

- **Procedure to Start Project:**

- Once you have downloaded pyshop.zip and extracted all of its contents.
- Next step is to open the extracted folder with IDE(we have used Visual studio code).
- Next install django using command `pip install django==2.1`(We have used django version 2.1, it will produce best result if you install the same version)

```
PS D:\Nirma\4th_Sem\Python\Assignment> pip install django==2.1
Collecting django==2.1
  Downloading Django-2.1-py3-none-any.whl (7.3 MB)
    |#####| 7.3 MB 19 kB/s
Requirement already satisfied: pytz in c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
django==2.1) (2020.5)
Installing collected packages: django
Successfully installed django-2.1
```

- Now check you are inside your extracted folder
- Next we have to run the command `python manage.py runserver` to launch our django server.

```
PS D:\Nirma\4th_Sem\Python\Assignment> python manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).
April 10, 2021 - 15:02:53
Django version 2.1, using settings 'pyshop.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

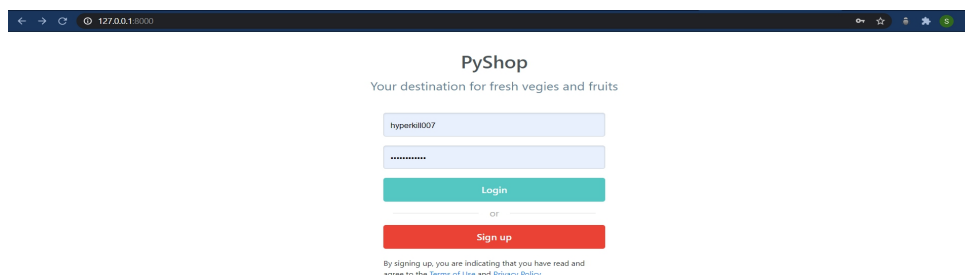
- Now click on the IDE generated server link(<http://127.0.0.1:8000/>) to go to the home page of our project.

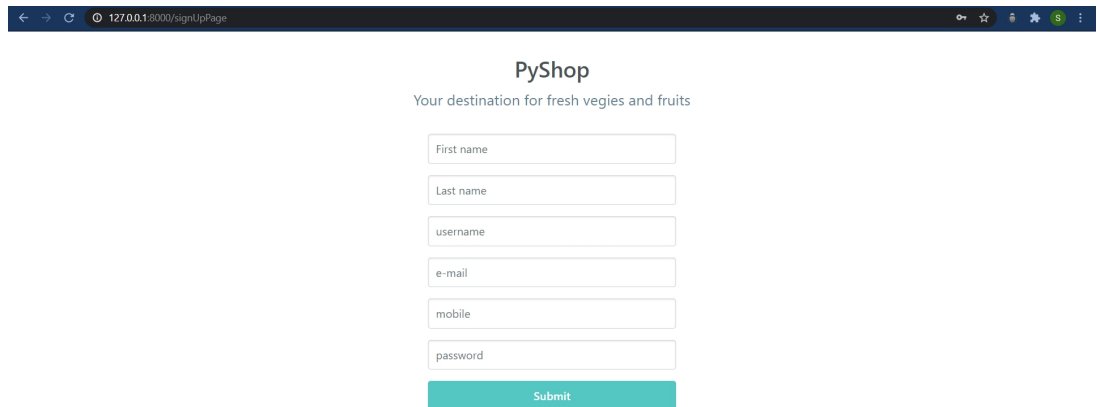
- **Features:**

We have 2 types of profiles: user and admin.

Let's first explore the user side of the project.

- The webpage you travelled on clicking on the link from IDE will take you to the User login/signup page where you can create new users.
- Login/Signup page → user can login into existing accounts or sign up for a new account. If you try to login with the wrong user it will display “User Not Found”. All the users are saved in the backend.





