daya tarik tersendiri bagi siapa saja yang melihatnya. Tanaman hias menjadi salah satu hobi yang diminati masyarakat Indonesia, sehingga terbentuknya komunitas tanaman hias di Indonesia semakin banyak. Platform ini akan memudahkan bagi orang-orang untuk melakukan transaksi jual beli tanaman hias yang diinginkan tanpa khawatir untuk mencari lokasi dan harga tanaman hias yang akan dibeli, serta menjadi pusat berbagi informasi mengenai tanaman hias.

2.2 Rumusan Masalah

Rumusan masalah website Orplant adalah sebagai berikut:

- Bagaimana membuat suatu wadah khusus untuk transaksi bagi para pecinta tanaman hias ataupun yang membutuhkan?
- Bagaimana membuat suatu wadah khusus bagi para pecinta tanaman hias ataupun yang membutuhkan untuk saling bertukar informasi?

2.3 Tujuan

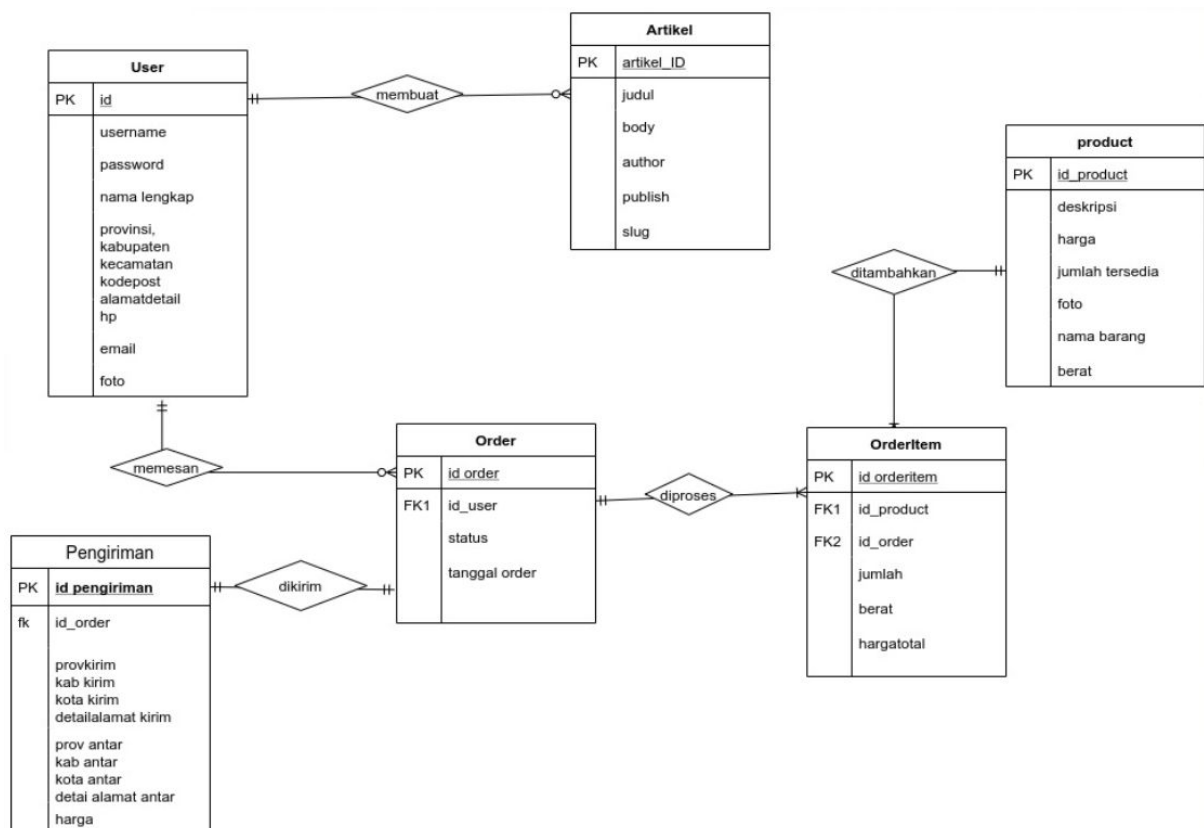
Website ini bertujuan untuk memudahkan masyarakat pada umumnya dan komunitas pecinta tanaman hias pada khususnya untuk menjangkau sarana transaksi tanaman hias lebih mudah dan menjadi sarana pengetahuan terkait tanaman hias.

2.4 Solusi Singkat

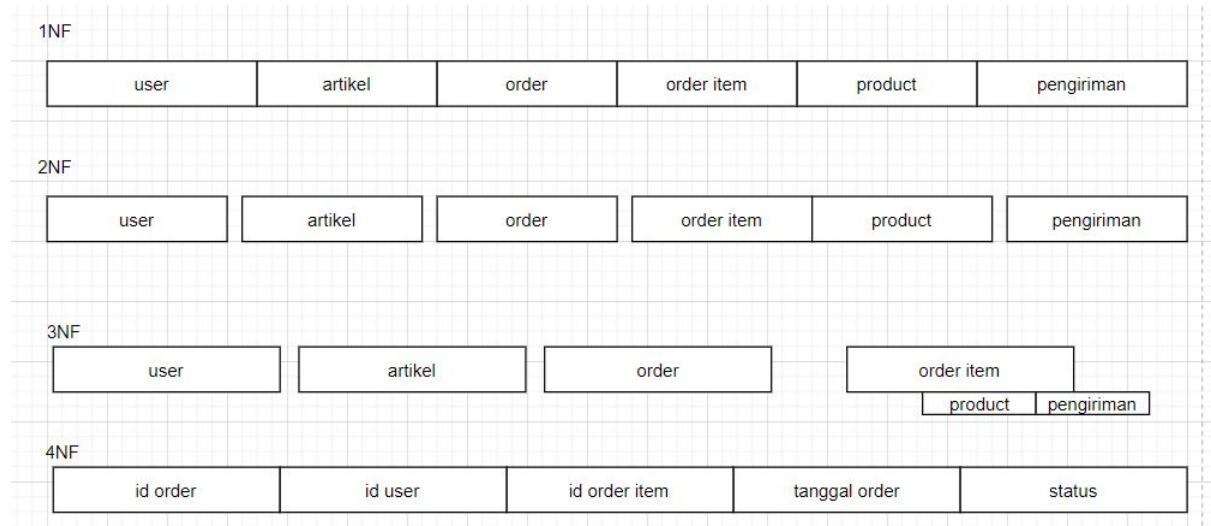
Kami menghadirkan Orplant sebagai solusi platform khusus tanaman hias untuk memudahkan masyarakat pada umumnya dan komunitas pecinta tanaman hias pada khususnya untuk menjangkau sarana transaksi tanaman hias lebih mudah dan menjadi sarana pengetahuan terkait tanaman hias.

3 Skema Basis Data

3.1 Entity-Relational Diagram



3.2 Diagram Skematik



1nf Normalisasi ini digunakan untuk menghilangkan null.

2nf dilakukan dengan memisahkan antar kolom dengan tujuan mempermudah identifikasi kolom.

3nf adalah proses penghilangan informasi yang redundan atau tidak penting sehingga data lebih bersih.

4 nf untuk mempercepat proses transaksi, terdapat primary key asli dari transaksi dan sisanya foreign key untuk akses masing-masing datanya.

