|  |  |  |
| --- | --- | --- |
|  | **Atelier 3 :**  **Applications web avancées** | **Année Universitaire 2023/2024** |

**Objectifs**

Le but de cet atelier est la manipulation des deux méthodes de récupération et d’invocation des données.

Soit le diagramme suivant :

|  |
| --- |
| **Product** |
| * Id   1   * Name * Description * Price * Stock |

|  |
| --- |
| **Category** |
| * Id   \*   * Name * Lastname * dateofbirth |

|  |
| --- |
| **Catalogue** |
| * Id   \*   * Name |

1

\*

\*

\*

\*

**Figure 1: Diagramme de classe**

**Partie 1 : CRUD Produit**

1. Créer la base de données « ***atelier3*** »
2. Créer une interface pour gérer les produits. La table suivante montre les détails :

|  |  |
| --- | --- |
| **Nom de la table** | **Product** |
| id | Auto\_increment |
| Name | String (20) |
| Description | String (200) |
| Price | Float |
| Stock | Integer |

1. Générer le modèle ainsi que le fichier de migration correspondant.
2. Ajouter le code suivant dans le fichier de migration, la fonction up.
3. Générer la table « ***Product***»
4. Générer le contrôleur de ressources associé à notre modèle « ***Product*** »

**NB :** Il faut maintenant remplir les différentes fonctions avec leurs traitements respectifs :

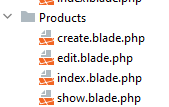
Nous avons comme correspondance :

* **index :** gère l'affichage de la page de listing des produits
* **create :** gère l'affichage de la page de création d'un produit
* **update :** gère la création d'un produit (suivant un formulaire de création par exemple)
* **show :** gère l'affichage de la page d'un produit
* **edit :** gère l'affichage de la page d'édition d'un produit
* **update :** gère la modification d'un produit (suivant un formulaire d'édition par exemple)
* **delete :** gère la suppression d'un produit.

1. Lister les routes de l’application

php artisan route:list

1. Modifier le fichier ***routes/web.php*** en ajoutant la ligne permettant de générer les routes de ressources pour les produits.
2. Ré-lister les routes de l’application.
3. Créer les vues de notre application dans le dossier view/products :



1. Créer le fichier ProductFactory qui correspond au modèle **« product** » :
2. ajouter le code suivant au fichier ProductFactory
3. Créer le fichier ProductSeeder qui correspond au modèle **« product** » :
4. ajouter le code suivant au fichier ProductSeeder :
5. Ajouter le code suivant au fichier DatabaseSeeder :
6. Exécuter maintenant la commande :

