P_PRO - PWA

Mathias Rogey
Santiago Sugrañes
CIN2a
ETML
Du 16.04.2021 au 02.06.2020

Table des matières

Description

PWA (Progressive Web App) est une application web qui consiste à faire apparaître des page web à l'utilisateur de la même manière qu'une application native sur desktop ou téléphone.

Ceci permet de combiner les fonctionnalités offertes par les navigateurs web avec les avantages de l'expérience offerte par les appareils mobiles.

Une PWA se consulte comme un site web classique, pour chaque plateforme, avec ou sans connexion internet. Ceci permet donc d'avoir une expérience similaire, voir exacte, à une application desktop/mobile avec légèreté.

Cahier des charges

Réaliser une application PWA.

Analyse

Opportunités

Pour la réalisation de notre application PWA, nous avons décidé de nous mettre un défi en faisant une vraie application que nous pouvons utiliser pour un serveur faisant office de pont pour ensuite interagir avec un robot dans le jeu Minecraft.

Pour pouvoir réaliser cette application, nous avons dû utiliser NodeJS, Three.js, le langage Lua et des WebSockets.

Réalisation

Réalisation Application

Nous avons créé une page web PWA qui communique avec un serveur à l'aide de WebSockets. Un robot du mod ComputerCraft Tweaked a été programmé en Lua afin d'agir comme un client auprès du même serveur.

Le client web envoie une commande, par exemple "avancer tout droit". Ensuite, le serveur interprète la commande et le broadcast sur l'ensemble des clients connectés au serveur.

Lorsque le Robot reçois cette commande via le web socket, il l'interprète, exécute la commande donnée et envoie un paquet contenant sa nouvelle position ainsi que les informations du block se trouvant devant, dessus en en dessous de lui au serveur.

Lorsque le serveur a fini d'interpréter le paquet du robot, nous l'envoyons au client qui s'occupe de modifier la position du robot de son côté.

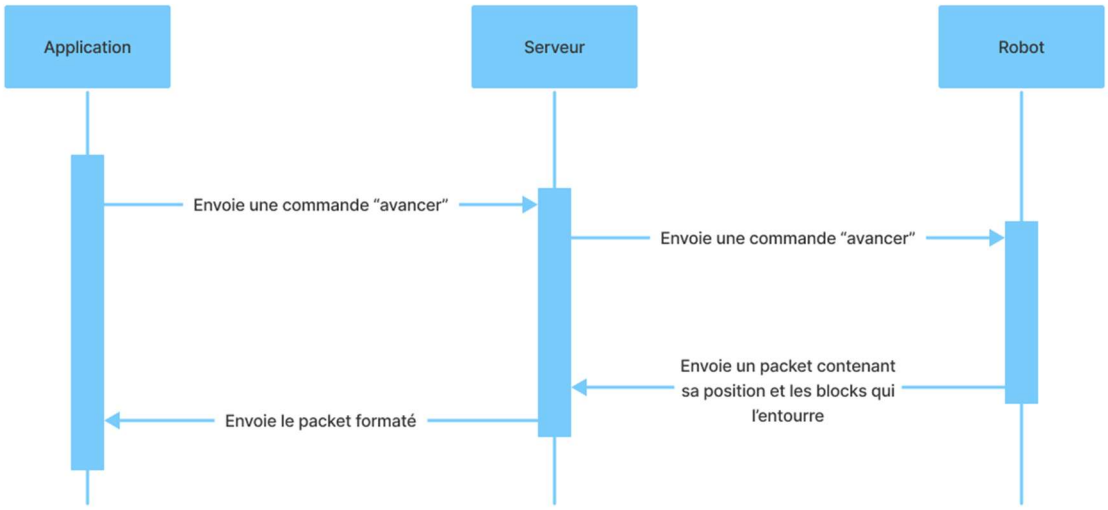

Figure 1 : Diagramme de séquence

Lorsque le client reçoit la nouvelle position, elle doit update le monde que nous répliquons grâce à Three.js. Ceci demande de bouger le robot, créer le monde pendant que le robot bouge, stocker le monde dans une base de données locale pour que l'utilisateur n'ait pas à re-mapper son monde à chaque fois qu'il lance l'application. Cette « base de données » n'est malheureusement pas très fiable, car nous utilisons le localstorage, qui agit comme le cache.

Réalisation PWA

Pour la réalisation de la PWA, nous devions d'abord nous documenter sur ce que c'est et comment cela marche, car aucun de nous deux en avions fait. Pour ceci, nous avons lu beaucoup de documentation de la part de Google et de Firefox.

Pour pouvoir transformer notre site web en PWA, nous devions utiliser une extension chrome nommée « Lighthouse ». Elle nous permet de générer un rapport de notre site web et permet de voir toutes les améliorations nécessaires pour afin d'atteindre notre objectif.

Pour remplir notre objectif, nous avons décidé de nous concentrer sur la version desktop de notre application car nous ne voulions pas perdre de temps à faire du CSS responsif sur téléphone.