4 Implementasi

4.1 Proses Implementasi

1. Membuat home orplant
2. Membuat user dengan registrasi form
3. Membuat user autentifikasi (login/logout)
4. Membuat Fitur Artikel
5. Membuat tampilan list barang dari toko (store, cart belanja, checkout belanja)

4.2 Hasil Implementasi

- 1 Membuat home orplant

The screenshot shows the Visual Studio Code interface with two files open: `index.html` and `base.html`. The `index.html` file is the active editor, displaying HTML code for a home page. The code includes a header with a navigation bar, a main content area with a welcome message and a list of features, and a footer with social media links. The `base.html` file is also visible, showing a common header and footer structure. The Explorer sidebar on the left shows the project structure, including a `static` folder for images and a `templates` folder for HTML files. The status bar at the bottom indicates the current line and column (Ln 21, Col 25).

```
1  {% extends "base.html" %}
2  {% load static %}
3
4
5
6  {% block content %}
7  <div class="jumbotron-fluid">
8  <div style="color: rgb(27, 66, 27); font-family: 'Times New Roman', Times, serif;">
9  <h1>Selamat Datang di Orplant! </h1>
10 <img style="height:50px; src='{% static "img/Logo_Orplant.png" %}' alt="">
11 <h2>About Us</h2>
12 <p>Orplant adalah platform tanaman hias yang menjadi wadah para pencinta tanaman
13 <hr class="my-4">
14 <img style="height:50px; src='{% static "img/Logo_Orplant.png" %}' alt="">
15 <h3>Contact Us</h3>
16 <div style="margin-left: 10px;">
17 
18 </div>
19 <div style="margin-left: 10px;">
20 
21 </div>
22 <div style="margin-left: 10px;">
23 
24 </div>
25 </div>
26
27 </div>
28
29 </div>
30
31 </div>
32
33 </div>
34
35 </div>
36
37 </div>
38
39 </div>
40 </div>
```



Selamat Datang di Orplant admin !



About Us



Selamat Datang di Orplant admin !



About Us

Orplant adalah platform tanaman hias yang menjadi wadah para pencinta tanaman hias untuk melakukan transaksi jual-beli (Store) dan berbagi informasi mengenai tanaman hias (Artikel).



