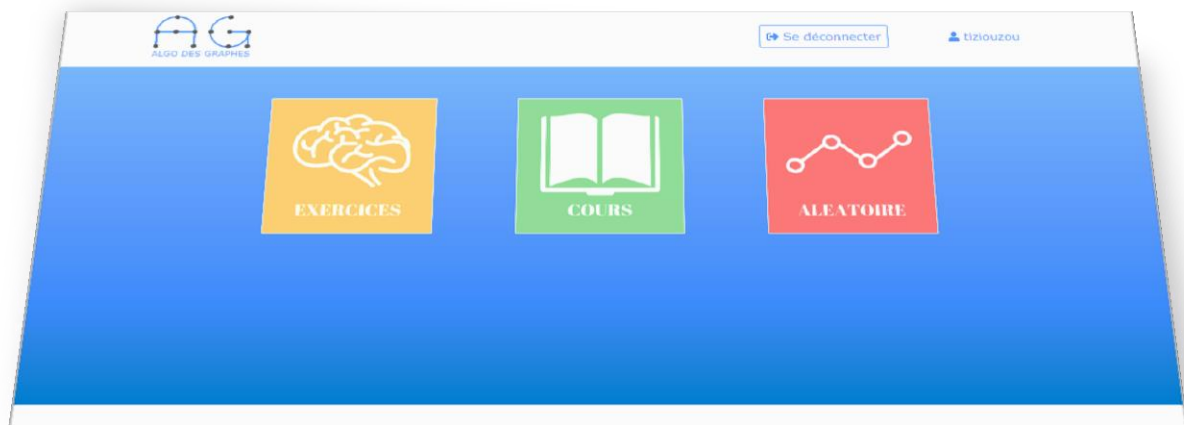


Projet : Réalisation d'un site web pour les cours d'algorithmique des graphes



Réalisé par :

- Thierno Souleymane DIALLO
- Saïd HADIM
- Saïd BRAHMI
- Antoine RENOU
- Kevin LINDAMBI YANGU
- Soufiane DERMOUMI

Encadrants :

- Rumen Andonov , Equipe GenScale, INRIA/IRISA
- Sébastien François, Equipe GenScale, INRIA/IRISA

Formation : L3 informatique

Année universitaire : 2017/2018

1. BUT DU PROJET

Notre projet a pour but de concevoir un site web dédié au cours d'Algorithme des graphes dispensé en Licence 3 Informatique. Ce site a pour vocation de permettre aux étudiants de pouvoir appliquer et de s'exercer sur les différentes notions abordées en cours notamment la recherche de plus courts chemins avec l'algorithme de Dijkstra et de Bellman Ford ainsi que les notions sur les cliques, les ACM et les composantes fortement connexes afin de pouvoir mieux les préparer à leurs examens.

Pour éviter de devoir écrire des variantes d'exercices, nous avons utilisé un système aléatoire qui génère à la volée, exercices et solutions. Il s'agit bien entendu d'exercices reposant sur des questions à choix multiples principalement.

Les objectifs à atteindre sont donc les suivants :

- proposer un design pour le dit site et réaliser son intégration
- ajouter un mécanisme d'inscription/identification pour les utilisateurs du site
- créer la routine de génération d'exercices
- faire en sorte de conserver des statistiques qualitative en fonction d'exercice résolu/échoué

2. OUTILS UTILISES

Pour développer ce site, nous avons utilisé plusieurs outils tels que :

1. Django : un cadre de développement web open source qui rend le développement web simple et rapide ouvert à python et qui offre un design pragmatique. La version utilisée est Django 2
2. Python : ça nous a servi à gérer tout le côté serveur, parfaitement intégré à Django, il a fallu acquérir au préalable un minimum de connaissances sur cet outil pour pouvoir profiter un maximum de Django. On a utilisé la version 3.6
3. PostgreSQL : nous avons fait notre choix sur ce système de gestion de base de données car la prise en main est simple et est très adapté pour la gestion de données des projets relativement simples. La version utilisée est PostgreSQL 10.3
4. HTML : ce langage des balises est proposé par Django pour la gestion de nos pages web. La version utilisée est HTML5
5. CSS : on a utilisé cette feuille de style pour pouvoir améliorer le design de nos pages, ajouter des fonts etc. la version utilisée est la dernière à jour proposée par Django.
6. Javascript : on a dû utiliser ce langage pour dynamiser nos pages notamment pour les graphes générés.
7. Networkx : ce paquet fournit des classes pour le graphique des objets, des générateurs pour créer des algorithmes notamment pour les graphes.
8. Vis.js : pour l'affichage des graphes qui ont été créés avec Networkx

9. JQuery : Pour faciliter les différentes animations sur les sites en écrivant des scripts simples.
10. Canvas : ça nous a permis de créer nos propres images de background pour éviter d'utiliser les images soumises à des droits d'auteur.
11. Fontawesome : pour créer les petites icônes pour illustrer quelques fonctionnalités comme la connexion, la déconnexion etc.

