

Projet développement mobile

Projet développement mobile.....	1
1. Présentation.....	2
1.1 Équipe.....	2
1.2 Contexte.....	2
1.3 Objectifs.....	2
2. Expression des besoins.....	2
2.1 IDE.....	2
2.2 Versioning.....	2
2. Déroulement du projet.....	3
2.1 Organisation.....	3
2.2 Documentation.....	3
3. Présentation des vues.....	3
3.1 Vue 1.....	3
3.2 Vue 2.....	3
3.3 Vue 3.....	3
4. Entrées/Sorties de nos outils.....	4
4.1 Entrées.....	4
4.2 Sorties.....	4
5. Diagrammes.....	4
5.1 Diagrammes de Gantt.....	4
5.2 Diagrammes de classes.....	6
5.3. Diagrammes de fonctionnalités.....	7
5.4. Spécifications fonctionnelles détaillées.....	7
6. Fonctionnalités implémentées.....	8
7. Problèmes rencontrés.....	9

Historique

Version	Date	Auteur	Commentaires
1.0	19/03/2019	TELLIER Noémie DULERY Quentin	Présentation projet
2.0	29/03/2019	TELLIER Noémie	Amélioration du cahier des charges

1. Présentation

1.1 Équipe

◆	TELLIER	Noémie	Chef de projet / Développeur
◆	NORTH	Justin	Lead Developer / Développeur
◆	DULERY	Quentin	Scrum Master / Développeur
◆	TRICHARD	Dimitri	Développeur
◆	CLIGNY	Benjamin	Développeur
◆	VOYER	Florent	Développeur

1.2 Contexte

Le projet concerne un drone marin de surface qui a actuellement la possibilité d'être contrôlé manuellement grâce à une télécommande et à une caméra disposée dessus.

1.3 Objectifs

Ce projet a pour but de réaliser une application mobile permettant à l'utilisateur de simuler une trajectoire, visualiser la position du drone sur une carte, créer une trajectoire grâce au placement de plusieurs waypoints et contrôler le drone marin grâce à l'accéléromètre du téléphone.

Contrairement à une télécommande, il n'y a pas obligation de contact visuel avec le drone car nous avons accès à sa position GPS sur l'application.

L'application mobile développée devra être disponible sur iOS et Android.

2. Expression des besoins

2.1 IDE

Pour le développement de l'application, nous allons utiliser :

- Xcode pour iOS,
- Android Studio et IntelliJ IDEA pour Android.

2.2 Versioning

Afin de travailler tous ensemble sur le même dossier nous allons utiliser GitHub.

https://github.com/justinnth/Projet_Smartphone

Avec une bonne gestion de projet, cet outil est très pratique et facile à utiliser.

Pour interagir au sein de l'équipe nous utilisons Discord. L'avantage est de pouvoir discuter et de créer des groupes afin de séparer les différents sujets à traiter.

2. Déroulement du projet

2.1 Organisation

Le projet va utiliser la méthode agile Scrum. Cette méthode permet de mieux comprendre chaque aspect du projet, de mieux s'orienter et de voir l'avancée du projet.

Cette méthode permet d'avoir une bonne communication dans l'équipe grâce à la réalisation de sprint où chacun pourra présenter son travail et interagir avec le reste de l'équipe sur certaines difficultés ou différents changements dans le projet.

Afin de visualiser et de répartir les tâches, nous allons utiliser Trello

- <https://trello.com/b/LZtZdzYf>

2.2 Documentation

Pour ce projet, nous avons réalisé différents documents afin de faciliter la réalisation. Les livrables disponibles pour ce projet :

- ➔ Cahier des charges
 - Présenter l'équipe, les objectifs et la réalisation du projet
- ➔ Diagramme de fonctionnalités
 - Présenter le fonctionnement de l'application
 - Présenter les possibilités de l'utilisateur aux travers de l'application
- ➔ Diagramme de classe
 - Présenter le schéma technique du projet
- ➔ Spécification Fonctionnelle Détaillée
 - Comprendre les maquettes des différentes vues de l'application
 - Expliquer chaque maquette et les actions possibles sur chacune d'elle

