

HEIG-VD

CRÉATION D'UNE CARTE INTERACTIVE ET DE SON
INFRASTRUCTURE PERMETTANT L'AFFICHAGE DE VOTATIONS
GÉO-LOCALISÉES.

Cahier des charges

Auteur :

Jérémie CHÂTILLON

Enseignant :

Pier DONINI

Mandant :

Arcanite Solutions Sàrl - www.arcanite.ch

Contact :

Lionel MARTIN - lionel.martin@arcanite.ch

Johan PARATTE - johan.paratte@arcanite.ch

12 avril 2019



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD

www.heig-vd.ch

Table des matières

1	Introduction	2
2	Mise en contexte	2
3	Buts	2
4	Technologies	2
5	Travail à réaliser	2
5.1	Workpackage 1 : "Votation Map"	3
5.1.1	Fonctionnalités de bases	3
5.1.2	Si le temps le permet	3
5.2	Workpackage 2 : "Infrastructure serveur"	3
5.2.1	Fonctionnalités de bases	3
5.2.2	Si le temps le permet	4
5.3	Workpackage 3 : "Interface utilisateur"	4
5.3.1	Fonctionnalités de bases	4
5.3.2	Si le temps le permet	4
6	Cas d'utilisations	4
7	Interface graphique	5

1 Introduction

Ce document est le cahier des charges du travail de Bachelor de fin d'études en informatique logiciel de la HEIG-VD département TIC.

2 Mise en contexte

Ce travail permettra de visualiser une carte du monde ; en fonction de jugements/votes associés à des positions géolocalisées, émis par des utilisateurs. Il existe déjà des outils permettant de faire une cartographie par un nuancé de couleurs (Heat map) en fonction de la fréquence d'une information. Il existe aussi des cartes qui permettent de juger un lieu (Choropleth map). Cependant, il n'existe pas d'outil qui produise les deux. Ce qui consisterait de pouvoir juger une position sur un critère définit (sur une échelle de 1 à 10 par exemple) et de faire une cartographie de tous ces jugements.

3 Buts

Le but de ce travail est de créer un outil générique permettant la création d'instance de votations accessible par tout le monde. Le but final de ce travail est de créer une instance permettant la visualisation de la propreté du monde. La visualisation se fera au travers d'une carte avec des nuancés de couleurs pour exprimer les différents seuils de propretés. La propreté sera définie par des votations émises par n'importe qui dans le monde en fonction de sa position géographique. La plate-forme doit être public et accessible par tout le monde. Elle a pour but d'être utilisée à grande échelle.

4 Technologies

Les contraintes technologiques sont d'utiliser le "Django" et une base de données SQL. Pour le reste, aucune contrainte n'est imposée mis à part d'utiliser au maximum des outils open source. Il est permis d'utiliser des librairies déjà existantes. L'infrastructure pour héberger l'application sera mise à disposition.

5 Travail à réaliser

Ce travail consiste en 3 Workpackages :

- Workpackage 1 : Création d'une "Votation Map" permettant l'affichage de votations liées à des positions géographiques.

- Workpackage 2 : Création d'une infrastructure serveur générique permettant la création d'instances de votations et le traitement de celles-ci.
- Workpackage 3 : Création d'une interface utilisateur pour que n'importe quelle personne puisse visualiser et soumettre des votations liées à la propriété d'un lieu (à condition d'être connecté).

5.1 Workpackage 1 : "Votation Map"

5.1.1 Fonctionnalités de bases

La carte à réaliser doit être générique et réutilisable dans n'importe quel autre projet, peu importe le type de votations qu'elle affiche. Les notes liées aux votations sont représentées par un nuancé de couleur qui apparaît au-dessus la carte du globe. Il faut définir un système qui calcule l'interpolation ou l'agrégation ainsi que la portée (distance) des différentes votations afin de faire un affichage cohérent.

5.1.2 Si le temps le permet

La carte doit être un paquet logiciel utilisable par n'importe quel autre développeur. Elle devra être mise à disposition au travers d'un "packet manager" (tel que "NPM"¹ par exemple) et respecter les conventions de déploiement.

5.2 Workpackage 2 : "Infrastructure serveur"

5.2.1 Fonctionnalités de bases

Ce Workpackage consiste à réaliser une application serveur répondant aux attentes du WorkPackage 3². Il faut aussi qu'il soit générique en ce qui concerne les types de votations.

N'importe qui pourra faire une demande pour créer une nouvelle instance de votations et la paramétrer. Le paramétrage se définit par la création ou réutilisation de règles de validité/impacte/-poids/etc. des votations.

Il est nécessaire de stocker toutes les votations pour pouvoir établir un historique. Les votations sont supprimées si et seulement si leur instance associée est supprimée.

Une instance appartient à un et un seul utilisateur. Cependant, il a la possibilité de partager les règles de paramétrage et d'en copier depuis une autre instance déjà existante.

1. NPM(Node Packet Manager): <https://www.npmjs.com/>

2. Cf. chap 5.2.2 Workpackage 3 : "Interface utilisateur"

5.2.2 Si le temps le permet

Ajout de règles.

Optimisation des recherches en utilisant des Datas-Frames.

Permettre la visualisation des votes dans le temps.

