

ASSIGNMENT: BACKEND DEVELOPMENT WITH LARAVEL

Name : NURUL HAFIZAH BINTI TAHA

Course: SEC Bootcamp

Date : 29 December 2025

Assignment :

1. Part A : Project Setup
2. Part B : Model & Migration
3. Part C : Controller & Routes
4. Part D : API Testing with Bruno

This file is the compilation of screenshots required for the assignment

PART A : PROJECT SETUP

.env file for DB configuration

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "OPEN EDITORS". The ".env" file is selected in the sidebar.
- Editor:** Displays the contents of the ".env" file in the main editor area. The file contains environment variables for database configuration and session settings.
- Terminal:** Shows the command "hp artisan make:model products -m" being run, followed by two informational messages indicating successful model and migration creation.
- Bottom Status Bar:** Provides system information including weather (25°C, mostly cloudy), search bar, taskbar icons, and system status (language: ENG US, date: 12/23/2025).

```
products > .env
17 LOG_CHANNEL=stack
18 LOG_STACK=single
19 LOG_DEPRECATIONS_CHANNEL=null
20 LOG_LEVEL=debug
21
22
23 DB_CONNECTION=pgsql
24 DB_HOST=127.0.0.1
25 DB_PORT=5432
26 DB_DATABASE=products_db
27 DB_USERNAME=postgres
28 DB_PASSWORD=postgres
29
30 SESSION_DRIVER=database
31 SESSION_LIFETIME=120
32 SESSION_ENCRYPT=false
33 SESSION_PATH=/
34 SESSION_DOMAIN=null
35
```

```
hp artisan make:model products -m
[INFO] Model [C:\Users\Surface Pro 7 Plus\Documents\SEC Bootcamp\adnexio-bootcamp\backend-laravel\products\app\Models\products.php] created successfully.
[INFO] Migration [C:\Users\Surface Pro 7 Plus\Documents\SEC Bootcamp\adnexio-bootcamp\backend-laravel\products\database\migrations\2025_12_22_234552_create_products_table.php] created successfully.
```

Successful PHP artisan migrate output

The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, ...
- Search Bar:** Untitled (Workspace)
- Explorer:** Shows the project structure:
 - products
 - app
 - > Http
 - > Models
 - products.php
 - User.php
 - > Providers
 - > bootstrap
 - > config
 - database
 - > factories
 - > migrations
 - 0001_01_01_000000_create_products_table.php
 - 0001_01_01_000001_create_users_table.php
 - 0001_01_01_000002_create_products_table.php
 - 2025_12_22_234552_create_products_table.php
 - > seeders
 - .gitignore
 - > public
 - > resources
 - > routes
- Editor:** The current file is 2025_12_22_234552_create_products_table.php, which defines a database migration for creating a products table. The code includes schema definitions for name, description, price, and stock.
- Terminal:** The terminal window shows the command "php artisan migrate" being run, resulting in the message "INFO Running migrations." followed by "2025_12_22_234552_create_products_table 619.30ms DONE".
- Bottom Status Bar:** Shows the current branch is master*, 1 cm of rain Saturday, and the date is 12/23/2025.



PART B : MODEL & MIGRATION

Screenshot showing migration ran successfully

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows the project structure under the "products" folder:
 - app
 - Http\Controllers
 - Models
 - Providers
 - bootstrap
 - config
 - database
 - factories
 - migrations
 - 0001_01_01_000000_create_users_table
 - 0001_01_01_000001_create_cache_table
 - 0001_01_01_000002_create_jobs_table
 - 2025_12_22_234552_create_products_table
 - seeders
 - .gitignore
 - public
 - resources
- Editors:** Several files are open in tabs:
 - ProductsController.php
 - .env.example
 - README.md
 - api.php
- Terminal:** The terminal window shows the command `php artisan migrate` being run, with the output:

```
PS C:\Users\Surface Pro 7 Plus\Herd\backend-laravel\products> php artisan migrate
[INFO] Nothing to migrate.
```
- Bottom Status Bar:** Includes icons for file operations, search, and system status (language, battery, date).