3. Présentation des vues

3.1 Vue 1

- Représentation de la trajectoire à partir des données NMEA (Simulator)+ Infos Vitesse
 - Tracé du parcours du drone
 - Affichage de la vitesse du drone durant le trajet

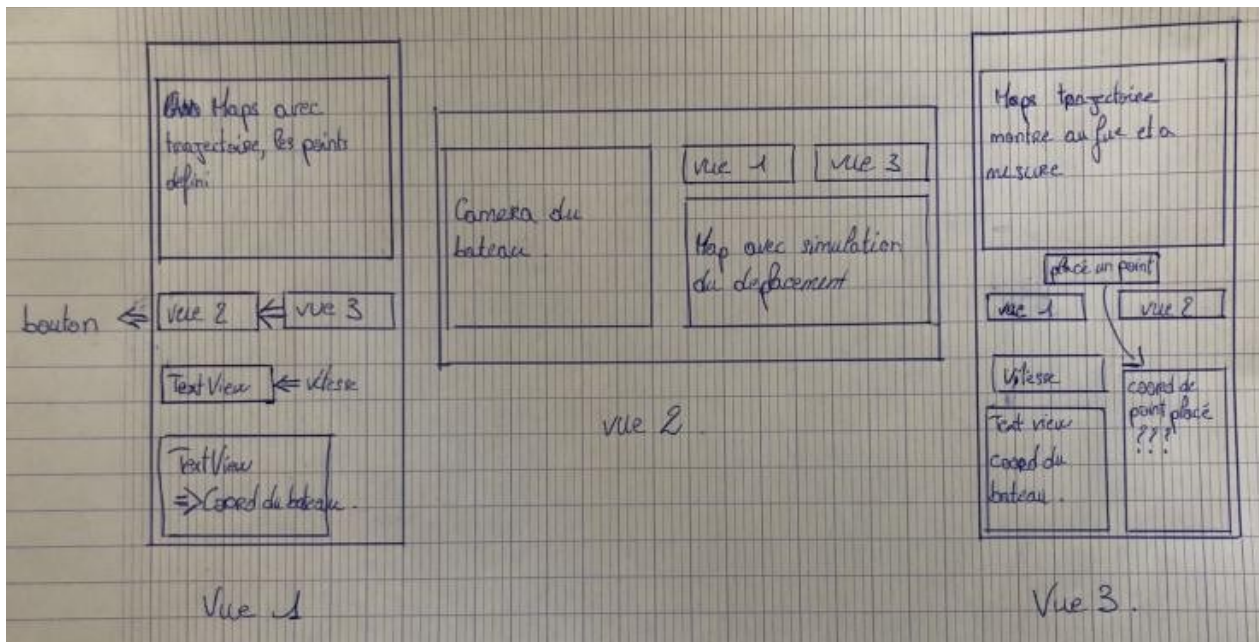
3.2 Vue 2

- Simulation du déplacement d'un drone virtuel à partir des données de l'accéléromètre
 - Orientation du téléphone / Répercussions sur la trajectoire du drone
 - Bouton HOME : le drone rentre au point de départ
 - Bouton URGENCE : arrêt immédiate du drone / Vitesse = 0

3.3 Vue 3

- Définition d'une trajectoire à partir de waypoints
 - Positionner des waypoints
- Définition de la vitesse sur chaque waypoints

- Choix de la vitesse entre chaque waypoint
- Transmission des données trajectoires (en fichier json ou trames NMEA (*Sleuth*))
 - Récupération de la trajectoire