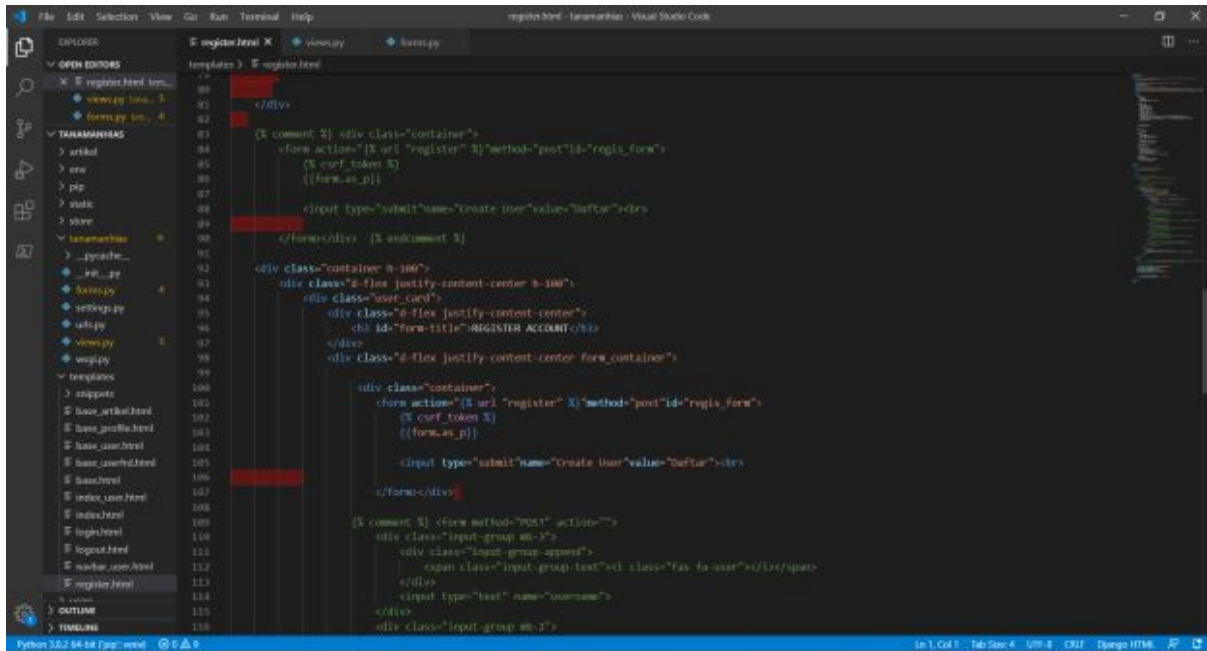
Contact

 Orplant

 @Orplant

 Orplant

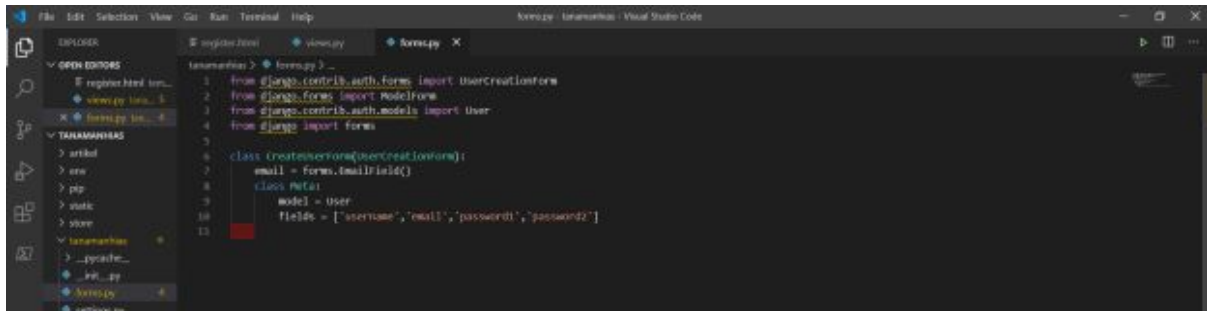
2. Membuat user registrasi



This screenshot shows the Visual Studio Code editor with the `register.html` file open. The file contains HTML code for a registration form. The code includes a container for the form, a form action pointing to the `register` endpoint, and input fields for email, password, and username. The form is styled with Bootstrap classes.

```
<!-- register.html -->
<div class="container">
  <form action="{% url 'register' %}" method="post" id="regis_form">
    <csrf_token %>
    {{form.as_p}}
    <input type="submit" name="create user" value="daftar"></div>
</form></div> <!-- endcomment -->

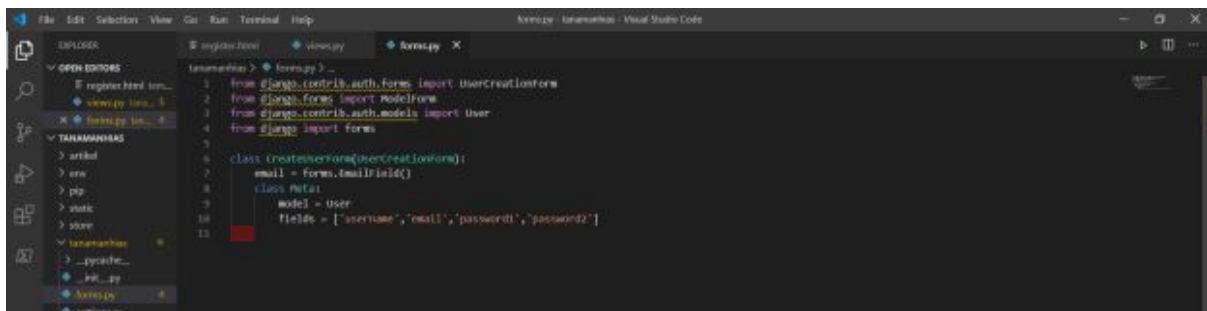
<div class="container h-100">
  <div class="d-flex justify-content-center h-100">
    <div class="d-flex justify-content-center">
      <div class="card">
        <div class="card-body">
          <div class="d-flex justify-content-center">
            <div class="card">
              <div class="card-body">
                <div class="d-flex justify-content-center">
                  <div class="container">
                    <form action="{% url 'register' %}" method="post" id="regis_form">
                      <csrf_token %>
                      {{form.as_p}}
                      <input type="submit" name="Create User" value="daftar"></div>
                    </form></div>
                    <div class="input-group">
                      <div class="input-group-append">
                        <div class="input-group-text"><div class="fas fa-user"></div></div>
                      <div>
                        <input type="text" name="username">
                      </div>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```



This screenshot shows the Visual Studio Code editor with the `forms.py` file open. The file contains a Django form class named `createuserform` that inherits from `UserCreationForm`. The form includes fields for email, password, and username.

```
from django.contrib.auth.forms import UserCreationForm
from django.forms import ModelForm
from django.contrib.auth.models import User
from django import forms

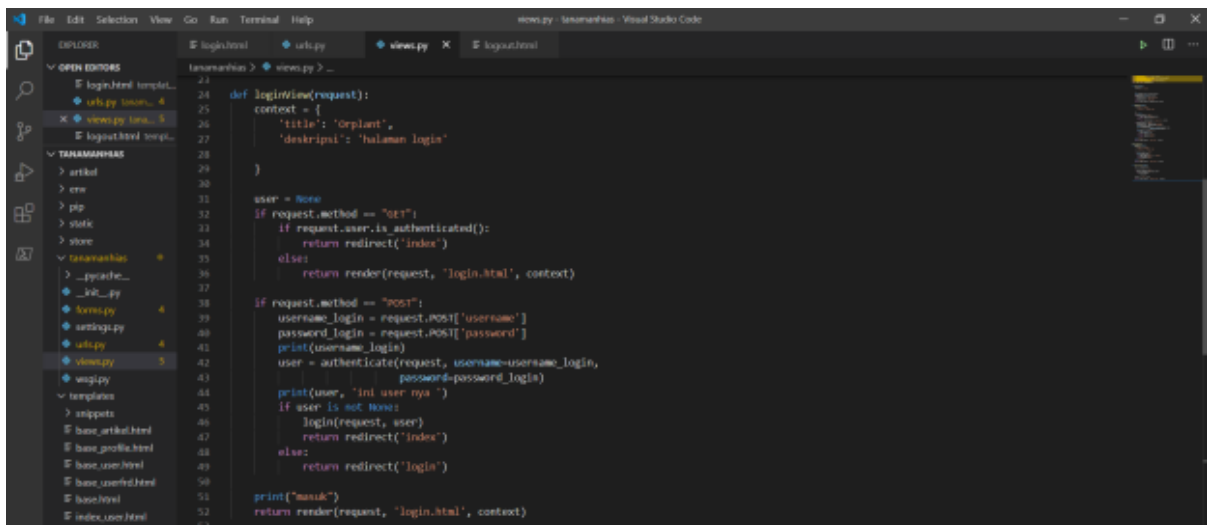
class createuserform(UserCreationForm):
    email = forms.EmailField()
    class Meta:
        model = User
        fields = ('username', 'email', 'password1', 'password2')
```



This screenshot shows the Visual Studio Code editor with the `forms.py` file open. The file contains a Django form class named `createuserform` that inherits from `UserCreationForm`. The form includes fields for email, password, and username.

```
from django.contrib.auth.forms import UserCreationForm
from django.forms import ModelForm
from django.contrib.auth.models import User
from django import forms

class createuserform(UserCreationForm):
    email = forms.EmailField()
    class Meta:
        model = User
        fields = ('username', 'email', 'password1', 'password2')
```

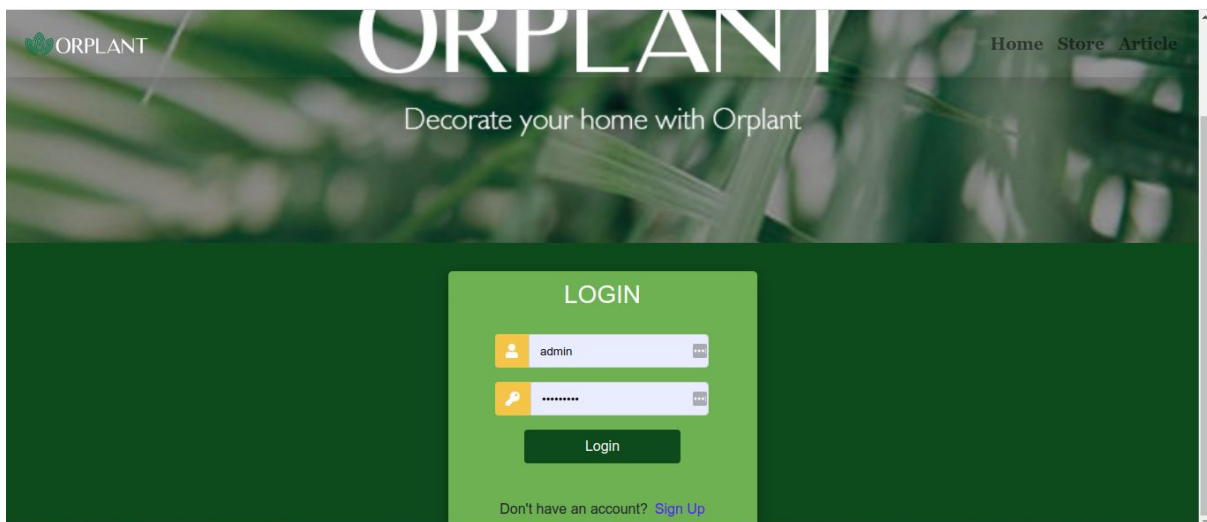



```
def login(request):
    context = {
        'title': 'Orplant',
        'deskripsi': 'halaman login'
    }