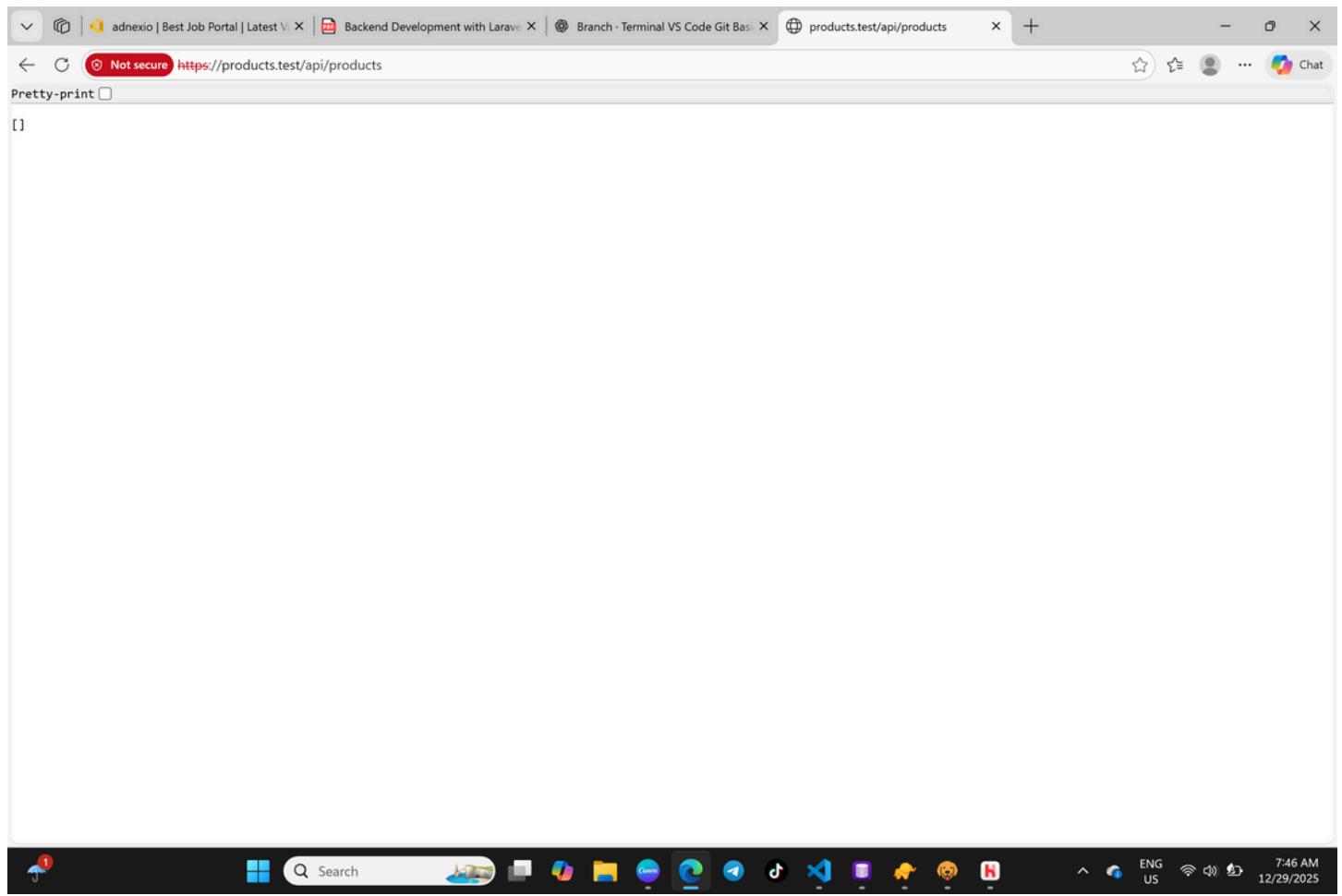
PART D : API TESTING WITH BRUNO

GET : Display Empty Products List

Bruno Display

The screenshot shows the Bruno API testing interface. On the left, there's a sidebar with 'Collections' and a search bar. The main area has a title 'Backend Laravel - Products' and a sub-section 'GET Display e...'. Below this, it says 'http://products.test/api/products'. The 'Body' tab is selected, showing 'No Body'. To the right, under 'Response', it shows a single line of JSON: '1 []'. At the top right, there are various icons for safe mode, local host, and other settings. The bottom of the window shows a taskbar with icons for file operations, search, and system status.

Browser Display



PART D : API TESTING WITH BRUNO

POST : Create a Product

Bruno Display

The screenshot shows the Bruno API testing tool interface. The top navigation bar includes 'Collection', 'Edit', 'View', 'Window', and 'Help'. A user icon 'bruno' is in the top left. The main title is 'Backend Laravel - Products'. Below it, there are tabs for 'GET Display e...' and 'POST Create a...'. The current tab is 'POST Create a Product'. The URL is 'http://products.test/api/products'. The 'Body' tab is selected, showing a JSON payload:

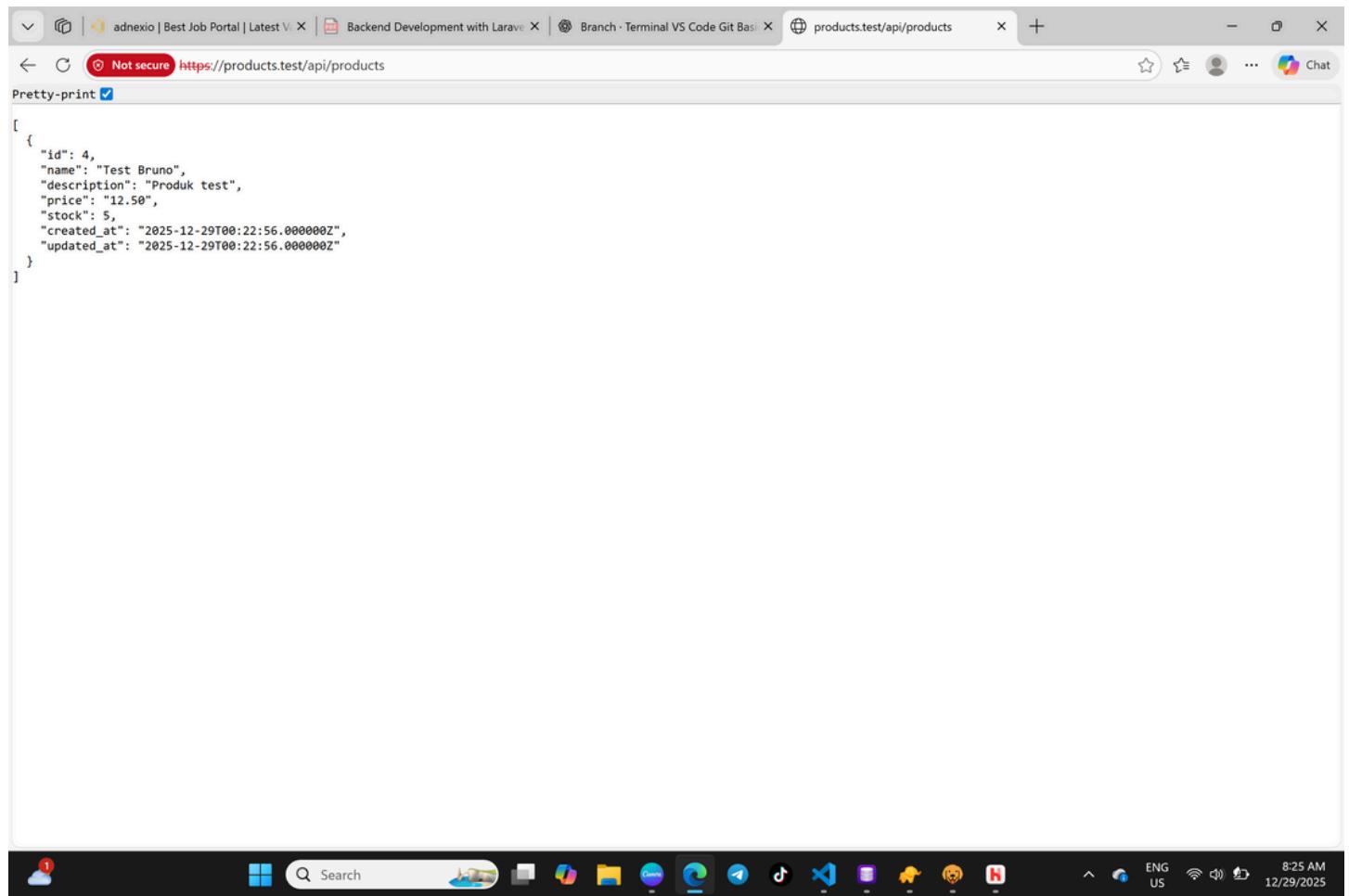
```
1  {
2   "name": "Test Bruno",
3   "description": "Produk test",
4   "price": 12.50,
5   "stock": 5
6 }
```

The 'Response' tab shows the API response in JSON format:

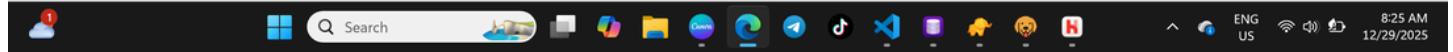
```
1  {
2   "success": true,
3   "data": {
4     "name": "Test Bruno",
5     "description": "Produk test",
6     "price": 12.5,
7     "stock": 5,
8     "updated_at": "2025-12-29T00:22:56.000000Z",
9     "created_at": "2025-12-29T00:22:56.000000Z",
10    "id": 4
11  }
12 }
```

The status bar at the bottom indicates 'v2.15.1'.