4. Entrées/Sorties de nos outils

4.1 Entrées

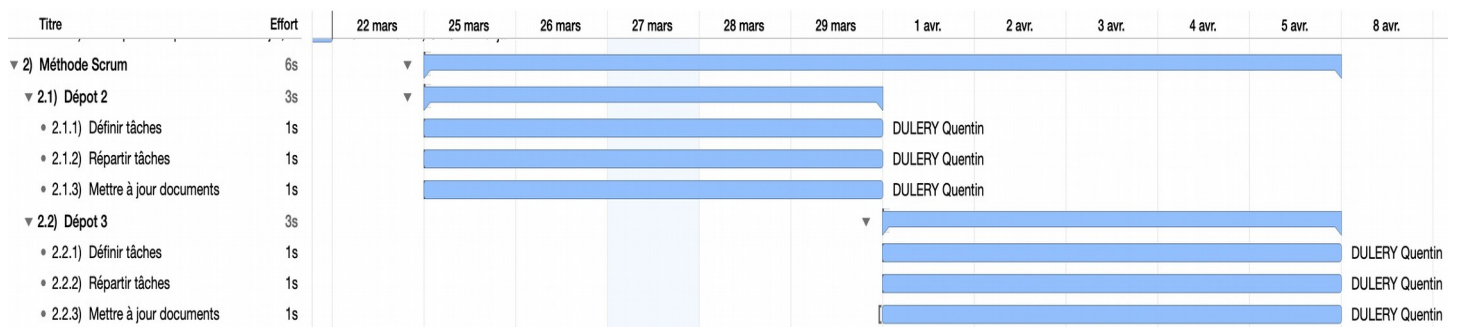
- trames NMEA
- ajout waypoints (trajectoire) par l'utilisateur

4.2 Sorties

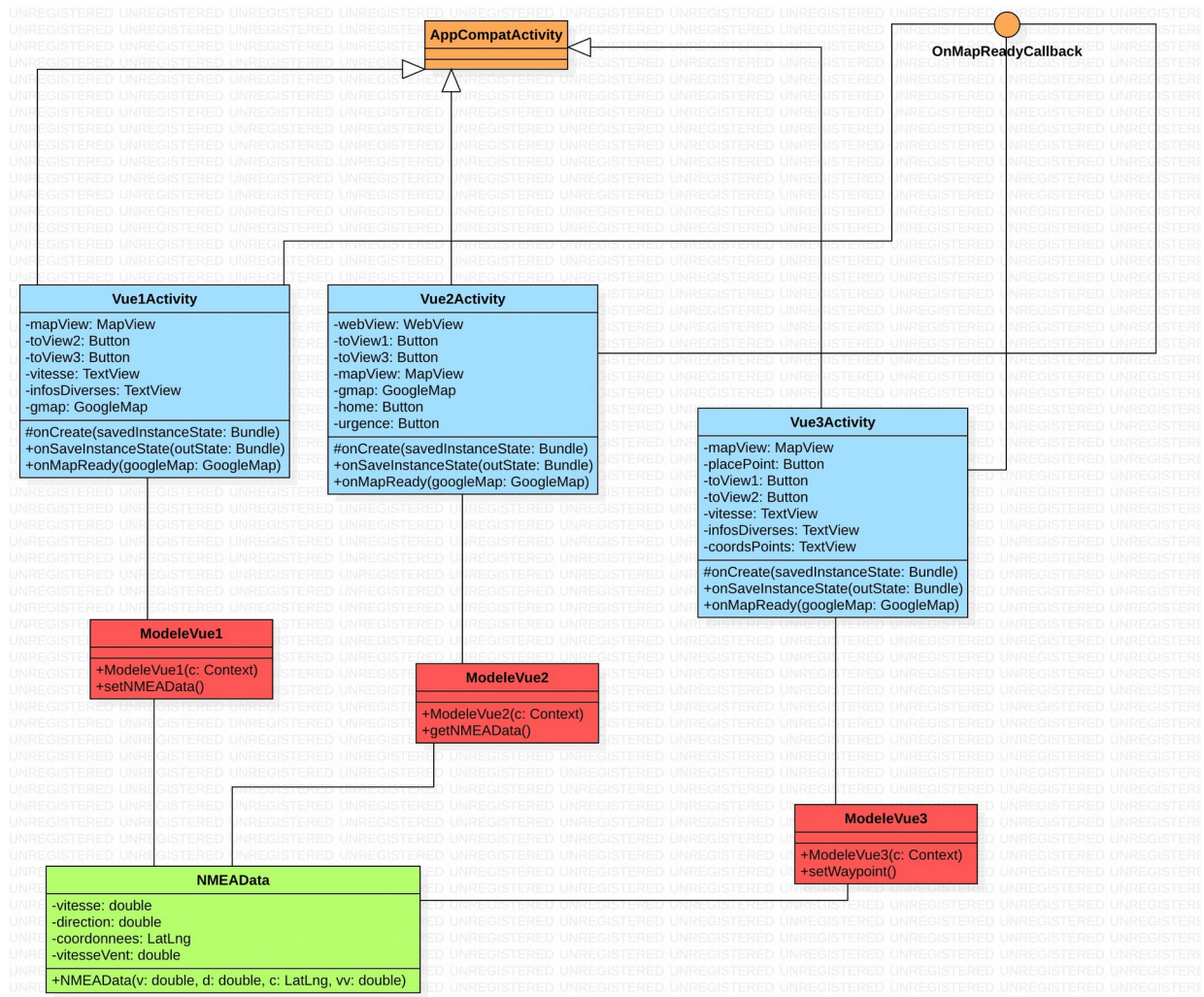
- fichier json
- envoi de la trajectoire au drone