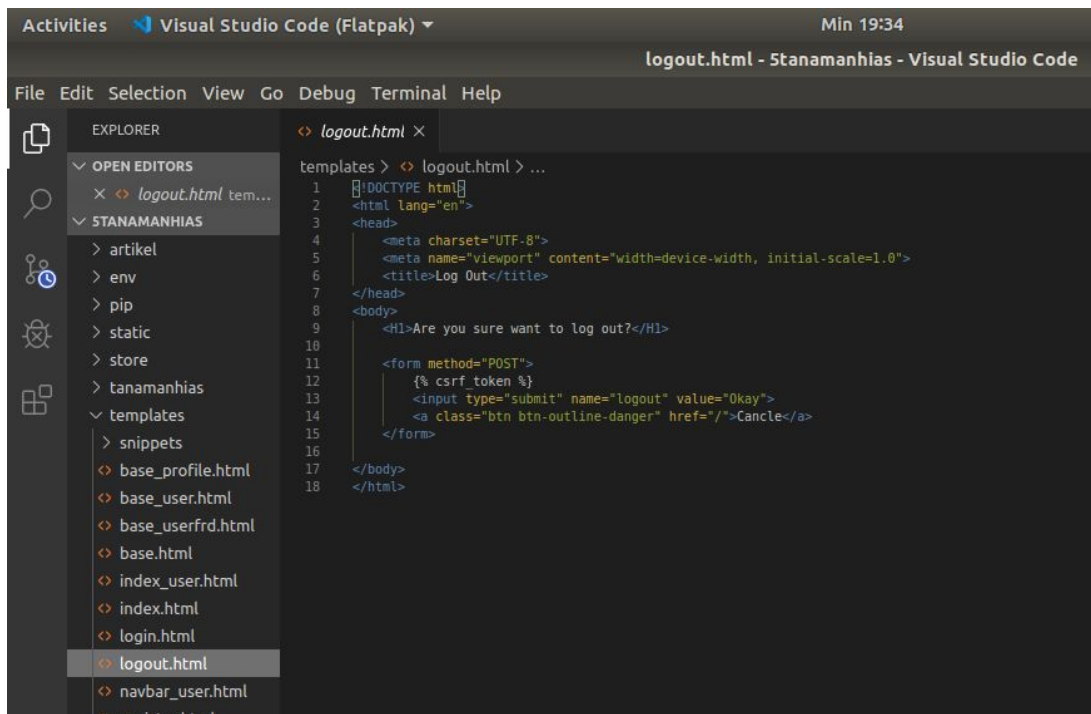
    user = None
    if request.method == "GET":
        if request.user.is_authenticated():
            return redirect("index")
        else:
            return render(request, "login.html", context)

    if request.method == "POST":
        username_login = request.POST['username']
        password_login = request.POST['password']
        print(username_login)
        user = authenticate(request, username=username_login,
                             password=password_login)
        print(user, "ini user nya ")
        if user is not None:
            login(request, user)
            return redirect("index")
        else:
            return redirect("login")

    print("masuk")
    return render(request, "login.html", context)
```



Logout



Visual Studio Code (Flatpak) interface showing the `logout.html` file in the `templates` folder. The code is as follows:

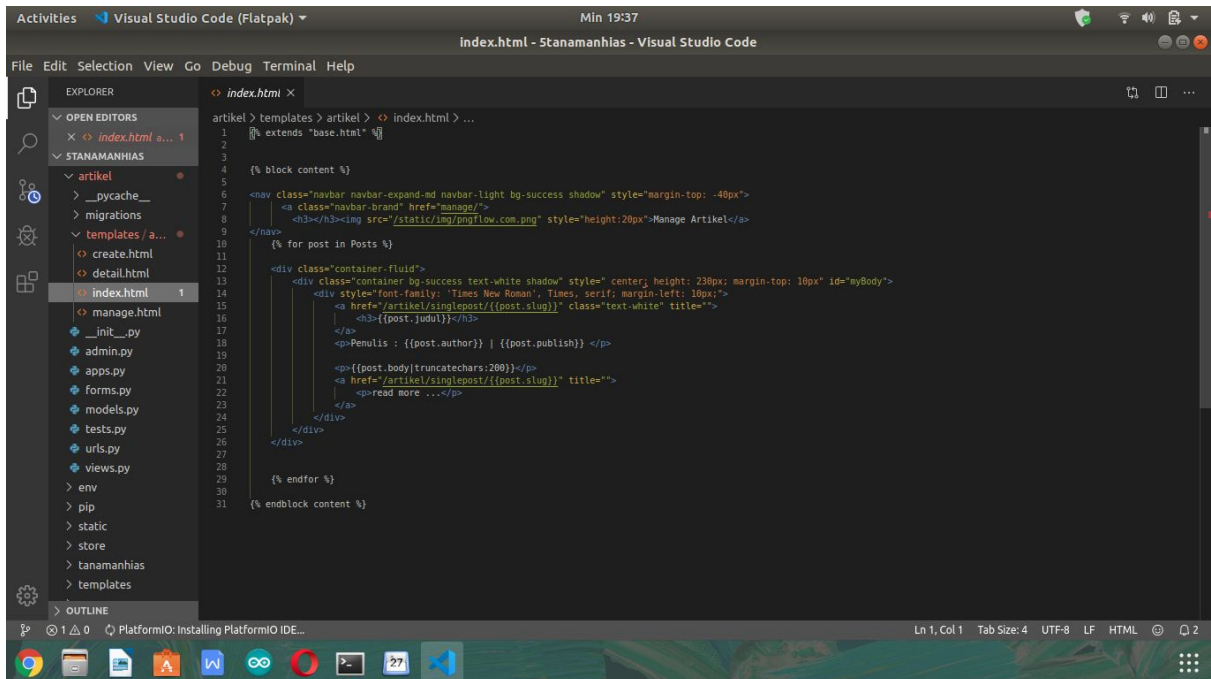
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Log Out</title>
7 </head>
8 <body>
9   <h1>Are you sure want to log out?</h1>
10
11   <form method="POST">
12     { csrf token %}
13     <input type="submit" name="logout" value="Okay">
14     <a class="btn btn-outline-danger" href="/">Cancel</a>
15   </form>
16
17 </body>
18 </html>
```

Are you sure want to log out?

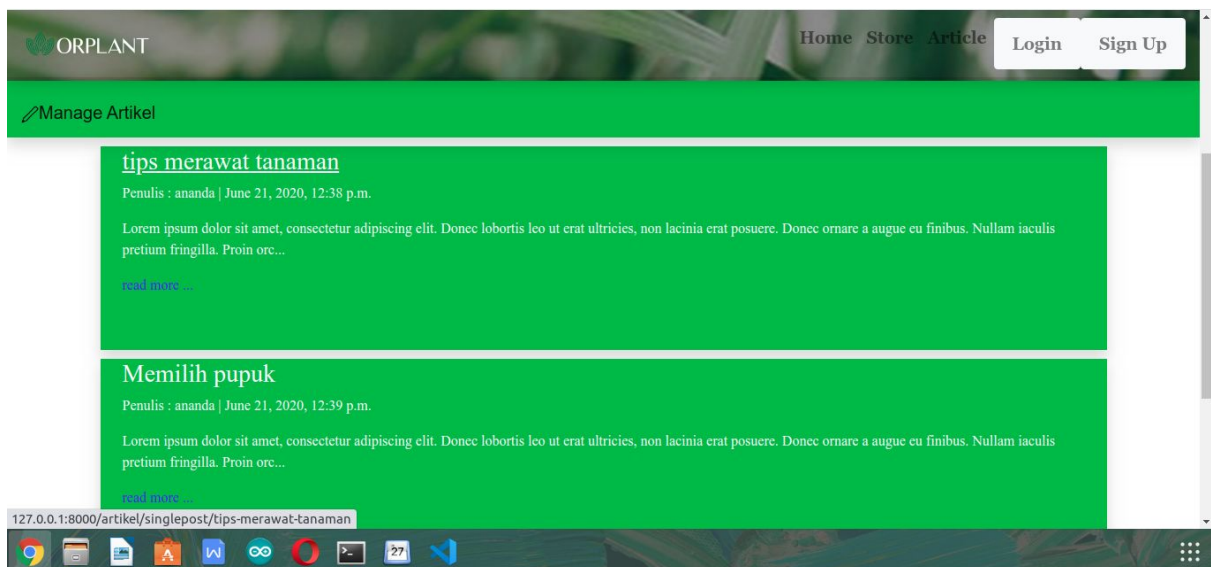
[Cancel](#)

