

Rapport de projet master 1^{ère} année

Gestion de traces GPS à l'aide de Google Earth



Etudiant :

Baptiste Moineau

Responsables de projet :

Cédric Chateau

Nathalie Corson

Antoine Dutot

Yoann Pigné

Enoncé du projet

Le travail demandé consistera à :

- Fournir une interface de visualisation des données GPS (et des fonds de cartographie, lorsqu'ils sont disponibles).
- Analyser à partir de ces données l'évolution des vents et les restituer sur le fond de carte.
- Donner des outils, à partir de ces données, permettant d'analyser et d'interpréter la performance des compétiteurs (statistiques, etc.).

Plan

INTRODUCTION	3
1. PRESENTATION DU PROJET	3
1.1 Les besoins exprimés	3
1.2 Les outils à disposition	3
1.2.1 Google Map	4
1.2.2 Google Earth	4
1.2.3 Librairie Mootools	5
2. REALISATION	6
2.1 Présentation de l'interface graphique	6
2.2 Présentation des fonctionnalités	7
2.2.1 Sélection de la trace	7
2.2.2 Lecture de la trace	7
2.2.3 Menu d'orientation	8
2.2.4 Les autres fonctionnalités	9
2.3 Difficultés rencontrées et solutions apportées	9
2.3.1 La rotation sous Google Map	9
2.3.2 L'API Google Earth	10
2.3.3 Ordonnancement des calculs	11
2.3.4 Lissage de la courbe	11
3. PERSPECTIVES D'EVOLUTION	12
CONCLUSION	13

INTRODUCTION

Le sport moderne tel que nous le connaissons demande toujours plus de performance, d'exploit et de spectacle. Ceci est tellement bien représenté par cette bulle grandissante que constitue le sport-business avec comme figure de proue le football. Dans cette optique, de nombreuses techniques de biomécanique, d'études physiques et physiologiques sont menées. Ce phénomène subit une expansion dans tous les domaines et touche dorénavant des sports moins populaires que le football ou le basket. C'est ainsi que la ligue de voile de Normandie a ressenti le besoin de disposer d'une solution logicielle pour permettre d'étudier les performances, mieux les expliquer et l'utiliser en tant qu'outil pédagogique auprès des plus jeunes.

1. PRESENTATION DU PROJET

1.1 Les besoins exprimés

La ligue de voile de Normandie, par l'intermédiaire de M. Cédric Chateau, a pris contact avec l'université du Havre afin de demander la création d'un outil pédagogique pour former ses navigateurs. Cette demande a donc fait l'objet d'un projet que j'ai décidé de réaliser avec l'aide de Mme Nathalie Corson, M. Antoine Dutot et M. Yoann Pigné.

Le principe du logiciel demandé est de pouvoir visualiser sur une carte le parcours préalablement enregistré par un GPS lors d'une course en mer.

M. Cédric Chateau a formulé un certain nombre de besoins :

- Le logiciel doit se placer dans le domaine du grand public, il doit donc être facile d'accès même pour des personnes non-expérimentées en informatique. Il doit aussi être utilisable sur les plates-formes les plus courantes à savoir Windows XP ou Vista.
- Le logiciel doit pouvoir tracer une dizaine de traces en même temps. Ces traces doivent se chevaucher dans le temps.
- Le logiciel doit permettre d'orienter la carte selon plusieurs axes afin d'améliorer la pédagogie à savoir : sens du vent, sens du parcours ou vers le nord. L'utilisateur peut aussi orienter la carte selon son désir et faire des zooms.
- Le logiciel doit éviter de placer des points dont les coordonnées sont incohérentes avec les autres.
- Le logiciel doit enfin permettre de visualiser en temps réel le déplacement des bateaux.

1.2 Les outils à disposition

A partir de l'analyse des besoins exprimés, j'ai dû faire de nombreux choix pour réaliser le travail demandé.

Tout d'abord, il a fallu que je trouve un moyen de représenter les traces sans connaître à l'avance à quel endroit du monde elles pouvaient se trouver. Il fallait donc utiliser un logiciel de cartographie libre d'accès. Google et Geoportail proposent tous les deux des APIs¹ disponibles pour le grand public. Après m'être documenté sur les deux, j'ai choisi Google et son outil Google Map. Cette API est disponible pour deux langages de programmation, à

¹ API= Application Programming Interface ou Interface de Programmation : ensemble de fonctions, procédures ou classes mises à disposition des programmes informatiques par une bibliothèque logicielle, un système d'exploitation ou un service. (Wikipedia)

savoir le Java et le Javascript. Me sentant plus à l'aise dans la programmation web, j'ai codé le projet en HTML/Javascript.