5. Diagrammes

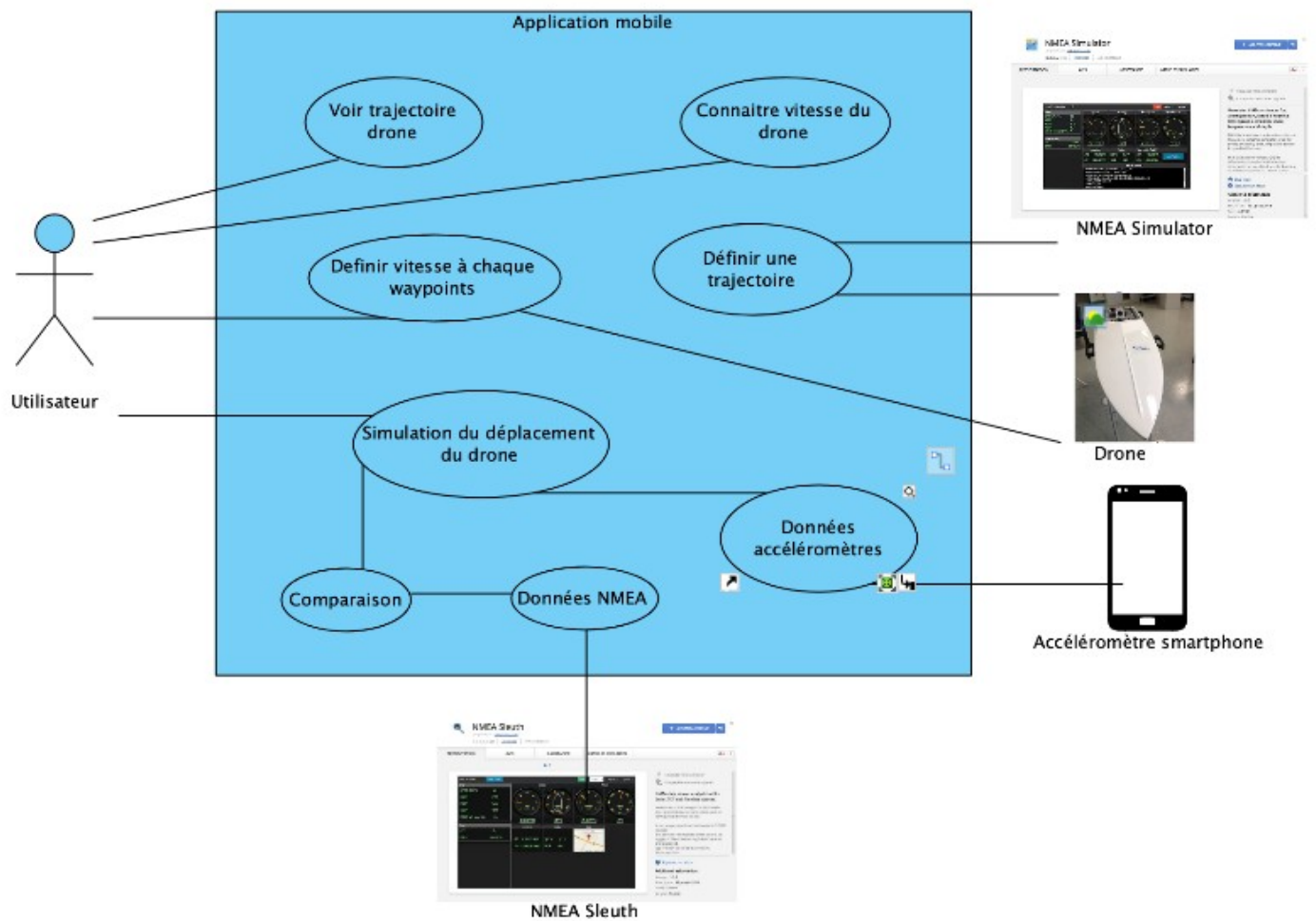
5.1 Diagrammes de Gantt



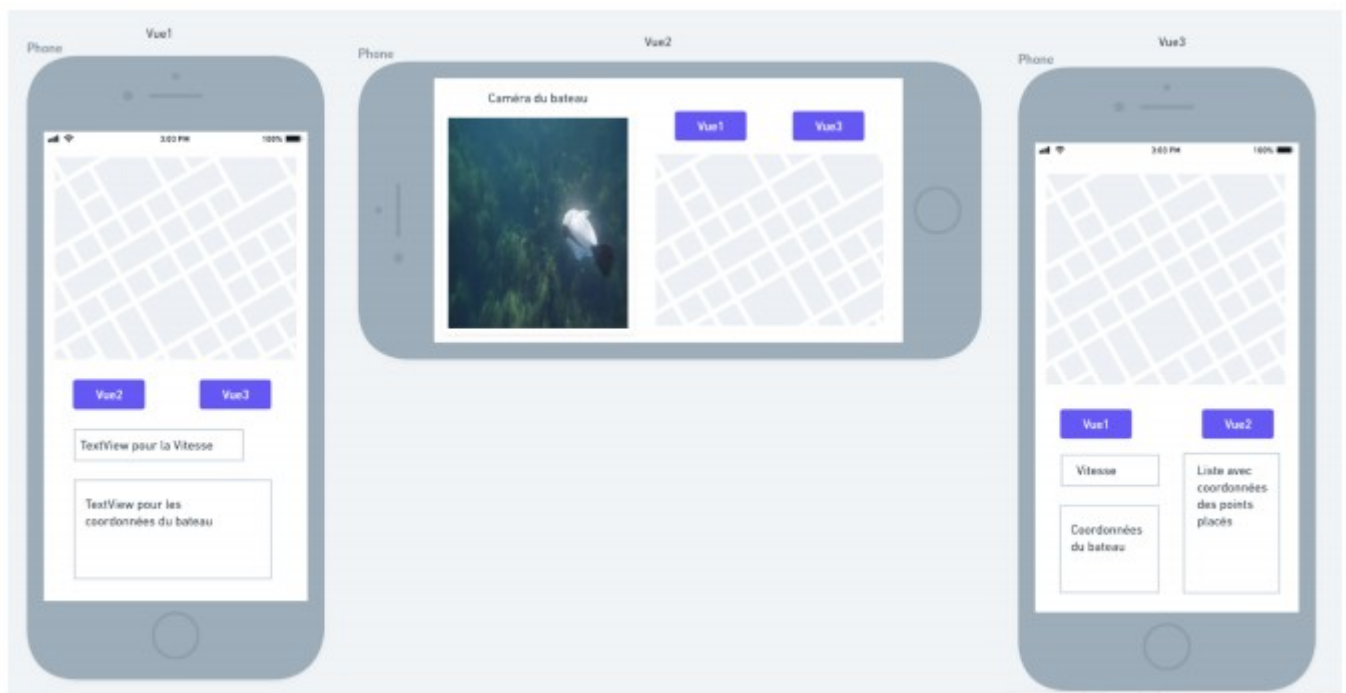
5.2 Diagrammes de classes



5.3. Diagrammes de fonctionnalités



5.4. Spécifications fonctionnelles détaillées



Les trois vues comportent deux boutons permettant l'accès aux autres vues, ainsi que deux TextView : un pour connaître la vitesse du drone et un pour connaître ses coordonnées.

Vue 1 : Simulation

- Elle permet de simuler un trajet du drone grâce à des trames NMEA.
- Cette vue comporte également une map qui représente la trajectoire suivie par le drone.

