|  |
| --- |
| melon  **Motto**: Find anything, sell anything. |

**INTERNET PROGRAMMING FINAL PROJECT 2019**

TEAM MEMBERS:

1) ABDULLAEV AZAMAT – u1710008

2) KHOLIKOV KHAMIDULLOKH – u1710015

3) URAZBOYEV ABDURAKHMON – u1710022

4) INADULLAEV MAKHMUDJON – u1710047

5) MAKHMUDKHUJAEV SAIDAKBAR – u1710073

**GitHub Repository:** <https://github.com/iuthub/design-project-liquidremix>

**Website URL:** <http://melonuz.herokuapp.com/>

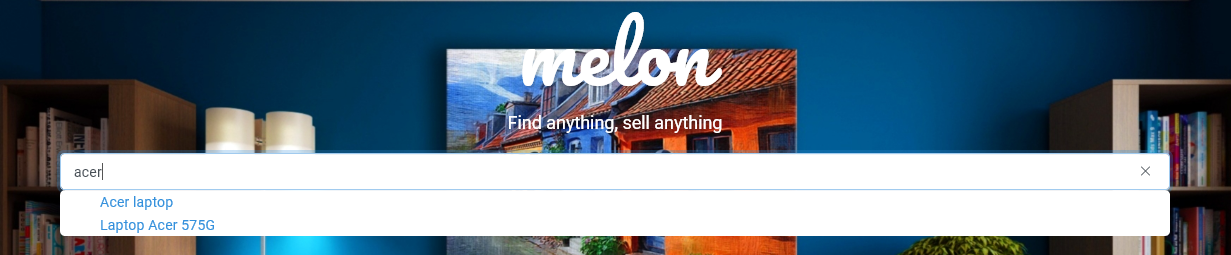
**BRIEF DESCRIPTION.**

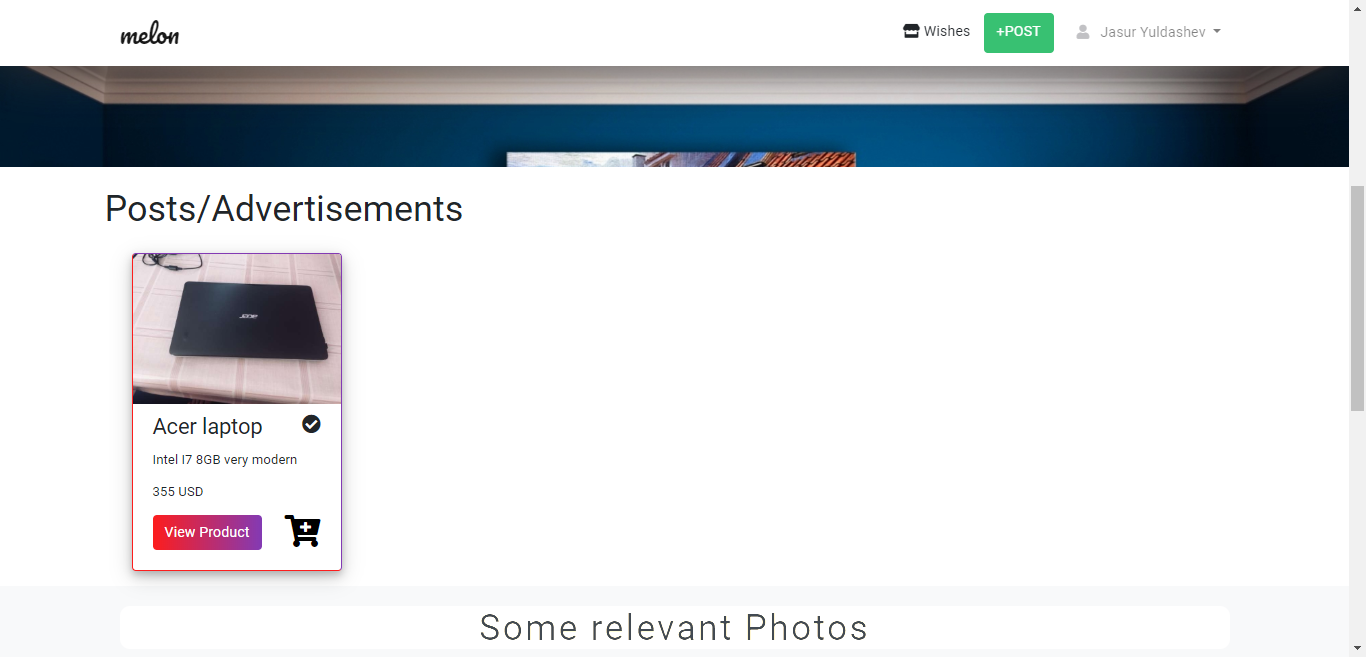
The website is a service for posting ads/posts of different categories (8) such as electronics, jobs, vehicles, home/estate etc, along with photo uploads. This service is free (for now). When users post a new ad this will be sent the admin for verification (as some users may post rubbish data). After the admin verifies the ad it will be published online for the public. Then anyone will be able to see the ad/post and contact the user (contact info is given in the post page). After the users finds buyer or a new customer he can go and delete that post from his account, it will be deleted from the service.

**FEATURES / CONTRIBUTIONS**

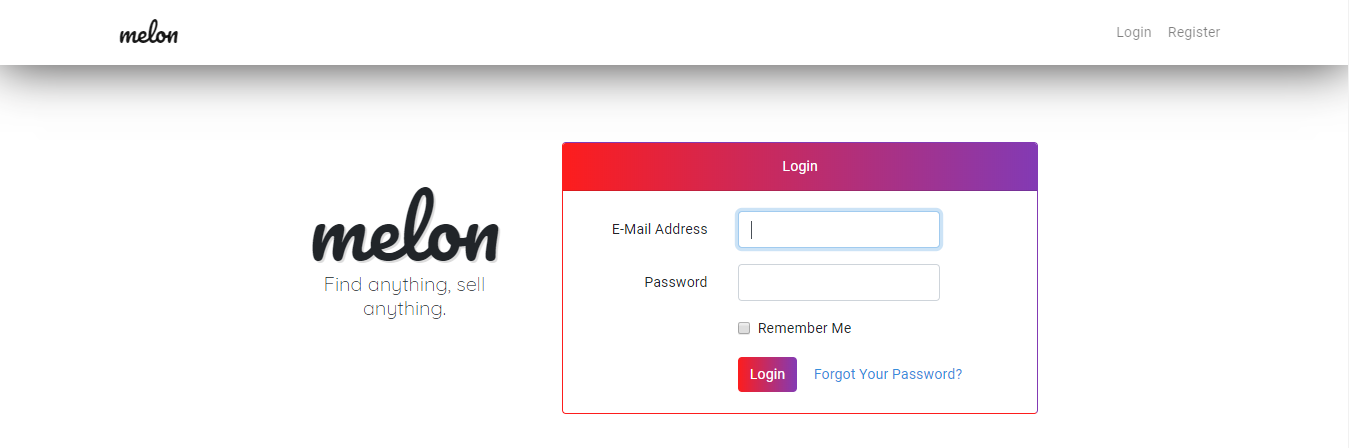
**Contributed by:**

**u1710008 – ABDULLAEV AZAMAT** – I worked both with backend and frontend. To start with the landing page, implemented the interactive live search function using jQuery ajax. To implement this I used keyup event, and sent GET request to /api/search/{key} so I did it using a small REST API implemented using Laravel. As you can see below, there is keyword “acer” and it dynamically shows the results, which are clickable and opens up the page of that product.



