Rapport AWS

Niels Merceron
Pierre Vermeulen
Alexis Guigal
Manel Azgag

Table des matières

Chapitre 1

Introduction

Nous sommes un groupe avec peu de notions de web, donc pour nous familiariser avec les langages et autre technologie nous avons fait le choix de faire un site simple et plus particulièrement un site de to do list. Il y a beaucoup de ressource en ligne nous décrivant comment faire une to do list et donc nous aiderons tout au cours du développement de notre site pour savoir quel service implémenter de notre site ou non, si c'est à notre portée. Pour autant, faire ce site relèvera de quelques challenges de conception vu que nous ne sommes pas familiers avec beaucoup de technologie pour développer des sites web. On peut noter par exemple la gestion du back peut être vite contraignante, ou l'aspect sécurité qui doit être fait de manière la plus propre possible pour ne laisser aucune faille.

Chapitre 2

Technologie utilisée

2.1 Front end

2.1.1 Svelte

C'est un framwork simple et proche syntaxiquement du javascript standard. Il est conçu pour compiler le code au moment de la compilation plutôt qu'au moment de l'éxécution, il offre donc de très bonnes performances. Le framework est également bien documenté avec une documentation claire et précise.

2.1.2 Tailwind avec DaisyUI

C'est un framwork qui est bien plus simple que les deux autres. Il est également très facile à intégrer avec n'importe quel type de stack.

2.2 Back end

2.2.1 Nodejs/expressjs

C'est un environnement d'éxécution pour JavaScript construit sur le moteur Javascript de Chrome. Il est extrêmement populaire et il est utilisé par de très nombreuses entreprises. Il est également très simple à prendre en main.

2.2.2 Mongodb

MongoDB est un système de gestion de bases de données NoSQL, qui utilise un format de stockage de données basé sur JSON. Contrairement aux

bases de données relationnelles, MongoDB n'utilise pas de tables et de schémas fixes, mais stocke les données dans des collections flexibles qui peuvent être modifiées sans avoir à définir un schéma préalablement. Cette approche permet une grande flexibilité pour gérer des données complexes et des schémas évolutifs, ce qui est particulièrement utile pour les applications modernes basées sur le cloud. De plus, MongoDB offre des performances élevées, une évolutivité horizontale facile et une grande disponibilité, ce qui en fait une solution de choix pour les entreprises qui cherchent à développer des applications à grande échelle. En raison de ses avantages, MongoDB est très utilisé et dispose d'une documentation de qualité, ce qui facilite son adoption et son utilisation.

2.2.3 JWT

2.3 Sécurité

utilisation bibliothèque crypto-js coté client
côté serveur bcrypt +crypto + TLS + helmet

Chapitre 3

Déroulement du projet

3.1 Back

3.1.1 Création des routes

- Signup - POST

Exemple de requête :

```
{
   "username": "bonjour",
   "email": "test@fafa.com",
   "password": "drfffffffff"
}
```

Exemple de réponse :

```
{
   "token": "eyJhbGciOiJIUzI1Ni5cCI6...2v0RWSOkv4"
}
```

- Login - POST

Exemple de requête :

```
{
   "email": "test@fafa.com",
   "password": "drfffffffff"
}
```

Exemple de réponse :

```
{
   "token": "eyJhbGciOiJIUzI1R5cCI6...2v0RWSOkv4"
}
```

- Createtodo - POST

Exemple de requête :

```
{
   "title": "tesvqhtret1",
   "description": "gfzghfgrfeafezrgezg"
}
```

Exemple de réponse :

```
   {"title":"tesvqhtret1",
   description":"gfzghfgrfeafezrgezg",
   "completed":false,
   "createdAt":"2023-04-28T23:57:41.194Z",
   _id":"644c5ef0c7ff2ca10c44521c","__v":0}
```

- Gettodo - GET

Exemple de requête :
Exemple de réponse :

```
[
   {
   "_id": "644c5647cd148f99d95d98a7",
   "title": "test1",
   description": "gfzghezrgezg",
   "completed": false,
   "createdAt": "2023-04-28T23:26:51.777Z",
   "__v": 0
   },
   {
   "_id": "644c5ef0c7ff2ca10c44521c",
   "title": "YES",
   "description": "mod",
   "completed": false,
   "createdAt": "2023-04-28T23:57:41.194Z",
   "__v": 0
```

```
  }
]
```

3.1.2 - Del - DEL

Exemple de requête :
Exemple de réponse :

```
{
  "message": "Todo Deleted"
}
```

3.1.3 - Modify - PUT

Exemple de requête :
Exemple de réponse :

```
{
  "_id": "644c5ef0c7ff2ca10c44521c",
  "title": "YES",
  "description": "mod",
  "completed": false,
  "createdAt": "2023-04-28T23:57:41.194Z",
  "__v": 0
}
```

Chapitre 4

conclusion/ce qu'on a appris

Chapitre 5

Conclusion