4.artikel

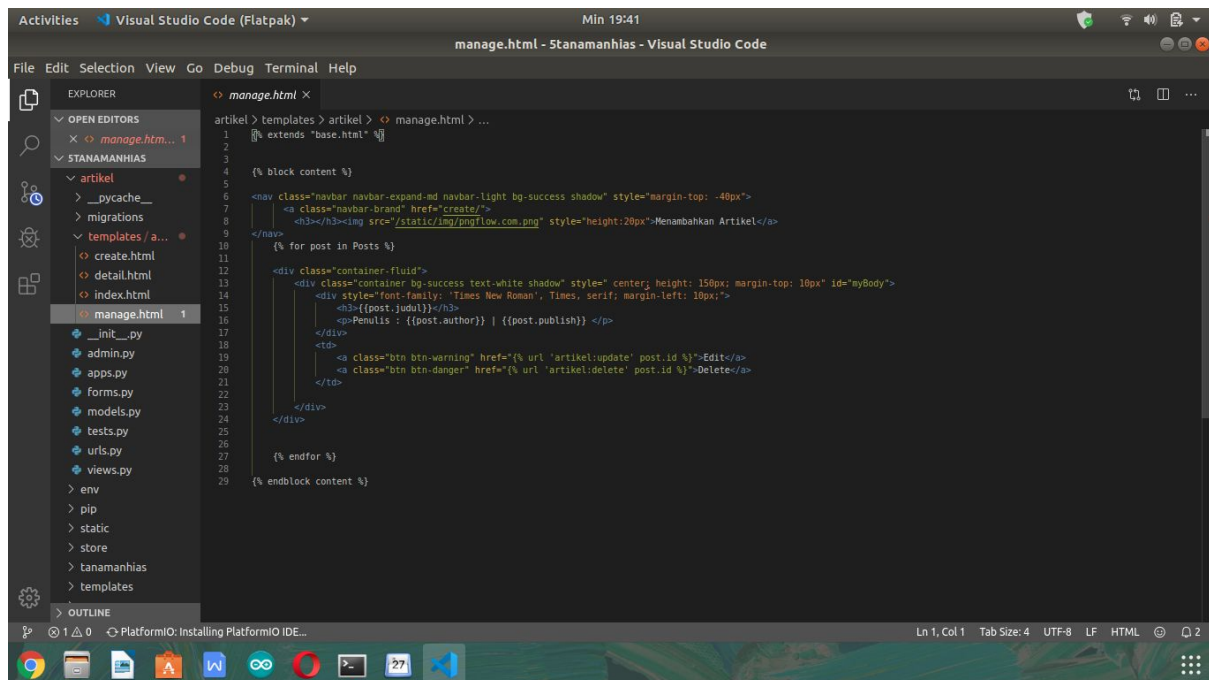
laman depan artikel



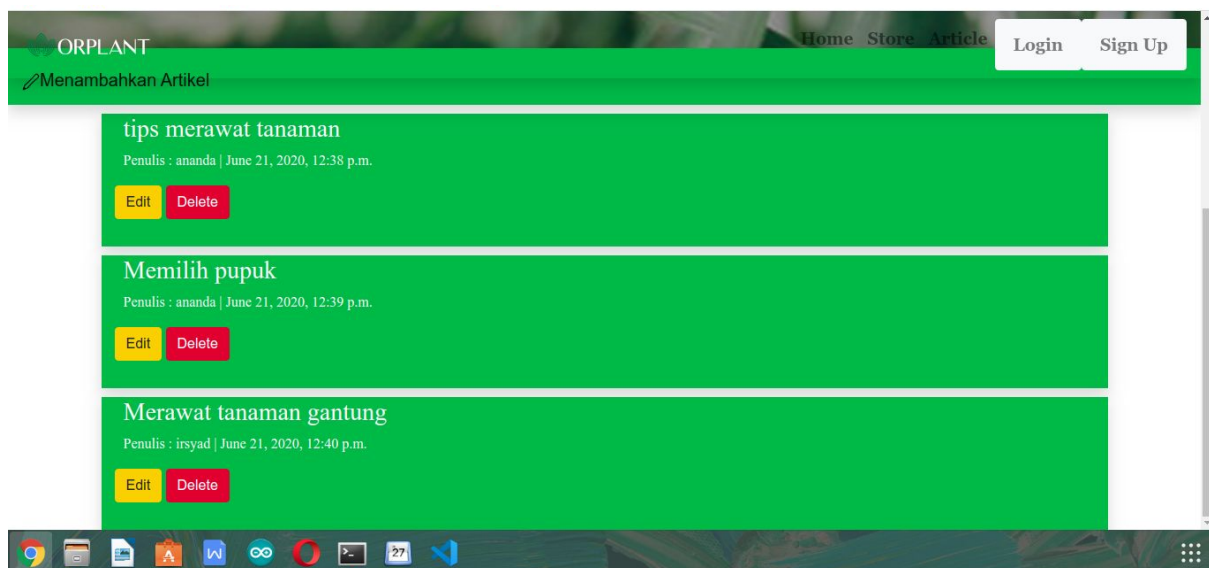
```
1  {% extends "base.html" %}
2
3
4  {% block content %}
5
6  <nav class="navbar navbar-expand-md navbar-light bg-success shadow" style="margin-top: -40px">
7    <a class="navbar-brand" href="manage/">
8    <h3></h3>Manage Artikel</a>
9  </nav>
10
11  {% for post in Posts %}
12
13  <div class="container-fluid">
14    <div class="container bg-success text-white shadow" style="center; height: 230px; margin-top: 10px; margin-left: 10px; id="myBody">
15      <a href="/artikel/singlepost/{{post.slug}}" class="text-white" title="">
16        <h3>{{post.judul}}</h3>
17      </a>
18      <p>Penulis : {{post.author}} | {{post.publish}}</p>
19      <p>{{post.body|truncatechars:200}}</p>
20      <a href="/artikel/singlepost/{{post.slug}}" title="">
21        <p>read more ...</p>
22      </a>
23    </div>
24  </div>
25
26  {% endfor %}
27
28  {% endblock content %}
```