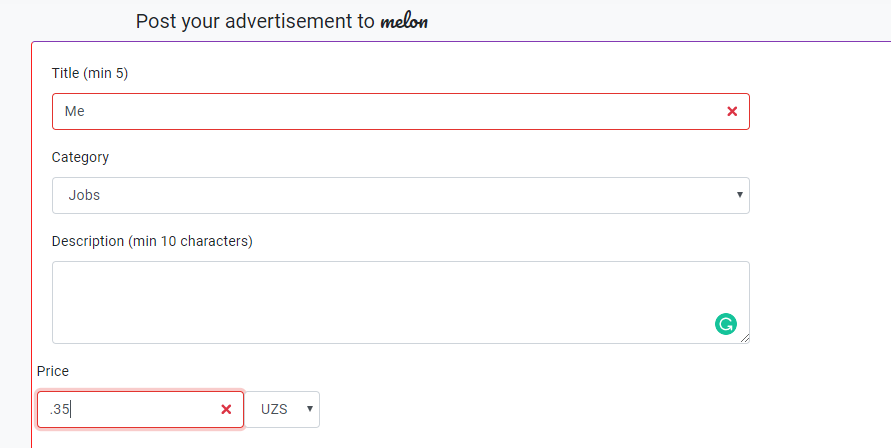


Then I implemented authentication both the frontend (bootstrap/css) and backend along with user input validation.



Developed almost all Eloquent ORM models: User, Post, PostPhoto,Feedback, including controllers to handle all CRUD operations throughout the website. Connected the models using Eloquent relationships: oneToMany etc.

Made manipulations and user validations on create post page. Validations are handled on the frontend using jQuery and RegEx (regular expressions).



Developed backend handling for handling file (image) upload using Laravel’s native features. Below is snippet from HomeController.php

File upload (front-end):

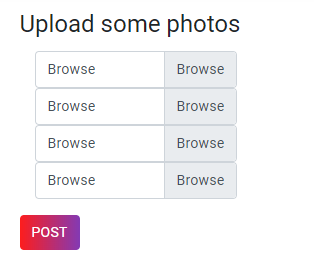
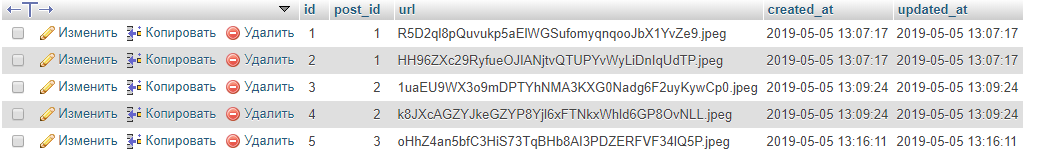
 

Table that maps photos to posts. Used oneToMany relationsip.

Used jQuery’s ajax to connect to external web service for fetching exchange rates from USD->UZS as shown in the below screenshot.





Implemented hosting. Deployed the website to heroku.

And developed many many more Laravel, jQuery, css features. In summary I worked both on backend and frontend!!!

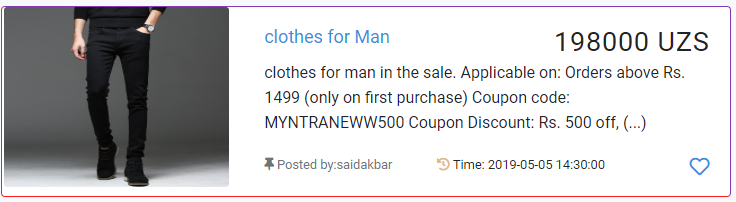
**U1710073 – MAKHMUDKHUJAEV SAIDAKBAR** – I made the main ui system using bootstrap/css.

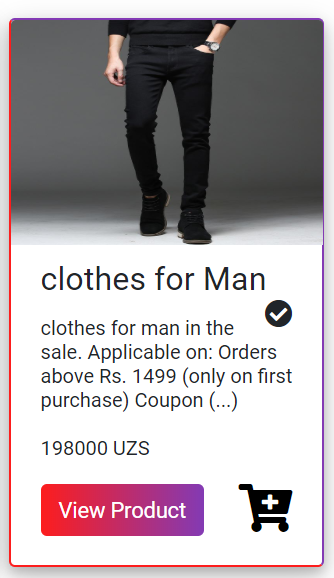
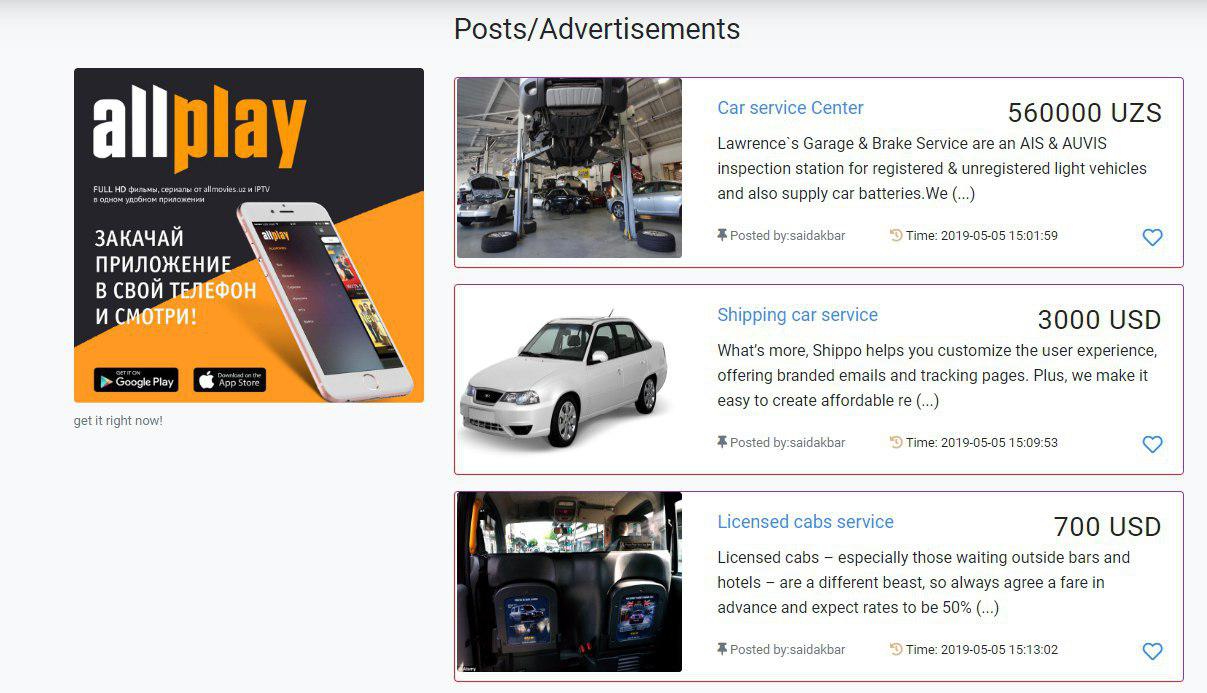
Made some of the database connection to read information using Eloquent ORM, made the Feedback model both frontend and backend to handle the database storage of the Contact form.



