**GadgetSeller**

Project Report

Team Members

Khalid Masum

**180042133**

Afnan Jarif

**180042139**

Asfi Rahman

**180042116**

**Introduction:** Gadgetseller is an e-commerce project. Where both shop owner and General user can buy or sell products.

**Technology used:**

Programming Language: PHP (With the framework Laravel)

Dependency Manager: Composer

DBMS: Used Eloquent, a layer over RDBMS. So it is compatible with MySQL, Postgres, SQLite and SQL server

Front End: HTML, CSS (With Bootstrap), Javascript

Debug Assistant: Tinker and Telescope

**Installation:**

Following steps are to be followed to install the project in any computer.

* Download and install PHP from the official website.

<https://www.php.net/downloads>

* Download and install the dependency manager Composer

<https://getcomposer.org/download/>

* Download and install node

<https://nodejs.org/en/>

* Install laravel

composer global require laravel/installer

* Clone *our project* from git

git clone <https://github.com/iut-web-builders/gadgetstore.git>

cd gadgetstore

* Install dependencies

composer update

npm install

* Find the .env.example file and edit it based on the database you want to run it on, save it as .env
* Run following commands:

php artisan key:generate

php artisan migrate

php artisan storage:link

php artisan serve

This should fire up the project in your local host. However you may need to change configuration of your php based on how it is requested on terminal.

**Features:**

Point of view based features:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature Name** | **Guest** | **Logged in General User** | **Mod** | **Admin** |
| View Product | Yes | Yes | Yes | Yes |
| Search Product | Yes | Yes | Yes | Yes |
| Add to cart | No | Yes | No | No |
| Add Product To Sellable | No | Yes | Yes | Yes |
| Authenticate | Yes | No­1 | No1,2 | No1,2 |
| Apply filter on product, by brand and category | Yes | Yes | Yes | Yes |
| Edit own product | No | Yes | Yes | Yes |
| Edit profile, containing contact number, address and profile picture | No | Yes | Yes | Yes |
| Approve mod | No | No | Yes | Yes |
| Remove users | No | No | Yes | Yes |
| Add new brand and category | No | No | Yes | Yes |
| View own orders | No | Yes | No | No |
| View all orders | No | No | Yes | Yes |

Other features:

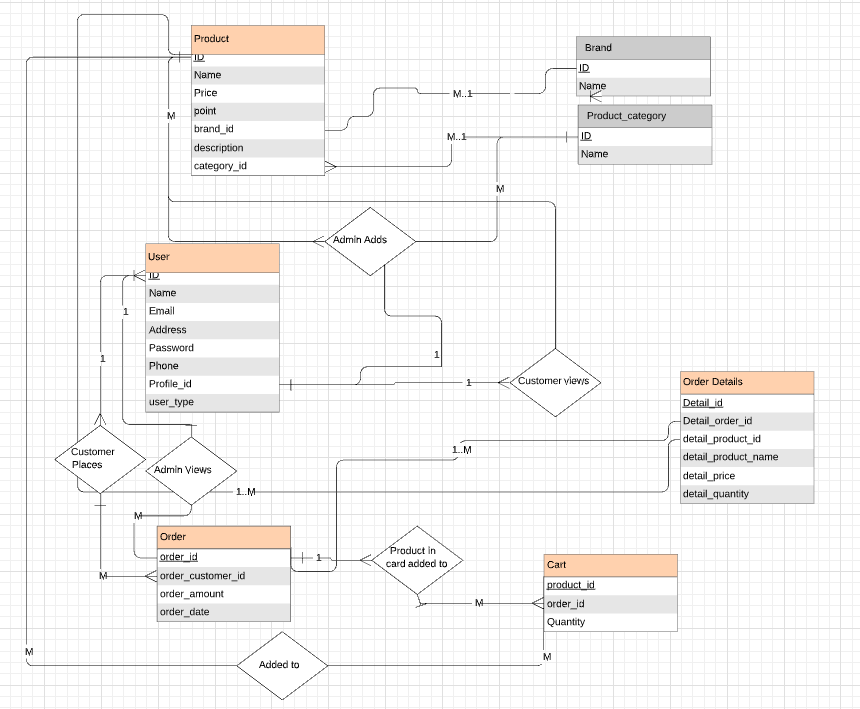
* Password guard, needs to have at least 8 characters.
* Password is stored hashed.
* Product has a description, image and price.
* Form data verification on every post request.

**Designing:**

The project follows Laravel’s format. Which is:

* Views contain all the front end related files
* Controllers contain all the main controlling functions
* Models connect to database
* Database, is some migration script that is run separately to install the database files
* A web.php file connectiong front end with the back end (Client with Server).

**Simplified ER Diagram:**

****

What’s in the complex version is, there are two of everything, one for main product and one for user products, except for or brand and category. Since the code is still in the Shameless Green version. We believe the code can easily be decomposed into it, however implementing the feature was our main purpose, since the code was good enough. Following the XP principles, we moved on.

**Conclusion:**

The project looks a very simple one. It indeed was very easy. All is about how to query, and what to query. With some storage techniques. However even though they are simple, there were so many things to do, many features to implement and many new things to learn.