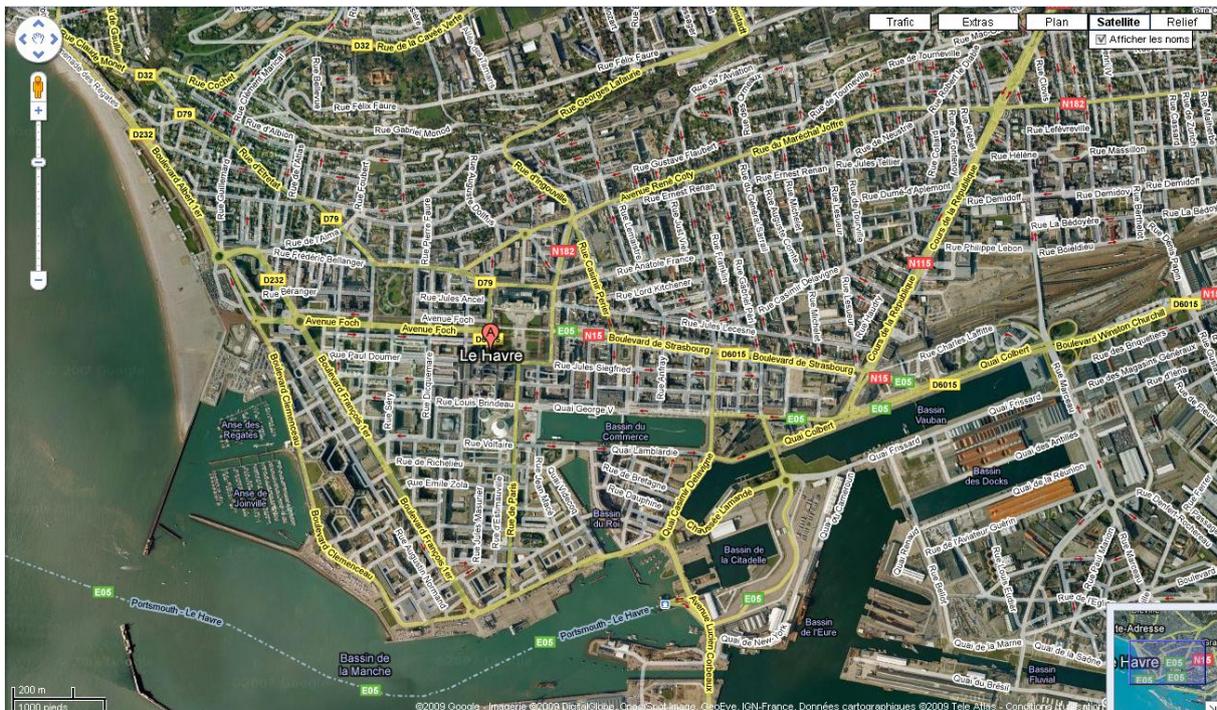
1.2.1 Google Map

Google Map est une application de plus en plus répandue sur internet. Disponible depuis plusieurs années, l'API est très fiable. La seule condition d'utilisation est que l'accès aux cartes ne soit pas privatif ou pour un usage professionnel. Le cadre du projet étant associatif, la charte d'usage est respectée.

Pour commencer, il faut s'inscrire sur le site de Google afin d'obtenir une clé d'utilisation. Cette clé doit être placée en entête du site pour permettre l'affichage des cartes.

Une multitude de fonctionnalités est ensuite disponible et permet, entre autres, de placer des marqueurs ou zoomer sur la carte. L'avantage de cette API est qu'elle reconnaît automatiquement les fichiers GPX issus des GPS. De ce fait, l'affichage du parcours est presque automatique.

J'ai utilisé cet outil pour réaliser la première version du projet mais un élément a bloqué ma progression. Plus de précisions sur ce problème dans la suite de ce rapport (partie 2.3.1).