PyShop

Your destination for fresh vegies and fruits

First name

Last name

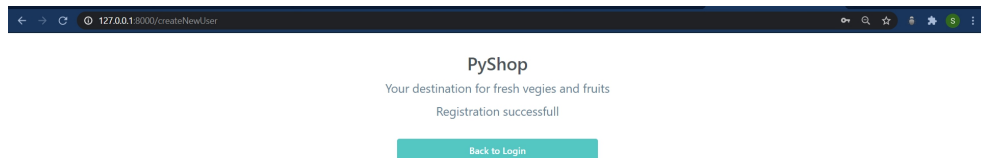
username

e-mail

mobile

password

Submit



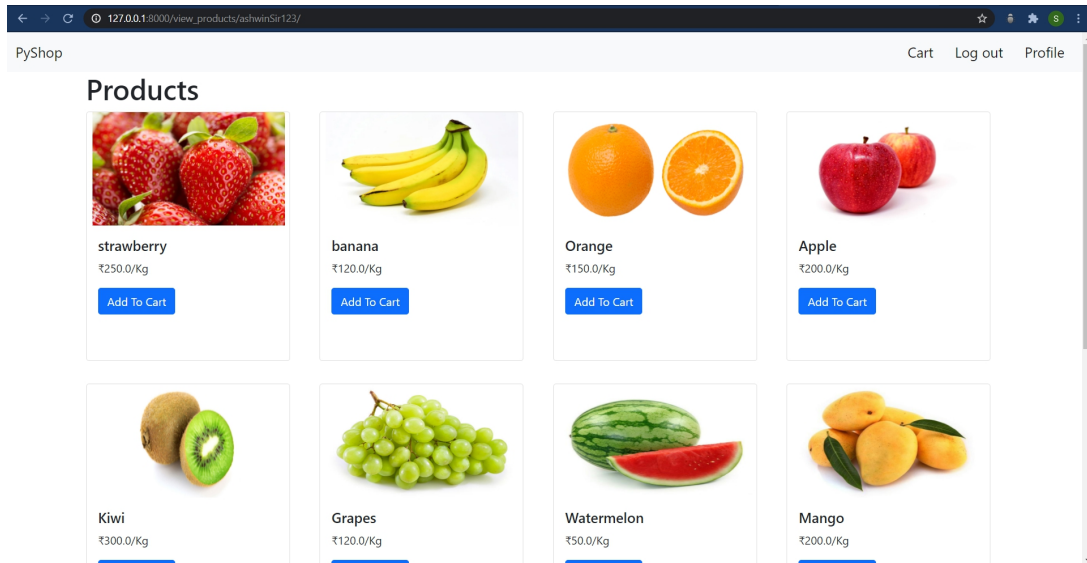
PyShop

Your destination for fresh vegies and fruits

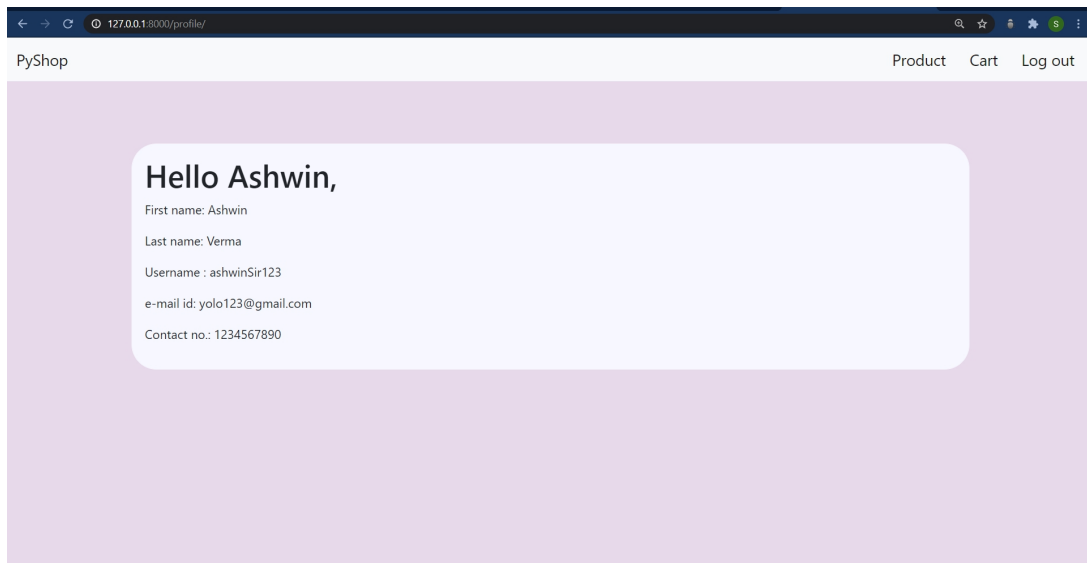
Registration successfull

Back to Login

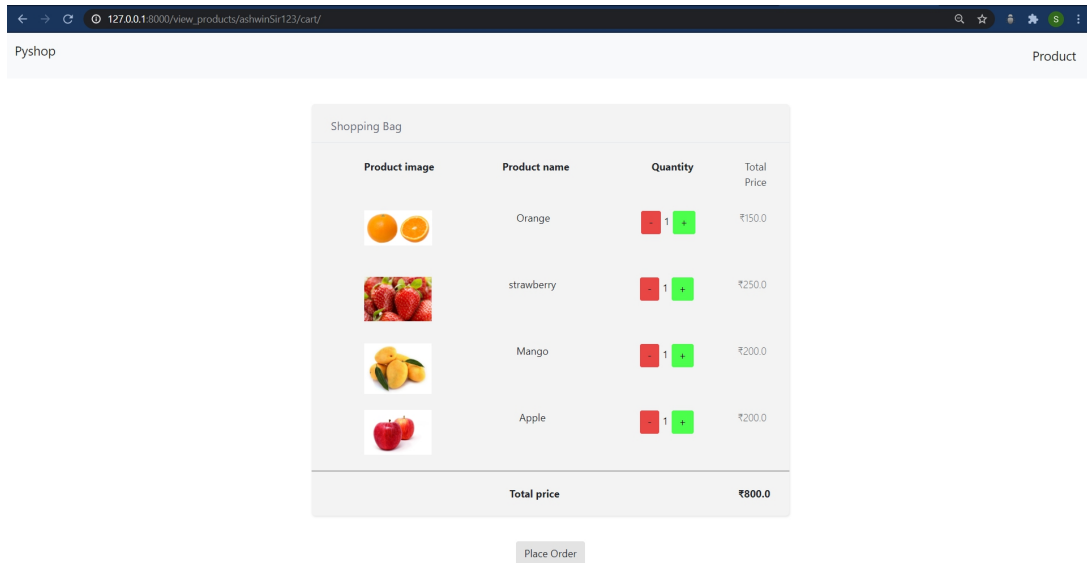
-
- As you move to any other page, you will see the difference in url also. You will see your login username in the url as well and if you tried to change the username you will be logged out.
 - Products(home) page → this is the first page you will see after logging in your account. It displays all the products offered by the website for sale. Users can select multiple products from the home page which will automatically be added to the user's cart. This page allows the user to navigate to 'cart', 'profile' and 'logout button'.



- Profile page → it will display the user's profile details. Users can navigate to the 'Product' page, 'Cart' page or log out from the account.



- Cart page → this page displays details of our ordered(selected) products. Users can add more products or remove some products. Once a user has decided on the order he/she can place it by clicking on the "Place Order" button. This site also lets users navigate back to the Products page by clicking Product button which is at right corner.



- Checkout window → on clicking the place order button of the cart user is sent to the checkout window. This window asks details required for delivering the products. All the fields are mandatory to be filled. Once the user has filled in all the details he/she can place the order by clicking on Submit button.