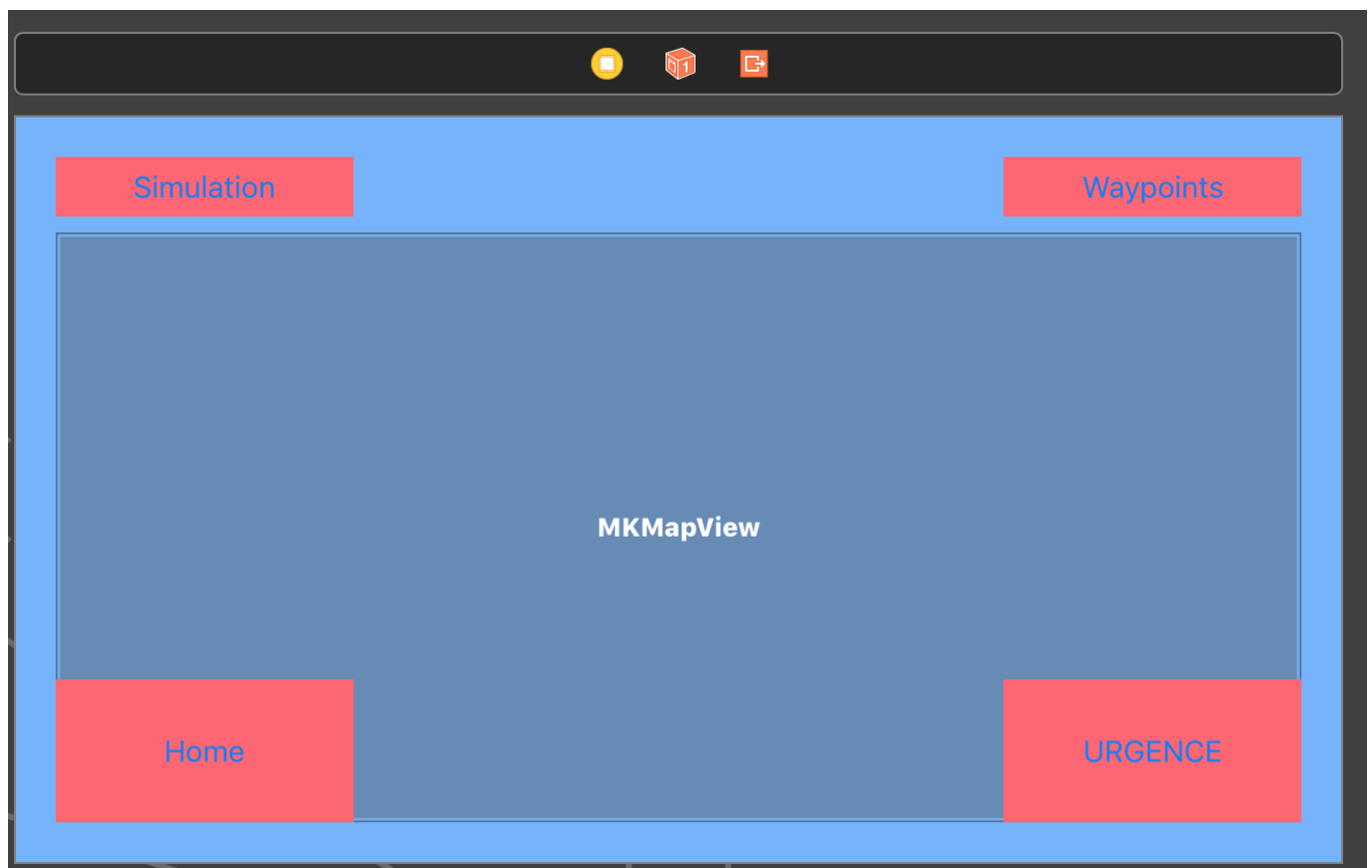
Vue 2 : Pilotage

- Elle permet de piloter le drone via les accéléromètres du téléphone.
- On y retrouve la caméra du drone ainsi qu'une map permettant de suivre sa trajectoire.

Vue 3 : Waypoints

- Elle permet à l'utilisateur de définir la trajectoire du drone à partir de waypoints qu'il placera sur la map.
- Elle comporte également un autre TextView listant les points placés et leurs coordonnées.

Vue Pilotage – Version iOS



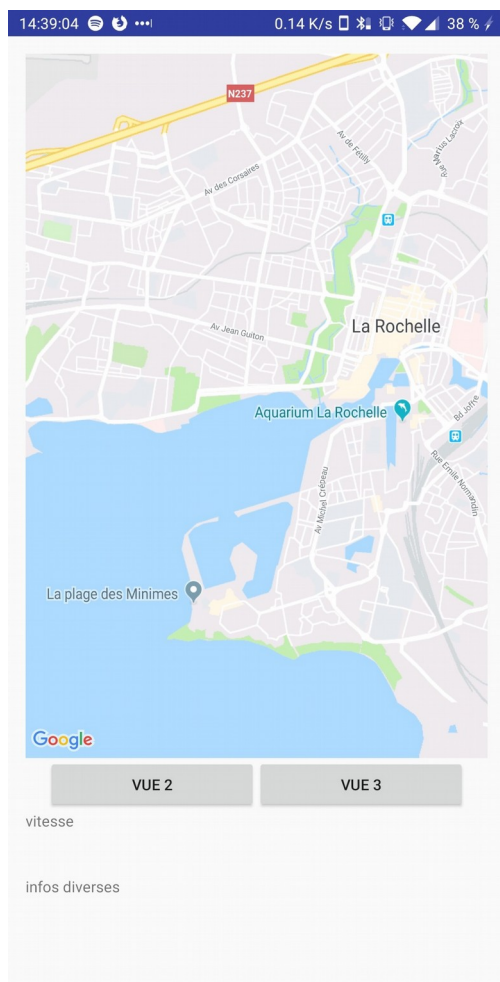
6. Fonctionnalités implémentées

- **Android**
 - Dans les 3 vues : Map + 2 boutons d'accès aux autres vues
 - Boutons HOME et URGENCE : lors de l'appui, affichent des Toasts
 - Positionner des waypoints, les afficher et tracer le trajet
- **iOs**
 - Dans les 3 vues : Map + 2 boutons d'accès aux autres vues
 - Boutons HOME et URGENCE : affichage seulement

7. Problèmes rencontrés

- **Android**
 - **Socket :**
 - Nous n'arrivons pas à se connecter au simulateur afin de récupérer les trames NMEA.
- **iOs**
 - **Socket :**
 - Nous n'arrivons pas à se connecter au simulateur afin de récupérer les trames NMEA en swift donc nous pensons essayer en objective-C.

Vue Simulation - Android



Vue Waypoints - Android



Vue Pilotage - Android

