

## Atelier 07

### Objectifs d'apprentissages :

- Utilisation de la bibliothèque Bcrypt

### Travail a réalisé

1. Créer un dossier sous le nom "Atelier07-express-bcrypt".
2. Lancer la commande de création de fichier package.json
3. Installer les packages suivants :

- express
- mongoose
- nodemon
- cors
- dotenv
- bcrypt

- 4- Créer le fichier .env puis déclarer les variables d'environnement suivante :

Nom de variable	Description
PORT	3000
URL_MONGOOSE	Chaine de connexion vers MongoDB/dbusers

- 6- Créer un fichier nommé server.js :

- Récupérer les variables d'environnement
- Permettre l'analyse des requêtes JSON.
- Permettre la réception des requêtes asynchrone.
- Connexion à la base de données MongoDB.
- Les requêtes qui commencent par /api/user seront redirigées vers le fichier user.js dans le dossier routes.
- Démarrer le serveur dans le port 3000 (valeur déjà défini dans le fichier .env).

- 7- Dans un dossier nommé models, créer un fichier nommé UserModel :

- Définir le schéma nommé UserSchema avec les caractéristiques

Nom du champ	Description
login	Type chaîne de caractère Champ Obligatoire Champ Unique
firstname	Type chaîne de caractère
lastname	Type chaîne de caractère
email	Type chaîne de caractère

	Champ Obligatoire Champ Unique
password	Type chaîne de caractère

- Exporter par défaut le modèle associé à ce schéma.

8- Dans le fichier user.js, créer les routes suivantes :

Chemin	Méthode	Description
<b>/add</b>	POST	- Ajout d'un utilisateur dans la base de données. - Les informations de l'utilisateur sont fournies dans le corps de la requête. - Le mot de passe doit être haché avant le stockage
<b>/checklogin</b>	GET	- retourne un message si le login et mot de passe fournis sont correctes. - Les informations d'authentification (login, password) sont fournies dans le corps de la requête.
<b>/checkemail</b>	GET	- retourne le message "SUCCESS" si le mail et mot de passe fournis sont corrects sinon le message "ECHEC" est retourné. - Les informations d'authentification(mail, password) sont fournies dans le corps de la requête.

9. Tester votre API