

Practical Laravel Book Images

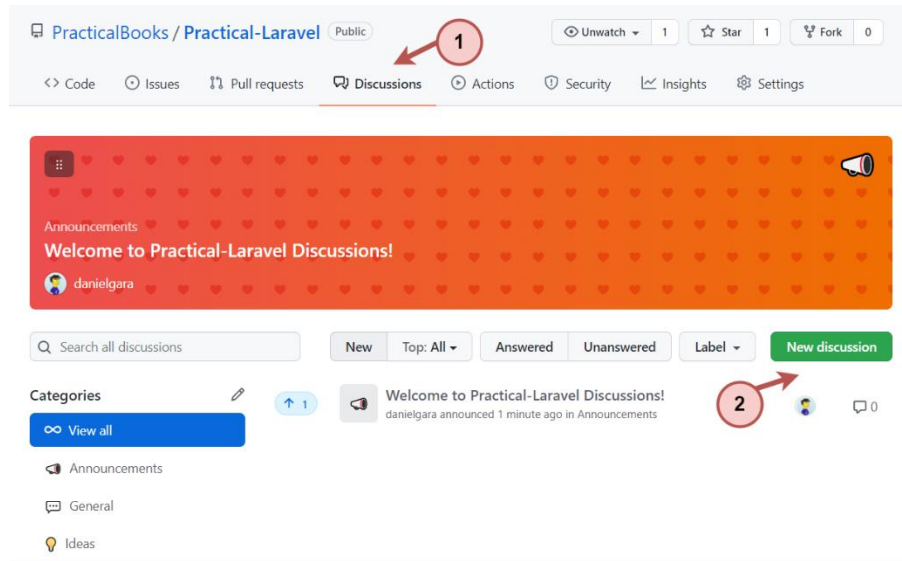


Figure P-1. Discussion zone of the GitHub repository.

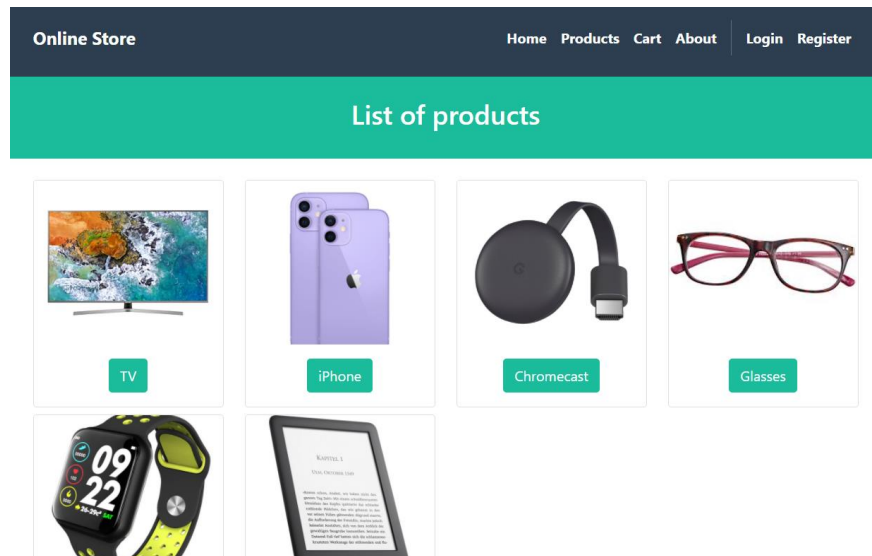


Figure 1-1. List of products page.

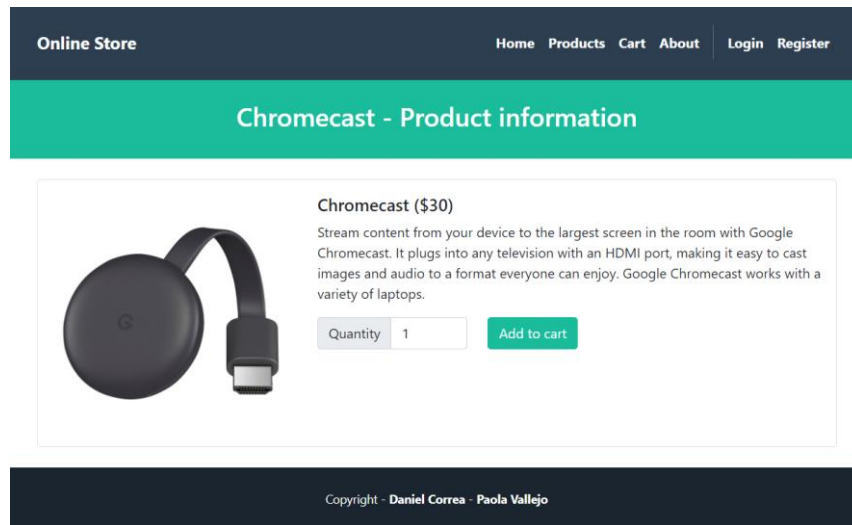


Figure 1-2. Product page.

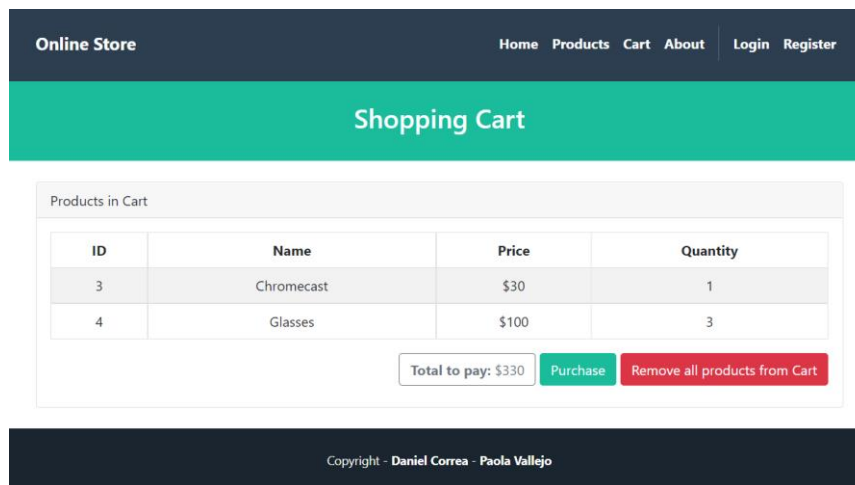


Figure 1-3. Shopping cart page.

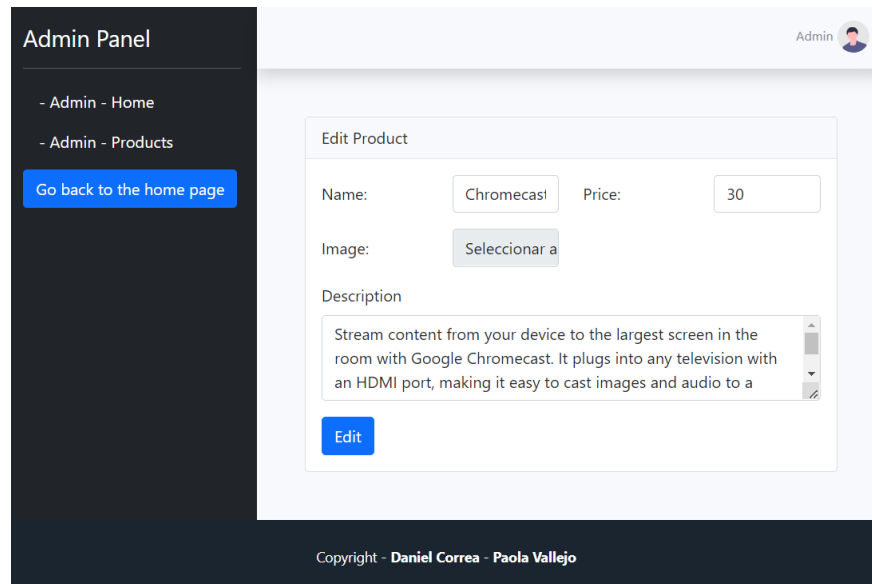


Figure 1-4. Admin panel page.

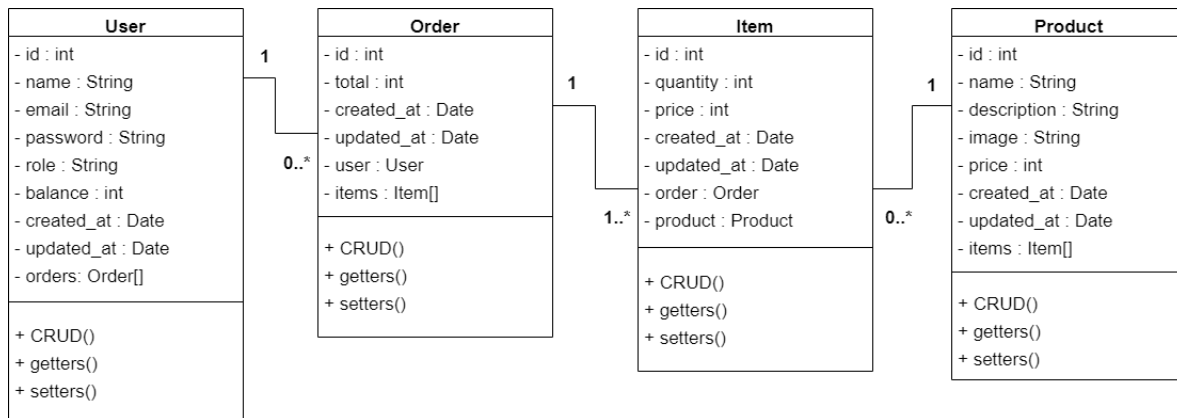


Figure 2-1. Online Store class diagram.

```

PS C:\Users\yo> php --version
PHP 8.0.11 (cli) (built: Sep 21 2021 18:26:06) ( ZTS Visual C++ 2019 x64 )
Copyright (c) The PHP Group
Zend Engine v4.0.11, Copyright (c) Zend Technologies
  
```

Figure 3-1. Checking PHP version.

```

PS C:\Users\yo> composer --version
Composer version 2.1.12 2021-11-09 16:02:04
  
```

Figure 3-2. Checking composer version.

```

PS C:\xampp\htdocs\onlineStore> php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Wed Feb 9 11:04:59 2022] PHP 8.0.11 Development Server (http://127.0.0.1:8000) started
  
```

Figure 3-3. Running Laravel project.