Google Map centré sur Le Havre

1.2.2 Google Earth

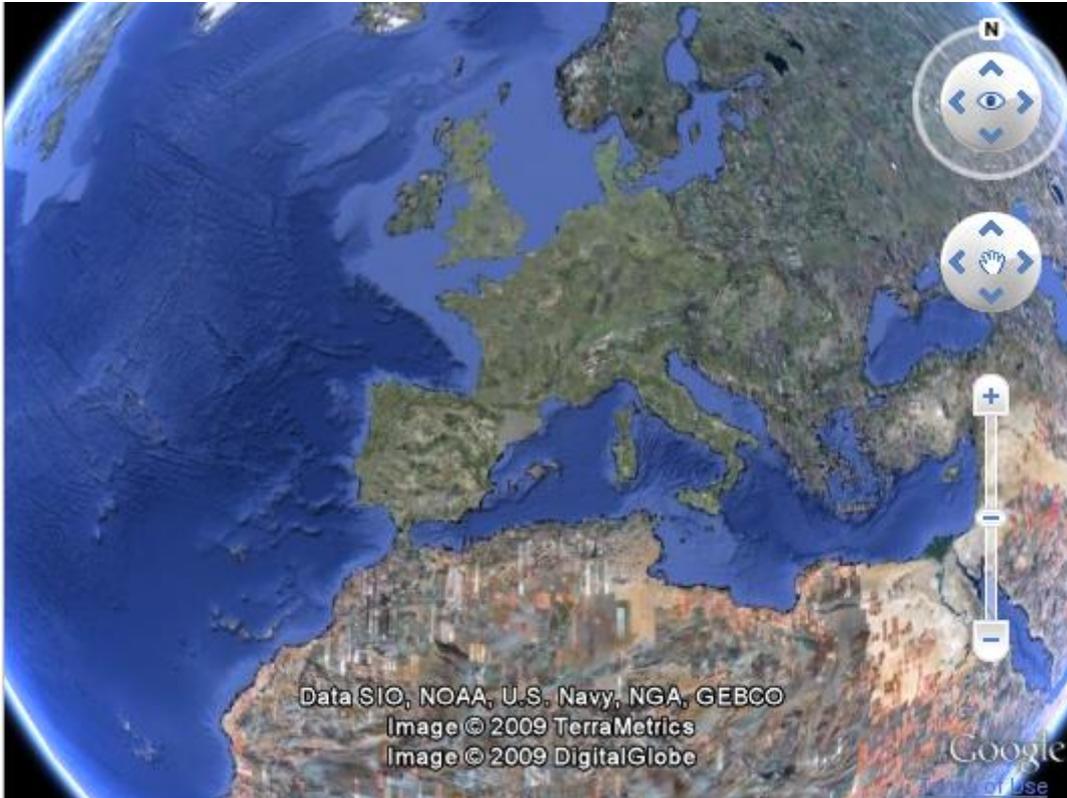
Disponible à l'origine sous forme de logiciel, Google Earth prend à son tour, le chemin de l'internet. Grâce à une extension installée sur le navigateur internet², l'application permet d'afficher des cartes en 3 dimensions. Les conditions d'utilisation sont les mêmes que pour son grand frère Google Map donc là encore, il n'y a aucun soucis dans le cadre de ce projet.

Par contre la mise à disposition pour le grand public est beaucoup plus récente. De ce fait, beaucoup de fonctionnalités présentes pour Google Map ne sont pas accessibles. Les deux

² Cette extension est installée en natif sur le navigateur de Google : Chrome. Pour les autres navigateurs, le plugin est disponible à cette adresse : <http://code.google.com/intl/fr/apis/earth/>.

APIs ayant été développées par deux équipes différentes, l'uniformisation est seulement en cours de réalisation. Il est possible de convertir une carte en 2 dimensions vers une carte en 3 dimensions mais les fonctions non reconnues sont alors ignorées.

Pour des raisons techniques, j'ai tout de même choisi de migrer sur ce support.



