

Rapport d'avancement de mi-mandat

Titre du projet : The Last Stand

Rapport de l'étudiant 1 :

Nom : Jonathan Robinson-Roberge

Fonctionnalité réalisée : Implémentation des librairies *frontend* (Phaser 3) et *backend* (Colyseus.js) utilisées, la première pour rendre le contenu visuel et la physique de notre jeu et la seconde pour synchroniser et sérialiser l'envoi et la réception des données pour rendre le contenu réactif en temps réel. Pour démontrer la synchronisation en temps réel de deux clients sur le serveur, il a été démontré que plusieurs carrées de couleurs différentes apparentées au joueur les contrôlant pouvait se déplacer sur le canevas de chaque client simultanément.

Le plus grand défi de cette réalisation : Le plus grand défi pour y arriver fut d'essayer de comprendre comment faire fonctionner ensemble deux librairies complètement nouvelles et d'adapter les solutions proposées par les différentes documentations et tutoriels pour les adapter à ce projet en particulier.

Emplacement dans le code: La ligne la plus significative en lien avec cette fonctionnalité se trouve dans le tag le plus rapproché en temps de la rencontre : <https://github.com/cryptoblivious/The-Last-Stand/releases/tag/v0.0.0-alpha>, dans le fichier `c61/sprint1/doc/the-last-stand/src/server/rooms/MatchOrchestrator.ts` à la ligne 30. On y voit un message `req_action` reçu par le serveur d'un client. Le serveur va donc aller dans son `state` (qui représente la structure de données sérialisables par *Colyseus.js*, pour y récupérer l'objet `gameEntity` correspondant au client qui vient d'envoyer la requête. L'action sera faite, qui apportera un changement dans le `state`, qui modifiera les données du `state`, qui seront ensuite rendues dans le `update` de tous les clients, puisque tous les clients partagent le même `state`.

Développement à venir : En guise de développement à venir que je juge important est de remplacer les rectangles par des *sprites* animés qui représenteront les personnages

des usagers. Je vais prendre en main cette partie du projet pour me familiariser avec la librairie Phaser 3 et la façon avec laquelle elle gère les *spritesheets* et leurs animations. Ainsi que d'agrémenter les fonctionnalités du jeu pour avoir quelque chose d'amusant et de complet.

Le plus grand défi de ce nouveau développement : Selon moi le plus grand défi sera de choisir ce que je vais implémenter, sachant que je n'aurai pas le temps de tout intégrer avant la remise finale, je dois trouver une bonne balance entre le nombre de fonctionnalités et la qualité de celles-ci.

Rapport de l'étudiant 2 :

Nom : Andrzej Wisniowski

Fonctionnalité réalisée : Mise en place d'une logique de session d'utilisateur par cookie basée sur l'authentification par Google en passant par divers middlewares permettant de faire le pont entre les droits d'accès Google et la base de données MongoDB Atlas.

Le plus grand défi de cette réalisation : Le plus grand défi de cette réalisation a été de comprendre de quelle façon chaque middleware doivent être implémentés de concert afin de permettre à la fonctionnalité de session par cookie d'être fonctionnelle. Bien que l'utilisation d'outils externe pour accélérer le développement d'un produit soit certe efficient, l'intégration de plusieurs outils de la sorte présente une complexité de détermination de bonne façon d'utiliser tous ces outils de concert, étant donné qu'un fournisseur d'outil n'a pas nécessairement de démonstration de comment intégrer leur outil avec tous les autres outils utilisés.

Emplacement dans le code : Cette fonctionnalité est démontrée dans le code de la meilleure façon à deux emplacements distincts; le premier étant là où les middlewares sont initialisés correctement pour être utilisé par le serveur afin de rendre possible les sessions utilisateurs par cookie, et le deuxième étant là où la logique d'authentification spécifique à Google est mise en place. Les chemins d'accès qui seront présentés prochainement seront valides pour le commit en date de la présentation de l'avancement, soit celui relié à la version v0.0.0-alpha du projet.

En ce qui concerne ce premier point, les middlewares pertinents sont implémentés dans le fichier `The-Last-Stand/C61/sprint1/doc/the-last-stand/src/server/arena.config.ts`. Les lignes pertinentes à ce sujet sont les lignes 6 à 12 (pour les importations de middlewares), les lignes 19 à 27 pour certaines configurations, et les lignes 46 à 101 pour ce qui est de l'initialisation méthodique et ordonnée des middlewares.

Pour ce qui est de l'implémentation de la logique d'authentification avec Google, le code pertinent peut être trouvé à l'emplacement 60 à 115. Ce code présente la structure de code attendue par l'API de Google pour permettre la reconnaissance de l'utilisateur, ainsi que le code nécessaire pour ajouter les informations concernant l'utilisateur à la base de données, autant dans la collection "sessions" que la collection "users".

Développement à venir : Intégration d'une logique de gestion de la présence ou non d'un utilisateur en fonction de sa présence dans l'application (online/offline status system), d'une logique d'ajout d'amis (friendlist system), et d'une logique de communication entre utilisateurs (chat system).

Le plus grand défi de ce nouveau développement : La mise en place de ces fonctionnalités nécessitera des informations en temps réel concernant les utilisateurs, ainsi qu'une interface utilisateur intuitive et facile à utiliser. Il sera donc nécessaire de mettre en place une logique de communication en temps réel entre les clients et le serveur, et cela à l'échelle de l'application au complet. De plus, l'interface utilisateur permettant d'accéder à ces fonctionnalités devra être claire et simple pour que les utilisateurs puissent facilement ajouter des amis, évaluer leur présence, et communiquer avec d'autres utilisateurs.

Estimation du chemin parcouru :

Nous estimons que nous sommes environ à 50% de l'objectif que nous nous étions fixé. Nous avons vu très grand, en restant réaliste, nous croyons que nous pourrions probablement livrer environ 80% de notre planification globale.

