

# Akenasia

## Table of Contents

Synthèse .....	1
Projet .....	1
Membres .....	1
Présentation .....	1
Aspects techniques .....	2
Modélisation .....	3
Features .....	4
Accueil .....	4
Liste de lieux .....	4
Chronomètre .....	5
Coups limités .....	6
Personnaliser sa liste .....	7
Map .....	7
Historique des positions .....	8

## Synthèse

### Projet

Jeu mobile centré sur l'utilisation des coordonnées GPS de son appareil. Le nom de notre jeu est Akenasia.

### Membres

- Sefkan BEYAZ
- Alexandre RIPERT
- Cathy SIWATALA
- Paul GORLICKI

### Présentation

Akenasia est un jeu qui utilise l'emplacement GPS de l'utilisateur. Au début de chaque partie, le joueur choisit une destination vers laquelle il doit se rendre. Il dispose d'un bouton qui lui permet d'actualiser sa position et de vérifier s'il a bien atteint son objectif. S'il est suffisamment proche de son objectif, la partie est terminée. Chaque appui augmente le nombre de tentatives et il y a un nombre de tentatives limitées alors attention !

Notre but est de pouvoir livrer un produit fini fonctionnel capable d'afficher le joueur sur une carte. Sa position sera affichée et actualisée lorsqu'il appuiera sur un bouton. Il y aura la possibilité de choisir son lieu parmi une liste ou alors d'ajouter soi même un lieu. Il est également prévu d'avoir différents modes de jeu comme : contre la montre ou encore avec des tentatives limitées.

Notre équipe est assez inexpérimentée dans le domaine des applications mobiles mais le défi est intéressant et nous avons la volonté de réaliser un jeu fun, agréable à jouer et intuitif.

Nous présentons ce jeu comme un jeu dans le style de Pokemon GO mais en réalité cela se traduit juste par l'utilisation de l'emplacement GPS, il n'y a pas de créatures à attraper ou d'arènes à conquérir. Nous encourageons les joueurs à découvrir des lieux et à se balader en ville tout en essayant de battre leur record. Il est également possible d'affronter ses amis en regardant qui a obtenu le meilleur score ou le temps le plus rapide. Pour cela il faut passer par le service Google de géolocalisation et autoriser l'application à accéder à son emplacement.

Notre jeu s'adresse à tout type de joueur. Lorsqu'on lance le jeu, on a tout d'abord le choix entre différents lieux enregistrés dans une liste. Une fois un lieu choisit, on doit sélectionner le mode de jeu : chrono ou tentatives. Il faut ensuite se déplacer jusqu'à son objectif et appuyer sur le bouton pour déclencher l'actualisation de sa position. Une fois la partie terminée, on peut choisir un autre lieu et le schéma se répète. Il pourrait aussi y avoir la possibilité d'ajouter soi-même un lieu et de gérer sa liste de lieux.

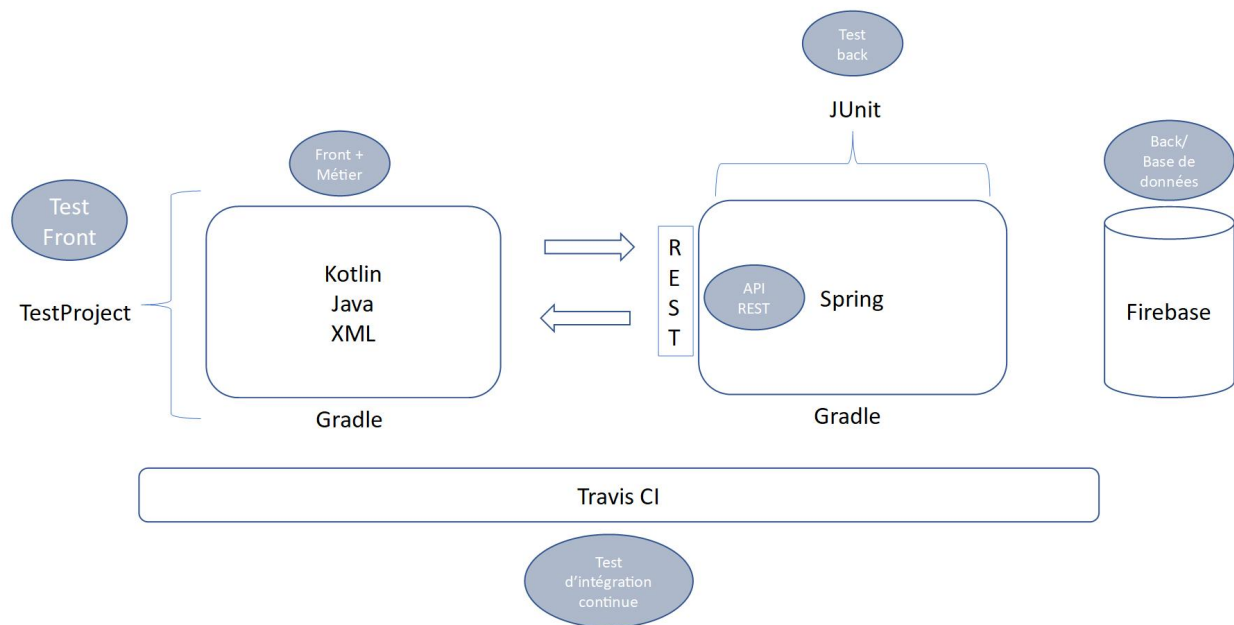
Pour faire connaître le jeu et gagner de nouveaux joueurs nous allons principalement utiliser les réseaux sociaux, en parler à notre entourage et les encourager à diffuser le jeu. Le monde doit connaître l'existence d'Akenasia !