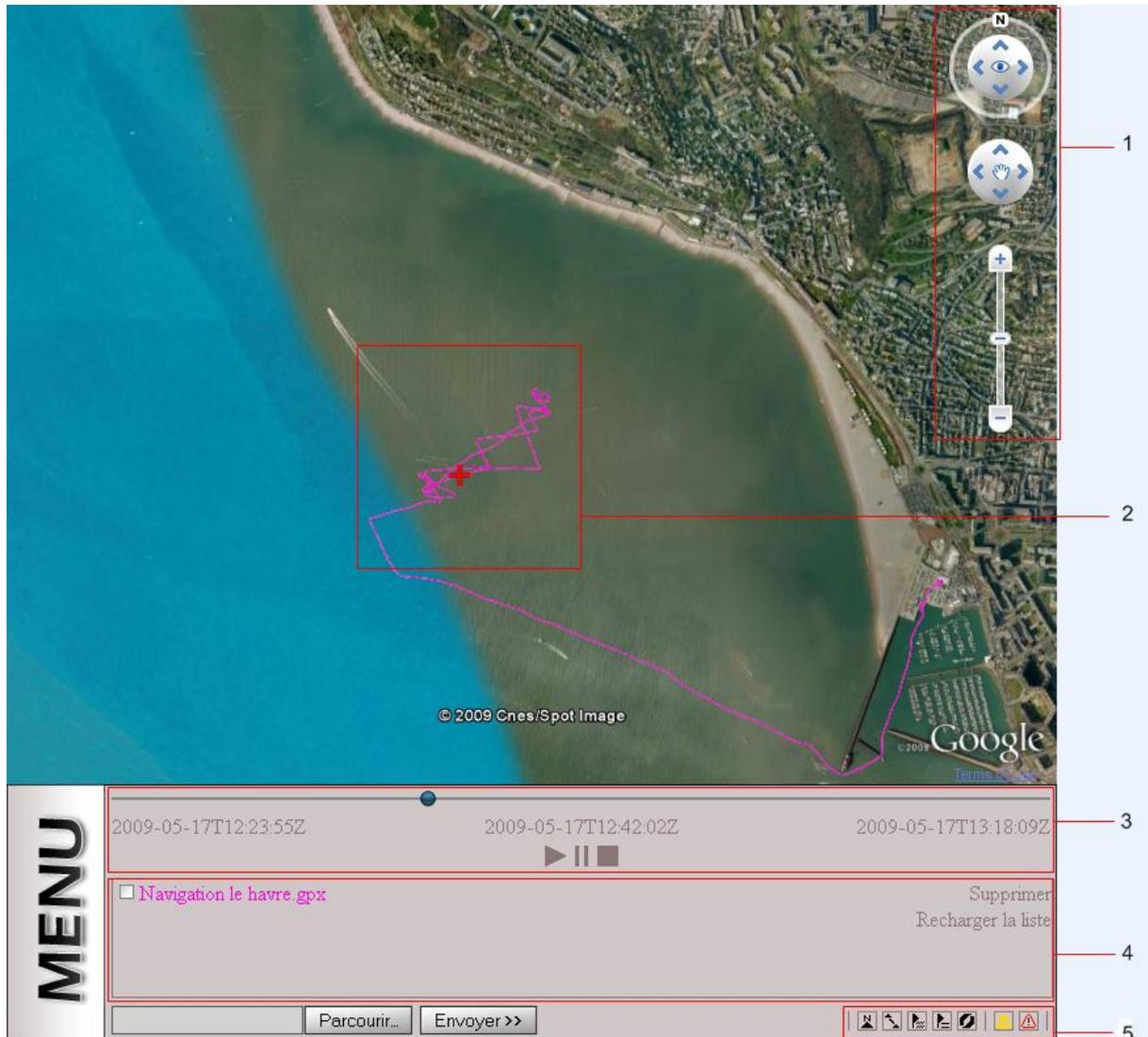
Google Earth centré sur la France

1.2.3 Librairie Mootools

Ayant la possibilité de réaliser le projet en programmation web, j'ai choisi d'utiliser la librairie Javascript Mootools. Cette librairie présente de nombreux points très utiles pour réaliser le travail. Elle permet de faire des requêtes HTML/PHP sans recharger la page ou encore de donner un dynamisme à la page. Pour ma part, le principal intérêt que j'ai trouvé à cette librairie est la facilité avec laquelle il est possible d'ordonner les tâches. En effet, le Javascript est un langage interprété et donc plusieurs calculs peuvent être effectués en parallèle. Or dans le cadre de ce projet, certains calculs ne devaient débiter que lorsque les calculs précédents étaient terminés.

2. REALISATION

2.1 Présentation de l'interface graphique



Interface de l'application

1 : Contrôle de la position sur la carte. De haut en bas : l'orientation selon les points cardinaux, le positionnement géographique et le zoom.

2 : Une trace représentée avec un bateau représenté par la croix.

3 : La barre de temps. Elle permet de déplacer le bateau sur la trace en fonction du temps. Le curseur peut être cliqué et déplacé. Les boutons de lecture permettent le déplacement en temps réel du bateau. Une seconde dans la réalité est égale à une seconde sur le logiciel. Un bouton avance rapide peut être rajouté.

4 : La gestion des traces. Grâce au formulaire en dessous, il est possible d'ajouter des fichiers GPX au logiciel. Ceux-ci sont listés dans le cadre. Si la liste est trop longue, une barre de défilement verticale apparaît. En cliquant, sur « Supprimer », une confirmation est demandée avant que la suppression du fichier soit définitive.