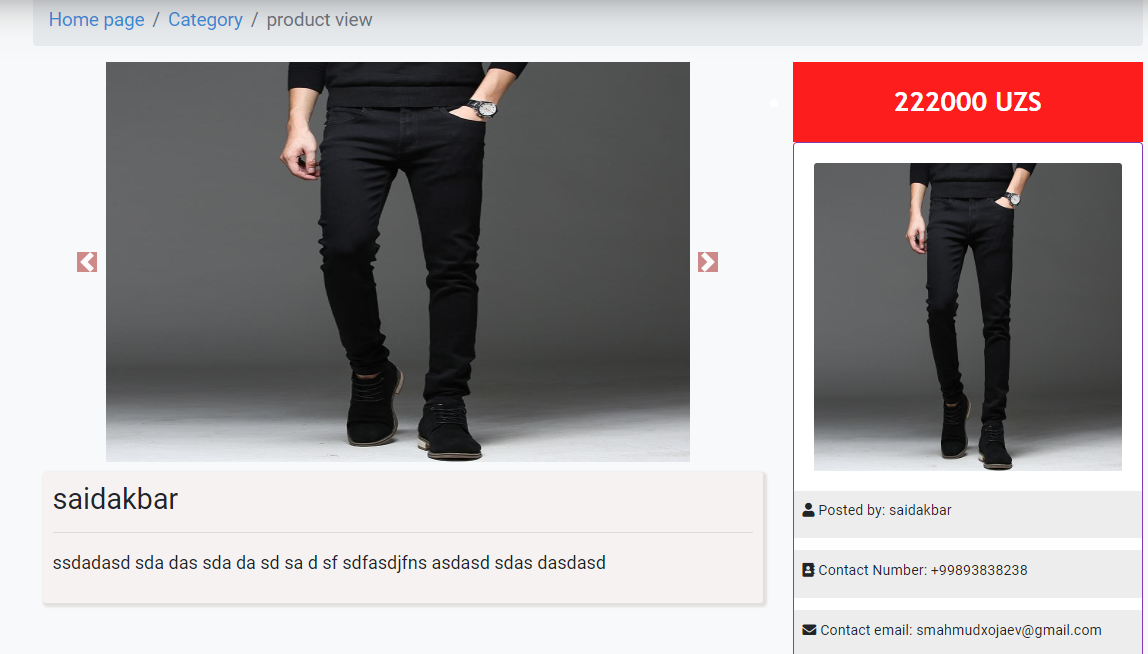
My main work is directed to handle client side issues and create our own reusable HTML/CSS skeleton. I contributed some feature for DB in order to handle client side error which had some conflict while we were working .

I created cards which we used to show info about product and directs a specific page in order to get whole information by using hyper link. Then, I did some tricks in JQuery in order to handle “like -> wishes” view. I did some modification in fetching data from Mysql using Eloquent’s approaches.



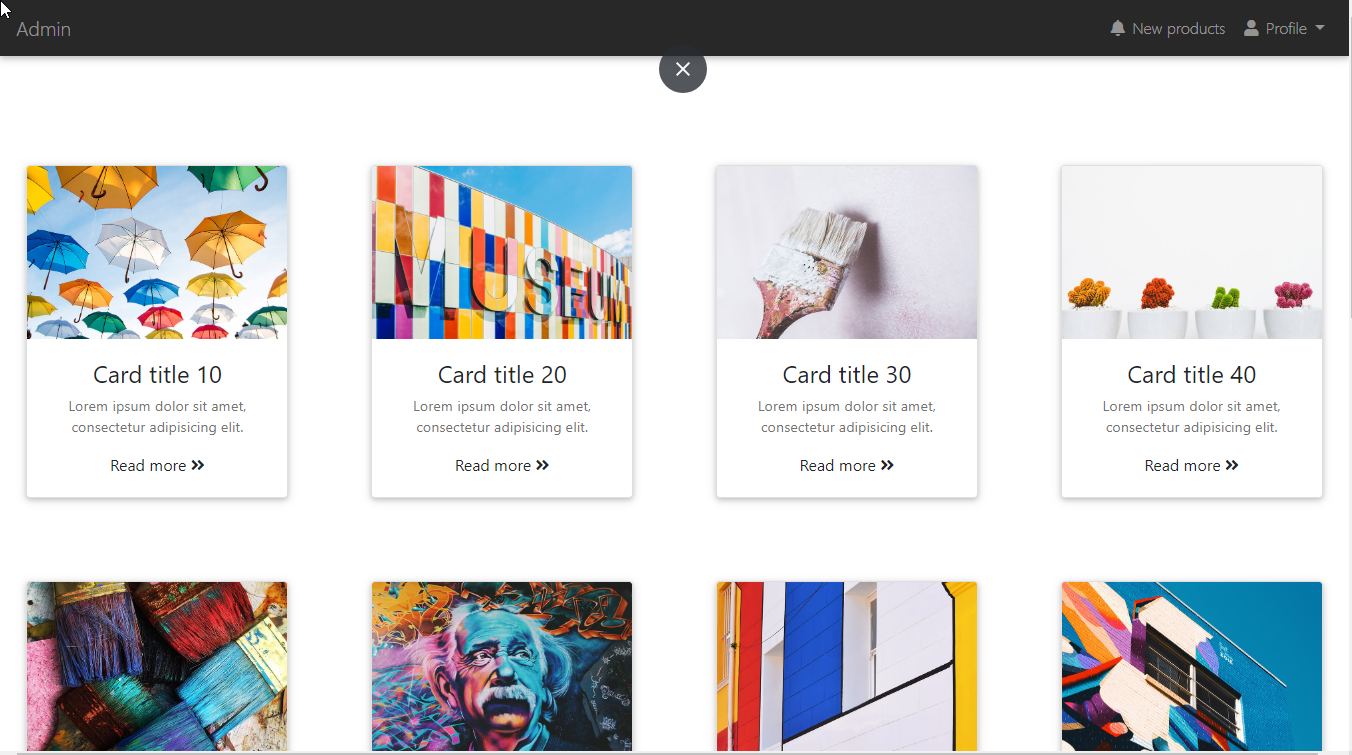


In real world, we can see adverts everywhere even in websites. In the same fashion , I have used external links in order to do some adverts which up brings us some amount money in future. I also created page called “category->item” which shows below:

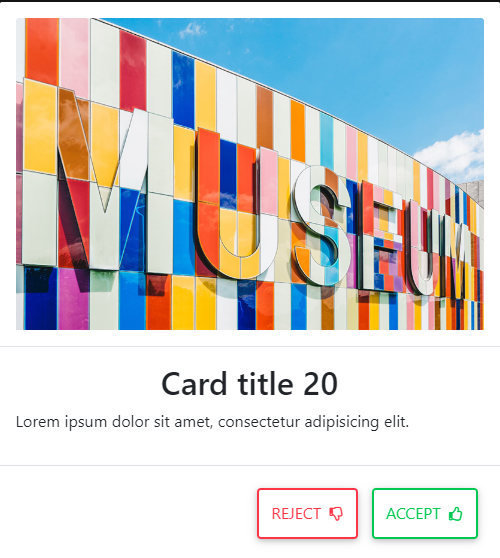


**u1710047 – INADULLAEV MAKHMUDJON**

I worked on UI/UX which comes to the point of Frontend and Backend side.

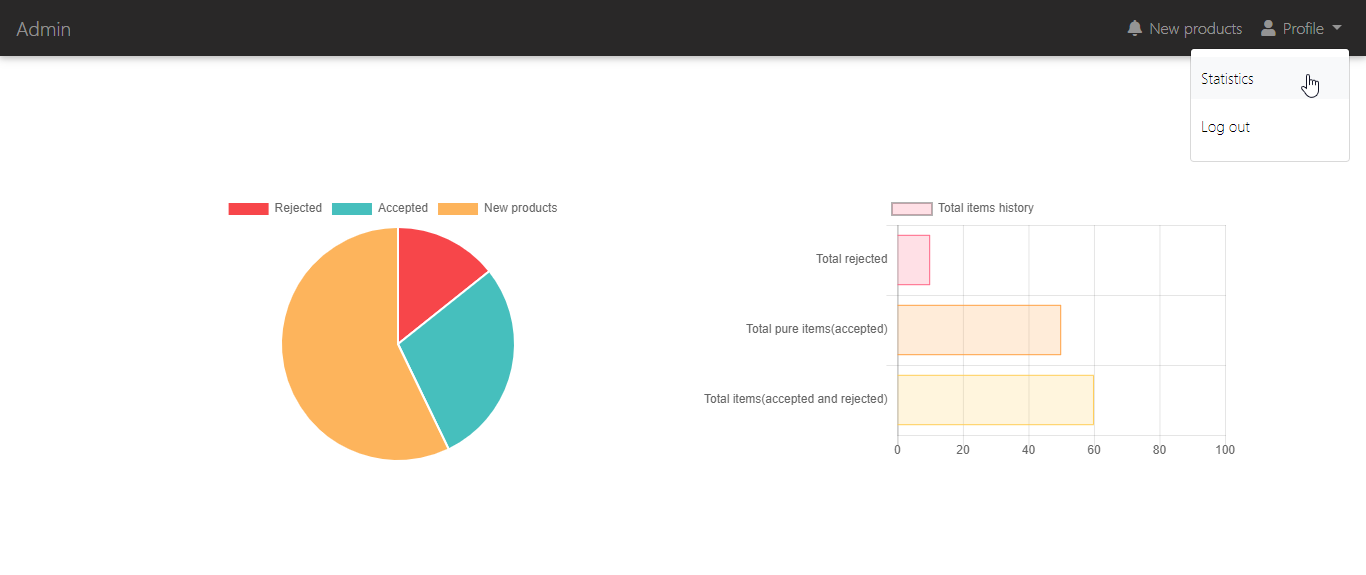


This is the admin page which designed and demanded higher afford on UX side after items populated. And come that point which describes the details of individual items.



You will see that page after clicking the item.

However, on admin page we see the statistics which illustrates the quantity of rejected, accepted and new products on it.