Enter Address for This Order

Address

Deliver to*
Aswin Verma

Apartment, unit, suite, or floor #
A-206, Subha Apartments

City*
Ahmedabad

State*
Gujarat

Postal code*
123456

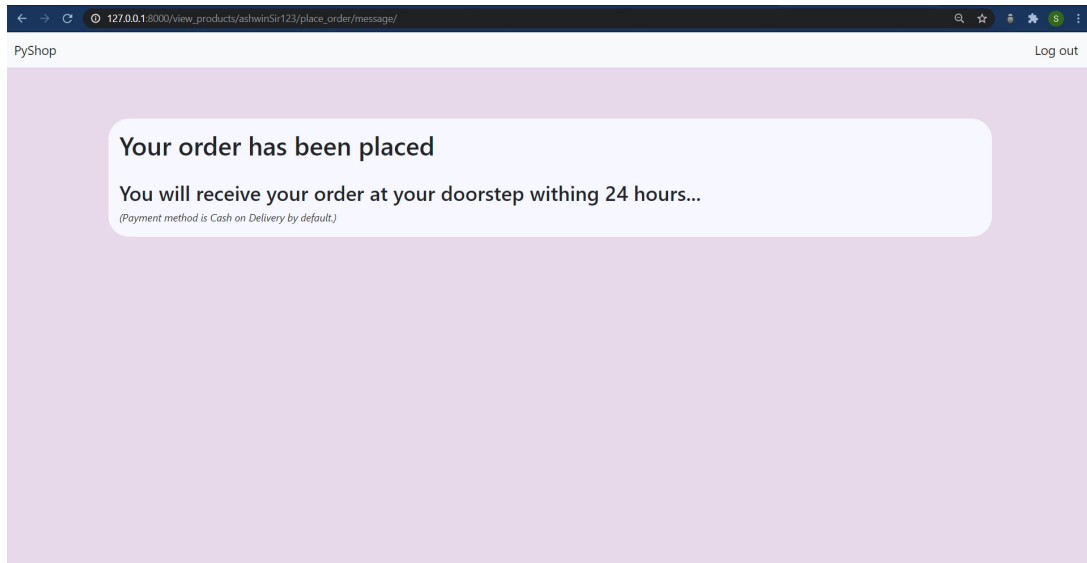
Country*
India

Submit

Place Order

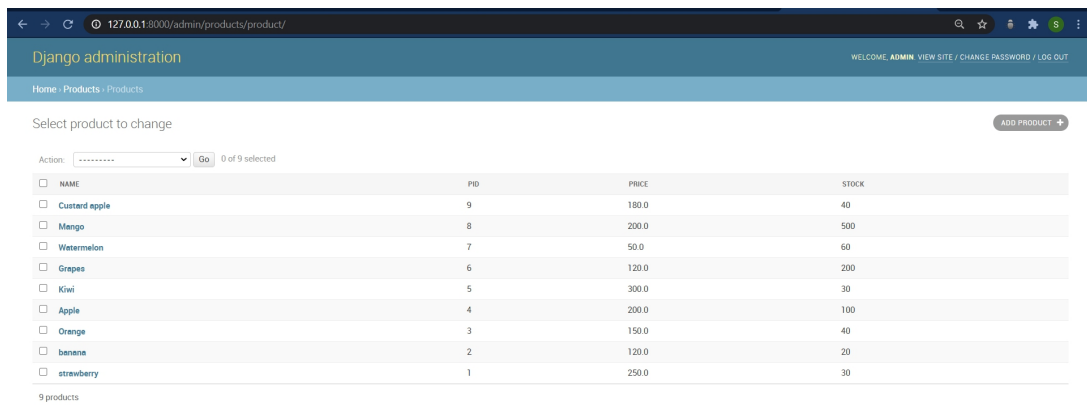
* = required field

- Message window → after placing an order website takes you to a webpage that contains an auto generated message that confirms your order. You can logout of your account by clicking on the logout button.



Next let's see the admin side.

- Admin side facilities are provided by django and for accessing it you need to copy the server link provided by IDE on a new tab and concatenate it with "/admin". This will take you to django's default functionality for admins.
- Now you can enter with username:admin and password:1234.
- You can also create your own superuser(from terminal using) but for simplicity use the account given above.
- We have created 4 Entities: Carts, Products, Offers and Customers.
- Admin account can be used to modify and authorise all details in the database.
- Admin can add, update or delete products from the website.



✧ URLs:

```
from django.contrib import admin
from django.urls import path, include
from user import views as uv
from products import views as pv
from cart import views as cv

urlpatterns = [
    path('', uv.log_in_view),
    path('admin/', admin.site.urls),
    path('view_products/<str:username>/', uv.showProduct, name='products'),
    path('view_products/<str:username>/cart/', cv.cart, name='cart'),
    path('createNewUser', uv.createNewUser),
    path('signUpPage', uv.signUpPage),
    path('view_products/<str:username>/place_order', cv.placeOrder),
    path('logout/', uv.user_logout, name='logout'),
    path('profile/', uv.user_profile, name='profile'),
    path('view_products/<str:username>/place_order/', cv.placeOrder),
    path('view_products/<str:username>/place_order/message/', cv.message),
    path('view_products/<str:username>/cart/', cv.cart, name='cart'),
]
```

✧ Our Business logic .py files

- user/view.py
- cart/view.py
- pyshop/view.py