5 : Le menu sous forme d'icônes. 7 icônes sont disponibles : Orientation vers le nord, Orientation dans le sens des balises, Orientation dans le sens du vent moyen, Orientation dans le sens du vent réel, Rotation d'un demi-tour et deux icônes de positionnement de balises.

2.2 Présentation des fonctionnalités

Toutes les fonctionnalités sauf le contrôle de position sur la carte ont été codées au cours du projet.

2.2.1 Sélection de la trace

A chaque chargement de la page, l'application Google Earth est réinitialisée. Le temps de chargement est donc assez lent et dépend énormément de la connexion internet. Ce soucis intervient au niveau de l'ajout ou la suppression des traces. En effet, celles-ci sont stockées sur le serveur et il est donc nécessaire d'utiliser un formulaire PHP pour effectuer ces actions. Afin d'éviter les rechargements, j'ai décidé de me concentrer sur la possibilité d'ajouter et de retirer des traces dynamiquement. Pour cela, j'ai utilisé l'AJAX³ qui est largement facilité par la présence de la librairie Mootools.

Malheureusement, les formulaires de fichiers ne sont pas gérés en AJAX. De ce fait, lors d'un ajout, une popup miniature s'ouvre, upload le fichier GPX dans le répertoire prévu à cet effet sur le serveur et se referme immédiatement.

La suppression et l'affichage du listing sont gérés sans soucis.

Après chaque ajout ou suppression, la liste est rafraîchie pour être à jour. Cependant, j'ai décidé de rajouter le bouton « Recharger la liste » au cas où cette opération se passerait mal.



Bloc de gestion des traces

Une fois que l'utilisateur clique sur une trace, il faut en même temps la représenter sur la carte et pouvoir l'identifier. Un code couleur est donc associé aux deux pour faire ce lien.

Le code Javascript est ensuite activé pour lire le fichier sélectionné. Dans un premier temps, la trace GPX est décodée afin de dégager la liste des points successifs par lesquels passe le bateau. Les temps de passage sont eux aussi répertoriés. Grâce à ces données, l'API Google Earth permet de tracer la courbe et de placer le bateau sur la coordonnée de départ.

A noter que si une ou plusieurs traces sont déjà affichées, les temps de passage sont comparés et s'ils ne se chevauchent pas, la trace n'est pas acceptée. Les temps de début et fin de la section lecture sont calculés de la façon suivante :

- Temps de départ = temps de départ le plus élevé de toutes les traces affichées.
- Temps de fin = temps de départ le plus bas de toutes les traces affichées.

2.2.2 Lecture de la trace



Boutons de lecture de trace

³ AJAX : Asynchronous Javascript permet d'effectuer des requêtes PHP en background de la page et donc de ne pas recharger celle-ci.

Lorsqu'au moins une trace est affichée, l'utilisateur peut influencer sur la position des bateaux sur la trace en fonction du temps. Pour cela, plusieurs choix se présentent à lui.

- La barre de temps permet de positionner le bateau à un point précis. La position du curseur détermine l'avancement dans le temps.
- Les boutons de lecture, pause et stop permettent de lancer le film du déplacement du bateau. La durée de ce film est égale au temps séparant les deux temps de référence début et fin. Il n'est pas possible d'exporter cette vidéo pour l'instant mais il existe des logiciels très simples d'utilisation le permettant. HyperCam2⁴ dispose de cette fonctionnalité et peut filmer une zone de l'écran pour l'exporter en format AVI.

2.2.3 Menu d'orientation

Contrairement à Google Map, Google Earth dispose dans son API d'une fonction permettant l'orientation de la carte. Cette orientation étant une priorité dans la volonté pédagogique, c'est à cause de cela que j'ai décidé de migrer de Google Map assez simple d'utilisation vers Google Earth beaucoup plus difficile.

Grace à la fonction d'orientation, je n'avais donc plus qu'à déterminer les angles de rotation par rapport au Nord (0°) pour créer les différents boutons.

Le menu d'orientation comporte 5 icônes : (De gauche à droite)

