

Application des surnoms des enseignants avec Adonis

- Step11

Nous allons mettre en place le login.

pour cela, nous avons besoin :

- d'une nouvelle route
- d'un nouveau contrôleur et d'une méthode dans ce contrôleur
- d'un validateur

Routes

```
// Route permettant de se connecter
router.post('/login', [AuthController, 'login']).as('auth.login')
```

Validateur

Création du validateur via le CLI :

```
$ node ace make:validator auth
DONE: create app/validators/auth.ts
```

Le code du validateur `app/validators/auth.ts`:

```
import vine from '@vinejs/vine'

const loginUserValidator = vine.compile(
  vine.object({
    username: vine.string().minLength(4),
    password: vine.string().minLength(8),
  })
)

export { loginUserValidator }
```

Contrôleur AuthController

Création du contrôleur via le CLI :

```
$ node ace make:controller AuthController
DONE: create app/controllers/auth_controller.ts
```

```
import type { HttpContext } from '@adonisjs/core/http'
import { loginUserValidator } from '#validators/auth'
```

```
import User from '#models/user'

/**
 * Contrôleur pour l'authentification
 */
export default class AuthController {
  /**
   * Gérer la connexion d'un utilisateur
   */
  async login({ request, auth, session, response }: HttpContext) {
    // Récupère les données validées
    const { username, password } = await request.validateUsing(loginUserValidator)

    // Récupère l'utilisateur correspondant aux données saisies par l'utilisateur
    const user = await User.verifyCredentials(username, password)

    // Utilise le guard 'web' pour connecter l'utilisateur -> Voir le fichier
    // config/auth.ts
    await auth.use('web').login(user)

    // Affiche un msg à l'utilisateur
    session.flash('success', `L'utilisateur ${user.username} s'est connecté avec succès`)

    // Redirige vers la route ayant pour nom 'home'
    return response.redirect().toRoute('home')
  }
}
```

Bouton 'Se connecter' et 'Se déconnecter'

Dans le fichier `header.edge` nous allons ajouter un bouton "Se déconnecter" ainsi que l'affichage du username de l'utilisateur connecté.

```
@eval(await auth.check())
<header>
  <div class="container-header">
    <div class="titre-header">
      <h1>
        Surnom des enseignants
      </h1>
    </div>
    <div class="login-container">
      @if(auth.isAuthenticated)
        {{ auth.user.username }} ({{ auth.user.isAdmin ? 'admin' : 'user' }})
        <form action="#" method="post">
          {{ csrfField() }}
          <button type="submit" class="btn btn-logout">Se
déconnecter</button>
        </form>
      @else
```

```

<form action="{{ route('auth.login') }}" method="post">
  {{ csrfField() }}
  <input
    type="text"
    id="username"
    name="username"
    value="{{ old('username') || '' }}"
    placeholder="Nom d'utilisateur"
  />
  @inputError('username')
    <p class="text-danger">
      {{ $messages.join(', ') }}
    </p>
  @end
  <input
    type="password"
    id="password"
    name="password"
    value="{{ old('password') || '' }}"
    placeholder="Mot de passe"
  />
  @inputError('password')
    <p class="text-danger">
      {{ $messages.join(', ') }}
    </p>
  @end
  <button type="submit" class="btn btn-login">Se connecter</button>
</form>
@endif
</div>
</div>
<nav>
  <a href="{{ route('home') }}">Accueil</a>&nbsp;
  <a href="{{ route('teacher.create') }}">Ajouter un enseignant</a>
</nav>
</header>

```

Utilisateur inconnu

Si l'utilisateur saisit des mauvaises informations de connexion, il est important de lui afficher un message d'erreur.

Ajouter ce code au fichier `views/partial/flash.edge`

```

@error('E_INVALID_CREDENTIALS')

  <div class="alert alert-danger">
    {{ $message }}
  </div>
@end

```

Message de validation en français

Le mécanisme est déjà en place mais nous devons simplement traduire les mots "username" et "password".

```
{
  "shared": {
    "fields": {
      "gender": "Genre",
      "firstname": "Prénom",
      "lastname": "Nom",
      "nickname": "Surnom",
      "origine": "Origine",
      "sectionId": "Section",
      "username": "Nom d'utilisateur",
      "password": "Mot de passe"
    },
    "messages": {
      "required": "Le champ {field} est obligatoire.",
      "minLength": "Le champ {field} doit contenir au moins { min } caractères.",
      "email": "Le champ doit {field} être une adresse e-mail valide."
    }
  }
}
```

Etat à la fin de cette étape

Surnom des enseignants







Albert (user)

Se déconnecter

[Accueil](#) [Ajouter un enseignant](#)

L'utilisateur Albert s'est connecté avec succès

Liste des enseignants

Nom	Surnom	Options
Carrel Xavier	XCL	  
Charmier Grégory	GregLeBarbar	  

Prochaine étape

Dans la prochaine étape [step12](#), nous allons mettre en place le logout.