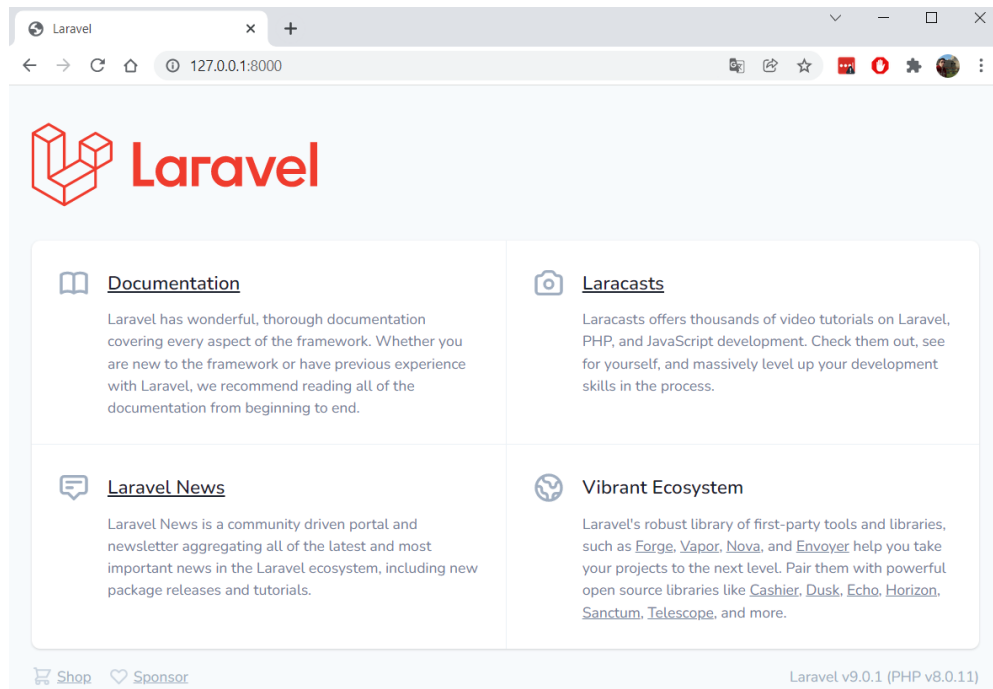


Figure 3-4. Laravel 9 default page.

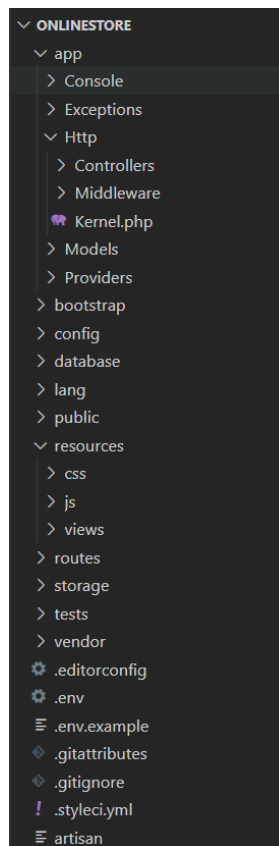


Figure 3-5. Laravel Project structure.

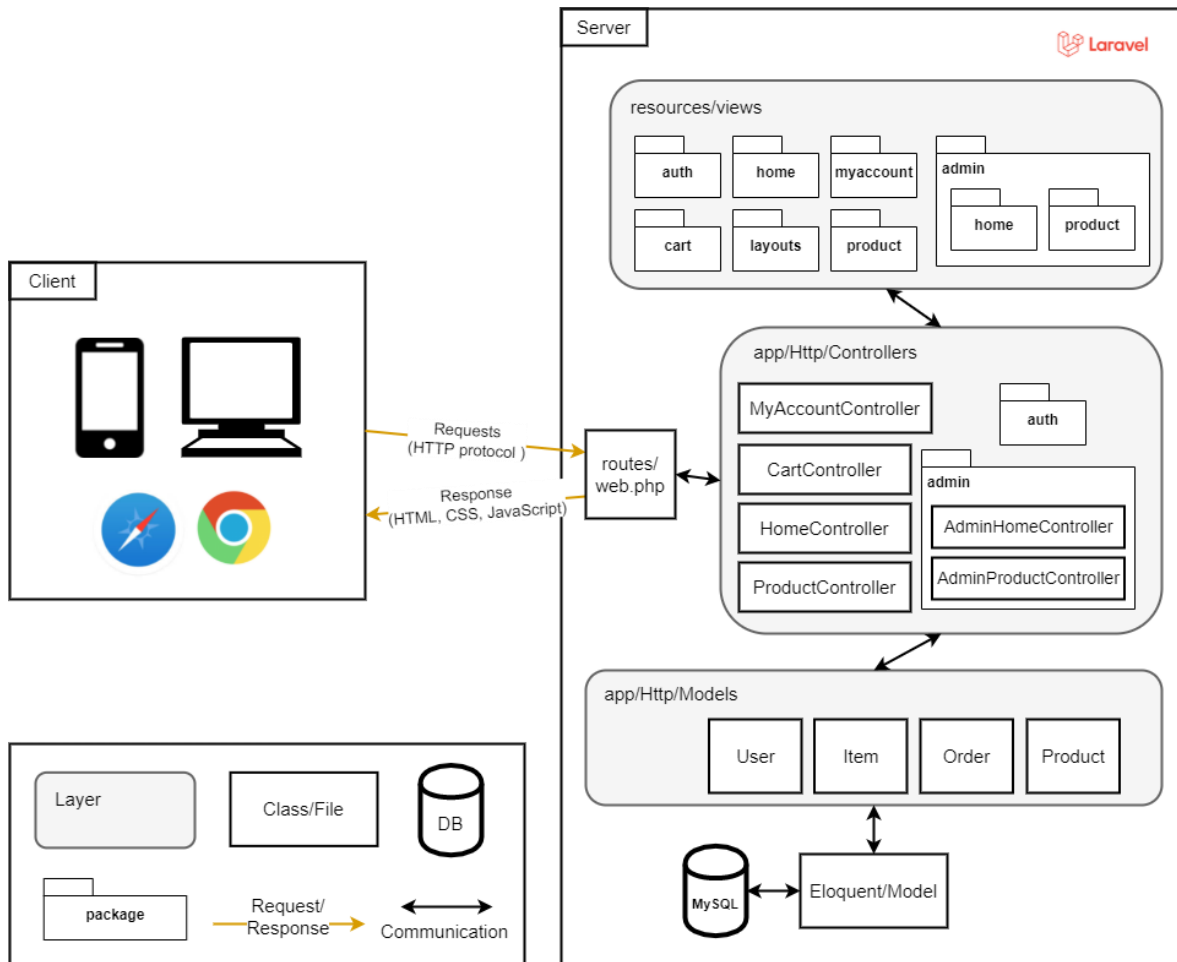


Figure 4-1. Online Store software architecture.

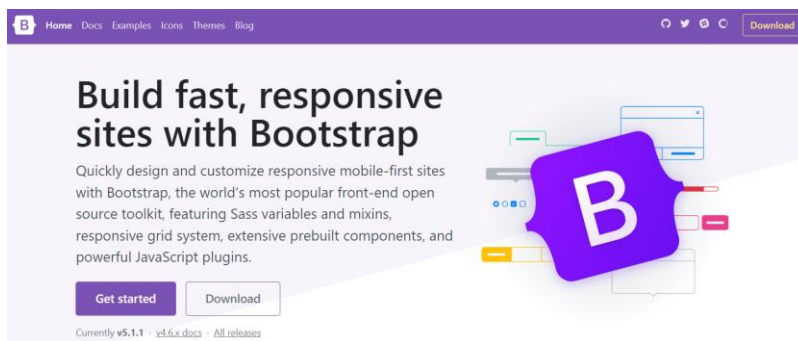


Figure 5-1. Bootstrap website.

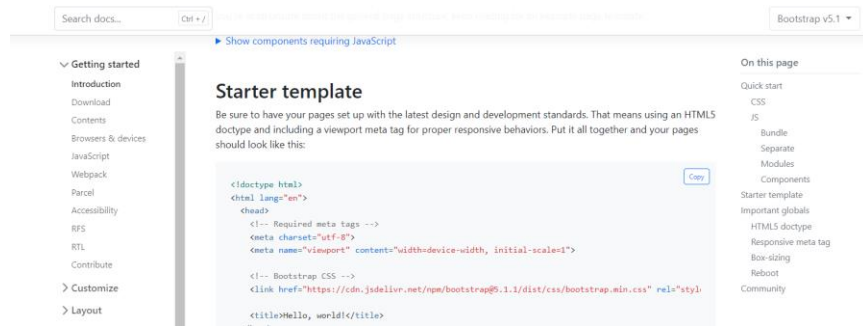


Figure 5-2. Bootstrap starter template.

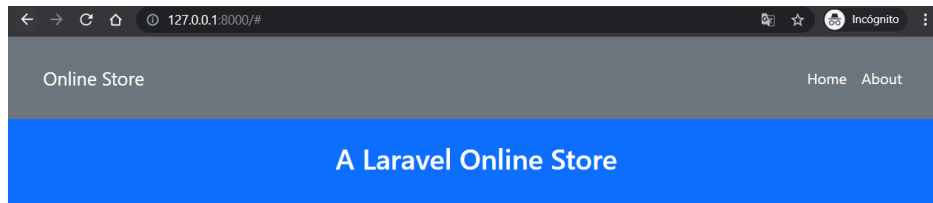
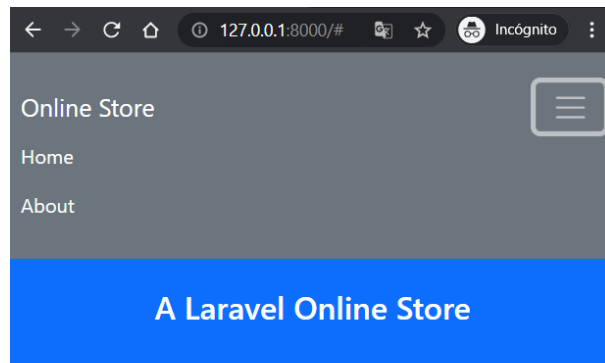


Figure 5-3. Application home page with the layout.



Welcome to the application

Figure 5-4. Application home page with reduced browser window width.

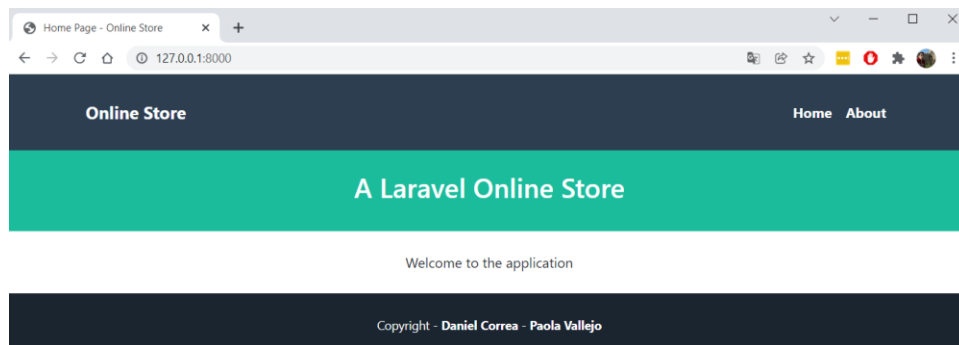


Figure 5-5. Application home page with the refined layout.

