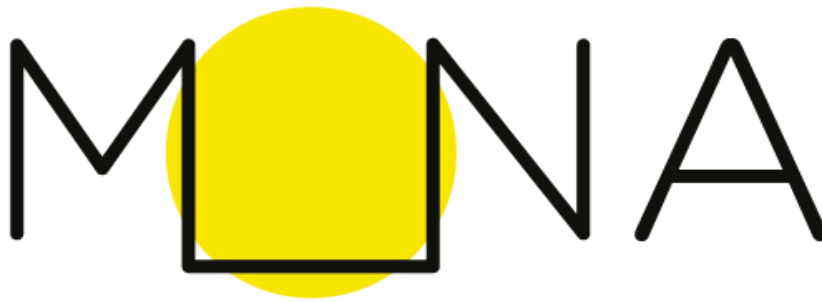


Sarah Heng

Matricule 20220310

Rapport final du projet informatique



Dans le cadre du cours IFT3150

Pour la session d'été 2024

Université de Montréal

sarah.heng@umontreal.ca

MONA est un organisme à but non lucratif qui veut promouvoir et rendre accessible l'art public à la communauté, qui organise des activités dont des activités de médiation culturelle pour montrer et informer sur l'art public et comprend une application mobile qui contribue à réaliser cette mission.

Le projet informatique consiste à participer au développement d'une application mobile permettant une chasse au trésor de l'art public afin de découvrir ce dernier. Mon but pour ce semestre est de faire la mise à jour pour la nouvelle interface de l'application et de corriger des bugs rencontrés.

Mon rôle, cet été 2024, était en développement mobile avec le framework Ionic, JavaScript/TypeScript, CSS, Vue.JS et la librairie OpenLayers. J'ai utilisé WebStorm, Google Chrome, XCode et Android Studio Code. Le framework Ionic est hybride, donc, un code est utilisé pour Android et iOS, mais il y a des bugs parfois comme le CSS qui ne s'affiche pas de la même manière pour Android et iOS.

Corélie, qui s'occupe du côté serveur, Lena et moi avons des rencontres à chaque deux semaines pour faire l'état de notre progression. J'ai eu beaucoup de rencontres avec Kim aussi qui m'a montrée le projet, coachée et beaucoup aidée pour le développement mobile, ce qui m'a permis de voler plus de mes propres ailes par la suite. Nous recevons du feedback de plusieurs membres de l'équipe testent l'application. Il y a eu des rencontres avec Barbara, la designer, aussi pour la nouvelle interface. La communication écrite est par Element.

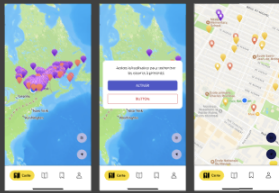
L'état des lieux au début du projet versus maintenant:

Au début, dans l'application, c'était l'ancienne interface, mais j'ai implémenté différentes composantes de la nouvelle interface (surtout la page Home). Il y avait 5 pages (comme j'ai lu dans le rapport de Gaspard, un ancien étudiant dans le projet) soit Découverte du jour, Liste, Carte, Collection et Autres. Maintenant, il y en a 4, car Découverte du jour a été enlevée dans la nouvelle interface.

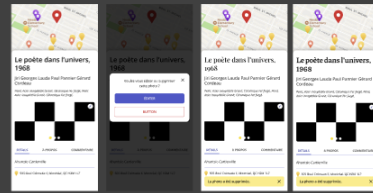
J'ai implémenté les parties terminées côté design sur Figma de la nouvelle interface. Je m'étais donnée comme but de finir la nouvelle interface Home pour ce semestre, qui est la Carte, qui est presque finie (Kim a fait Home - Disabled localization et j'ai fait d'autres fonctionnalités sauf Home – Filters flow qui n'est pas encore fini côté design).

Voici le design Figma pour la carte en ce moment:

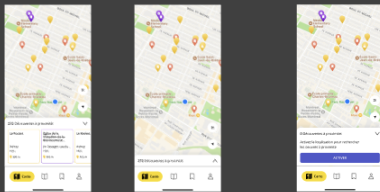
Home - Disabled localization



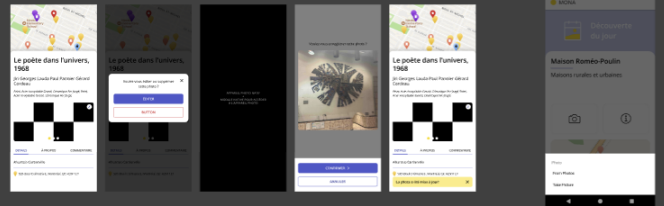
Home - Remove photo flow



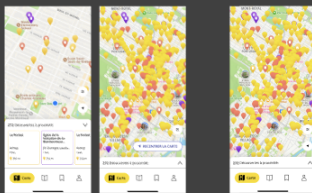
Home - Modale Map



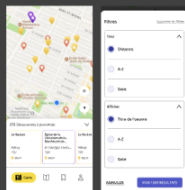
Home - Editing photo flow



Home - De-zoom Map



Home - Filters Flow



Ce que j'ai fait:

Setup du projet:

Au début, après la première rencontre, avec Guy Lapalme et Lena, et avant que ma camarade et moi recevions nos rôles, j'ai téléchargé et testé l'application. Ensuite, j'ai rencontré Kim, Corélie et 3 autres étudiantes en histoire de l'art participant au projet.

Ensuite, j'ai fait le setup avec Kim sur le MacBook qui était nouveau (alors, il n'y avait rien dessus). Cela a pris environ 2-3 semaines pour faire la configuration pour le projet, installer les dépendances, les logiciels reliés, etc. à cause de problèmes d'accès et d'autres choses dont on n'a pas pensé pour un nouveau MacBook. Il y avait plusieurs problèmes pour faire fonctionner le projet et pouvoir travailler dessus, par exemple, les émulateurs Android et iOS qui ne fonctionnaient pas ou faire un build dans Android Studio Code et Xcode. Cela demandait un peu de recherches et de "débugage" pour les faire fonctionner, comme l'installation de Homebrew pour installer Java 17 parce qu'Android Studio Code marche seulement avec Java 17 apparemment et de CocoaPod pour faire fonctionner un fichier s'appelant PodsFile pour XCode selon mes recherches. Cela fait que je comprends mieux comment marchent Android Studio Code et XCode maintenant. J'étais habituée à utiliser le système Windows, alors, cela m'a demandée un peu d'adaptation.

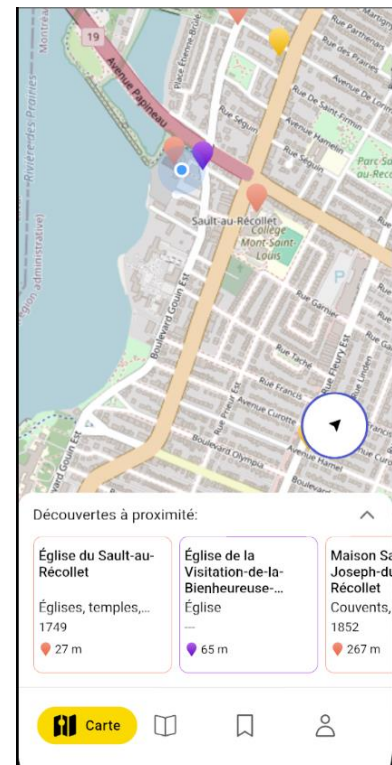
Développement:

Pour me familiariser avec le code:

Après avoir réussi à setup le projet, j'ai fait des issues pour débutants: • j'ai fait des toasts pour les messages d'erreurs des pages Login et Register qui seront éventuellement modifiées pour la nouvelle interface. Pour cela, j'ai appris comment utiliser les APIs et Postman, qui sert à voir comment marche un API. • J'ai corrigé un bug de titres dans la liste qui se collaient à droite et j'ai ajouté un key listener sur les forms pour qu'on puisse envoyer le formulaire en cliquant Enter. Ensuite, • j'ai modifié le pin de l'emplacement de l'utilisateur qui était une image par un point qui a un style dans la librairie OpenLayers et qui a un rayon changeant selon l'incertitude de l'estimation de l'emplacement de l'utilisateur.

Ce que j'ai implémenté ensuite:

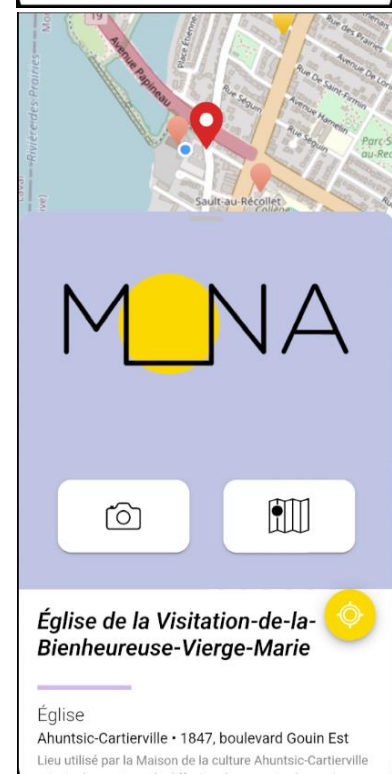
- Plus tard, j'ai modifié la barre de navigation pour correspondre à la nouvelle interface, ce qui m'a fait apprendre plus CSS, dont le box model.
- J'ai aussi fait que le bouton recentrer change d'apparence selon que le pin d'emplacement de l'utilisateur soit centré, décentré, mais dans l'écran, ou hors de l'écran.
- J'ai aussi implémenté la fenêtre de découvertes à proximité selon le design Figma.



- J'ai modifié les tailles de pins selon le zoom sur la carte selon environ ce que Lena et Barbara voulaient avec mes captures d'écran de tailles de pins à des zooms différents, car il y a eu du feedback que la taille des pins était trop grande.
- J'ai aussi modifié le focus du pin qui devient plus grand, rouge et centré en haut de l'écran avec sa modale description qui apparaît lorsque le pin est sélectionné en cliquant ou de la fenêtre de découvertes à proximité ou de la liste de découvertes.

Puis, • j'ai corrigé des bugs dont les images qui ne s'affichaient pas après une mise à jour de dépendances avec Kim.

Ce que j'ai appris et les difficultés rencontrées:



Au début, j'ai beaucoup lu la documentation avant de me mettre à coder... en pensant que cela me préparerait mieux au code... ensuite, j'ai réalisé que j'apprenais mieux en lisant la documentation qui correspond à ce que j'ai besoin dans le code, car, comme Kim et Lena ont mentionné, on n'utilise pas tout ce qui est dans la documentation. Malgré cela, je sens qu'avoir beaucoup lu la documentation m'a quand

même aidée, peut-être m'a donnée une petite lancée avec laquelle je continue le "momentum" en lisant la documentation lorsque j'en ai besoin en codant. Lire la documentation me permet de mieux comprendre le langage et d'avoir plus de contrôle/précision pour le développement des fonctionnalités. — D'ailleurs, j'ai découvert que je raffole de lire la documentation et de mettre en pratique ce que je lis... la documentation me permet d'exécuter la créativité.

J'ai toujours été curieuse sur comment on développait des cartes en informatique, alors, je me suis attaquée à apprendre OpenLayers. Je suis contente d'avoir appris et codé avec cette librairie, car je la trouve très utile.

J'ai gardé en tête les futures générations en codant en essayant de rendre mon code clair, simple et commenté afin qu'il soit plus facilement compris en requérant moins de connaissances du langage du code pour le comprendre (j'ai essayé de décrire ce que les fonctions/le code fait). En codant lors de ce projet, j'ai commencé à me dire qu'il faut que j'essaie de rendre les choses simples et efficaces (moins de code et/ou plus clair).

De plus, au début, des trucs qui me semblaient simples, finalement, demandaient plus de travail que je pensais puisqu'il faut en premier apprendre et comprendre le langage ou la librairie afin de savoir quelles fonctions ou quels mots utiliser ou seraient plus optimaux à utiliser dans le langage pour développer la fonctionnalité et, ensuite, parfois, les choses ne se passent pas comme on le pensait et il y a des bugs... ce qui m'a fait réaliser qu'il faut parfois comprendre comment fonctionne en arrière le framework/la librairie pour trouver des solutions aux bugs.

Conclusion:

Je sens que j'ai bien travaillé et que j'ai fait du bon progrès, surtout qu'au début du projet, j'étais novice et je ne connaissais pas bien GitHub, Ionic, Vue.JS et OpenLayers. Je sens que j'ai beaucoup appris et j'ai beaucoup aimé travailler dans ce projet informatique. Je me sens définitivement plus à l'aise avec l'implémentation/codage et je suis contente de travailler en équipe, partager des idées et voir les idées des autres. D'ailleurs, je pense que l'application pourrait donner de bons parcours pour marcher dehors/donner une raison/un but pour aller dehors qui est de visiter l'art public. Je resterai encore avec MONA ensuite et je continuerai

d'implémenter la nouvelle interface. J'aimerais finir d'implémenter la nouvelle interface dans le futur et de faire des tests automatisés. Aussi, si j'ai le temps dans le futur ou pour les futures générations, utiliser le state management Pinia pour optimiser comment sont gérées les données globales serait bien.