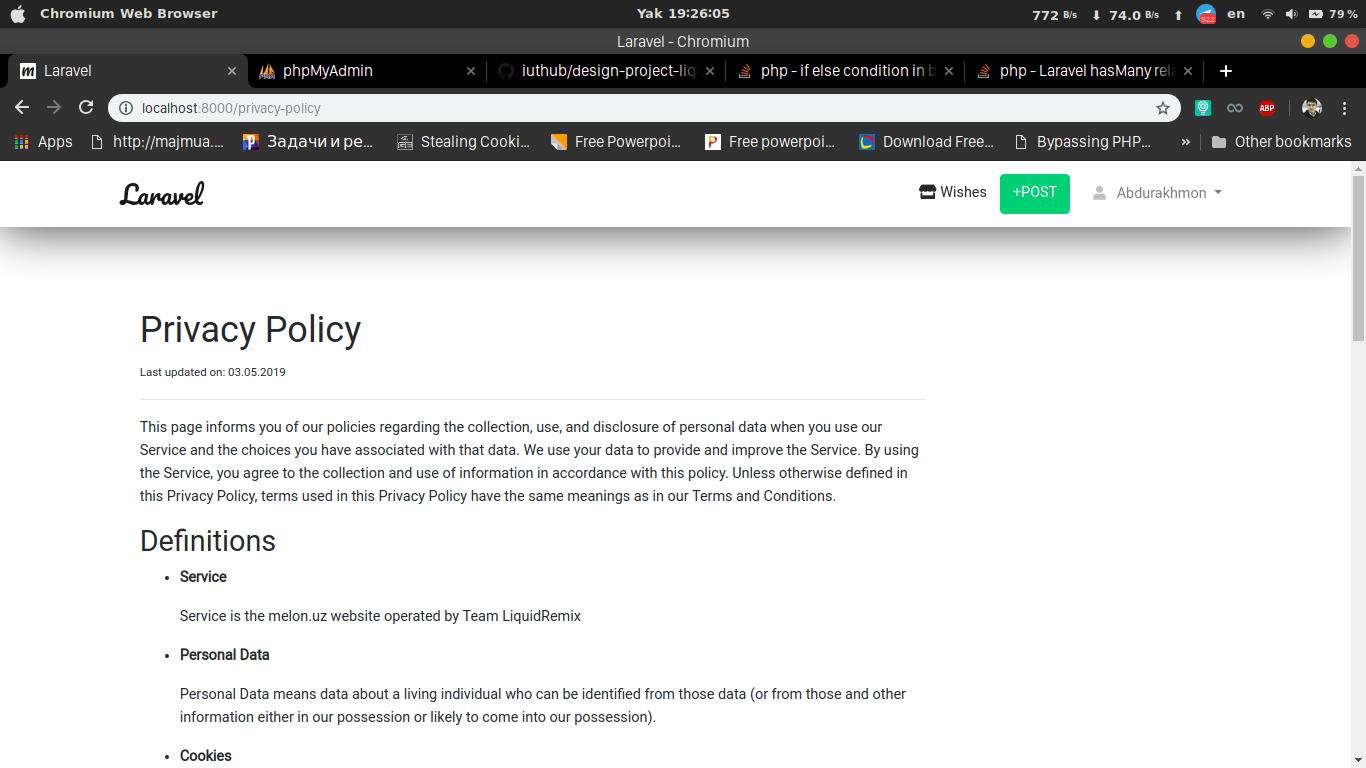


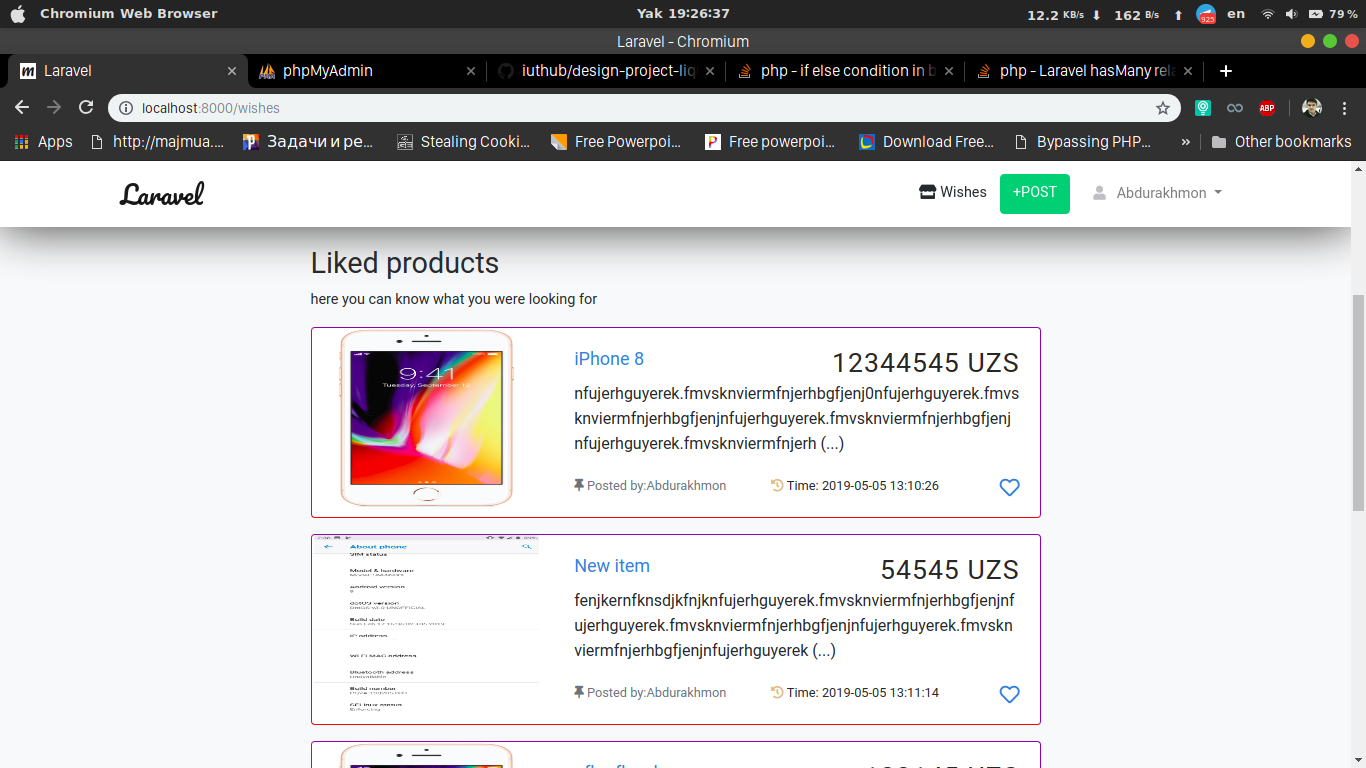
Similarly, they all connected to the database that user can see their product results individual. I covered the topics of Java Script, Jquery, database relations and all frontend technics and some tricky logic over the classes

**u1710022 – URAZBOEV ABDURAKHMON –** made Privacy policy page, implemented categories and logic for sorting posts by categories in category page. Implemented like feature for posts and wishlist logic to show liked posts. Wrote migration files for like and category saving in database. Implemented logic to to display liked and categorized posts.

Build route for privacy policy page. Wrote all policies.

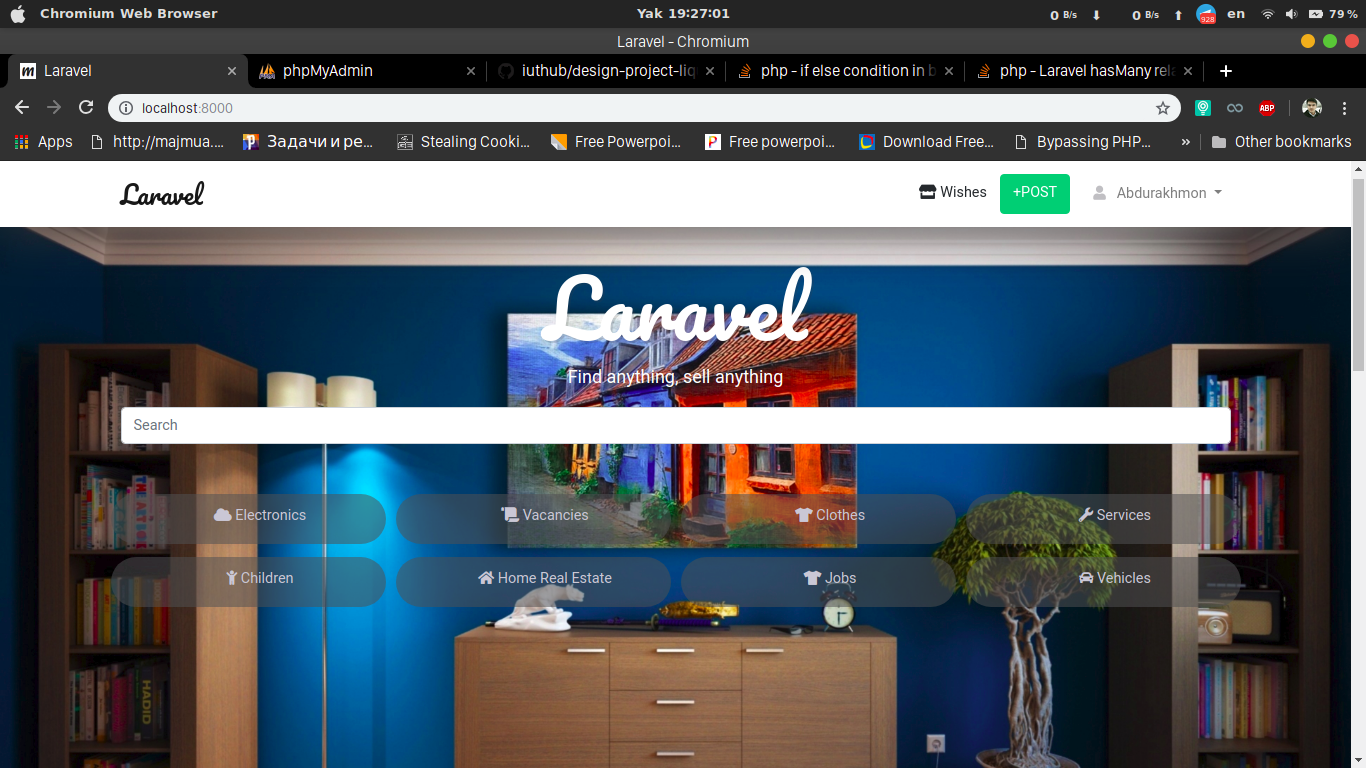
Added like function to created wishlist page with liked post. Implemented logic to look for likes of current user, and found post that was liked and listed them im wishlist page.