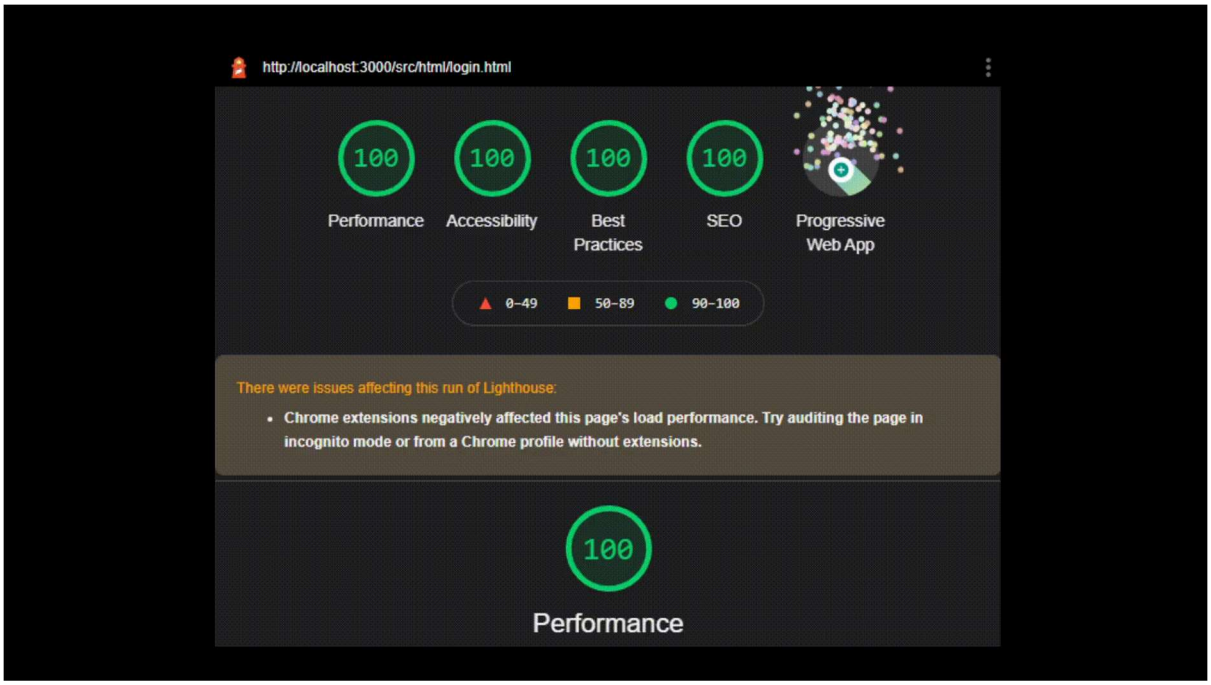

Figure 2 : Résultat Lighthouse

Lorsque notre application était validée par Lighthouse, le navigateur (dans notre cas, Brave) détecte alors automatiquement que notre site web est admissible à être téléchargé en tant qu'application PWA.

Figure 3: Télécharger l'application PWA

Le résultat final nous donne une application PWA qui tourne dans notre navigateur favori, sans internet, sur desktop comme sur téléphone.

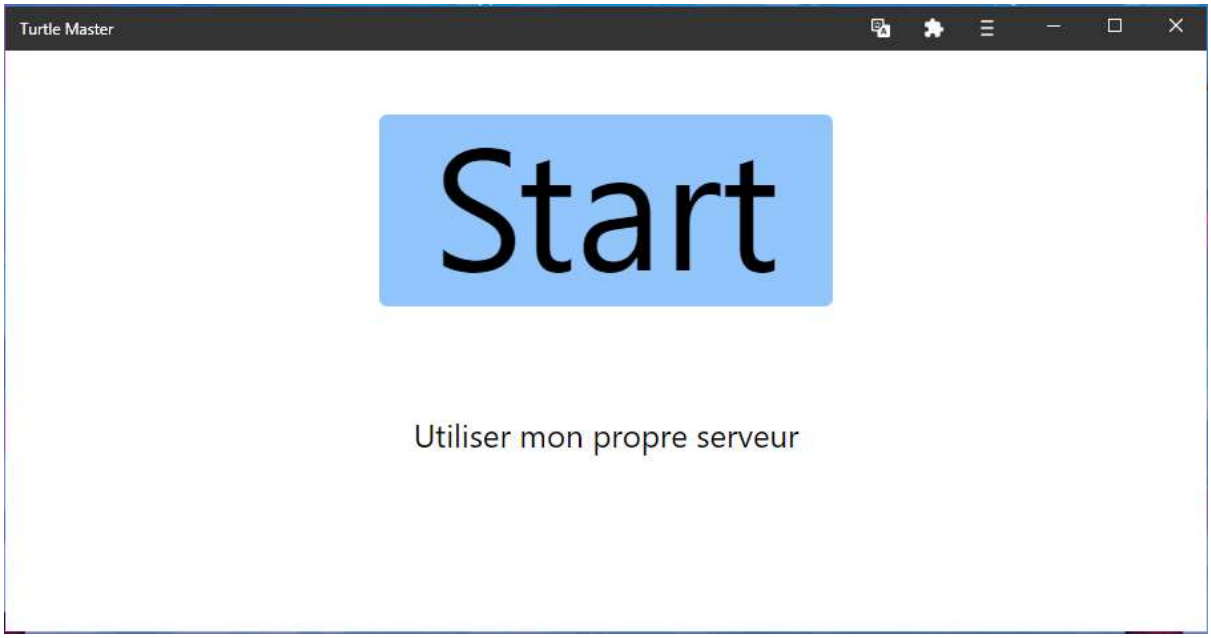

Figure 4: Application PWA final

Conclusion

Bilan personnel

Créer une application en partant de rien et en ayant des connaissances de bases dans les librairies et langages de programmation utilisés était un défi. Ceci nous a permis d'élargir notre horizon de connaissances et d'apprendre à utiliser des libraires JavaScript très populaires.

Annexes

Webographie

Three.js - Documentation

Lua - Documentation

CC : Tweaked - Documentation

PWA - Documentation Firefox, Documentation Google

Lighthouse - Documentation