Browser Display



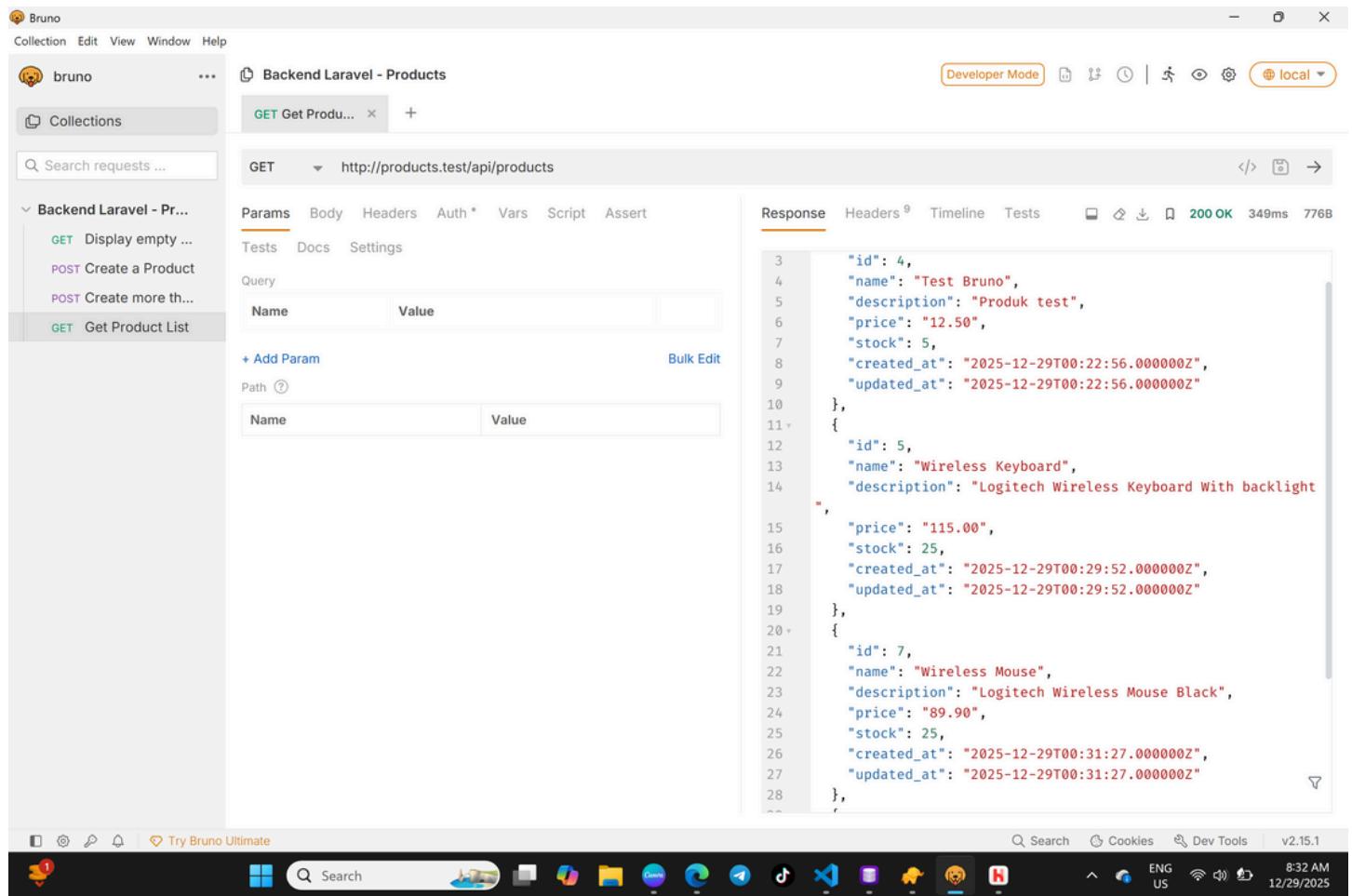
```
[{"id": 4, "name": "Test Bruno", "description": "produk test", "price": "12.50", "stock": 5, "created_at": "2025-12-29T00:22:56.000000Z", "updated_at": "2025-12-29T00:22:56.000000Z"}]
```



PART D : API TESTING WITH BRUNO

GET : Get All Products in List

Bruno Display



The screenshot shows the Bruno API testing tool interface. On the left, there's a sidebar with collections like 'Backend Laravel - Pr...' and a selected 'GET Get Product List'. The main area has tabs for 'Params', 'Body', 'Headers', 'Auth*', 'Vars', 'Script', and 'Assert'. A 'Query' section contains tables for 'Name' and 'Value' with a 'Bulk Edit' button. The 'Response' tab is active, showing a JSON response with three products. The first product is 'Test Bruno' with id 4, name 'Test Bruno', description 'Produk test', price '12.50', stock 5, created at '2025-12-29T00:22:56.000000Z', and updated at '2025-12-29T00:22:56.000000Z'. The second product is 'Wireless Keyboard' with id 5, name 'Wireless Keyboard', description 'Logitech Wireless Keyboard With backlight', price '115.00', stock 25, created at '2025-12-29T00:29:52.000000Z', and updated at '2025-12-29T00:29:52.000000Z'. The third product is 'Wireless Mouse' with id 7, name 'Wireless Mouse', description 'Logitech Wireless Mouse Black', price '89.90', stock 25, created at '2025-12-29T00:31:27.000000Z', and updated at '2025-12-29T00:31:27.000000Z'. The status bar at the bottom shows 'v2.15.1'.

```
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
{
  "id": 4,
  "name": "Test Bruno",
  "description": "Produk test",
  "price": "12.50",
  "stock": 5,
  "created_at": "2025-12-29T00:22:56.000000Z",
  "updated_at": "2025-12-29T00:22:56.000000Z"
},
{
  "id": 5,
  "name": "Wireless Keyboard",
  "description": "Logitech Wireless Keyboard With backlight",
  "price": "115.00",
  "stock": 25,
  "created_at": "2025-12-29T00:29:52.000000Z",
  "updated_at": "2025-12-29T00:29:52.000000Z"
},
{
  "id": 7,
  "name": "Wireless Mouse",
  "description": "Logitech Wireless Mouse Black",
  "price": "89.90",
  "stock": 25,
  "created_at": "2025-12-29T00:31:27.000000Z",
  "updated_at": "2025-12-29T00:31:27.000000Z"
},
```