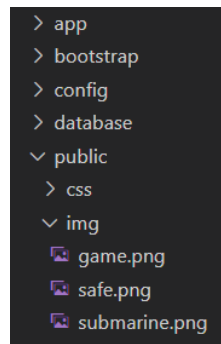


Figure 6-1. Project structure after storing the images.

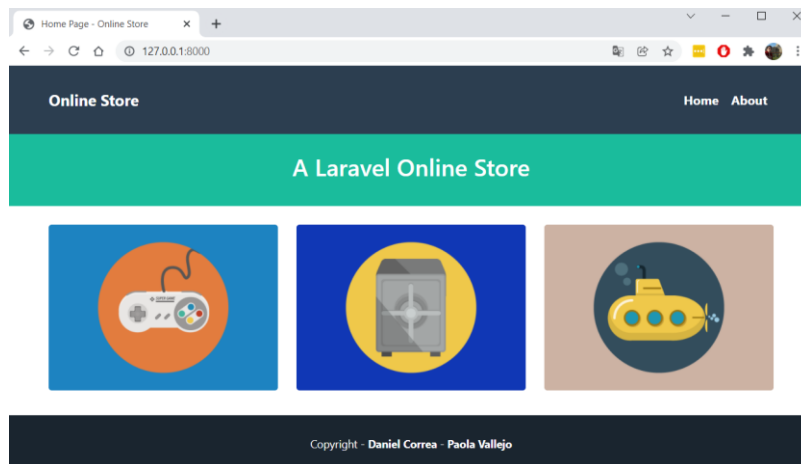


Figure 6-2. Online Store – Home page.

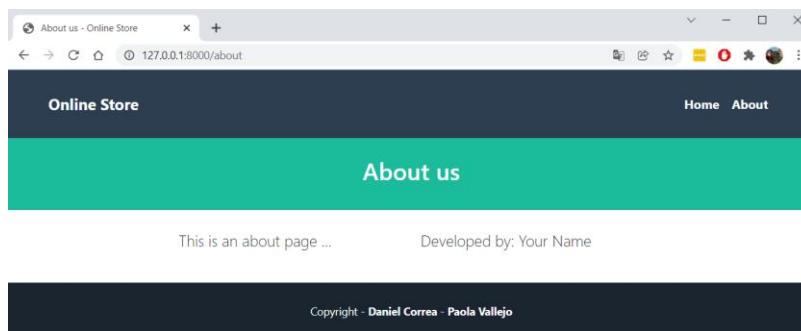


Figure 6-3. Online Store – About page.

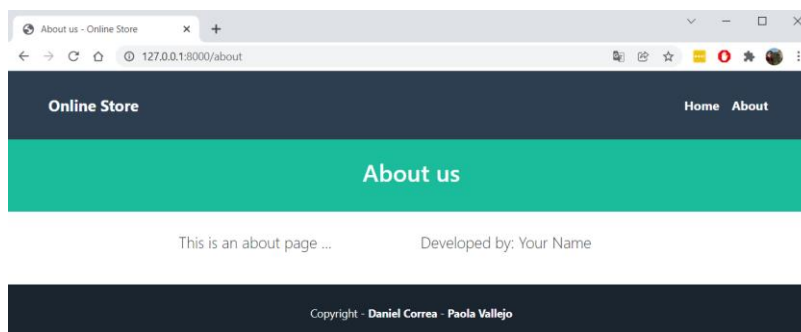


Figure 7-1. Online Store – About page.

```

PS C:\xampp\htdocs\onlineStore> ./vendor/bin/phpcs
FILE: ...xampp\htdocs\onlineStore\app\Http\Controllers\HomeController.php
-----
FOUND 17 ERRORS AFFECTING 17 LINES
-----
 1 | ERROR | [x] End of line character is invalid; expected "\n" but
   |       | found "\r\n"
 3 | ERROR | [x] There must be one blank line after the namespace
   |       | declaration
 8 | ERROR | [x] Line indented incorrectly; expected 4 spaces, found
   |       | 2
 9 | ERROR | [x] Line indented incorrectly; expected at least 4
   |       | spaces, found 2

```

Figure 8-1. Running PHP CodeSniffer.

```

PS C:\xampp\htdocs\onlineStore> ./vendor/bin/phpcbf
PHPCBF RESULT SUMMARY
-----
FILE                                     FIXED  REMAINING
-----
...lineStore\app\Http\Controllers\HomeController.php 17      1
-----
A TOTAL OF 17 ERRORS WERE FIXED IN 1 FILE
-----

Time: 209ms; Memory: 8MB

```

Figure 8-2. Fixing style errors with PHP CodeSniffer.

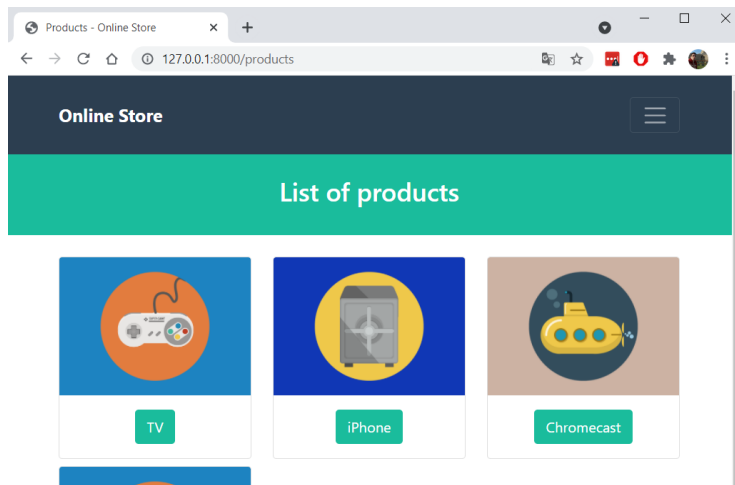


Figure 9-1. Online Store – Products page.

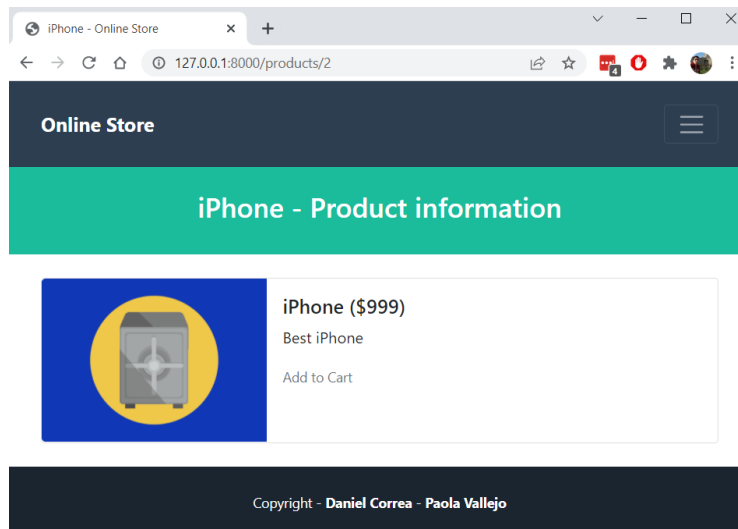


Figure 9-2. Online Store – iPhone product page.

id	name	description	image	price
1	TV	Best tv	game.png	1000
2	iPhone	Best iPhone	safe.png	999
3	Chromecast	Best Chromecast	submarine.png	30
4	Glasses	Best Glasses	game.png	100

Figure 10-1. Product table.

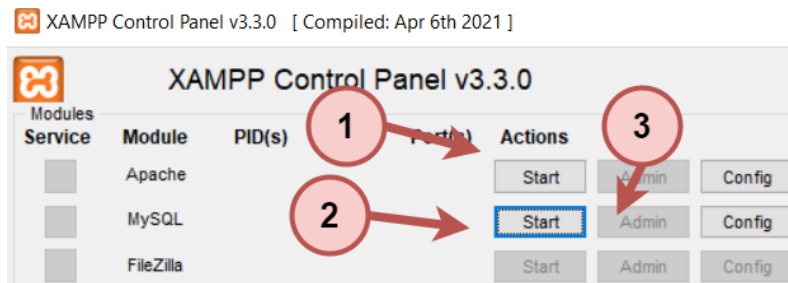


Figure 10-2. Starting MySQL module in XAMPP.

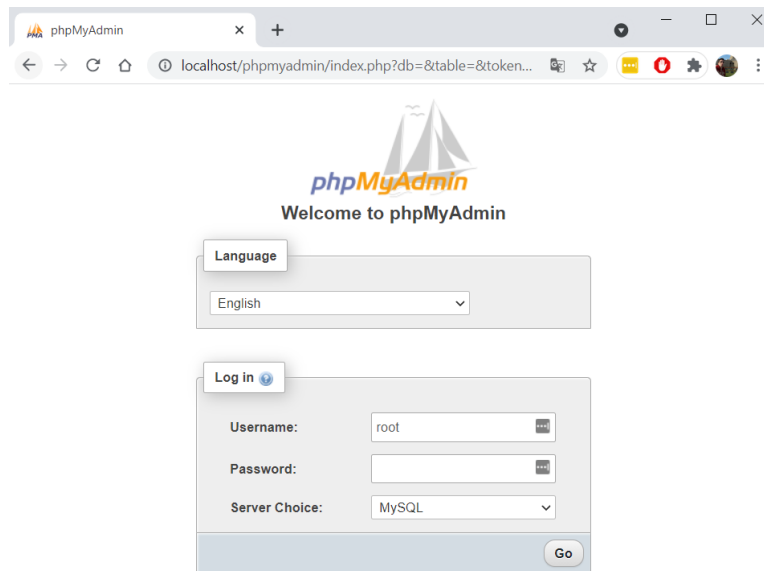


Figure 10-3. XAMPP phpMyAdmin application.

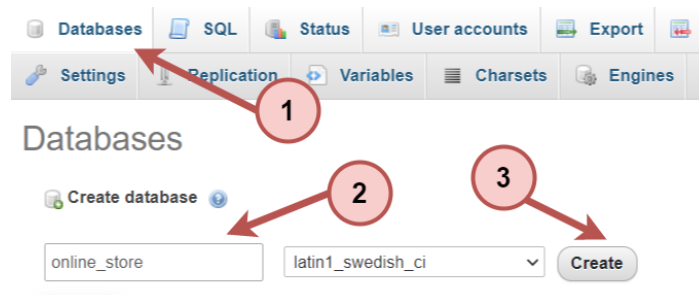


Figure 10-4. Database creation.

```
PS C:\xampp\htdocs\onlineStore> php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (26.54ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (24.25ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (24.61ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (34.73ms)
Migrating: 2022_02_11_153916_create_products_table
Migrated: 2022_02_11_153916_create_products_table (11.84ms)
```

Figure 11-1. Execution of Laravel migrations.