php artisan db:seed

Vérifier que les données ont été ajoutées à la base de données

1. Ajouter le code qui permet de récupérer la liste des produits

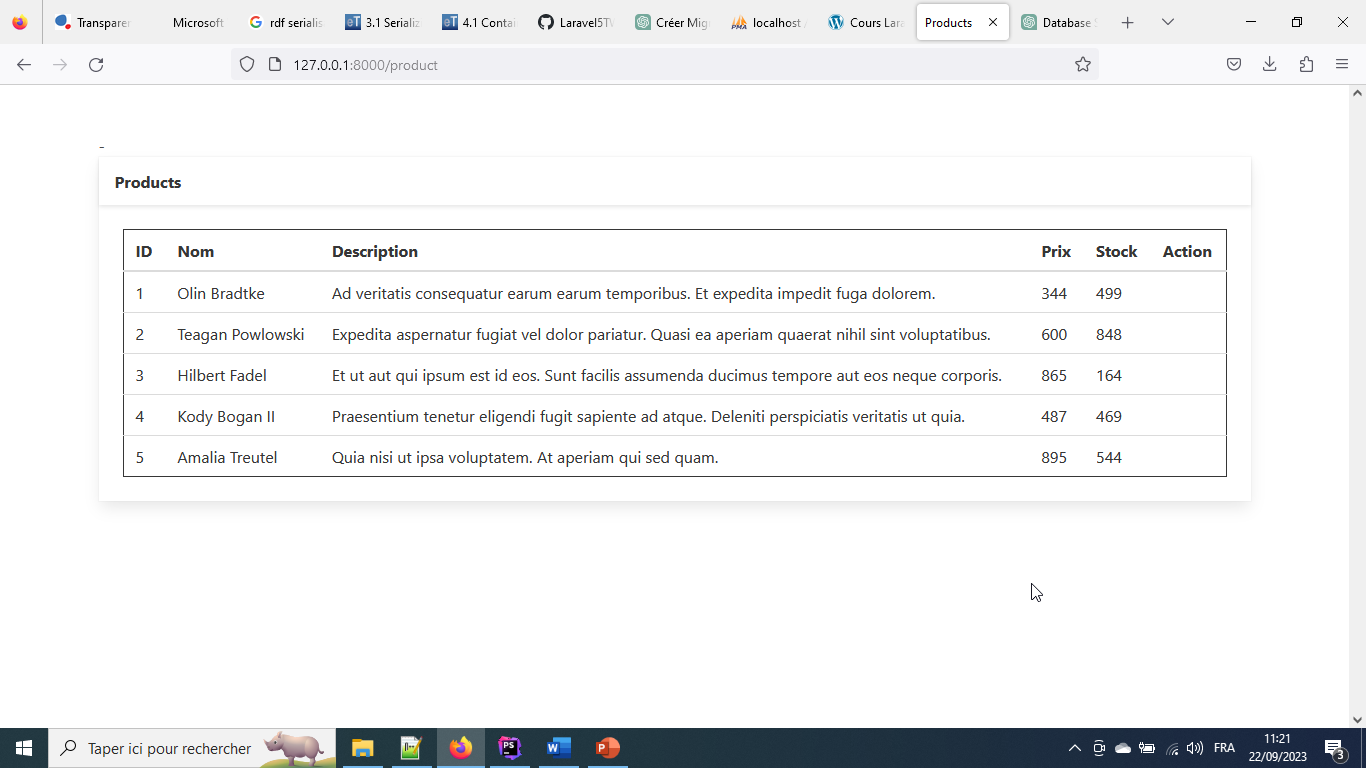
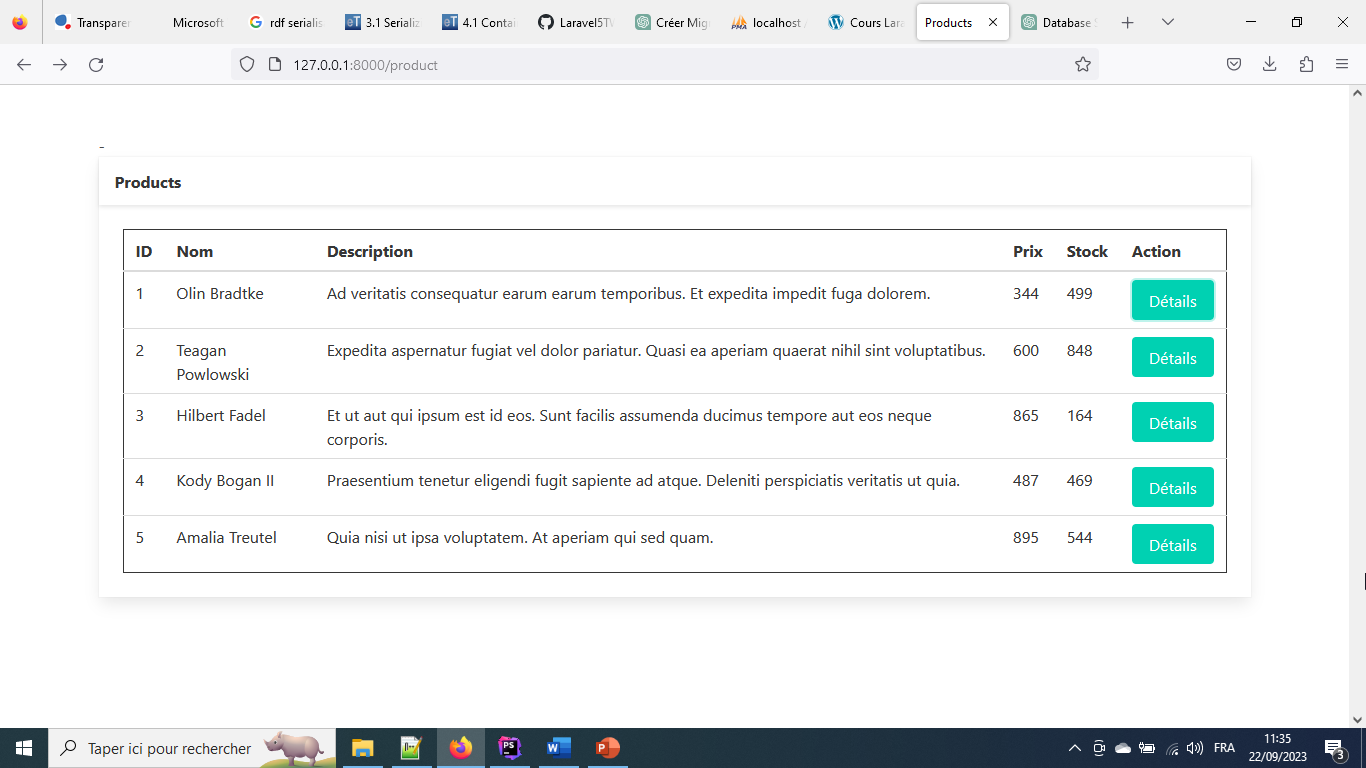


