



20 DECEMBRE 2022

Document présenté à

Jean-Christophe Demers

Dans le cadre du cours

420-C61-IN Projet Synthèse


RAPPORT FINAL

ESCAPEROOMMANAGER

CARLENS BELONY, MAXENCE GUINDON

CÉGEP DU VIEUX MONTRÉAL

255 rue Ontario H2X 1X6



Présentation du livrable

La version initiale du projet se voulait une application de gestion d'horaire pour des entreprises proposant des services. Notre client type était une entreprise de jeux d'évasion. Nous avons un volet pour la gestion quotidienne des horaires, et un volet pour la gestion des employés utilisant la plateforme en séparant l'accès en différents niveaux.

Cependant, nous avons dû réduire nos attentes en cours de projet et nous nous sommes concentrés à produire une application permettant d'assurer la gestion des horaires puisque c'était le cœur de notre projet. Nous croyons avoir réussi cet objectif.

Résumé du développement

Respect ou modification des objectifs

Nous avons réussi à atteindre nos objectifs pour ce qui est du cœur du projet puisque nous avons livré une application permettant de gérer des réservations. Cependant, nous avons dû laisser de côté beaucoup de choses pour y arriver. Notamment, le volet sur la gestion des utilisateurs du système.

Au niveau des objectifs pour le code, nous avons rempli la majorité de ceux-ci. Cependant, nous voulions implanter plusieurs algorithmes différents avec le patron de conception de stratégie pour utiliser différentes techniques de conception d'horaire. Toutefois, nous avons dû nous limiter à l'algorithme du *backtracking*. Également, nous avons pensé utiliser un *binary tree*, mais nous avons laissé tomber cette idée.

Difficultés rencontrées

La communication efficace entre la vue et le serveur fut difficile bien que nous ayons fait de notre mieux pour créer un bon design. La grosse difficulté venant de notre façon de traiter les horaires et les salles en SQL. Si c'était à refaire, nous utiliserions MongoDB pour la conservation des données de nos salles en particulier.

Une autre difficulté fut l'apprentissage de React et de MobX qui était au départ complexe à faire travailler ensemble. De plus, les sources en lignes furent assez discordantes les unes aux autres par rapport à leur utilisation, rendant l'apprentissage de ces deux modules plus complexes.

Finalement, nous pensions utiliser les variables de session de Flask, mais celle-ci se révéla inutilisable dans notre projet, puisque dès qu'on effectuait une connexion, elle s'effaçait.

Sources importantes pour le projet

- CNIL. (2017, Janvier 27). *CNIL*. Récupéré sur Les conseils de la CNIL pour un bon mot de passe : <https://www.cnil.fr/fr/les-conseils-de-la-cnil-pour-un-bon-mot-de-passe#:~:text=Un%20bon%20mot%20de%20passe%20doit%20contenir%20au%20moins%2012,est%20%C3%A9quip%C3%A9%20de%20s%C3%A9curit%C3%A9s%20compl%C3%A9mentaires%20!>
- Shvets, A. (2018, Décembre 5). *Strategy*. Récupéré sur refactoring guru : <https://refactoring.guru/design-patterns/strategy>
- Wang, X. (2022, Janvier 10). *What is HTTP middleware? Best practices for building, designing and using middleware*. Récupéré sur moesif blog : <https://www.moesif.com/blog/engineering/middleware/What-Is-HTTP-Middleware/#:~:text=Middleware%20is%20a%20design%20pattern,on%20the%20core%20business%20logic>
- <https://medium.com/algorithms-and-leetcode/backtracking-e001561b9f28>
- https://www.tutorialspoint.com/python_data_structure/python_backtracking.htm#:~:text=Practical%20Data%20Science%20using%20Python&text=Backtracking%20is%20a%20form%20of,not%20give%20the%20required%20solution
- <https://medium.com/algorithms-and-leetcode/backtracking-with-leetcode-problems-part-2-705c9cc70e52>
- <https://medium.com/algorithms-and-leetcode/in-depth-backtracking-with-leetcode-problems-part-3-b225f19e0d51>

Librairies utilisées

Nous avons utilisé les librairies Bcrypt pour l'encodage des mots de passe, my-SQL-connector pour communiquer avec la base de données SQL, React pour l'affichage du projet, MobX pour le middleware.