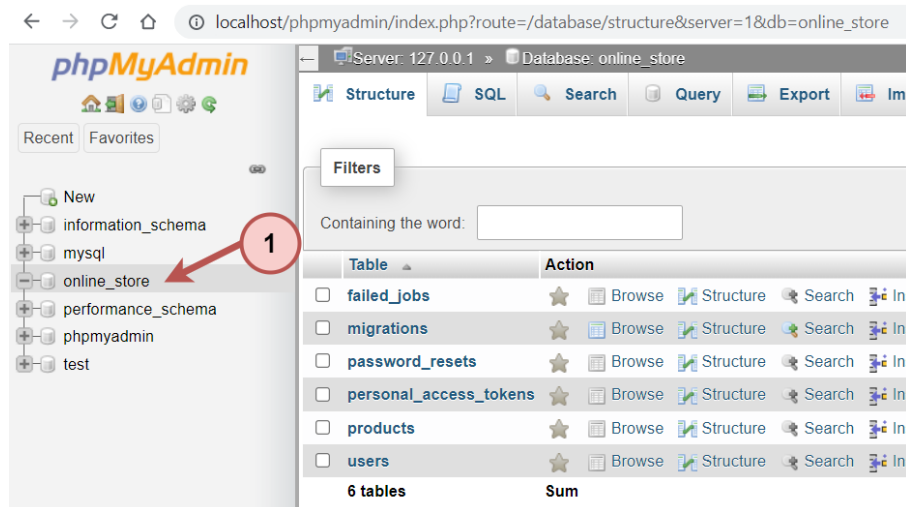


Figure 11-2. Accessing our database from phpMyAdmin.

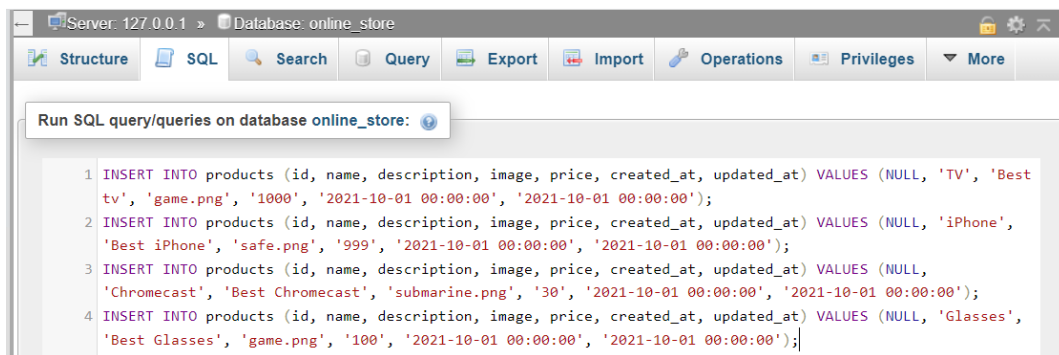


Figure 11-3. Inserting products through phpMyAdmin.

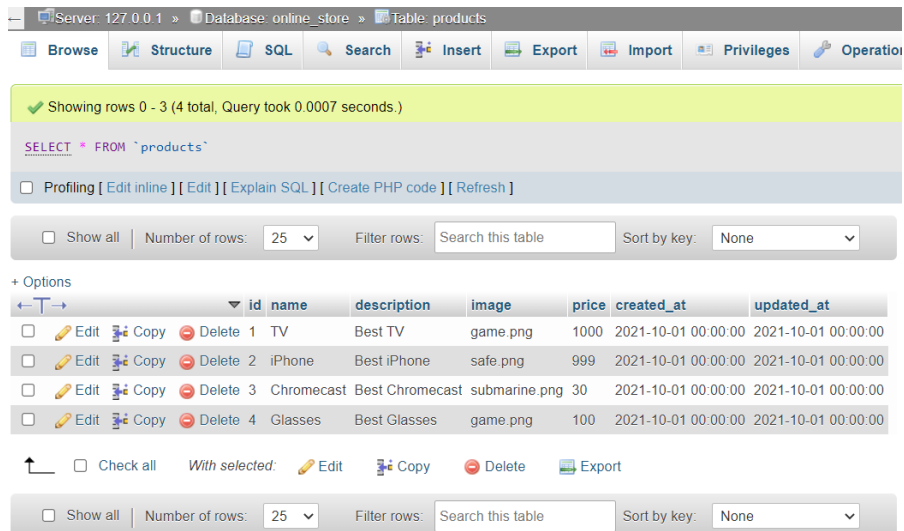


Figure 11-4. Products displayed in phpMyAdmin.

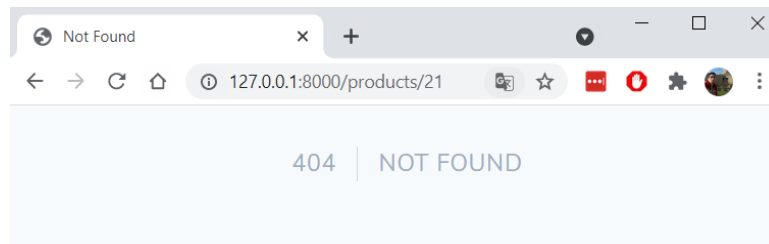


Figure 13-1. Accessing a product that does not exist.

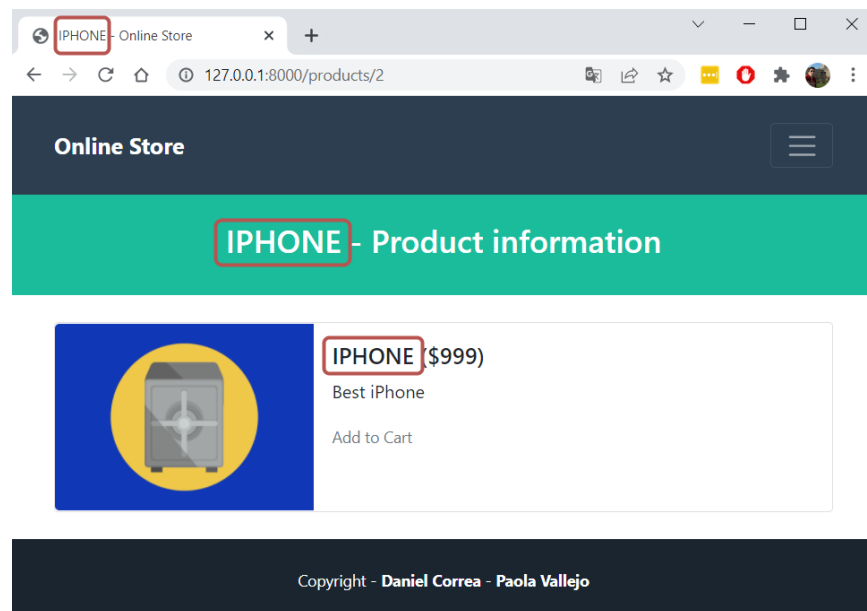


Figure 14-1. Accessing a product with the modified *getName* method.

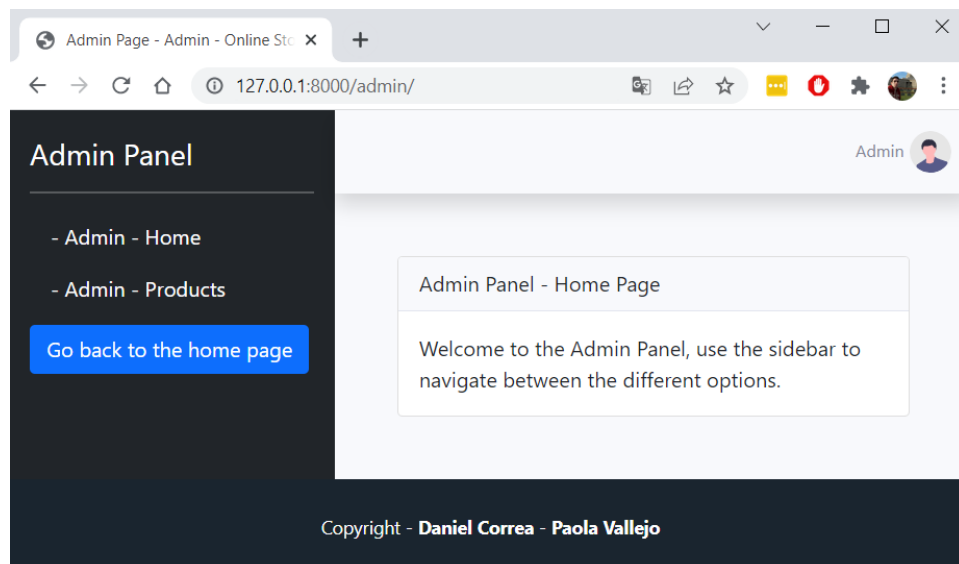


Figure 15-1. Online Store – Admin Panel.

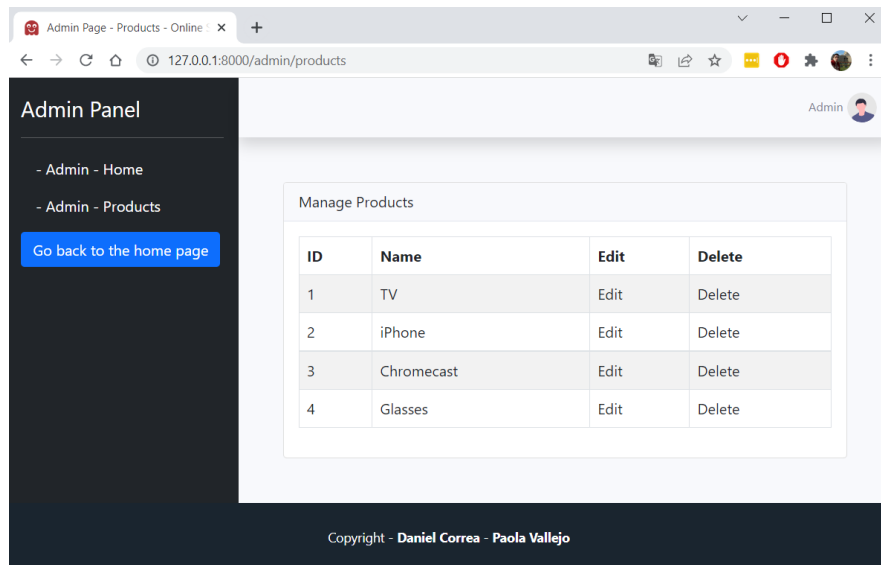


Figure 16-1. Online Store – Admin Panel – Products.

The screenshot shows a "Create Products" form. At the top, there is a red error message: "- The price field is required." Below this, the form has the following fields:

- Name:** A text input field containing the value "Mouse".
- Price:** An empty text input field.
- Description:** A text area containing the text "Minimalist mouse with laser".

At the bottom of the form is a blue "Submit" button.

Figure 17-1. Submitting a form with invalid data.