Laman Manage artikel

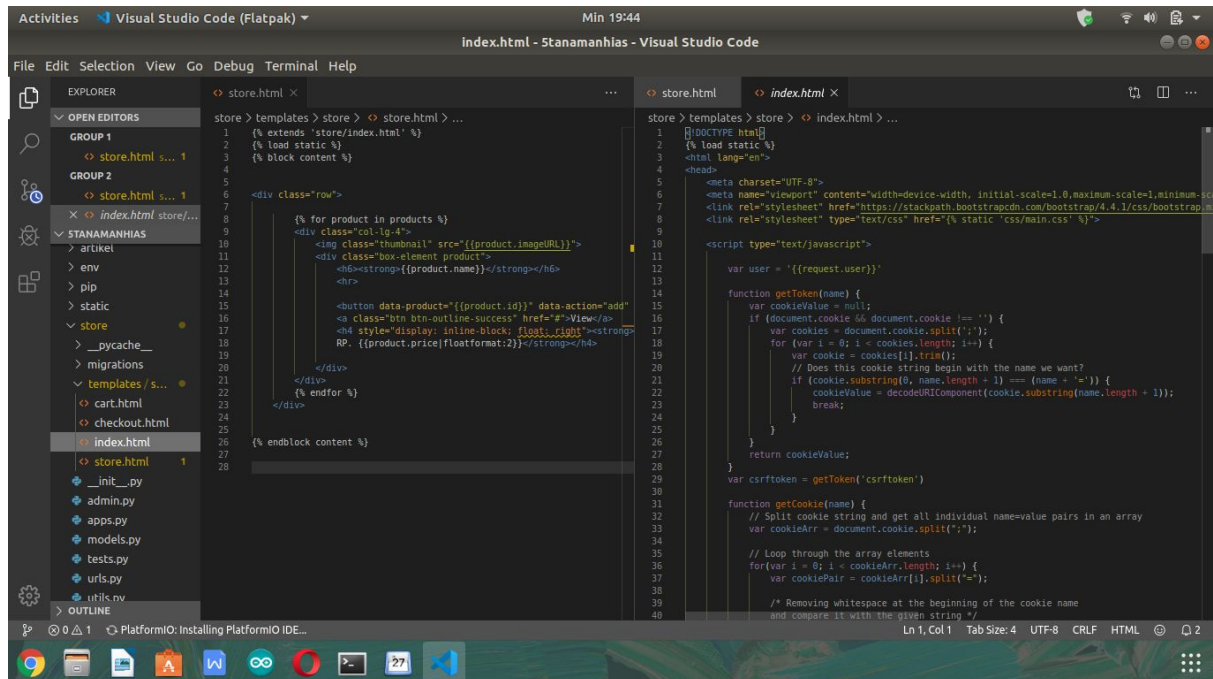


```
1 {% extends "base.html" %}
2
3
4 {% block content %}
5
6 <nav class="navbar navbar-expand-md navbar-light bg-success shadow" style="margin-top: -40px">
7   <a class="navbar-brand" href="/create/">
8     <h3></h3>Menambahkan Artikel</a>
9 </nav>
10 {% for post in Posts %}
11
12 <div class="container-fluid">
13   <div style="font-family: 'Times New Roman', Times, serif; margin-left: 10px;">
14     <h3>{{post.judul}}</h3>
15     <p>Penulis : {{post.author}} | {{post.publish}} </p>
16   </div>
17   <div>
18     <a class="btn btn-warning" href="{% url 'artikel:update' post.id %}">Edit</a>
19     <a class="btn btn-danger" href="{% url 'artikel:delete' post.id %}">Delete</a>
20   </div>
21 </div>
22 {% endfor %}
23
24 {% endblock content %}
```



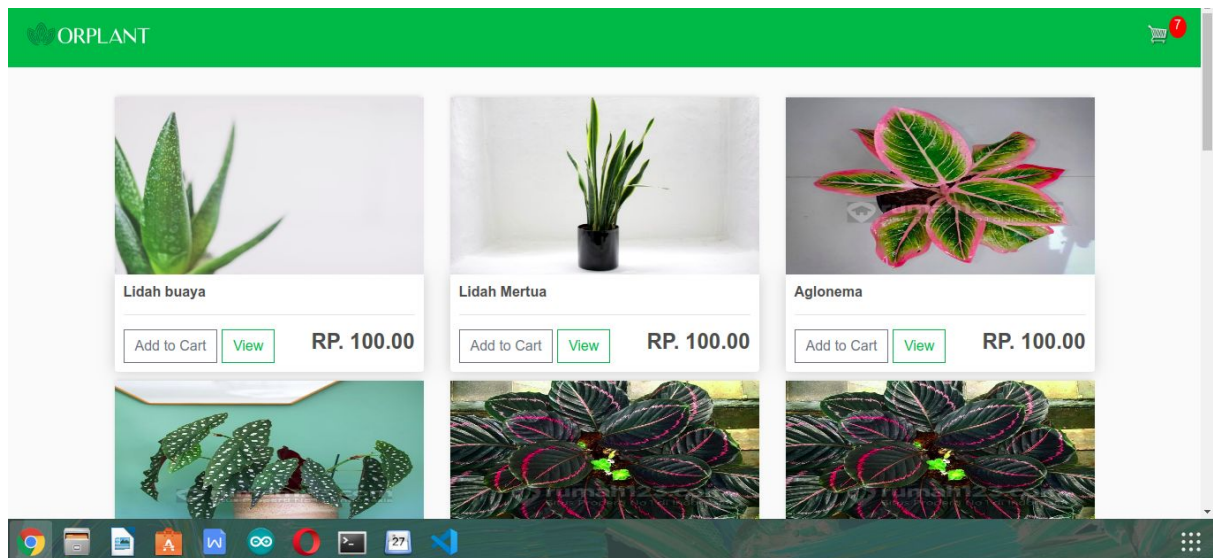
5. Orplant Store(store, cart, checkout)

Laman depan store



```
store.html
1  {% extends 'store/index.html' %}
2  {% load static %}
3  {% block content %}
4
5
6  <div class="row">
7
8      {% for product in products %}
9
10         <div class="col-lg-4">
11             
12             <div class="box-element product">
13                 <h6><strong>{{product.name}}</strong></h6>
14                 <hr>
15                 <button data-product="{{product.id}}" data-action="add"
16                     <a class="btn btn-outline-success" href="#">View</a>
17                 <h4 style="display: inline-block; float: right;">
18                     <strong>{{product.price|floatformat:2}}</strong></h4>
19             </div>
20         </div>
21     {% endfor %}
22 </div>
23
24 {% endblock content %}
25
26
27
28

index.html
1  <!DOCTYPE html>
2  {% load static %}
3  <html lang="en">
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1, minimum-scale=1">
7      <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css">
8      <link rel="stylesheet" type="text/css" href="{static 'css/main.css' %}">
9
10     <script type="text/javascript">
11         var user = '{{request.user}}'
12
13         function getToken(name) {
14             var cookieValue = null;
15             if (document.cookie % document.cookie != '') {
16                 var cookies = document.cookie.split(';');
17                 for (var i = 0; i < cookies.length; i++) {
18                     var cookie = cookies[i].trim();
19                     // Does this cookie string begin with the name we want?
20                     if (cookie.substr(0, name.length + 1) === (name + '=')) {
21                         cookieValue = decodeURIComponent(cookie.substr(name.length + 1));
22                         break;
23                     }
24                 }
25             }
26             return cookieValue;
27         }
28         var csrftoken = getToken('csrftoken')
29
30         function getCookie(name) {
31             // Split cookie string and get all individual name-value pairs in an array
32             var cookieArr = document.cookie.split(';');
33
34             // Loop through the array elements
35             for (var i = 0; i < cookieArr.length; i++) {
36                 var cookiePair = cookieArr[i].split('=');
37                 // Removing whitespace at the beginning of the cookie name
38                 // and compare it with the given string */
39             }
40         }
```



