

I. Plateforme Freelances

1. Solution proposee

- **Analyse du Problème**

Une startup souhaite créer une plateforme centralisée pour la gestion des profils de freelances. **Actuellement, les données sont partagées de manière non structurée (WhatsApp, e-mail), ce qui rend la recherche, le suivi, et la maintenance difficiles.**

- **La solution envisagés :**

Mettre en place un Web Service GraphQL qui permet de :

- ✓ Créer, modifier et supprimer un freelance
- ✓ Consulter la liste des freelances

- ✓ Consulter un freelance par identifiant

- **Entités et Relations**

Entité : Administrateur

- ✓ L'administrateur est responsable de la gestion des freelances sur la plateforme
- ✓ Un administrateur peut gérer plusieurs freelances

Entité : Freelance

- ✓ Chaque freelance est géré par un seul administrateur

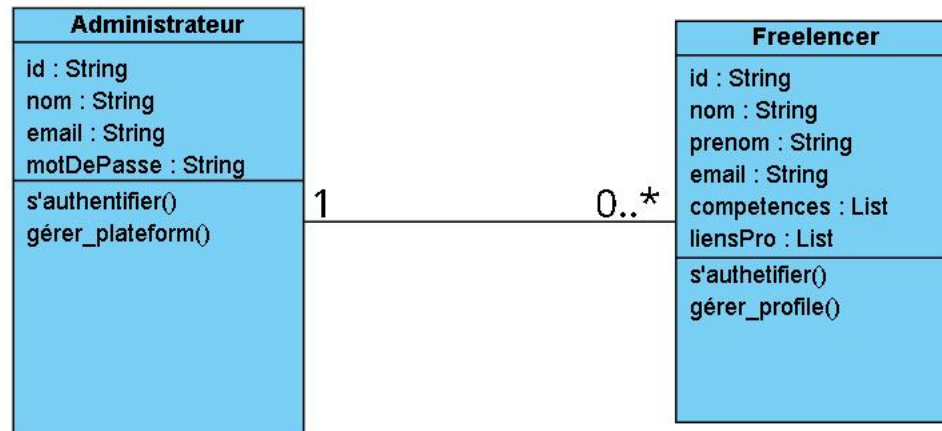
- **Fonctionnalités sous forme de Web Services**

- ✓ freelances: lister tous les freelances (query)
- ✓ freelance(id): retourner un freelance par ID (query)

- ✓ addFreelance(...): ajouter un freelance (mutation)
- ✓ updateFreelance(...): modifier un freelance (mutation)
- ✓ deleteFreelance(id): supprimer un freelance (mutation)

2. Diagramms

- Diagramme de classe:



• Schema des requetes && reponses du service

```
1 mutation {  
2   addFreelance(  
3     nom: "riahi",  
4     prenom: "arbya",  
5     email: "arbiariahi84@gmail.com",  
6     competences: ["java", "spring boot"],  
7     liensPro: ["https://www.linkedin.com/feed/"]  
8   ) {  
9     id  
10    nom  
11    prenom  
12    competences  
13    liensPro  
14  }  
15 }  
16
```

```
{  
  "data": {  
    "addFreelance": {  
      "id": "683210f8a3c1100c8fc3a2a1",  
      "nom": "riahi",  
      "prenom": "arbya",  
      "competences": [  
        "java",  
        "spring boot"  
      ],  
      "liensPro": [  
        "https://www.linkedin.com/feed/"  
      ]  
    }  
  }  
}
```

voir tous les freelances:

The screenshot shows the GraphQL IDE interface. The top bar includes the 'GraphiQL' logo, a play button, and buttons for 'Prettify', 'Merge', 'Copy', and 'History'. On the right, there is a '< Docs' link. The left pane contains a query written in lines 1 through 10:

```
1 query {  
2   freelances {  
3     id  
4     nom  
5     email  
6     competences  
7   }  
8 }  
9  
10
```

The right pane displays the JSON response, which is a list of three freelance objects. The response is as follows:

```
{  
  "data": {  
    "freelances": [  
      {  
        "id": "683210dea3c1100c8fc3a29f",  
        "nom": "riahi",  
        "email": "arbiariahi840@gmail.com",  
        "competences": [  
          "java",  
          "spring boot"  
        ]  
      },  
      {  
        "id": "683210f8a3c1100c8fc3a2a1",  
        "nom": "riahi",  
        "email": "arbiariahi840@gmail.com",  
        "competences": [  
          "java",  
          "spring boot"  
        ]  
      },  
      {  
        "id": "683212bfa3c1100c8fc3a2a3",  
        "nom": "riahi",  
        "email": "riahimaryem@gmail.com",  
        "competences": [  
          "finance",  
          "marketing"  
        ]  
      }  
    ]  
  }  
}
```

At the bottom of the interface, there is a section labeled 'QUERY VARIABLES'.

Modifier un freelanceur par son id:

GraphiQL

Prettify

Merge

Copy

History

< Docs

```
1 mutation {
2   updateFreelance(
3     id: "683212bfa3c1100c8fc3a2a3",
4     nom: "ala",
5     email: "ala@gmail.com"
6     competences: ["Node.js", "Express", "GraphQL"]
7   ) {
8     id
9     nom
10    competences
11  }
12 }
13
```

```
{
  "data": {
    "updateFreelance": {
      "id": "683212bfa3c1100c8fc3a2a3",
      "nom": "ala",
      "competences": [
        "Node.js",
        "Express",
        "GraphQL"
      ]
    }
  }
}
```

Supprimer un freelencer par son id :

```
1 mutation {  
2   deleteFreelance(id: "683212bfa3c1100c8fc3a2a3") {  
3     id  
4     nom  
5   }  
6 }  
7 |  
8
```

```
{  
  "data": {  
    "deleteFreelance": {  
      "id": "683212bfa3c1100c8fc3a2a3",  
      "nom": "ala"  
    }  
  }  
}
```

Supprimer tous les freelances

```
1 mutation {  
2   deleteAllFreelances  
3 }  
4 |  
5
```

```
{  
  "data": {  
    "deleteAllFreelances": "Tous les freelances ont été supprimés."  
  }  
}
```

3. Implementation && documentation : voir ce lien github