The screenshot shows the same "Admin Panel" as in Figure 16-1, but the main content area now displays the "Create Products" form. The form is filled with the following data:

- Name:** A text input field containing the value "SmartWatch".
- Price:** A text input field containing the value "325".
- Description:** A text area containing the text "Best SmartWatch in the world".

The blue "Submit" button is still present at the bottom of the form.

Figure 17-2. Online Store – Refined Admin Panel – Products.

ID	Name	Edit	Delete
1	TV	Edit	Delete
2	iPhone	Edit	Delete
3	Chromecast	Edit	Delete
4	Glasses	Edit	Delete
5	SmartWatch	Edit	Delete

Figure 17-3. New product listed.

```
PS C:\xampp\htdocs\onlineStore> php artisan storage:link
The [C:\xampp\htdocs\onlineStore\public\storage] link has been connected to [C:\xampp\htdocs\onlineStore\storage\app/public].
The links have been created.
```

Figure 18-1. Creating the storage symbolic link.

Admin Panel

- Admin - Home
- Admin - Products
- [Go back to the home page](#)

Admin

Create Products

Name: Price:

Image:

Description

Figure 18-2. Online Store – Create products form with image selection.

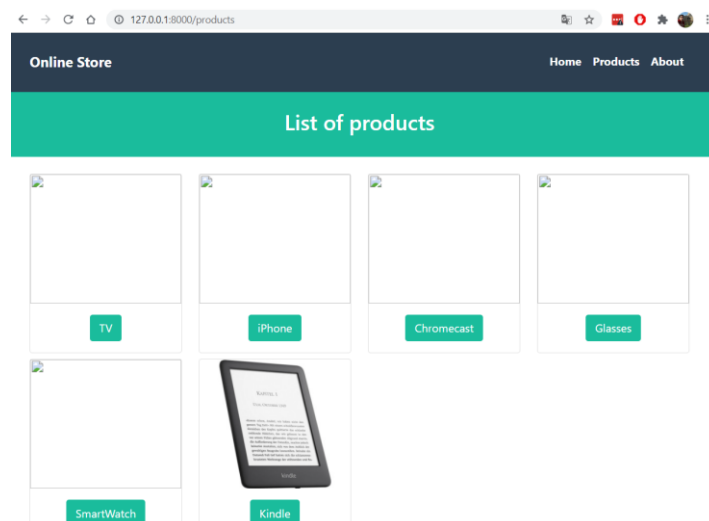


Figure 18-3. Online Store – List products with their uploaded images.













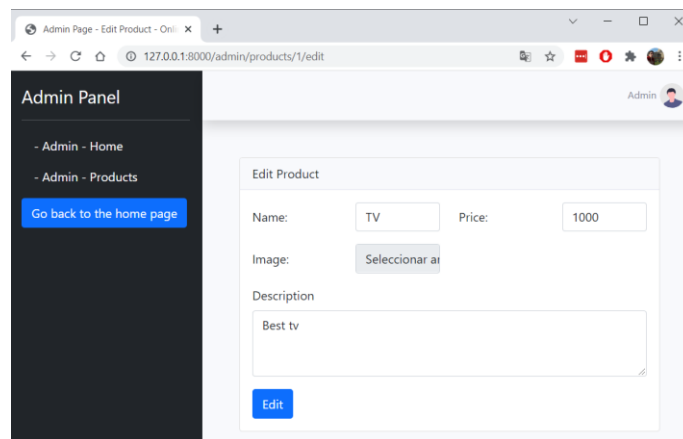
ID	Name	Edit	Delete
1	TV		
2	iPhone		
3	Chromecast		
4	Glasses		
5	SmartWatch		
6	Kindle		

Figure 19-1. Online Store – Refined Admin Panel – Products.



Admin Panel

- Admin - Home
- Admin - Products

[Go back to the home page](#)

Admin

Edit Product

Name: Price:

Image:

Description

Figure 19-2. Editing a product.

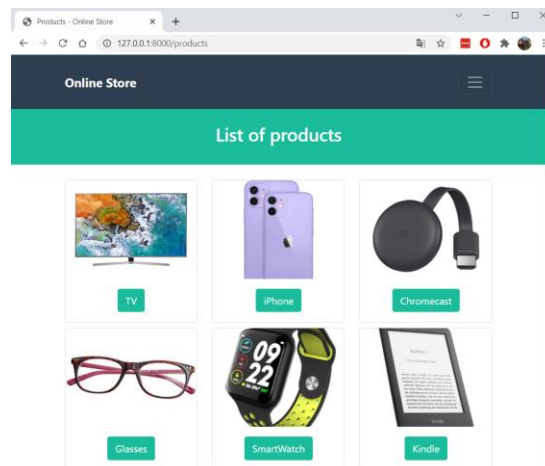


Figure 19-3. List products with proper images.

```

PS C:\xampp\htdocs\onlineStore> php artisan ui bootstrap --auth
Bootstrap scaffolding installed successfully.
Please run "npm install && npm run dev" to compile your fresh scaffolding.

The [layouts/app.blade.php] view already exists. Do you want to replace it? (yes/no) [no]:
> no

The [HomeController.php] file already exists. Do you want to replace it? (yes/no) [no]:
> no

Authentication scaffolding generated successfully.

```

Figure 21-1. Configuring laravel/ui.

Figure 21-2. Online Store – Register page.

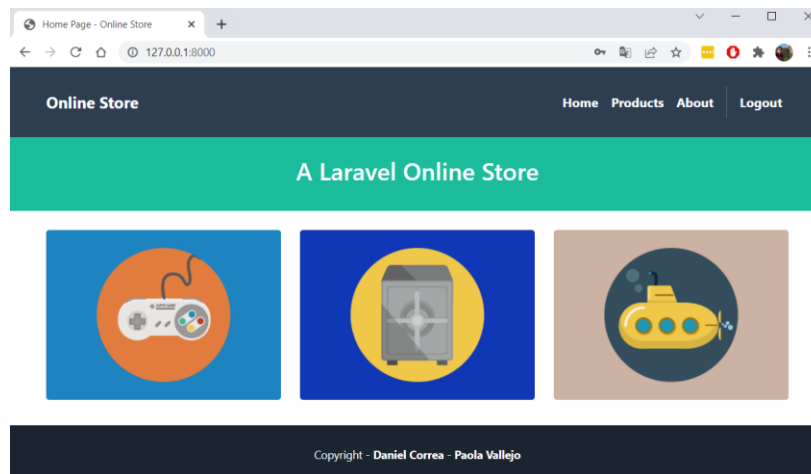


Figure 21-3. Online Store – Home page for authenticated users.

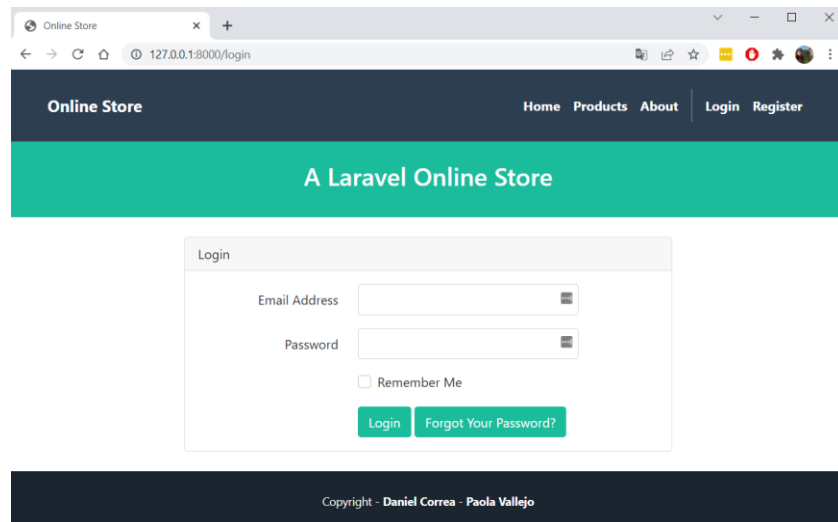


Figure 21-4. Online Store – Login page.

```
PS C:\xampp\htdocs\onlineStore> php artisan migrate
Migrating: 2022_02_12_140820_alter_users_table
Migrated: 2022_02_12_140820_alter_users_table (10.75ms)
```

Figure 22-1. Execution of alter users table migration.

created_at	updated_at	role	balance
2022-02-12 13:53:31	2022-02-12 13:53:31	client	0
2022-02-12 14:15:54	2022-02-12 14:15:54	client	5000

Figure 22-2. New columns and rows in the users table.

created_at	updated_at	role	balance
2022-02-12 13:53:31	2022-02-12 13:53:31	admin	0
2022-02-12 14:15:54	2022-02-12 14:15:54	client	5000

Figure 22-3. Editing a user's role.

```
PS C:\xampp\htdocs\onlineStore> php artisan tinker
Psy Shell v0.11.1 (PHP 8.0.11 - cli) by Justin Hileman
>>> $user = new App\Models\User();
=> App\Models\User {#3551}
>>> $user->setName('Daniel');
=> null
>>> $user->setEmail('daniel@danielgara.com');
=> null
>>> $user->setPassword(bcrypt('passwordVerySecret'));
=> null
>>> $user->setBalance(5000);
=> null
>>> $user->setRole('admin');
=> null
>>> $user->save();
=> true
>>> exit;
Exit: Goodbye
```

Figure 22-4. Create a user with Laravel Tinker.

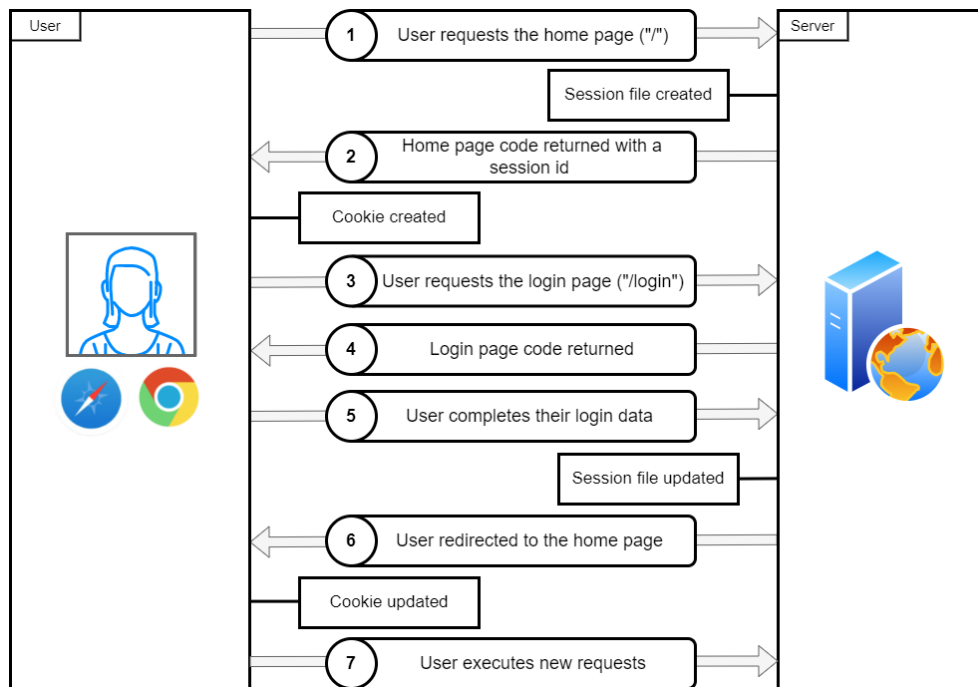


Figure 24-1. Typical Laravel sessions operation.