- Orienter vers le nord : Oriente la carte vers le nord, l'angle est de 0° .
- Orienter dans le sens des balises : Cette fonctionnalité n'est disponible que si les balises de courses sont placées. Ce point sera explicité dans la partie suivante (2.2.4). Grâce aux balises positionnées, le logiciel peut calculer le coefficient directeur de la droite passant par deux points sur la carte. L'angle en radian de cette droite par rapport à l'axe des abscisses est l'arctan du coefficient directeur trouvé. Avec un produit en croix, le logiciel converti ensuite cet angle en degrés puis effectue un calcul pour connaître l'angle par rapport au nord.
- Orienter dans le sens du vent moyen (Cette fonctionnalité n'est pas encore codée) : Le vent moyen est la moyenne des vents réels calculée tout au long du parcours. Les calculs sont ensuite similaires à ceux effectués pour le vent dans le sens des balises. Le principal problème est que la courbe n'est pas lissée et donc que les vents réels ne peuvent pas être calculés. Cette difficulté sera exposée dans la suite du rapport (2.3.4).
- Orienter dans le sens du vent réel (Cette fonctionnalité n'est pas encore codée) : Cette fonctionnalité ne devrait marcher que lors de la lecture des traces. Théoriquement, cette fonctionnalité est inutile puisque l'angle du vent réel est sans cesse changeant et donc la carte ne serait pas stable. Le problème lié à cette fonctionnalité est le même que pour la précédente.
- Faire une rotation de 180° : Tous les angles sont calculés à partir de coefficients directeurs de droite. Or contrairement au vent, les droites n'ont pas de sens. Lors d'une réorientation, j'ai dû faire le choix du sens. Il est donc fort possible dans certains cas de se retrouver diamétralement opposé au résultat voulu. Ce bouton a pour objectif de corriger cette erreur. Toujours dans une optique pédagogique, il est peut-être nécessaire quelques fois de faire cette rotation afin de mieux visualiser des phases du parcours.



Menu d'orientation

⁴ HyperCam2 est disponible gratuitement sur le site 01.net :

http://www.01net.com/telecharger/windows/Multimedia/edition_video/fiches/1476.html

2.2.4 Les autres fonctionnalités

Lors des différentes réunions avec mes tuteurs de stage, nous avons convenu qu'il était indispensable de disposer d'un système de positionnement de balises de courses. Ceci pour calculer l'orientation de la course. Il s'avère qu'avoir la possibilité de positionner d'autres balises « événements » est un atout pédagogique. J'ai donc créé deux nouvelles icônes pour cela :

- Placement de balise de course : Permet de placer sur la carte deux balises de courses. Il est impossible d'en placer une troisième par la suite.
- Placement d'une balise « événement » : Permet de placer et nommer une balise sur la carte. Leur nombre est infini.

Le placement des balises est assez aisé à coder puisque Google Earth intègre dans son API des marqueurs (placemark). Il est à noter que chacune des balises peut être retirée par la suite en cliquant dessus. Si une balise de course est retirée, il sera alors possible d'en placer une nouvelle et le bouton orientation dans le sens des balises est désactivé.



Menu des balises

Afin de m'aider dans le codage du projet, j'ai créé une console d'affichage. Cette console s'affiche sur la droite du site et permet l'affichage de nombreuses informations comme les erreurs rencontrées, le détail de certains calculs ou encore les actions réalisées par l'utilisateur. Dans la version finale du projet, cette console n'est plus utile. Elle reste cependant accessible en rajoutant le mot clé « debug » en variable GET⁵ dans l'adresse internet du projet.