Stratégie explorée

Nous avons exploré la stratégie du *middleware* pour communiquer de l'application react à la base de données. Le but étant de valider les informations avec celle-ci avant de tenter des connexions de l'application à la base de données.

Événements importants et commentaire sur la session

Un événement important pour nous a été lorsque Fred nous a dit que notre projet pouvait être décliné en deux applications distinctes (une *backend* et l'autre *frontend*). Nous avons donc travaillé en ce sens depuis.

Pour les autres événements marquants, il y a eu la réussite de notre première connexion, lorsque nous avons pu enregistrer des réservations, les modifier, etc. et lorsque nous avons réussi à faire un affichage primitif déterminant les salles libres et celles qui étaient réservées.

Fonctionnalités

- Connexion — parfaitement fonctionnelles
- Enregistrement d'une nouvelle compagnie — parfaitement fonctionnelles
- Enregistrement d'un employé — semi-fonctionnelles
- Création, modification, suppression de réservations — parfaitement fonctionnelles
- Création, modification, suppression de centre — parfaitement fonctionnelles
- Création modification, suppression de salle — parfaitement fonctionnelles
- L'affichage d'un rapport de performance — parfaitement fonctionnelles
- Utilisation du Middleware — parfaitement fonctionnelles
- Vérification du mot de passe — parfaitement fonctionnelles
- Générateur d'horaire — semi-fonctionnelles
- Afficher le site selon le niveau d'accès de l'utilisateur — abandonné
- Restreindre les fonctionnalités possibles des utilisateurs selon leur niveau d'accès — abandonné

Améliorations possibles

Il est certainement possible d'améliorer notre code en React puisqu'il s'agissait d'une nouvelle façon de coder. Sinon, la classe GestionSysteme est une classe que nous pourrions améliorer pour la rendre plus efficace autant en termes de performance de code que de compréhension de celui-ci.

Auto-Évaluation individuelle

Maxence

Je pense que ce projet fut une bonne évaluation de mes aptitudes et compétences acquises durant la technique. Ce projet m'a forcé à pousser plus loin les concepts et compétences acquis en classe. Je pense que nous pouvons en tant qu'équipes être fiers du projet qu'on a livré. Sans être parfaits, nous sommes arrivés à livrer une version du projet qui est assez fidèle à l'idée de base que nous avions. Finalement en pourcentage je me donnerais une note de 80 % pour l'ensemble du projet puisque je crois avoir bien fait, mais j'aurais pu en donner plus. Pour le travail d'équipe, je crois que nous avons fait de notre mieux. Nous aurions définitivement pu mieux communiquer surtout en début de projet où nous nous sommes lancés un peu chacun de notre côté. Nous aurions aussi pu faire en sorte d'avoir plus de séances de code à deux pour ce qui nous aurait été bénéfique. Je crois que j'aurais pu mieux supporter l'équipe en ayant une meilleure documentation du code.

Carlens

Personnellement je nous donne un 11/10, car je ne pense pas qu'avec notre niveau d'expérience au cours de la session, le temps donné et la charge des autres travaux, nous aurions pu faire mieux. Nous avons tous les 2 poussé sur nos recherches et arrangé des bogues impossibles, et nous avons appris de nombreuses nouvelles choses, et l'un comprend le code de l'autre, ce n'est pas comme si chacun codait dans son coin (au début oui, mais on s'est corrigé rapidement par la suite). De mon côté, j'ai définitivement du travail à faire pour ce qui est de choisir une architecture, car j'ai dû la changer plusieurs fois au cours du projet, tandis qu'en avoir une bonne m'aurait facilité la vie vers la fin. De plus, j'aurais définitivement dû ajouter de la documentation en cours de route.