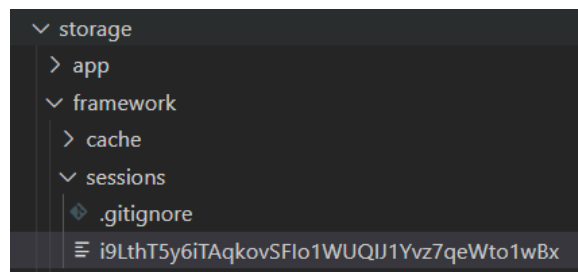


Figure 24-2. Generated session file.

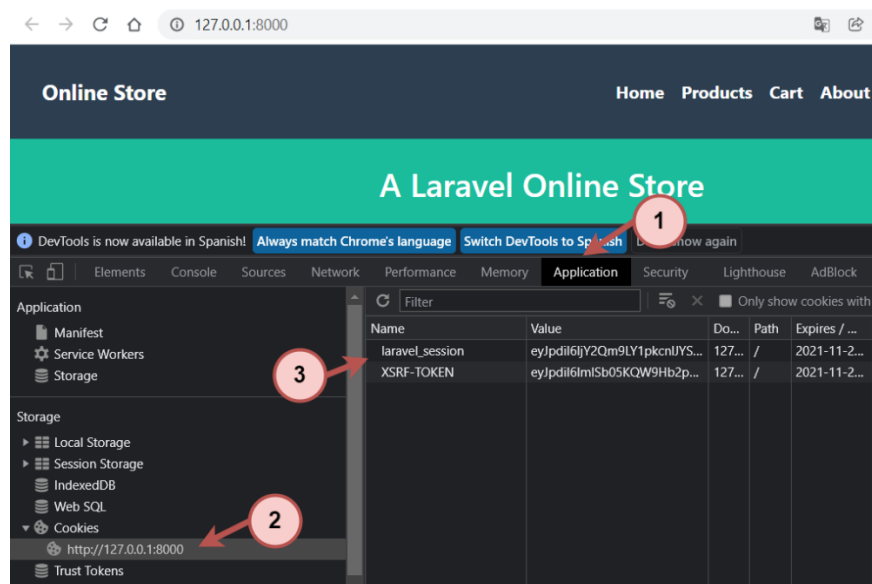


Figure 24-3. Generated cookie (Chrome DevTools).

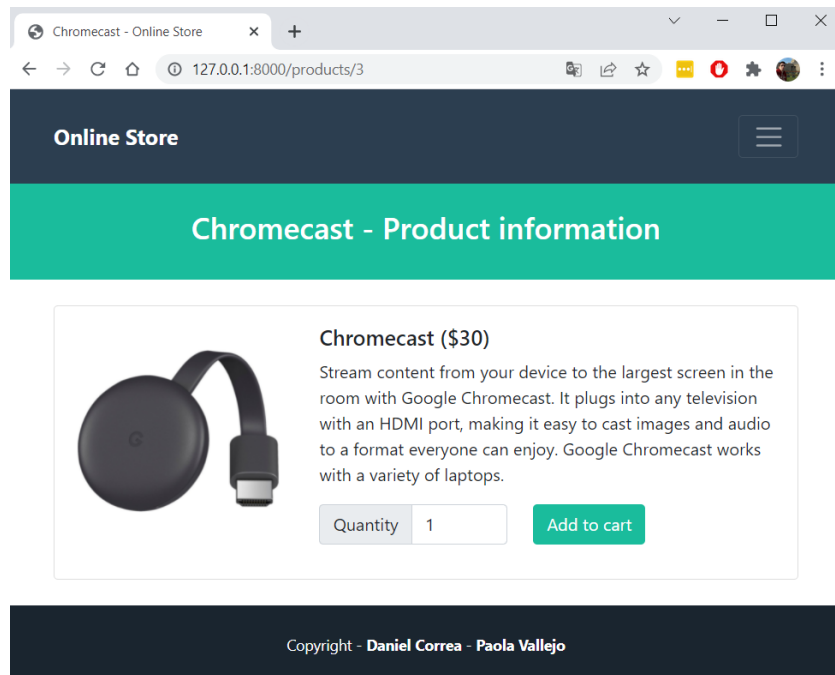


Figure 25-1. Product show view with add to cart button.

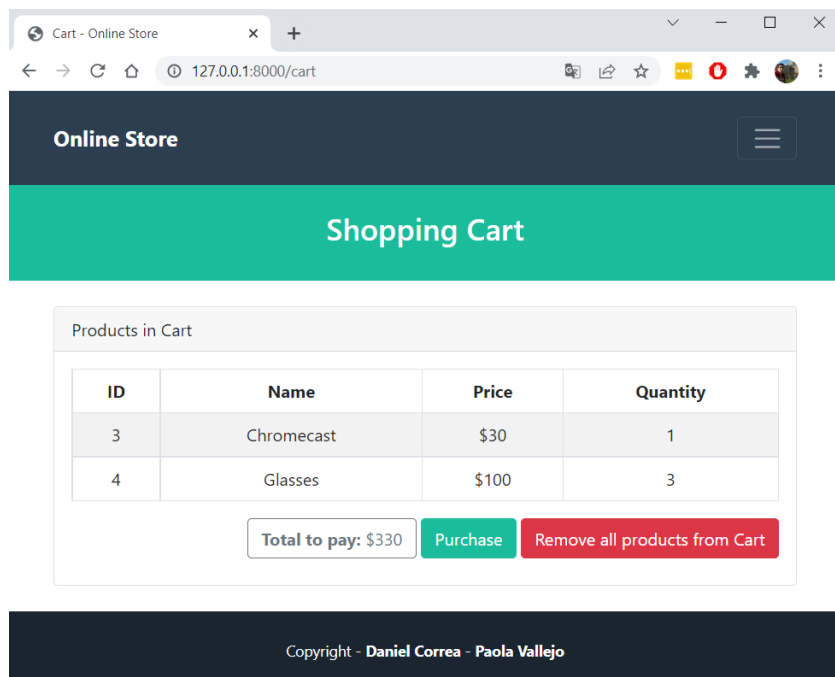


Figure 25-2. Online Store – Shopping cart page.

Order #1

Date: 2021-10-14

Total: \$1060

User id - name: 1 - Daniel

Qty	Product id - name	Price
1	1 - TV	1000
2	3 - Chromecast	30

Figure 26-1. Example of an invoice.

```
PS C:\xampp\htdocs\onlineStore> php artisan migrate
Migrating: 2022_02_12_152713_create_orders_table
Migrated: 2022_02_12_152713_create_orders_table (55.03ms)
Migrating: 2022_02_12_152850_create_items_table
Migrated: 2022_02_12_152850_create_items_table (170.14ms)
```

Figure 26-2. Execution of orders and items tables migrations.

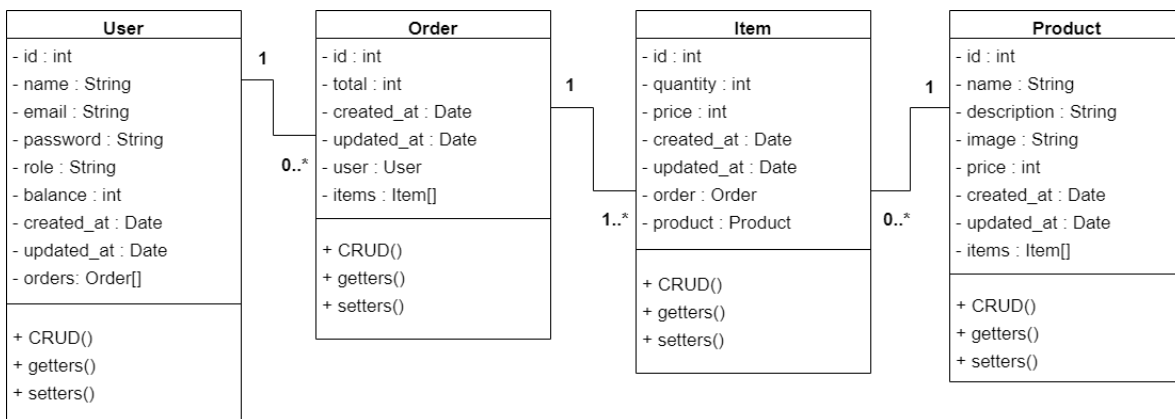


Figure 26-3. Original Online Store class diagram.

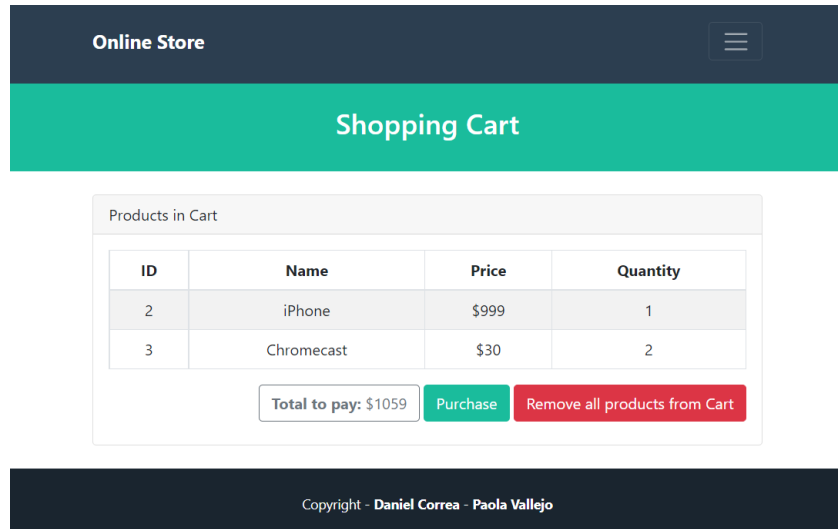


Figure 27-1. Shopping cart with some products.