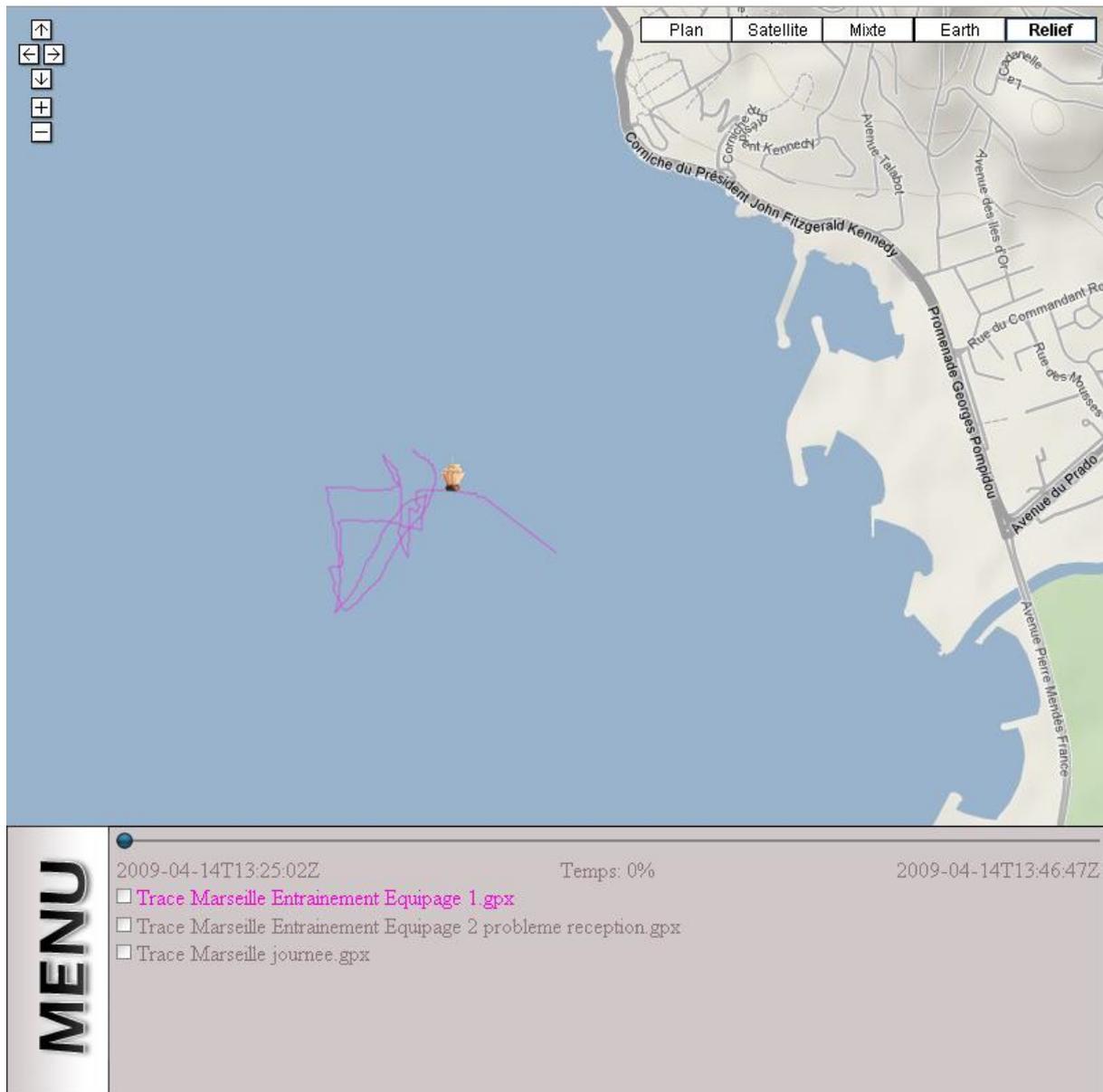
2.3 Difficultés rencontrées et solutions apportées

2.3.1 La rotation sous Google Map

Lorsque le projet m'a été proposé, je ne connaissais pas les APIs Google Map et Google Earth. A l'écoute des conseils de mes tuteurs et après quelques recherches sur internet, j'ai pensé que Google Map correspondait le mieux aux attentes. J'ai donc commencé à coder le projet avec cette application.

C'est alors que M. Cédric Chateau m'a informé de l'importance des réorientations afin d'aider l'enseignement auprès des élèves. Il s'avère que Google Map ne propose pas ce genre de fonctionnalités et l'outil ne correspondait alors plus. J'ai donc abandonné cette version pour proposer une nouvelle version avec comme support Google Earth.

⁵ Dans ce cas, si l'adresse du projet est du style : <http://leprojet.fr>, alors l'adresse de debug sera <http://leprojet.fr/?debug>.



Version 1 du projet avec Google Map

2.3.2 L'API Google Earth

L'API Google Earth est très récente et n'est disponible que depuis quelques mois pour les internautes. De ce fait, elle est encore très changeante et quelques fois elle n'est pas simple d'utilisation. Par exemple, pour changer l'icône d'un placemark, il faut écrire une dizaine de lignes de code. Heureusement que les fonctions que j'ai utilisées pour le projet sont assez basiques et ne subissent pas de modifications.

Le souci de compréhension de l'API est amplifié par la rareté relative des exemples et tutoriaux sur internet. Pour réaliser le projet, j'ai dû étudier le code source des démonstrations disponibles sur le site de Google.

Malgré tout cela, j'ai toujours réussi à obtenir le résultat que je souhaitais. L'API va continuer à évoluer et il sera peut-être nécessaire de retoucher le code dans quelques temps pour le simplifier ou pour le mettre à jour.

2.3.3 Ordonnancement des calculs

Deux problèmes d'ordonnancement se sont présentés à moi :

- L'application Google Earth est assez lourde au chargement et dépend de la qualité de la connexion internet. De ce fait, il fallait attendre la fin du chargement de la page pour débiter les calculs.