Figure 1: Page "**index.blade.php**"

1. Ajouter le code qui permet d’afficher les détails d’un produit



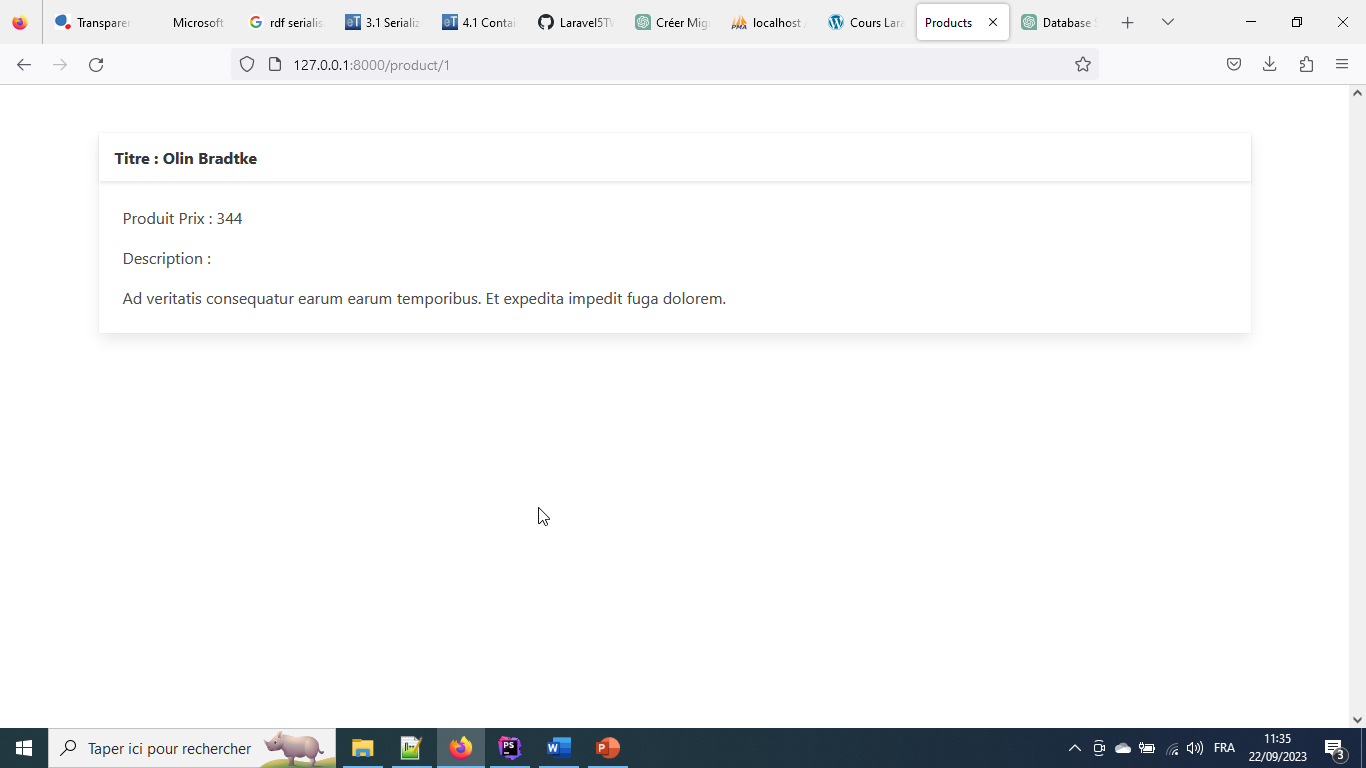
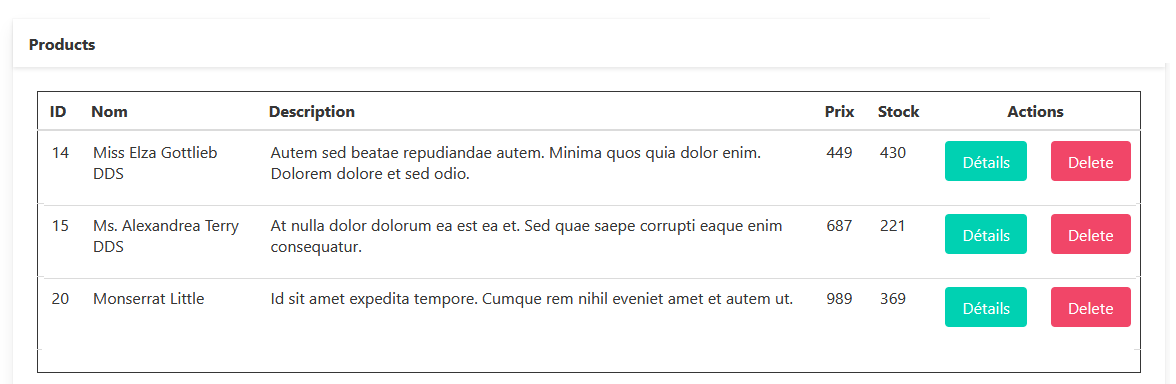


Figure 2: Page "**show.blade.php**"