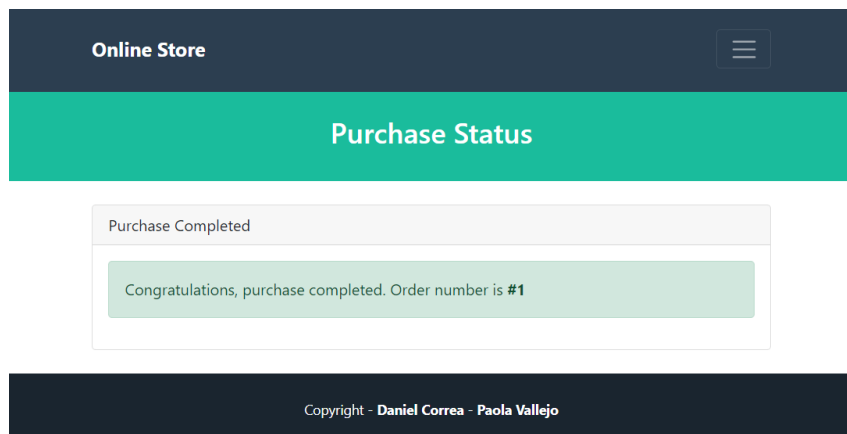


Figure 27-2. Purchase completed message.

id	total	user_id	created_at	updated_at
1	1059	1	2022-02-12 15:58:56	2022-02-12 15:58:56

Figure 27-3. New order row included in the database.

id	quantity	price	order_id	product_id	created_at	updated_at
1	1	999	1	2	2022-02-12 15:58:56	2022-02-12 15:58:56
2	2	30	1	3	2022-02-12 15:58:56	2022-02-12 15:58:56

Figure 27-4. New item rows included in the database.

```
PS C:\xampp\htdocs\onlineStore> php artisan tinker
Psy Shell v0.11.1 (PHP 8.0.11 - cli) by Justin Hileman
>>> Product::where('name', 'iPhone')->first()->getItems()[0]->getOrder()->getUser()->getName();
[!] Aliasing 'Product' to 'App\Models\Product' for this Tinker session.
=> "Daniel Correa"
>>> exit;
Exit: Goodbye
```

Figure 28-1. Use of Laravel tinker.

Online Store

My Orders

Order #1

Date: 2022-02-12 15:58:56

Total: \$1059

Item ID	Product Name	Price	Quantity
1	iPhone	\$999	1
2	Chromecast	\$30	2

Copyright - Daniel Correa - Paola Vallejo

Figure 28-2. Online Store – Orders page.

Online Store

My Orders

Seems to be that you have not purchased anything in our store =(.

Copyright - Daniel Correa - Paola Vallejo

Figure 28-3. Online Store – Orders page with no orders.

← → ↻ 🏠 127.0.0.1:8000/my-account/orders

Online Store

Home Products Cart About

My Orders

Order #1

Date: 2021-10-15 18:26:59
Total: \$1059

🔗	Messages	Timeline	Exceptions	Views 2	Route	Queries 9	Models 12	Mails	Gate	Session	Request	➡ GET my-account/orders 🔗
📄 9 statements were executed, 2 of which were duplicated, 7 unique												
select * from `users` where `id` = 1 limit 1 17.2ms 🔗 \vendor\laravel\framework\src\Illuminate\Auth\Eloquent												
select * from `orders` where `user_id` = 1 520µs 🔗 \app\Http\Controllers\MyAccou												
select * from `items` where `items`.`order_id` = 1 and `items`.`order_id` is not null 500µs 🔗 \app\M												
select * from `products` where `products`.`id` = 2 limit 1 570µs 🔗 \app\												
select * from `products` where `products`.`id` = 3 limit 1 470µs 🔗 \app\												
select * from `items` where `items`.`order_id` = 2 and `items`.`order_id` is not null 490µs 🔗 \app\M												
select * from `products` where `products`.`id` = 4 limit 1 470µs 🔗 \app\												
select * from `items` where `items`.`order_id` = 3 and `items`.`order_id` is not null 530µs 🔗 \app\M												
select * from `products` where `products`.`id` = 3 limit 1 470µs 🔗 \app\												

Figure 28-4. Online Store – Orders page with debug bar and lazy loading.

My Orders - Online Store

127.0.0.1:8000/my-account/orders

Online Store Home

My Orders

Order #1

Date: 2021-10-15 18:26:59

Total: \$1059

Messages	Timeline	Exceptions	Views 2	Route	Queries 4	Models 11	Mails	Gate	Session	Request
----------	----------	------------	---------	-------	-----------	-----------	-------	------	---------	---------

4 statements were executed

```

select * from `users` where `id` = 1 limit 1
select * from `orders` where `user_id` = 1
select * from `items` where `items`.`order_id` in (1, 2, 3)
select * from `products` where `products`.`id` in (2, 3, 4)

```

Figure 28-5. Online Store – Orders page with debug bar and eager loading.

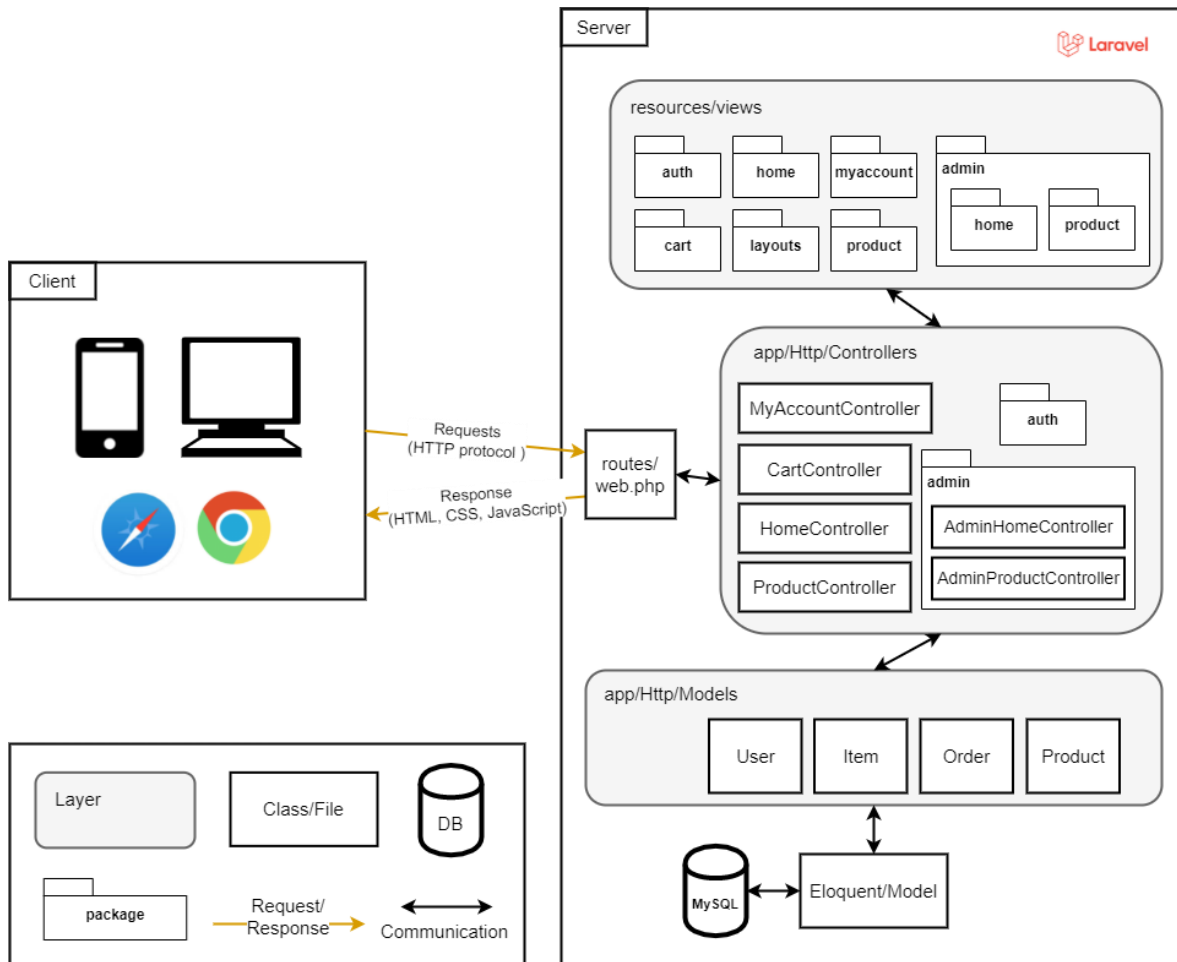


Figure 28-6. Online Store software architecture.

Sign up with Email

EMAIL ADDRESS

books@danielgara.com

PASSWORD

.....

☒ By clicking this I have read and agree to the [Terms & Conditions](#).

SIGN UP WITH EMAIL

Figure 29-1. Clever Cloud Sign up page.

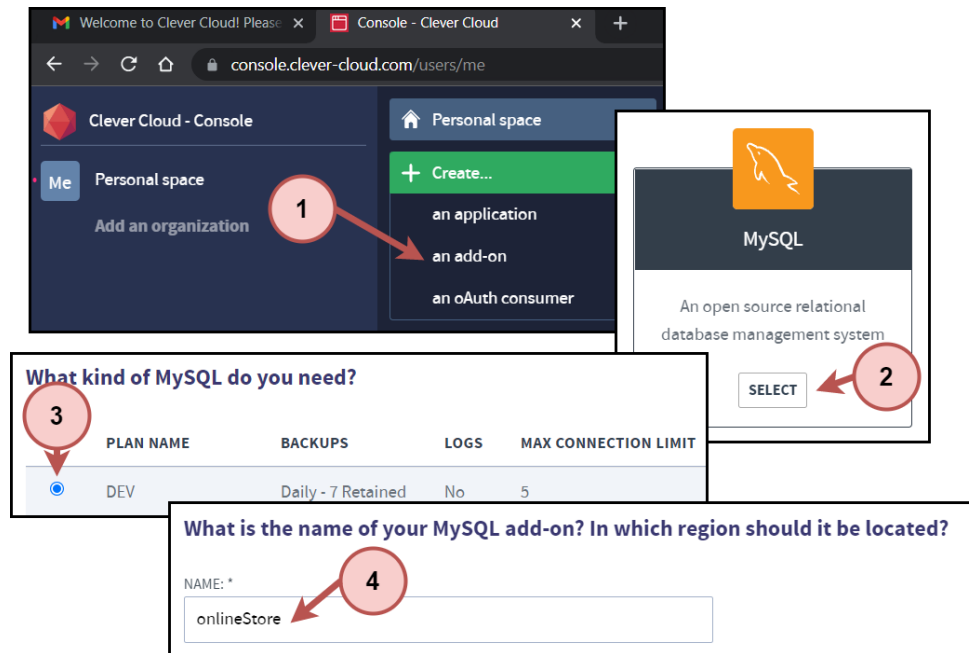


Figure 29-2. Clever Cloud MySQL add-on creation.