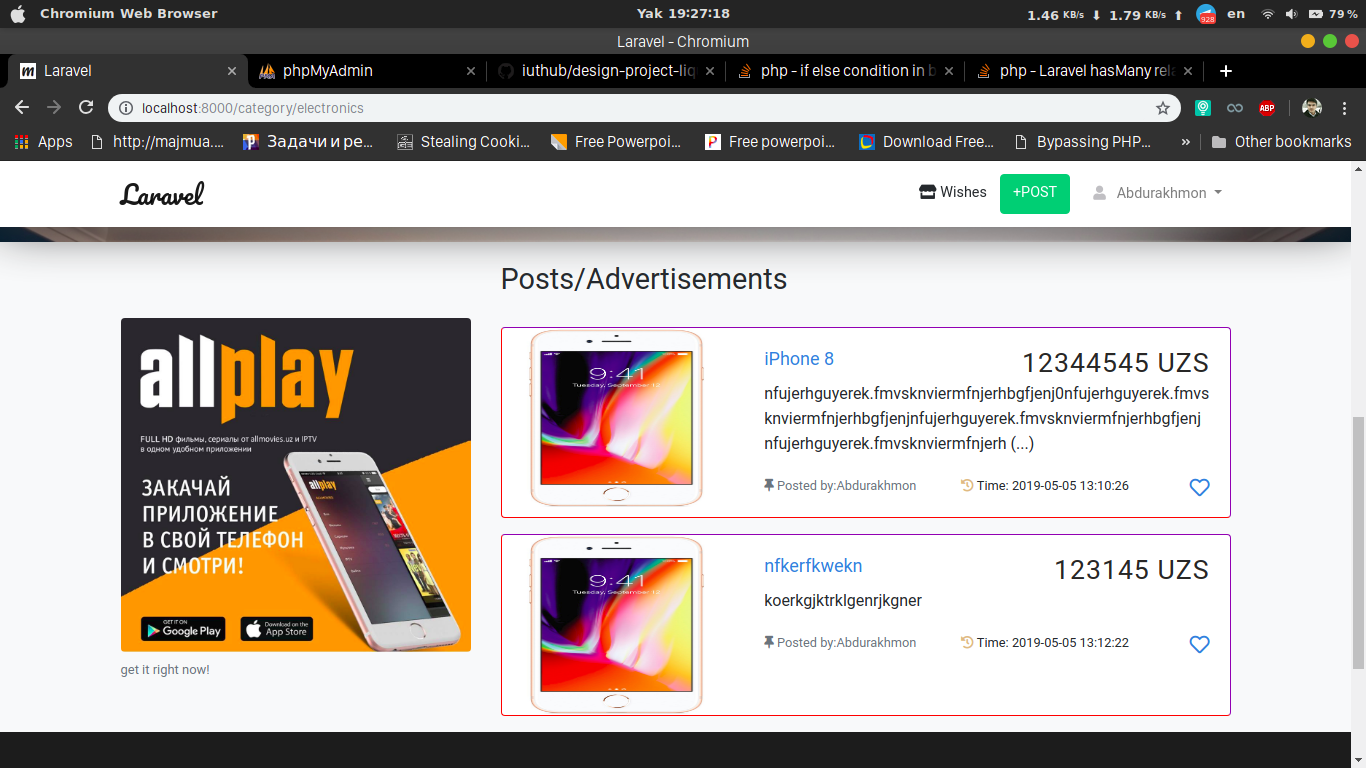
Implemented Logic for categories, to classify posts by category.

Every category has different routes. Category itself separates all posts of this category from others and displays it.