3. FONCTIONNALITÉS DÉVELOPPÉES

Dans ce projet, nous avons développé les fonctionnalités suivantes :

- Génération aléatoire des graphes
- Mécanisme d'inscription/identification des utilisateurs du site

4. FONCTIONNALITÉS MANQUANTES

Faire en sorte de conserver les statistiques qualitatives en fonction du type d'exercices résolu/échoué.

5. AMÉLIORATIONS POSSIBLES

- Amélioration côté CSS pour les pages.
- Permettre à l'utilisateur de choisir par exemple les poids sur les arcs.
- Utilisation des variables globales ou paramètres de fonctions pour la création des graphes (non utilisé ici, les valeurs sont écrites en dur) afin de créer des exercices d'un niveau de difficultés constant.

Exemple 1 : pour l'instant les questions sur Dijkstra et Bellman-Ford demanderont toujours le plus court chemin entre le nœud A et le nœud E.

Exemple 2 : la question sur les composantes fortement connexes comportera toujours 8 nœuds pour ne pas avoir à gérer la génération de plus de 8 couleurs différentes.

- Faire une deuxième requête AJAX au serveur pour obtenir la bonne réponse, si on observe la réponse actuelle du serveur avant de répondre à l'exercice on peut y lire la bonne réponse.
- Création de niveau de difficultés pour les exercices (plus de nœuds, d'arcs, changement de parcours, etc ...)

- S'assurer que les réponses proposées dans le QCM ne soient pas identiques (peut arriver très rarement pour les questions sur les composantes fortement connexes et l'arbre couvrant minimal.
- Gestion d'un système de score avec le script.
- Coloration en rouge du cycle négatif lorsqu'il est impossible d'utiliser l'algorithme de Bellman-Ford.
- Possibilité qu'un cycle négatif ne soit pas relié au graphe principal mais génère quand même la réponse "Utilisation de l'algorithme impossible" pour l'exercice sur Bellman-Ford.

6. DÉROULEMENT DU PROJET

Nous avons utilisé la méthodologie SCRUM, on a procédé par des sprints, à chaque fin de sprints, on testait les fonctionnalités développées puis on décidait de ce qui devrait être fait dans le prochain sprint.

L'équipe a été subdivisée en 2 parties. Un ScrumMaster était désigné à chaque sprint pour s'assurer du bon fonctionnement des sprints et des éventuelles interventions dans les deux groupes.

Pour le sprint 0 : on a vu ensemble les bases du produit qu'on devait développer, défini l'architecture puis on s'est mis d'accord sur la façon de travailler, les outils à utiliser etc.

Au sprint 1 : Étant donné qu'on avait pas utilisé ces technologies avant, on s'est fixé comme objectif de se documenter sur les différents outils à utiliser notamment par des tutos pour la prise en main de ces outils.

Au sprint 2 : le projet avait démarré, un groupe s'occupait de la partie génération aléatoire des graphes et un autre s'occupait de gestion des pages web tout en intégrant les codes dans Django en faisant appel à différents outils.

Aux Sprints 3, 4 : Améliorations du code, corrections des bugs.

Au Sprint 5 : Mise en commun de l'ensemble des codes développés et test global. On a vu aussi des éventuelles améliorations.

CONCLUSION

Lors de la réalisation de ce travail, nous avons eu le plaisir d'apprendre plusieurs technologies qu'on a pas forcément abordées en cours notamment le domaine du web. Dans ce sens, nous avons pu aboutir et atteindre notre objectif qui consistait à concevoir le site web pour le cours d'algorithme des graphes . Cette étude nous a aidé à approfondir et élaborer nos connaissances en méthode d'analyse et de développement Scrum ainsi qu'aux différents outils cités ci-haut. En utilisant différentes méthodes, à travers le processus effectué, nous avons acquis et développé des qualités, aptitudes et intérêts. Ce qui va refléter positivement sur notre cursus d'étude.

Quelques difficultés ont été rencontrées aussi durant la réalisation de ce projet notamment la familiarisation avec les différents outils et aussi le temps car ça n'a pas toujours été facile de concilier petits boulots et projets ainsi que de coordonner notre temps pour pouvoir travailler ensemble.

Enfin, nous tenons à remercier encore une fois nos chères formateurs et professeurs pour tout effort fournit de leur part durant ces années de formation, ainsi que chaque personne ayant contribué, de prêt ou de loin, à la réalisation de ce travail.