1. Ajouter le code qui permet de supprimer un produit :



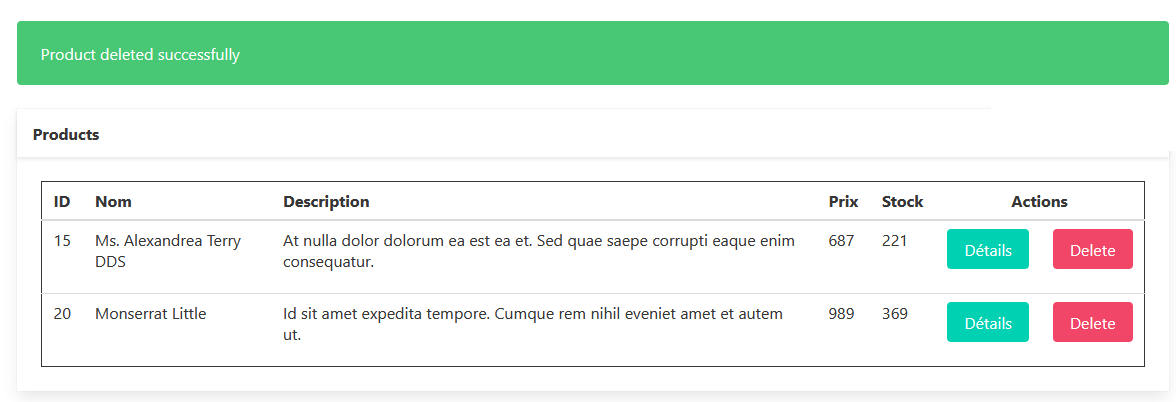


Figure 3: Page "**index.blade.php**"

1. Ajouter le code qui permet d’ajouter un produit:
2. Ajouter ce code dans le modèle Product

protected $fillable = [  
 'name',  
 'description' ,  
 'price',  
 'stock'  
];

**NB** : Le tableau fillable est utilisé pour définir quelles colonnes de la base de données peuvent être remplies en une seule fois lors de la création ou de la mise à jour d'un modèle.



Figure 4: Page "**index.blade.php**"

