

## DEV-WEB

### Mini projet

#### Objectif:

Réalisation d'un forum de discussion (questions/réponses).

Un utilisateur poste une question, les autres répondent à cette question.

Cette application se compose essentiellement de trois pages

Page	Description
<b><i>home.pug</i></b>	Cette page affiche la liste des questions au public, en plus, si l'utilisateur est authentifié il peut poster une nouvelle question
<b><i>question.pug</i></b>	Cette page affiche une question et la listes des réponses à cette question au public, en plus, si l'utilisateur est authentifié il peut poster une nouvelle réponse à cette question
<b><i>login.pug</i></b>	Cette page permet à un utilisateur de s'authentifier
<b><i>register.pug</i></b>	Cette page permet à un visiteur de s'enregistrer

#### Partie 1:

1.1- Reproduire la structure ci contre et initialiser un dépôt git dans le dossier forum avec un premier commit "initialisation du projet".

1.2- Créez les modèles suivants:

Users(id, nom, email, password)

Questions(id, user\_id, question, date)

Reponses(id, user\_id, question\_id, response, date)

1.3- Dans ***app.js*** écrire script pour établir une connexion à la base de données en utilisant mongoose

1.4- Dans userController créez une fonction ***addUser(req,res)*** pour insérer un nouvel utilisateur. Utiliser le module bcrypt pour crypter le mot de passe avant son insertion dans la collection.

1.5- Dans ***register.pug*** insérer un formulaire d'enregistrement d'utilisateur qui poste les données au route ***/users***, puis éditez les fichiers ***app.js*** et ***usersRouter.js*** pour gérer cette route.

1.5- Faire un commit avec le message "Partie 1 : Ajout d'un nouvel utilisateur".

```

forum
├── app.js
├── controllers
│   ├── questionController.js
│   ├── reponseController.js
│   └── userController.js
├── middleware
│   └── auth.js
├── models
│   ├── Question.js
│   ├── Reponse.js
│   └── User.js
├── routes
│   ├── questionsRouter.js
│   ├── reponseRouter.js
│   └── usersRouter.js
└── views
    ├── home.pug
    ├── login.pug
    ├── question.pug
    └── register.pug
  
```

### Enregistrement

Nom:

Email:

Mot de passe:

Confirmer Mot de passe

## Partie 2

Email:

Mot de passe:

## Partie 3

[Logout/login](#)

20-11-2023 reda@est.ovh

[Réponses](#)

18-11-2023 yahya@est.ovh

[Réponses](#)

3.4- faire un commit "Partie 3: Questions/réponses"

[Home](#) [Logout/login](#)

Une question ...

Votre réponse ici ....

Envoyer

20-11-2023 reda@est.ovh

une reponse .....

18-11-2023 yahya@est.ovh

une autre reponse .....