

Université Paris-Saclay

Compte-Rendu Projet Web

ATIA Safiya – BOSSUT Noémie – HERMAN Simon

I / Documentation orientée utilisateur

~voir dépôt GitHub~

II/ Documentation technique

1) Généralités

Le serveur a été développé en utilisant plusieurs langages de programmation :

- Python
- JavaScript

Ainsi que les langages de script web :

- HTML
- CSS

Le framework Django a permis une gestion de la base de données et de l'interface utilisateur-serveur. Dans ce cadre, plusieurs modules sont nécessaires à l'utilisation de l'application, en plus de Django :

- django-extensions
- biopython

L'utilisateur a la possibilité, s'il le souhaite, de passer par un environnement conda pour éviter l'installation des dépendances et pour s'assurer d'utiliser des versions compatibles entre elles.

2) Fonctions pour l'utilisateur

Les principales fonctions d'interaction avec le serveur via la ligne de commande sont groupées dans le répertoire “ scripts “ :

- load_data
- reset_db
- create_groups

Les fonctions load_data et reset_db sont dédiées à la gestion automatique du chargement des tables et des données pour les remplir. create_groups se charge d'initialiser les rôles de chaque visiteur du site web.

L'utilisateur est cependant invité à n'initialiser le serveur que via une seule ligne de commande :

```
python3 manage.py runscript init_server
```

La commande **runscript** tire parti du module Django-extensions pour faciliter la tâche à l'utilisateur sans connaissance de Django.

III/ Retour d'expérience

1) Contexte

Cette application web a été mise au point lors de l'UE "Projet Web". L'objectif étant de créer, en 6 semaines, un site fonctionnel d'annotation de séquences bactériennes. Si l'hébergement n'est pas à implémenter, la création de la base de données et sa gestion ainsi que la création de l'architecture et de l'interface utilisateur sont à construire à partir de zéro.

2) Organisation

Pour mener à bien ce projet, notre groupe s'est divisé en deux parties : Safiya & Simon sur la partie backend : gestion de la base de données, gestion des utilisateurs, interactions avec les utilisateurs (views, forms ...) et Noémie sur la partie frontend : architecture et interface utilisateur.

La partie back-end a été construite autour d'une unique application Django : genomeBact. Il a été tenté d'encapsuler au maximum les fonctions réutilisées fréquemment dans une source dédiée nommée selon les conventions de la communauté : utils.py. Les views, cœur de l'interaction entre la base de donnée et les utilisateur, ont été séparées en 4 grands groupes de fonctionnalités pour une meilleure maintenance au cours du développement. De la même manière les décorateurs de fonctions utilisés pour gérer les permissions utilisateurs ont été regroupés entre eux.

L'outil de visualisation a été développé en JavaScript, tirant parti d'une portion de code du projet Genoscapist pour l'adapter aux besoins du projet. Toutes les sources et permissions sont présentes au sein du Readme.

Pour la gestion des annotations entre annotateurs et validateurs, nous avons pris le parti de ne pas créer une page html pour chaque fonctionnalité mais plutôt de rendre l'interaction sur la page du gène dynamique en fonction de l'utilisateur connecté et des clics effectués. Ainsi de nombreux boutons sont présents sur la page html mais visibles seulement selon l'utilisateur (annotateur associé à ce gène, ...) et l'état du gène (en cours d'annotation, validé, etc).

Concernant la partie "frontend" : le parti pris a été de faire simple (deux couleurs d'accents, pages plutôt épurées) et d'apporter une note décalée (le projet OUAF) . ce qui nous a permis d'un côté, de nous rappeler qu'il s'agit d'un projet étudiant d'apprentissage, mais surtout d'apporter du plaisir pendant la phase difficile de la prise en main des outils. Il a également été important de ne pas réinventer la roue : les templates des pages ont été pris de codes disponibles en libre accès puis modifiés afin de convenir à notre utilisation, Bootstrap et son système de cartes nous a également été utile.

3) Résultats

Toutes les fonctionnalités obligatoires ont été remplies. Par manque de temps et de moyens, aucune des options n'a pu être mise en place. Bien que nous soyons satisfaits de notre travail, nous sommes conscients de ses limites.

Le temps de chargement des données et de téléchargement des FASTA est satisfaisant. Le format de téléchargement des fichiers FASTA respecte le format d'entrée .

Nous n'avons pas réussi à résoudre certains problèmes : bien que les fonctions de recherche de gènes soient fonctionnelles, l'application plante lorsqu'un trop grand nombre d'instances sont récupérées. Nous avons également fait des compromis sur l'interface utilisateur (pas d'affichage dynamique sans passer par un submit). De manière générale, et dû au manque de temps et de main d'œuvre, les fichiers de codes html ne sont pas propres : certains fichiers contiennent par exemple des lignes de javascript ou de css, alors que nous avons décidé de compartimenter les fichiers.

Nous avons tenté de prendre en compte au maximum les autorisations et accès des pages et fonctionnalités selon les utilisateurs avec des conditions dans le html mais également au sein même des fonctions en python. Cependant, par manque d'expérience, il n'est pas exclu qu'un utilisateur expérimenté trouve des failles dans notre site afin d'outrepasser ses permissions.

Certaines fonctionnalités comme la suppression d'objets dans la base de données nécessiteraient une confirmation avant suppression; par manque de temps cela n'a pas été implémenté.

4) Retours personnels

Bien que ce projet a été considéré comme difficile et que nous avons fortement senti l'absence d'un quatrième membre dans le groupe, nous avons su utiliser les forces de chacun afin d'être productifs, ce qui nous a permis de développer nos compétences en gestion de projets et travail d'équipe, et nous a forcé à communiquer efficacement afin de garder le bon cap.

Ce projet plus que les autres, nous a permis voir l'importance de la planification rigoureuse et de la gestion du temps pour garantir un livrable dans les délais impartis. Enfin, nous sommes reconnaissants d'avoir pu développer nos compétences techniques en développement web et d'avoir mieux compris les défis et les opportunités qui y sont liées.