Browser Display

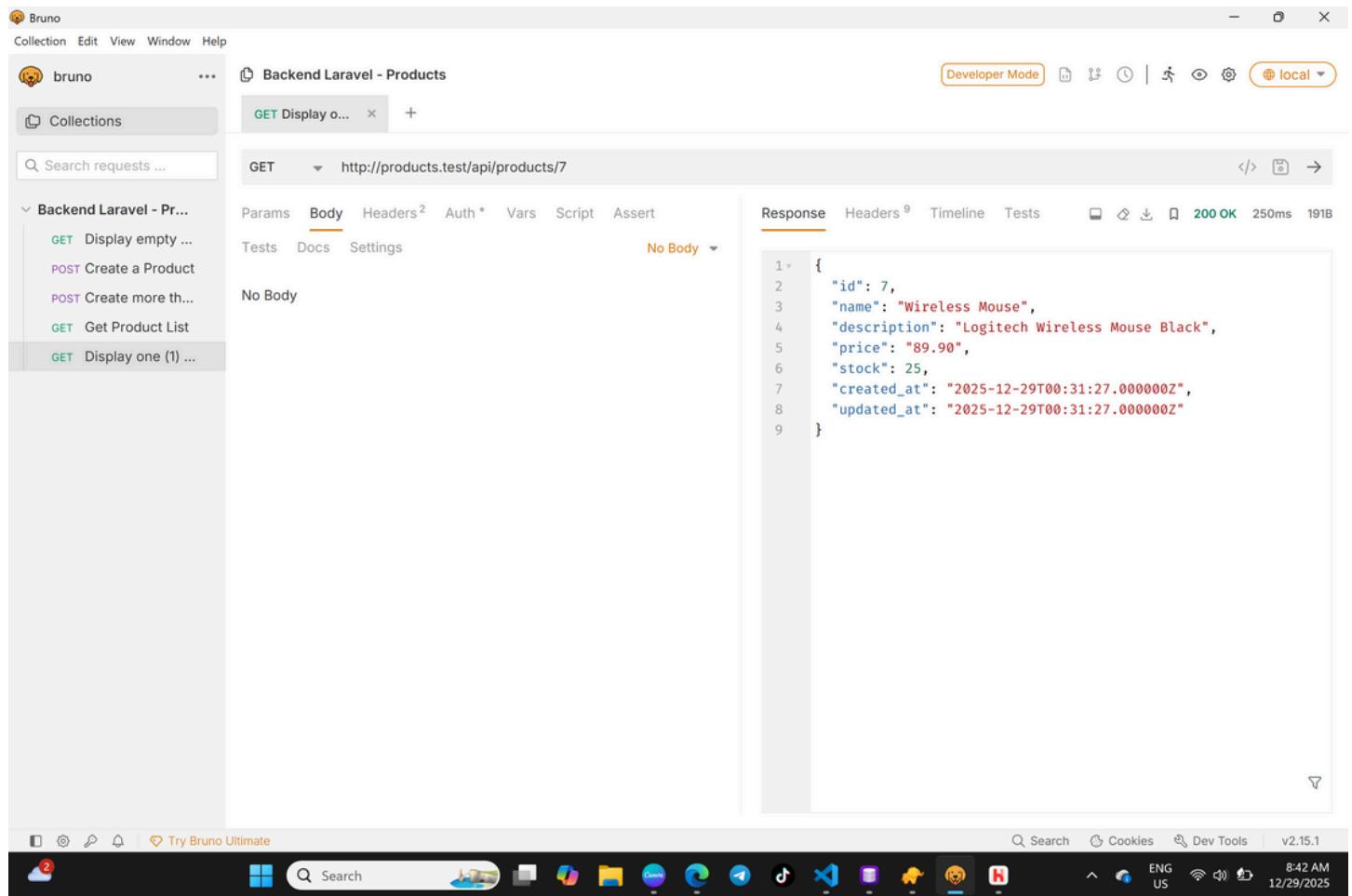
```
[  
  {  
    "id": 4,  
    "name": "Test Bruno",  
    "description": "produk test",  
    "price": "12.50",  
    "stock": 5,  
    "created_at": "2025-12-29T00:22:56.000002Z",  
    "updated_at": "2025-12-29T00:22:56.000002Z"  
  },  
  {  
    "id": 5,  
    "name": "Wireless Keyboard",  
    "description": "Logitech Wireless Keyboard With backlight",  
    "price": "115.00",  
    "stock": 25,  
    "created_at": "2025-12-29T00:29:52.000002Z",  
    "updated_at": "2025-12-29T00:29:52.000002Z"  
  },  
  {  
    "id": 7,  
    "name": "Wireless Mouse",  
    "description": "Logitech Wireless Mouse Black",  
    "price": "89.90",  
    "stock": 25,  
    "created_at": "2025-12-29T00:31:27.000002Z",  
    "updated_at": "2025-12-29T00:31:27.000002Z"  
  },  
  {  
    "id": 6,  
    "name": "Wireless Keyboard",  
    "description": "Logitech Wireless Keyboard No backlight",  
    "price": "185.00",  
    "stock": 25,  
    "created_at": "2025-12-29T00:30:54.000002Z",  
    "updated_at": "2025-12-29T00:30:54.000002Z"  
  }  
]
```