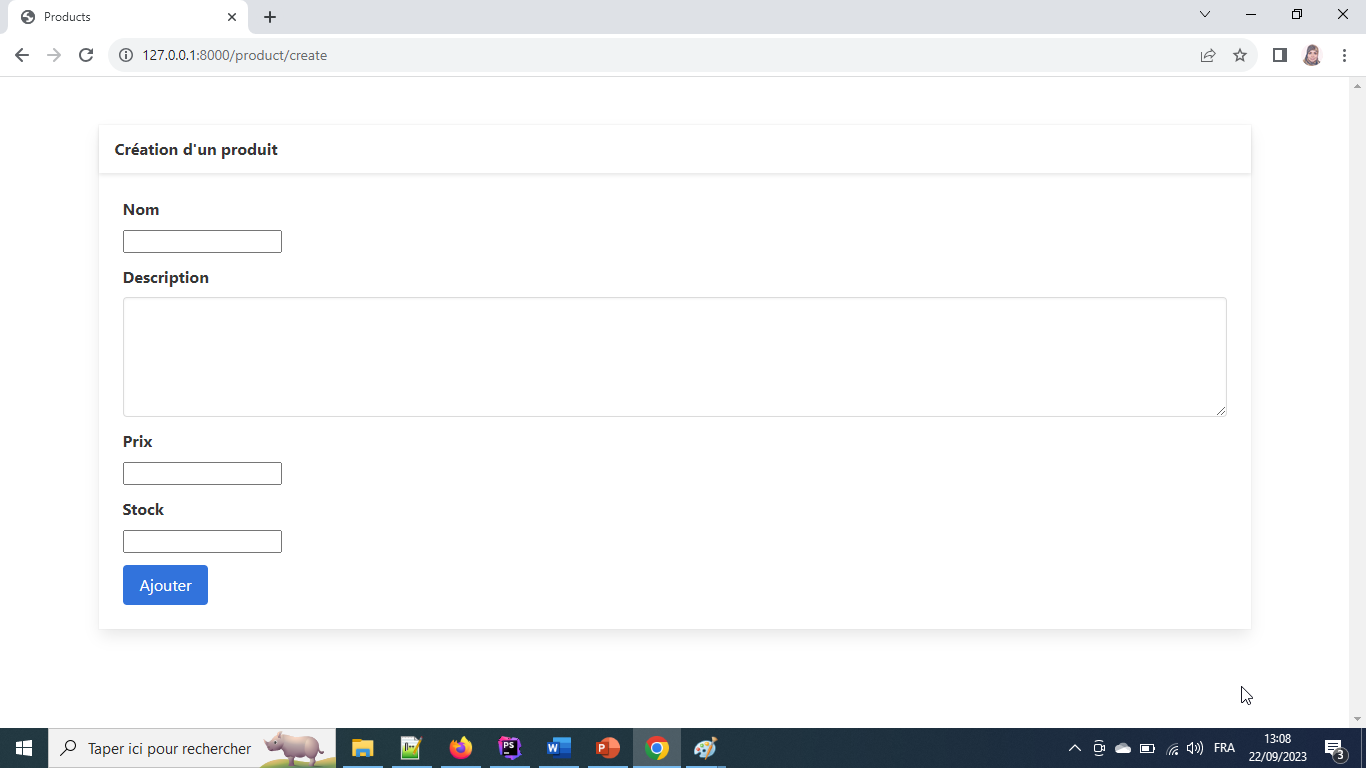


Figure 5: Page "**create.blade.php**"

1. Ajouter le code qui permet