

Technical Documentations

Dania Abutair

19th April, 2021



Introduction:

The website is an online marketplace for people who need to sell their stuff whether used or new, and you can swipe it if you don't want to sell it, this website helps people to sell or exchange unneeded stuff with other people who need this stuff. I had the idea of this website after I saw lots of people posting on Facebook and Instagram to sell their objects, so I decided to make a website where all of them can gather to sell or swipe, this would make it easier to find and advertise the needed stuff.

Technologies:

Backend:

Php /laravel

Mysql

Frontend:

Html

CSS

Bootstrap

JavaScript

functional specifications:

- .Dashboard for admins with different role admin and super admin to manage all resources
- .Dashboard for user to manage his products or posts
- .User can post his product on website, but first the user must be login
- .login and register system for both user and admin

- .The user can browse the public side without login or register on website
- .User can connect to another user to buy his product, in this case the user must be login

Security:

- Cross-Site Request Forgery (CSRF) Protection on Laravel
- Hashing password
- Authenticating for user and admin (guard, session, middleware)

Obstacles:

I built my website on laravel, a new framework for me, so I had to do a lot of search on google, watch courses and read documentation , first I learned laravel version 5.8 and after that I learned laravel version 8 to do my master project.

The most significant code excerpts and their justification:

I used resource routing, Laravel resource routing assigns the typical create, read, update, and delete ("CRUD") routes to a controller with a single line of code. This single route declaration creates multiple routes to handle a variety of actions on the resource.

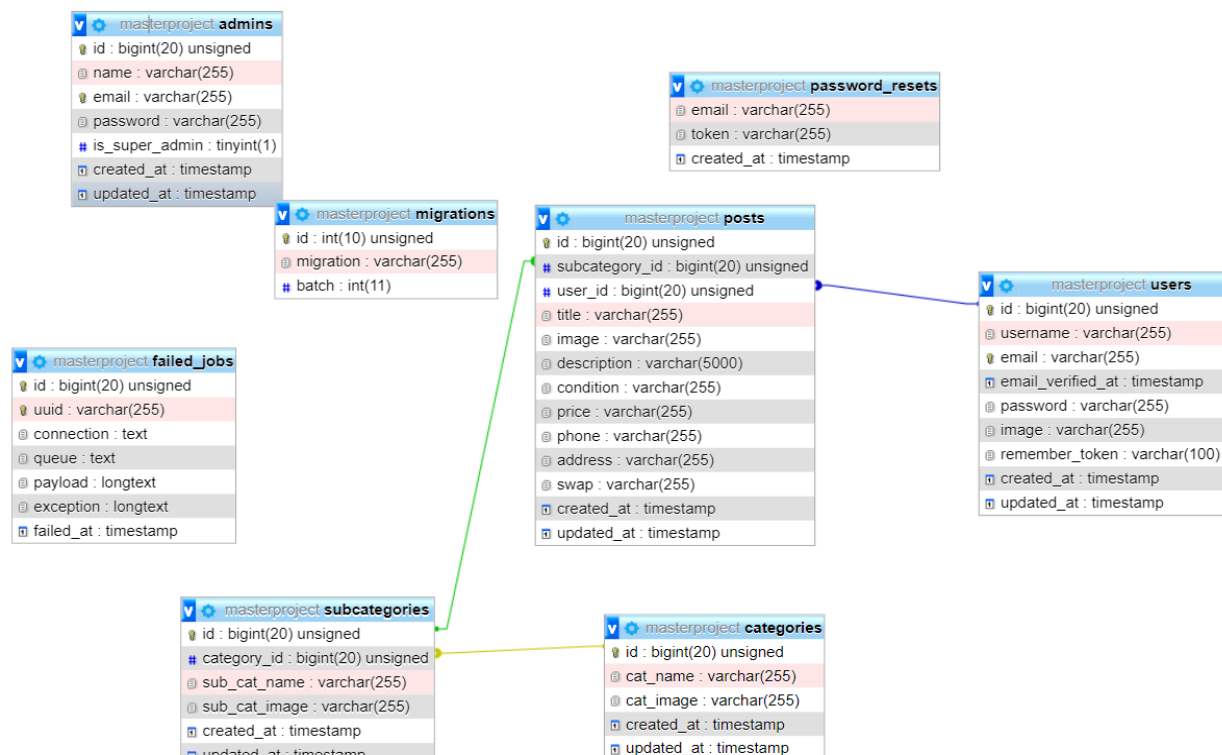
The test case of most representative feature:

User can post his product on the website but first he must login or register on the website, first the user clicks on the button post that takes him to another page to create his product, the user must fill information about his product, and he can check all the posts in his dashboard.

Description of the watch over security vulnerabilities:

I separate the users in two tables Admin and User to be more secure and with two different login systems.

My Database:



Laravel Structure

Project Directory

```
|--app
  |--console
  |--Exceptions
  |--Http
  |--Providers
|--bootstrap
  |--Cache
|--config
|--database
|--public
  |--CSS
  |--fonts
  |--img
  |--js
|--console
|--Exceptions
|--Http
|--Providers
|--resources
  |--views
  |--layouts
    |--partials
    |--footer-scripts.blade.php
    |--head.blade.php
    |--header.blade.php
    |--nav.blade.php
    |--mainlayout.blade.php
|--routes
  |--api.php
  |--channels.php
  |--console.php
  |--web.php
|--storage
|--tests
|--vendor
|--editorconfig
|--env
|--env.example
|--styleci.yml
|--artisan
|--composer.json
|--composer.lock
|--package.json
|--package-lock.json
|--phpunit
|--README.md
|--webpack.mix
```