PART D : API TESTING WITH BRUNO

GET : Get ONE (1) Product from List

Bruno Display

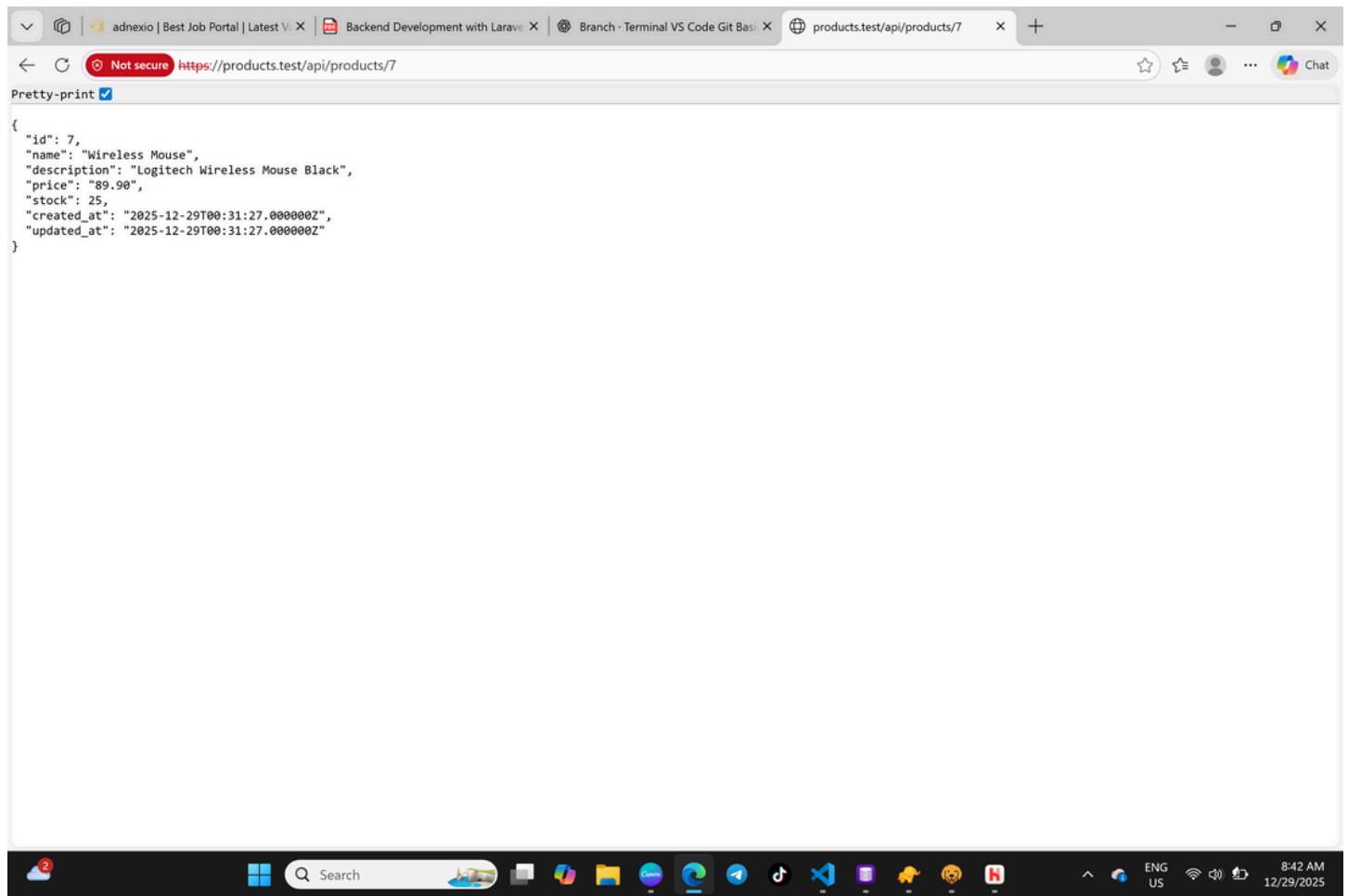


The screenshot shows the Bruno API testing tool interface. On the left, there's a sidebar with a tree view of collections, including "Backend Laravel - Pr..." which is expanded, showing several requests: "Display empty ...", "Create a Product", "Create more th...", "Get Product List", and "Display one (1) ...". The main area shows a request titled "GET Display o...". The URL is "http://products.test/api/products/7". The "Body" tab is selected, showing "No Body". The "Response" tab shows the JSON response:

```
1 {  
2   "id": 7,  
3   "name": "Wireless Mouse",  
4   "description": "Logitech Wireless Mouse Black",  
5   "price": "89.90",  
6   "stock": 25,  
7   "created_at": "2025-12-29T00:31:27.000000Z",  
8   "updated_at": "2025-12-29T00:31:27.000000Z"  
9 }
```

The status bar at the bottom indicates "v2.15.1".