- Certains calculs ne doivent démarrer que lorsque les calculs précédents sont terminés. Or le langage Javascript ne gère pas les opérations de cette manière et le logiciel rencontrait alors quelques bugs. Par exemple, le déplacement des bateaux ne doit pas commencer avant que la trace soit finie d'être affichée. Ceci était possible avant et provoquait un figement total de la page.

Afin de résoudre ces problèmes, j'ai décidé d'utiliser une bibliothèque Javascript que je connais très bien puisque je l'ai déjà utilisée à de nombreuses reprises : Mootools. Cette librairie est simple à prendre en main et a permis de régler tous les problèmes précédemment cités. De plus, grâce à elle, j'ai pu ajouter d'autres fonctionnalités comme l'ajout et le retrait des fichiers GPX sans recharger la page. Pour terminer, il est à noter que le curseur de temps est géré par Mootools.

2.3.4 Lissage de la courbe

La direction des bateaux en mer dépend du vent mais aussi d'autres paramètres comme les vagues, la proximité des côtes etc. De ce fait, le coefficient directeur entre deux points consécutifs d'une même trace peut varier énormément et il n'est pas fiable pour déterminer le vent réel. Afin de lisser la courbe et donc de rajouter des points cohérents, il faut utiliser la technique des splines cubiques. Cette technique consiste à construire des nouveaux points à intervalle régulier pour rendre la courbe dérivable. Une fois cette opération effectuée, il est ensuite possible de calculer les bissectrices de la courbe et donc le temps réel aux points voulus. J'ai donc construit le code permettant de calculer les points intermédiaires. Trois problèmes se posent alors :

- La difficulté d'interprétation des valeurs calculées pour en conclure une direction du vent.
- Le lissage n'est pas idéal et présente encore des valeurs incohérentes pour le calcul.
- Le temps de calcul est assez long.

3. PERSPECTIVES D'EVOLUTION

A la vue de mon projet, de nombreux points restent à travailler afin d'obtenir un résultat plus complet.

- Les traces en format GPX ne sont pas reconnues directement par Google Earth. Pour l'instant je suis obligé de feinter en continuant d'utiliser les fonctions incluses dans l'API Google Map. Il faut donc surveiller les prochaines évolutions de l'API Google Earth afin de voir si des possibilités sont envisageables de ce côté.

- Dans certains cas, il serait intéressant de pouvoir afficher qu'une partie des traces afin de favoriser les calculs et d'éliminer les données inutiles comme le temps d'attente avant la course ou le retour au port. Pour cela, le logiciel doit effacer la trace déjà affichée, sélectionner l'intervalle de points intéressants et les afficher de nouveau. Cette manipulation pourrait être simplifiée avec la prochaine idée exposée.

- Le calcul de la spline cubique est assez long et dépend énormément du nombre de points intermédiaires que l'on souhaite prendre. Pour résoudre ce problème, il faudrait effectuer ce calcul en amont lors de l'ajout d'une trace sur le site et stocker les données pour ne plus avoir à les calculer. Ceci pourrait apporter une simplicité supplémentaire au niveau de la lecture du fichier de trace par le logiciel et améliorerai grandement le temps d'affichage puisque le fichier de trace ne serait plus parsé à chaque fois. Je pense que cette évolution est majeure et doit figurer comme une priorité. Elle peut être réalisée grâce à un applet⁶ en Java.

- Malgré le calcul des splines cubiques, la détermination des vents est encore impossible du fait des trop nombreux changements de cap du voiliste pendant la course. Je n'ai pas trouvé l'idée pour résoudre ce problème.

- En partant du principe que le logiciel connaît l'échelle de la carte, la distance entre deux points et les temps de passage, il peut facilement déterminer la vitesse de celui-ci à un instant T. Cette information peut se révéler intéressante pour les analyses des traces.

⁶ Un applet : morceau de code en Java qui peut être exécuté dans un navigateur internet.

CONCLUSION

Ce projet a été une véritable opportunité pour moi d'appliquer mes connaissances sur un domaine pratique non scolaire. J'ai dû faire face à une demande assez pointue, analyser le contexte pour en ressortir les meilleures solutions possibles. Mes tuteurs m'ont laissé la chance d'être libre dans mes choix mais aussi face à mes problèmes. Ceci m'a amené dans une recherche de solutions pratiques.

J'espère pouvoir apporter à la ligue de voile un outil pédagogique qui correspond à leur volonté. Certains ajustements seront nécessaires suite à l'utilisation du logiciel. En effet, il se peut qu'au cours de son utilisation, certaines fonctionnalités fassent défaut et il faudra alors les ajouter.

Je tiens finalement à remercier Mme Corson et Mrs Chateau, Dutot, Pigné pour ce projet.