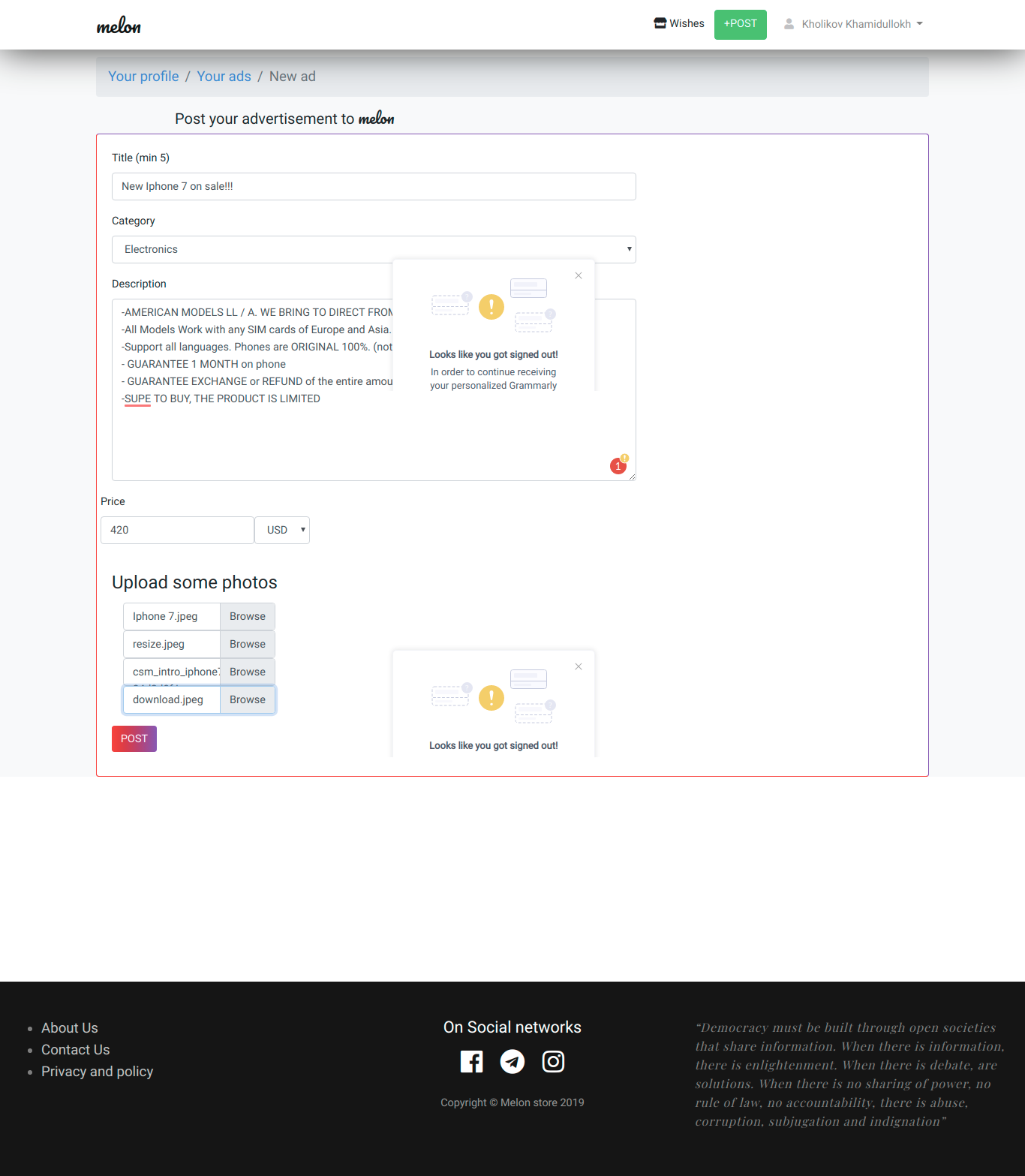


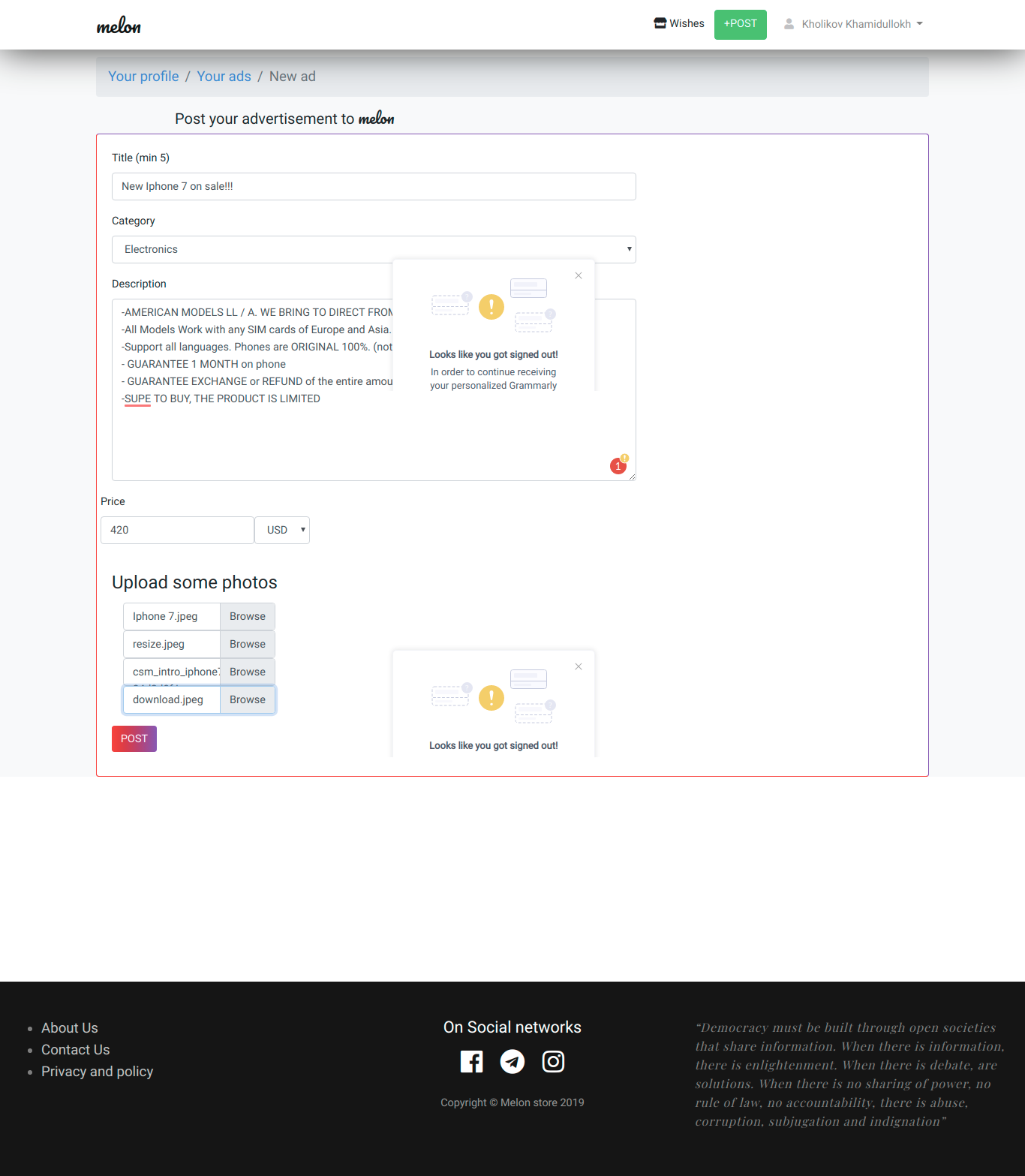
In every category page, posts related to this category will be listed so that that it will display relative information as title and description of object, time of post, author and etc.

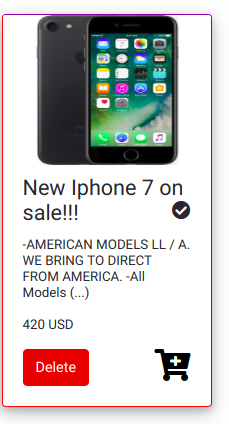
All file containers are link for file page that has whole information about the post.