Browser Display



PART D : API TESTING WITH BRUNO

PUT : Update A Product

Bruno Display

The screenshot shows the Bruno API testing tool interface. The top navigation bar includes 'Collection', 'Edit', 'View', 'Window', and 'Help'. On the left, there's a sidebar with a user icon labeled 'bruno' and sections for 'Collections' and 'Search requests ...'. The main workspace has tabs for 'Backend Laravel - Products' (active), 'Display on...', and 'Update a ...'. Below these tabs, a search bar contains 'PUT http://products.test/api/products/6'. The 'Body' tab is selected, showing a JSON payload:

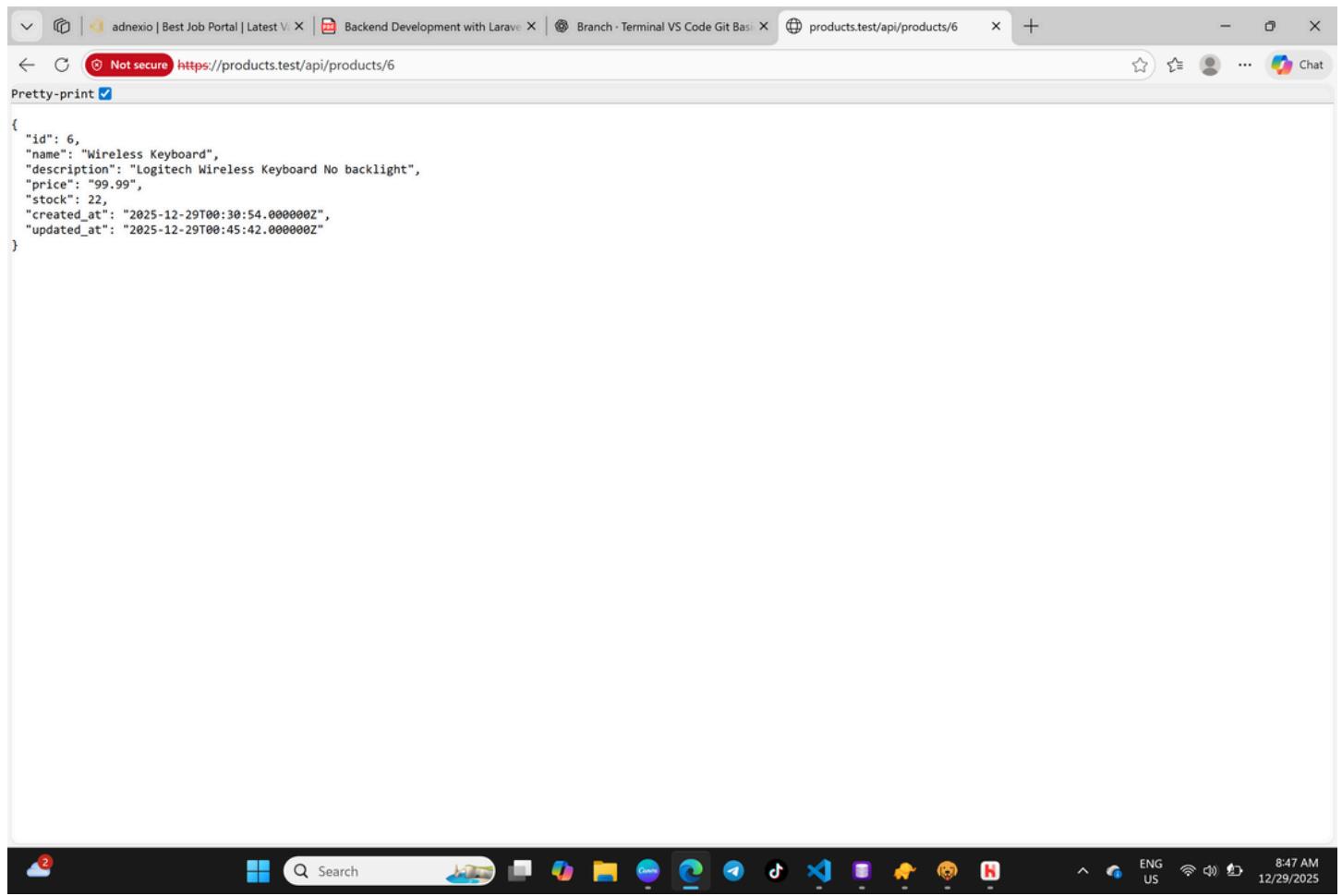
```
1 {  
2   "price": 99.99,  
3   "stock": 22  
4 }
```

The 'Response' tab shows the successful response in JSON format:

```
1 {  
2   "success": true,  
3   "data": {  
4     "id": 6,  
5     "name": "Wireless Keyboard",  
6     "description": "Logitech Wireless Keyboard No backlight",  
7     "price": 99.99,  
8     "stock": 22,  
9     "created_at": "2025-12-29T00:30:54.000000Z",  
10    "updated_at": "2025-12-29T00:45:42.000000Z"  
11  }  
12 }
```

The status bar at the bottom indicates 'v2.15.1'.