tampilan cart

Activities Visual Studio Code (Flatpak) Min 19:48

cart.html - Stanamanhias - Visual Studio Code

File Edit Selection View Go Debug Terminal Help

EXPLORER

store.html cart.html

store > templates > store > cart.html > ...

```
1 {% extends 'store/index.html' %}
2 {% load static %}
3 {% block content %}
4
5 <div class="row">
6
7     <div class="box-element">
8
9         <a class="btn btn-outline-dark" href="{% url 'store' %}">Continue Shopping</a>
10
11     <br>
12     <br>
13     <table class="table">
14
15         <tr>
16             <th>Items: <strong>{{order.get_cart_items}}</strong></th>
17             <th>Total: <strong>${(order.get_cart_total|floatformat:2)}</strong></th>
18         </tr>
19         <tr>
20             <td>
21                 <a style="float:right; margin:5px;" class="btn btn-success" href="{% url 'checkout' %}">Checkout</a>
22             </td>
23         </tr>
24     </table>
25
26 </div>
27
28 <div class="box-element">
29     <div class="cart-row">
30         <div style="flex:2"></div>
31         <div style="flex:2"><strong>Item</strong></div>
32         <div style="flex:1"><strong>Price</strong></div>
33         <div style="flex:1"><strong>Quantity</strong></div>
34         <div style="flex:1"><strong>Total</strong></div>
35     </div>
36     {% for item in items %}
37     <div class="cart-row">
38         <div style="flex:2"></div>
39         <div style="flex:2"><p>{{item.product.name}}</p></div>
40         <div style="flex:1"><p>${(item.product.price|floatformat:2)}</p></div>
41         <div style="flex:1">
42             <p class="quantity">{{item.quantity}}</p>
43         </div>
44     </div>
45 </div>
46 {% endblock content %}
```

PlatformIO: Installing PlatformIO IDE... Ln 1, Col 1 Tab Size: 4 UTF-8 CRLF HTML

Activities Visual Studio Code (Flatpak) Min 19:48

cart.html - Stanamanhias - Visual Studio Code

File Edit Selection View Go Debug Terminal Help

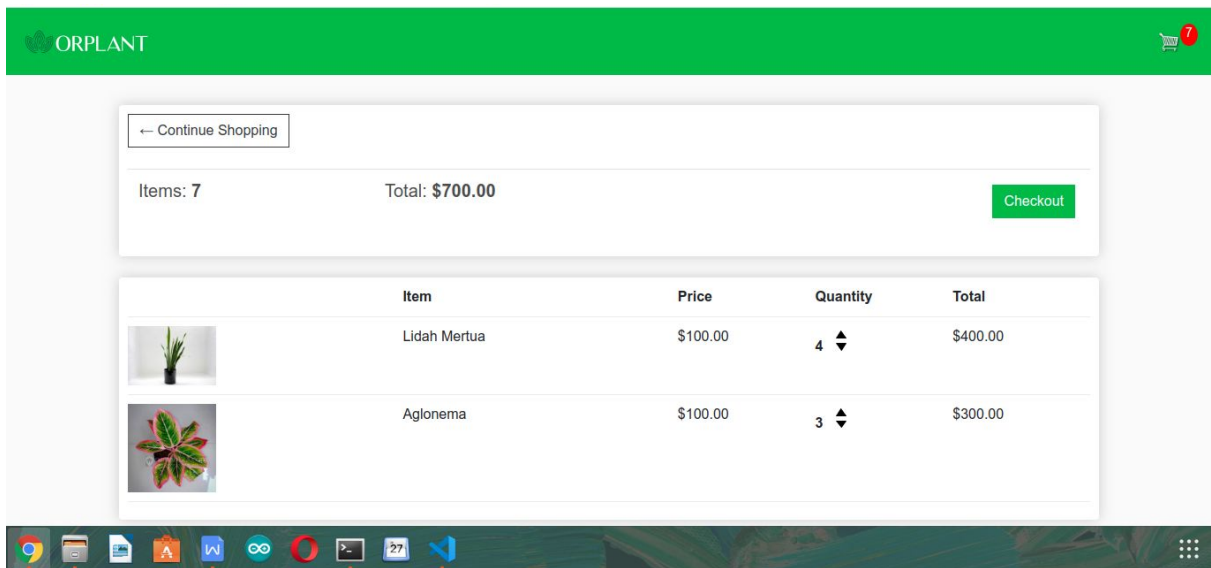
EXPLORER

store.html cart.html

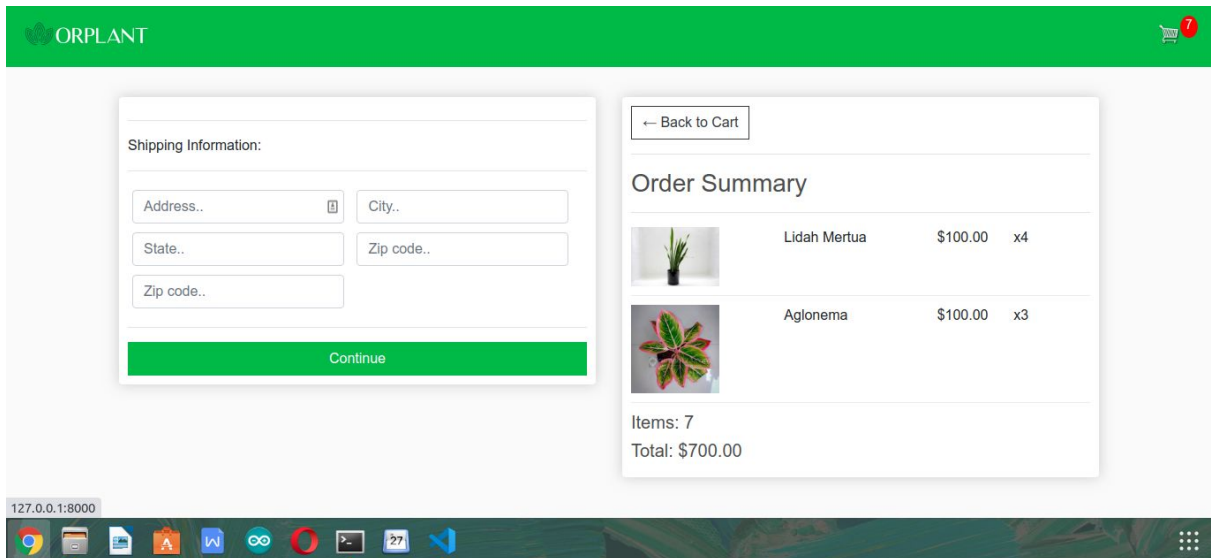
store > templates > store > cart.html > ...

```
15 <div class="row">
16     <div class="box-element">
17         <div style="flex:2"><strong>Item</strong></div>
18         <div style="flex:1"><strong>Price</strong></div>
19         <div style="flex:1"><strong>Quantity</strong></div>
20         <div style="flex:1"><strong>Total</strong></div>
21     </div>
22     {% for item in items %}
23     <div class="cart-row">
24         <div style="flex:2"></div>
25         <div style="flex:2"><p>{{item.product.name}}</p></div>
26         <div style="flex:1"><p>${(item.product.price|floatformat:2)}</p></div>
27         <div style="flex:1">
28             <p class="quantity">{{item.quantity}}</p>
29             <div class="quantity">
30                 
31                 
32             </div>
33         </div>
34     </div>
35     {% endfor %}
36 </div>
37 {% endblock content %}
```