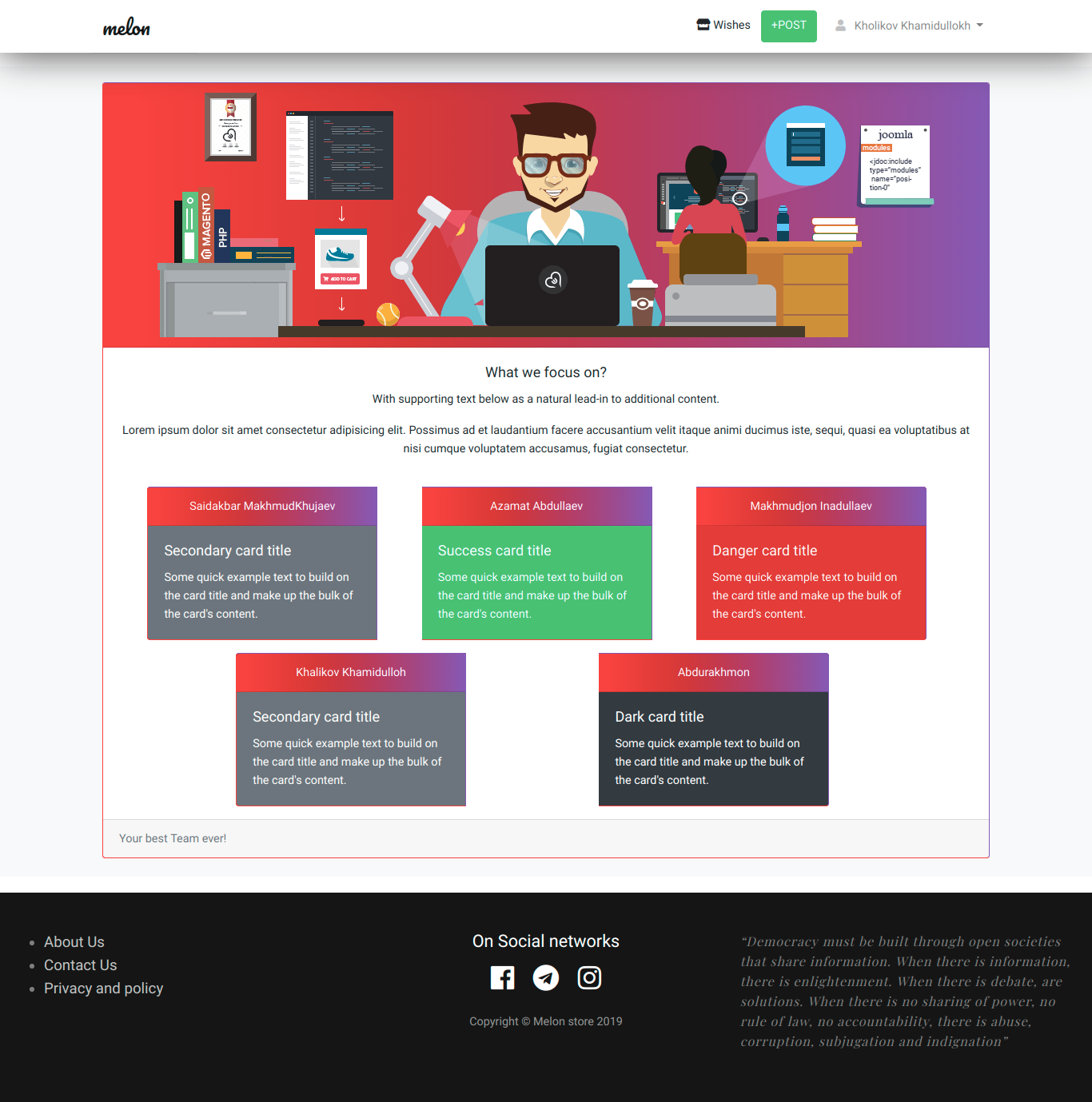


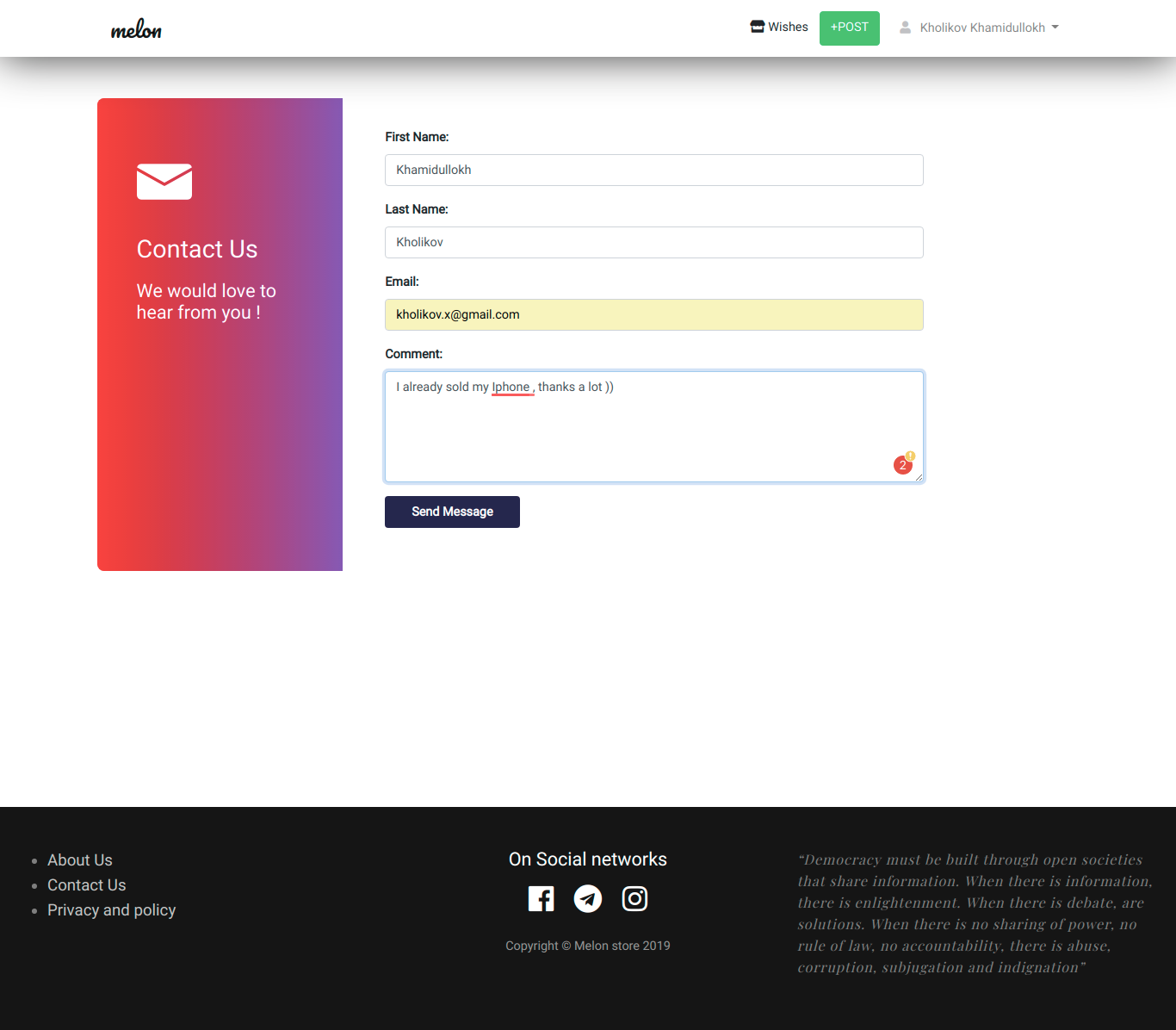
**u1710015 – KHOLIKOV KHAMIDULLOKH** – I made both backend and frontend of POST page using Bootstrap/css and I created database “posts” table which store all below posts in database.My main work was to create posts page and show on index page. I stored image url in database path: /storage/files/{{$post→photos→first()→url}} by using one to many relationships. And I also made about page. There are some description and team members. Then I created Contact US page and there are users can enter comments .











**PASSWORDS/ACCOUNTS**

**1. USER:**

login/e-mail-address: [jasur@gmail.com](mailto:jasur@gmail.com)

password: azamat98

**2. ADMIN:**

**Go to: /admin**