5.3 Workpackage 3 : "Interface utilisateur"

5.3.1 Fonctionnalités de bases

Il faut réaliser une interface utilisateur destinée au tout public. Elle doit être accessible par ordinateur ainsi que par smartPhone. Elle doit être simple à prendre en main et ergonomique.

Tout utilisateur peut visualiser la carte de votations. Afin de pouvoir émettre des votes, l'utilisateur doit posséder un compte et être connecté à l'application (Système de gestion utilisateurs). L'utilisateur peut décider d'utiliser un compte externe s'il ne veut pas en créer un (utilisation de "Google Sign-In"³ par exemple).

Un utilisateur connecté a aussi la possibilité de visualiser uniquement ses propres votes.

Le critère des votations sera la propriété du lieu où se trouve physiquement l'utilisateur. Comme la propriété est une notion abstraite, nous n'imposons pas de restrictions particulières. C'est à l'utilisateur de définir par lui-même si un lieu est propre ou pas.

Il faut posséder une connexion internet pour pouvoir accéder à l'application.

5.3.2 Si le temps le permet

Rendre l'application multilingue (en tout cas français et anglais).

Faire en sorte que l'application soit disponible sans connexion internet. C'est-à-dire que l'utilisateur puisse soumettre une votation peu importe l'état de sa connexion. Les données seront mises à jour dès qu'il sera à nouveau connecté à internet.

Permettre la visualisation des votes dans le temps.

6 Cas d'utilisations

Les cas d'utilisations sont fournis en fichier annexe⁴.

3. GoogleSign-In: <https://developers.google.com/identity/>

4. Cf. "Annexe : Cas d'utilisations"

7 Interface graphique

Les mokups ci-dessous représentent à quoi devrait ressembler l'interface utilisateur. D'autres mokups sont disponibles en annexe⁵.

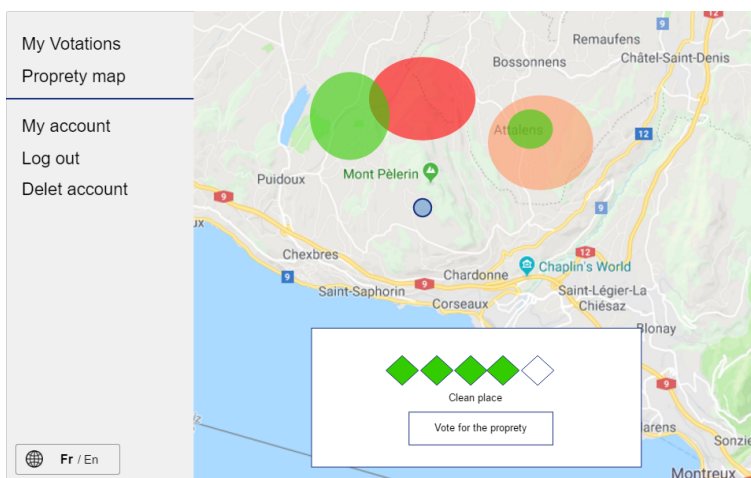


FIGURE 1 – Mokup carte utilisateur connecté

5. Cf. "Annexe Mokups interface utilisateurs"

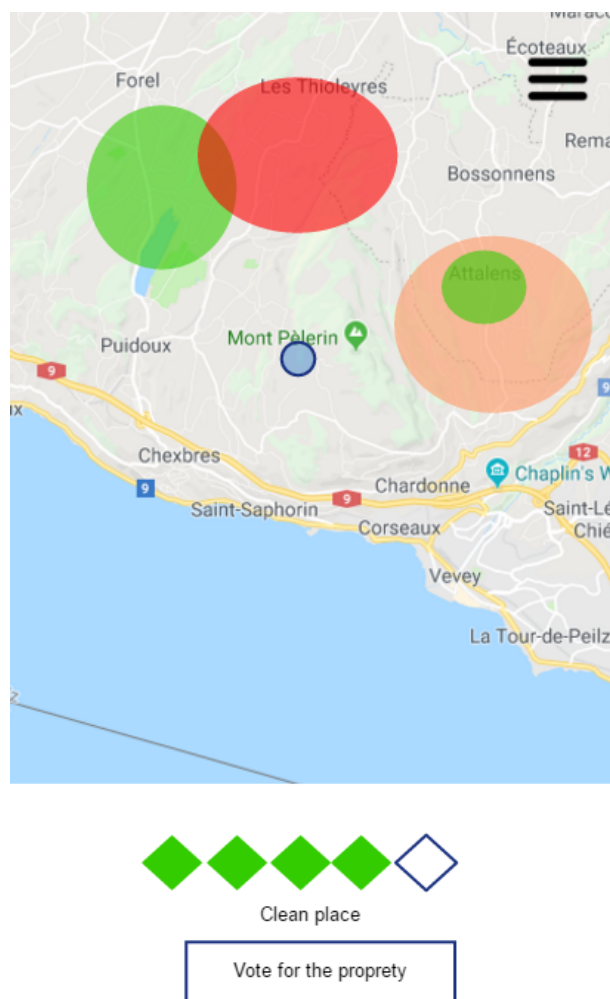


FIGURE 2 – Mukup carte utilisateur connecté version mobile