PlatformIO: Installing PlatformIO IDE... Ln 1, Col 1 Tab Size: 4 UTF-8 CRLF HTML



Laman Check Out



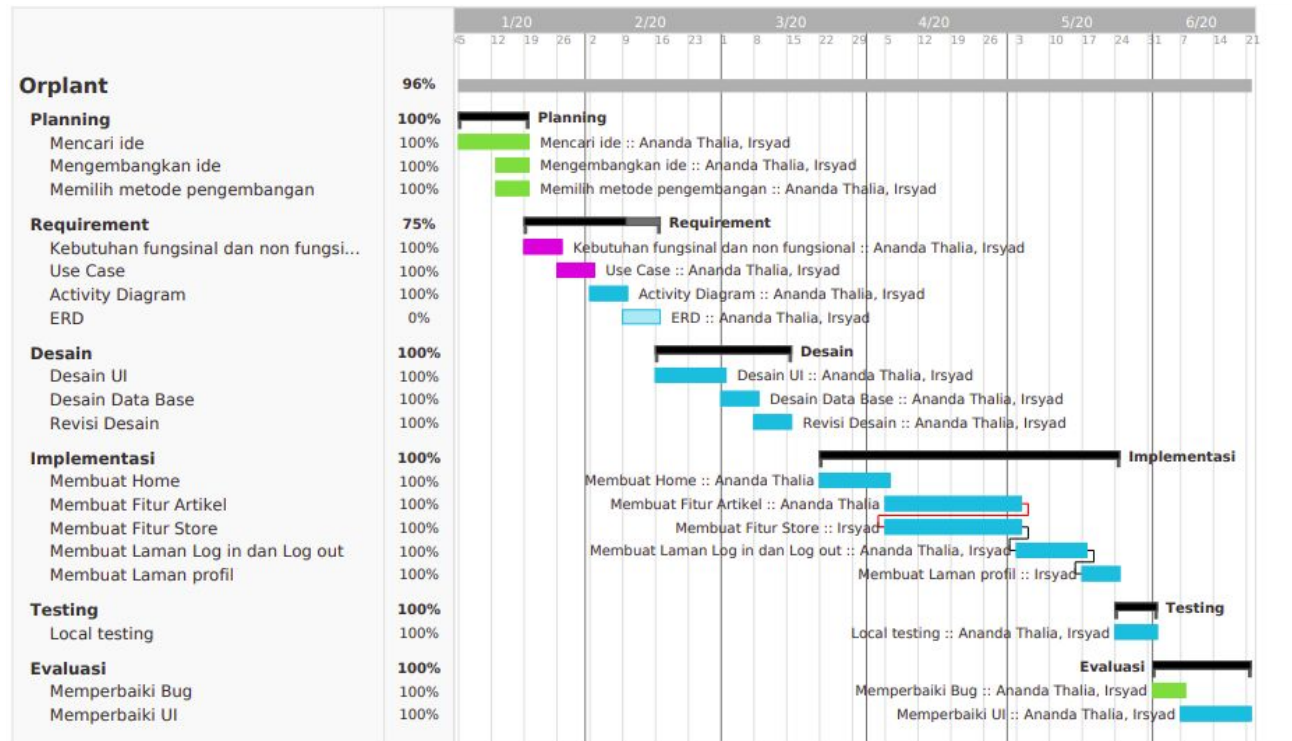
5 Pembagian Kerja dalam Kelompok

- Planning : Ananda Thalia, Irsyad M. Andre
- Desain : Ananda Thalia, Irsyad M. Andre
- Laman Home : Ananda Thalia
- Laman Sing in, Sing Out, dan Register : Ananda Thalia, Irsyad M. Andre
- Laman Profil : Ananda Thalia
- Fitur Artikel :Ananda Thalia
- Fitur Store : Ananda Thalia

6 Lampiran

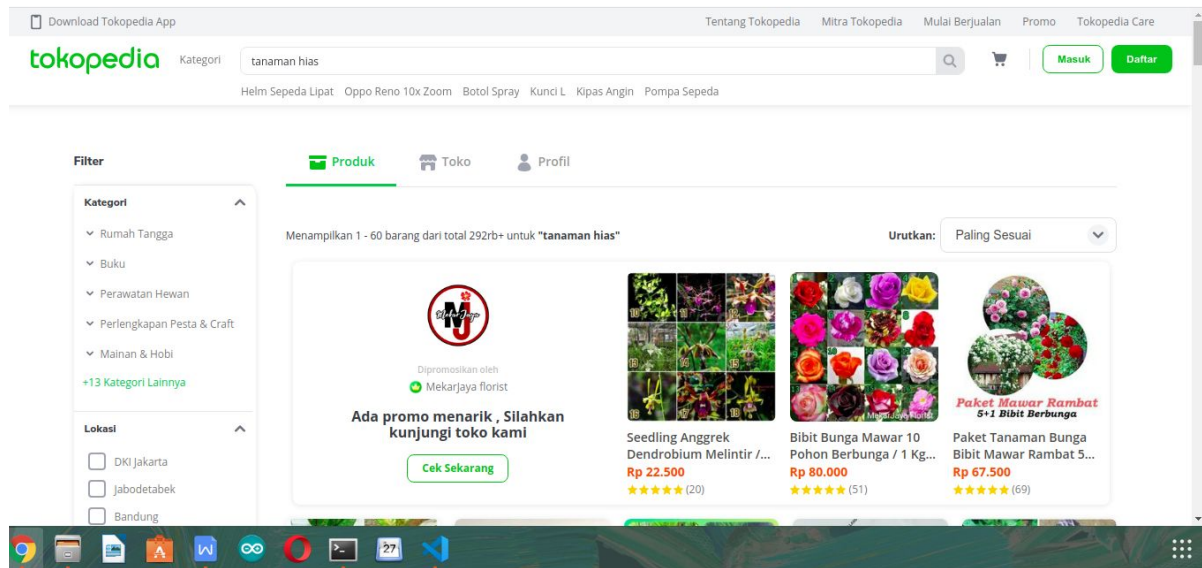
6.1 Log Activity Anggota Kelompok

Berikut adalah Gantt Chart yang memuat jobdesk setiap anggota dan timeline pengerjaannya:



6.2 Dokumentasi Studi Lapangan

Kami melakukan observasi pada media sosial dan e-commers:



Ananda
20+

Tanaman Hias

Grup Publik · 26,9 rb anggota

[Gabung Grup](#)

Tentang **Diskusi** Anggota Acara Media

Pengumuman Belum Dibaca · 9

Budi Sutomo
13 Desember 2019 ·

PENGUMUMAN: Untuk semua anggota grup, foto tanaman hias di album grup ini yang hasil foto saya. Silahkan jika perlu foto boleh di ambil dan digunakan untuk keperluan apa saja, tanpa perlu minta izin. Foto sengaja dibuat size besar dan tanpa watermark agar bisa digunakan. Foto sudah dikelompokan di album grup ini, per jenis tanaman hias. Silahkan di tengok album grupnya.

Bikin grup kita lebih menyenangkan dan bermanfaat. Caranya:
1. Posting foto tanaman minimal ada nama tan... **Lihat Selengkapnya**

Tentang

Grup ini diperuntukan bagi pecinta tanaman hias. Silahkan berbagi pengalaman berkebun atau memberikan informasi tentang seluk beluk tanaman hias... **Lihat Selengkapnya**

- Publik**
Semua orang bisa melihat siapa anggota grup ini dan apa yang diposting
- Terlihat**
Semua orang bisa menemukan grup ini.
- Jakarta**

Ananda
20+

Tanaman Hias

Kirim Pesan
Suka

4.016 orang mengikuti ini

[Kirim Pesan](#)

[Teras/Taman](#)

Foto [Lihat Semua](#)

Tanaman Hias
23 Mei pada 13.47 ·

Inilah agloanema kebanggaan bangsa Indonesia