Browser Display



Not secure https://products.test/api/products/6

Pretty-print

```
{  
    "id": 6,  
    "name": "Wireless Keyboard",  
    "description": "Logitech Wireless Keyboard No backlight",  
    "price": "99.99",  
    "stock": 22,  
    "created_at": "2025-12-29T00:30:54.000000Z",  
    "updated_at": "2025-12-29T00:45:42.000000Z"  
}
```



PART D : API TESTING WITH BRUNO

DELETE : Delete a Product

Bruno Display

The screenshot shows the Bruno API testing interface. On the left, there's a sidebar with a collection named "Backend Laravel - Products" containing various HTTP methods: GET, POST, PUT, and DEL. The "DEL" method is currently selected. The main area shows a request configuration for a DELETE operation to the URL `http://products.test/api/products/4`. The "Body" tab is active, showing a simple JSON payload with the number "1". The "Response" tab shows the result of the delete operation, which is a single digit "1". The top right corner indicates a status of "204 No Content" with a response time of "365ms". The bottom of the window shows a taskbar with various application icons.

TablePlus Display (Before Delete)

PostgreSQL 17.0 : TLS : PostgreSQL : products_db : ..

Items Queries History

Search for item...

Functions Tables

- cache
- cache_locks
- failed_jobs
- job_batches
- jobs
- migrations
- password_reset_to...
- products
- sessions
- users

products

id	name	description	price	stock	created_at	updated_at
4	Test Bruno	Produk test	12.50	5	2025-12-29..	2025-12-29..
5	Wireless Keyboard	Logitech Wireless Keybo...	115.00	25	2025-12-29..	2025-12-29..
6	Wireless Keyboard	Logitech Wireless Keybo...	105.00	25	2025-12-29..	2025-12-29..
7	Wireless Mouse	Logitech Wireless Mouse...	89.90	25	2025-12-29..	2025-12-29..

Details Assistant

Search for field...

id int8
6
name varchar
Wireless Keyboard
description text
Logitech Wireless Keyboard No backlight
price numeric
105.00
stock int4
25
created_at timestamp
2025-12-29 00:30:54
updated_at timestamp
2025-12-29 00:30:54

+ public Data Structure + Row 1 of 4 rows selected Filters Columns < >

Cloud icon 2 Search icon ENG US 9:13 AM 12/29/2025



TablePlus Display (After Delete)

The screenshot shows the TablePlus application interface for PostgreSQL 17.0. The main window displays a table named 'products' with three rows of data. The columns are: id, name, description, price, stock, created_at, and updated_at. The data is as follows:

	id	name	description	price	stock	created_at	updated_at
5	5	Wireless Keyboard	Logitech Wireless Keybo...	115.00	25	2025-12-29 00:31:27	2025-12-29 00:31:27
6	6	Wireless Keyboard	Logitech Wireless Keybo...	99.99	22	2025-12-29 00:31:27	2025-12-29 00:31:27
7	7	Wireless Mouse	Logitech Wireless Mouse...	89.90	25	2025-12-29 00:31:27	2025-12-29 00:31:27

The sidebar on the left lists various database objects: Functions, Tables, and specific tables like cache, cache_locks, failed_jobs, job_batches, jobs, migrations, password_reset_tokens, products, sessions, and users. The 'Products' table is currently selected. The right panel shows the detailed structure of the 'products' table, including its columns: id (int8), name (varchar), description (text), price (numeric), stock (int4), created_at (timestamp), and updated_at (timestamp). The 'name' column is currently set to 'Wireless Mouse'.

Browser Display (After Delete)

```
[{"id": 5, "name": "Wireless Keyboard", "description": "Logitech Wireless Keyboard With backlight", "price": "115.00", "stock": 25, "created_at": "2025-12-29T00:29:52.000000Z", "updated_at": "2025-12-29T00:29:52.000000Z"}, {"id": 7, "name": "Wireless Mouse", "description": "Logitech Wireless Mouse Black", "price": "89.90", "stock": 25, "created_at": "2025-12-29T00:31:27.000000Z", "updated_at": "2025-12-29T00:31:27.000000Z"}, {"id": 6, "name": "Wireless Keyboard", "description": "Logitech Wireless Keyboard No backlight", "price": "99.99", "stock": 22, "created_at": "2025-12-29T00:30:54.000000Z", "updated_at": "2025-12-29T00:45:42.000000Z"}]
```