de modifier un produit:

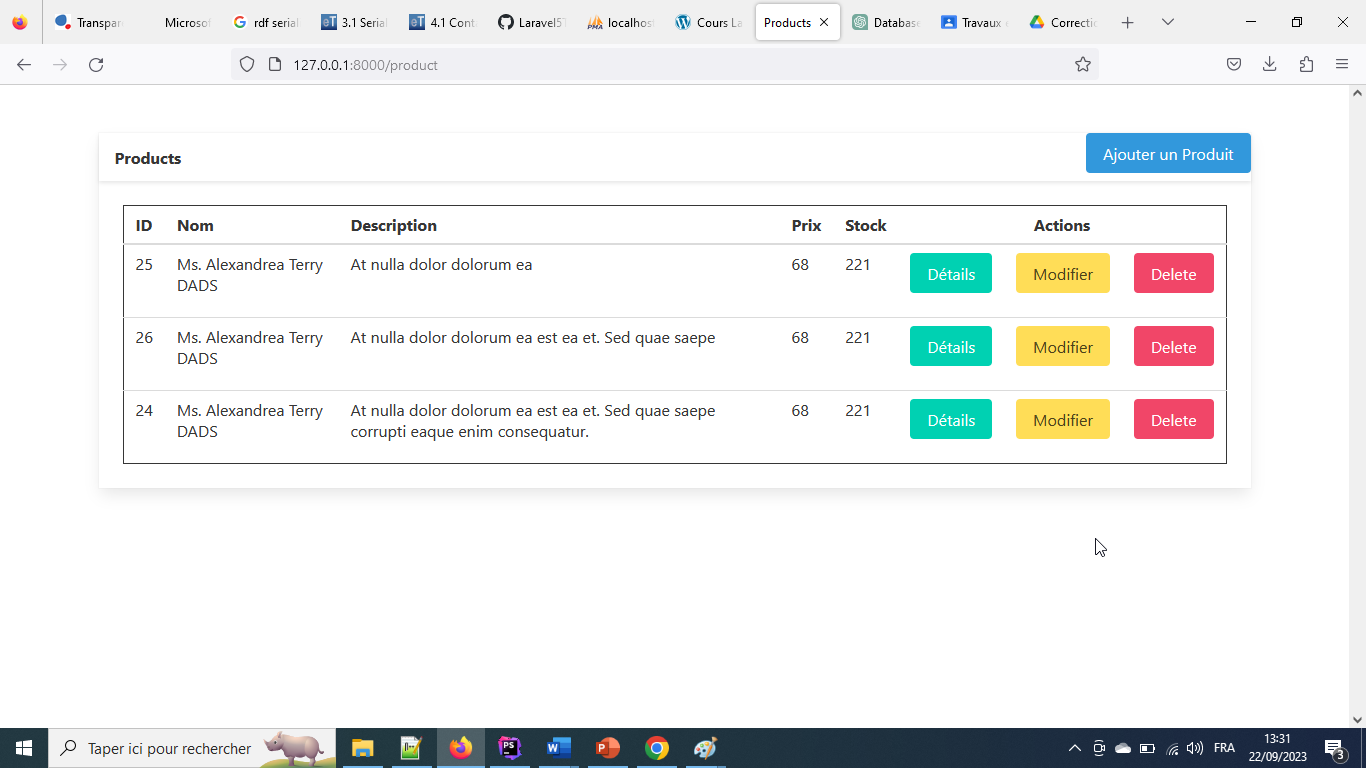


Figure 6: Page "**create.blade.php**"