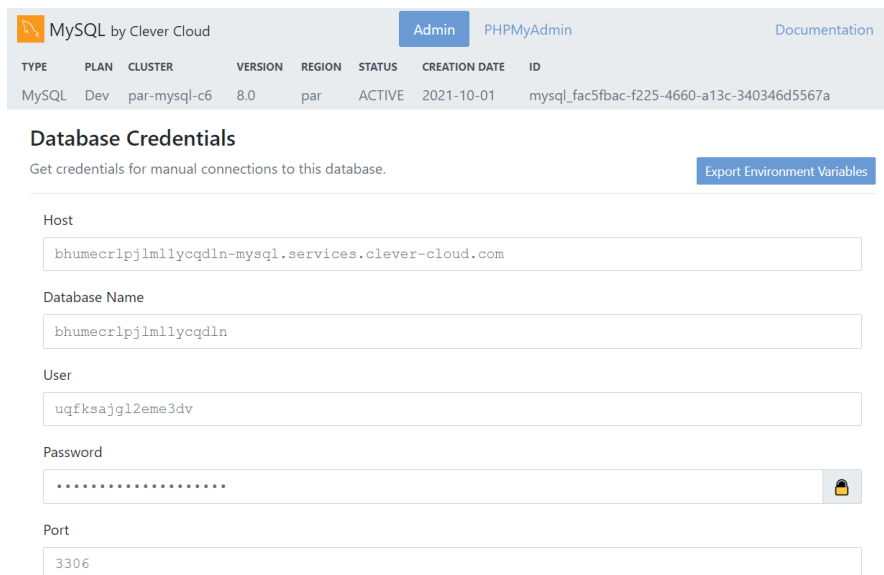


Figure 29-3. Clever Cloud MySQL add-on creation.

```

PS C:\xampp\htdocs\onlineStore> php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (694.42ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (716.37ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (765.13ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (1,131.34ms)
Migrating: 2022_02_11_153916_create_products_table
Migrated: 2022_02_11_153916_create_products_table (351.93ms)
Migrating: 2022_02_12_140820_alter_users_table
Migrated: 2022_02_12_140820_alter_users_table (345.62ms)
Migrating: 2022_02_12_152713_create_orders_table
Migrated: 2022_02_12_152713_create_orders_table (739.33ms)
Migrating: 2022_02_12_152850_create_items_table
Migrated: 2022_02_12_152850_create_items_table (1,114.81ms)

```

Figure 29-4. Execution of Laravel migrations over the cloud database.

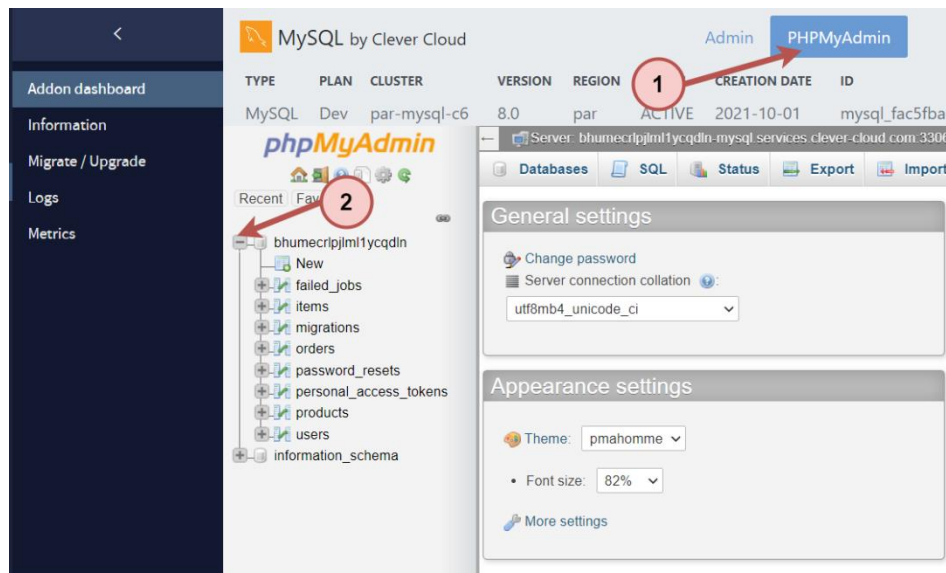


Figure 29-5. Accessing phpMyAdmin over the Clever Cloud database.

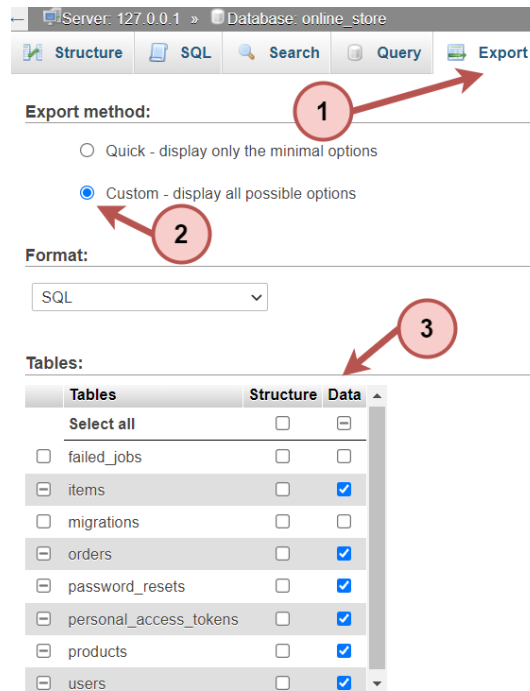


Figure 29-6. Exporting the data from the local database.

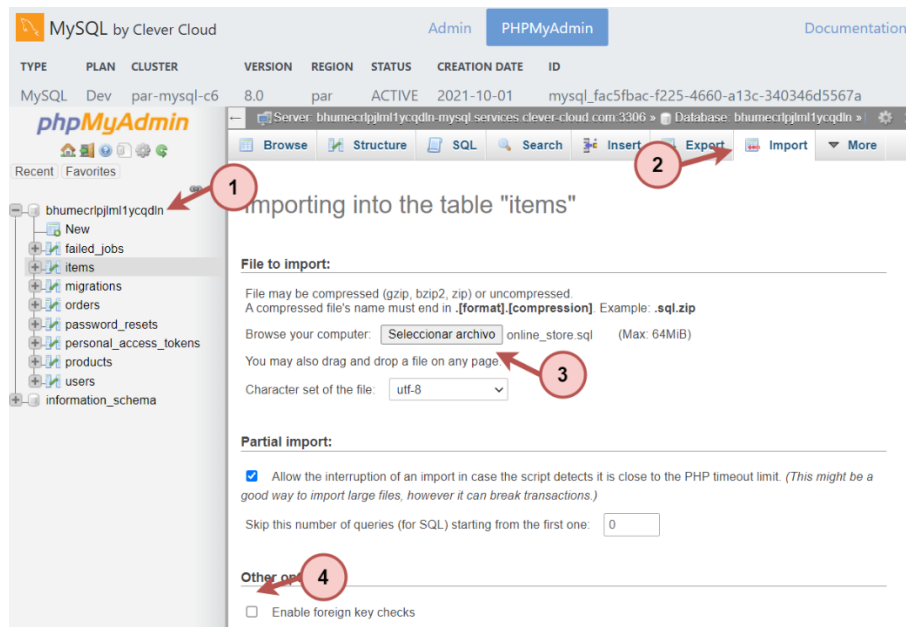


Figure 29-7. Importing the data to the cloud database.

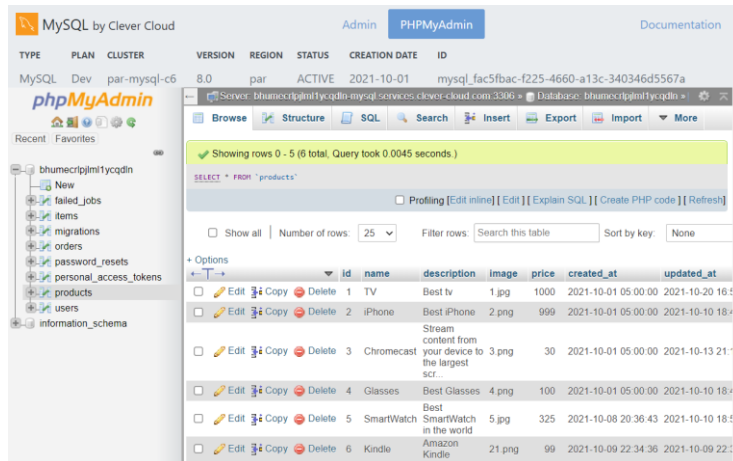


Figure 29-8. Browsing the *products* table.

```
PS C:\xampp\htdocs\onlineStore> heroku create
Creating app... done, powerful-savannah-94864
https://powerful-savannah-94864.herokuapp.com/ | https://git.heroku.com/powerful-savannah-94864.git
```

Figure 30-1. Creating a Heroku application.

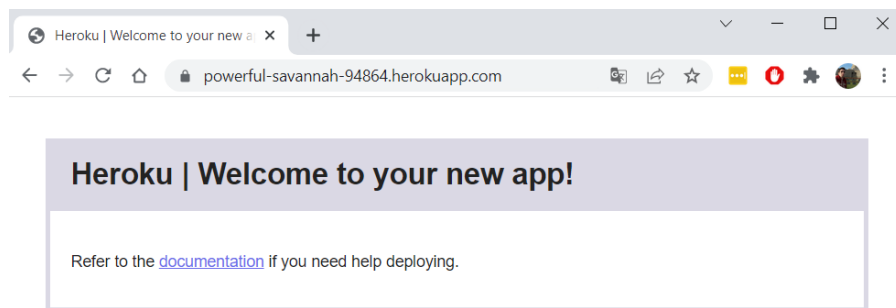


Figure 30-2. Accessing the Heroku application.

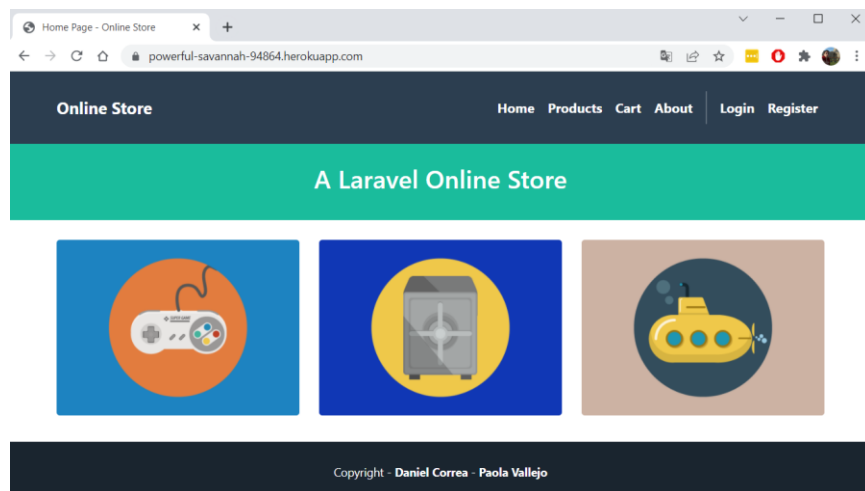


Figure 30-3. Online Store running over the cloud.