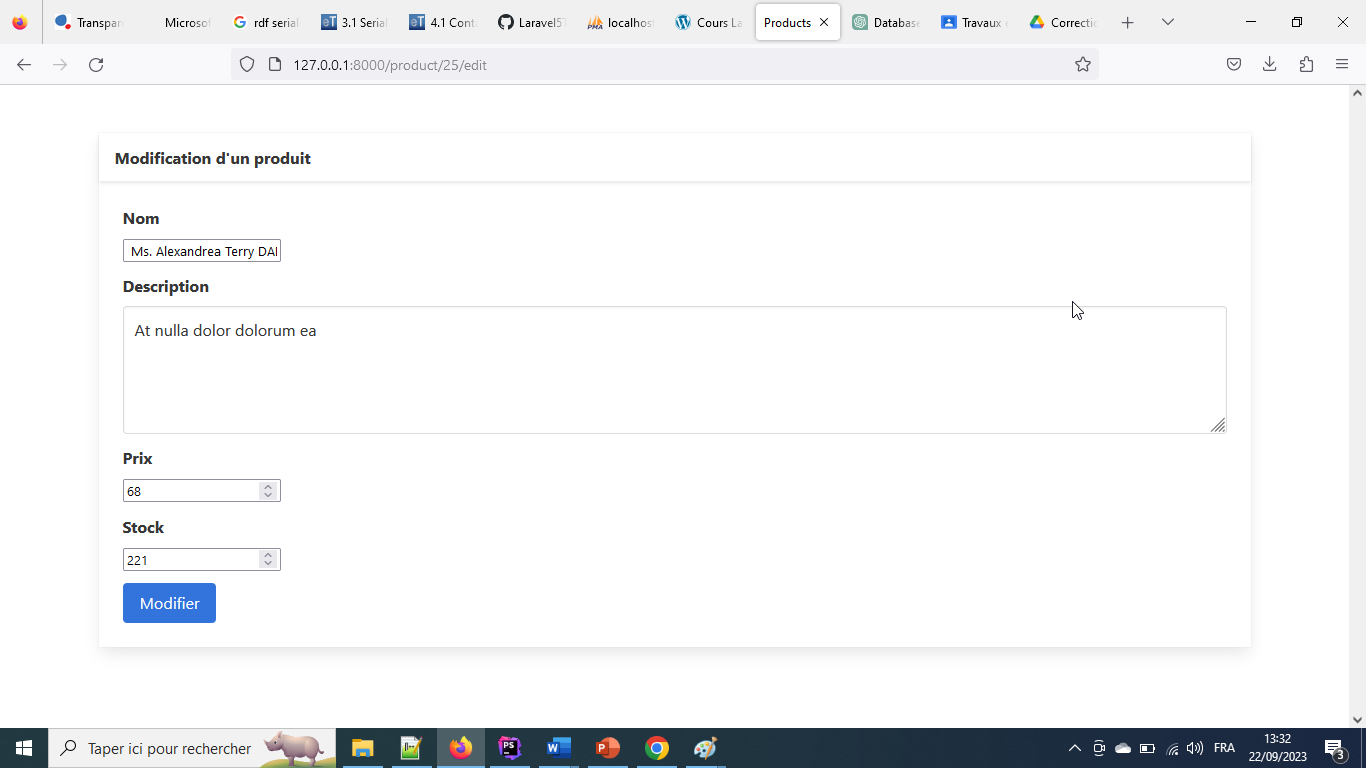


Figure 7: Page "**edit.blade.php**"