## Aspects techniques

Akenasia est une application mobile développée principalement en Kotlin, Java et XML. C'est un projet buildé avec Gradle. Nous réalisons des tests d'intégration continue avec l'outil Travis CI et des tests d'interface avec l'outil TestProject. Akenasia est reliée à une base de données mobile Firebase, accessible depuis une API REST Spring.

Type d'application : Mobile

Schéma architectural de l'application :



Plateforme technologique :

- Langages utilisés : Kotlin/Java/XML
- Frameworks de test : TravisCI/TestProject
- Frameworks de lien métier/API : AndroidStudio
- Framework de lien métier/persistence :

Plateforme opérationnelle :

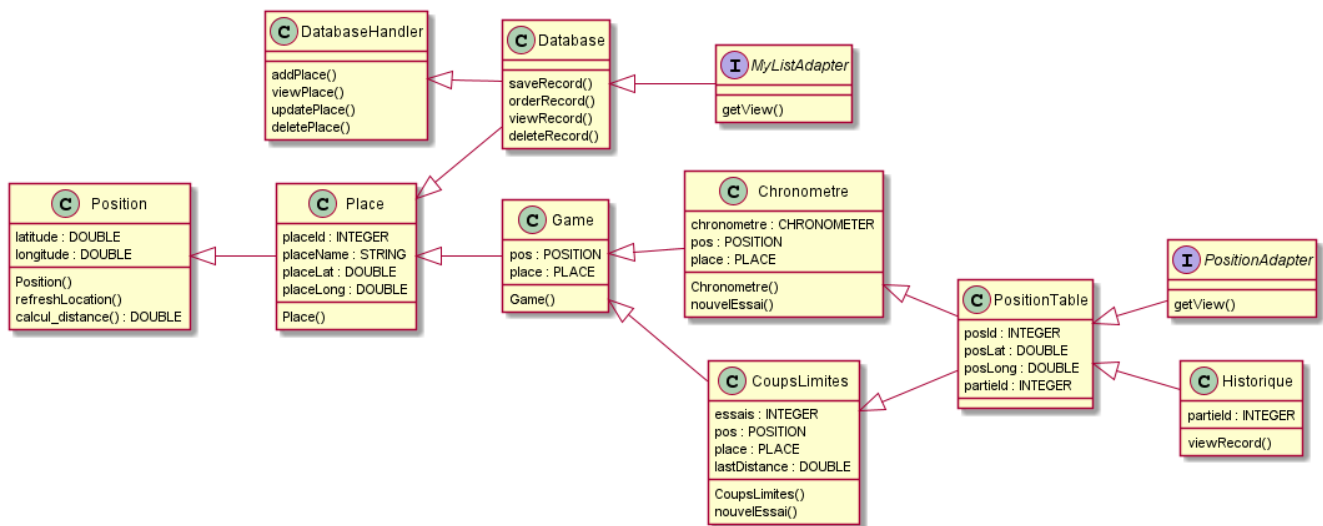
- Gestion de version : GitHub
- Build : Gradle
- Qualité de code : SonarQube/SonarCloud
- CI : TravisCI
- déploiement : TravisCI

Interconnexion des systèmes /API distantes : ?

Acquisition de données, usage intelligent des données : ?

## Modélisation

Modèle UML :



# Features

## Accueil

- Page d'accueil de l'application, il suffit de cliquer sur le bouton "Jouer" afin d'accéder à la liste des lieux.



Figure 1. Menu principal

## Liste de lieux

- Au début de la partie, l'utilisateur choisit dans la liste proposée un lieu qui sera sa destination. Une partie peut se jouer sous deux modes de jeux différents. Le premier mode de jeu est une partie chronomètre, le but est de rejoindre la destination le plus rapidement possible. Le deuxième mode de jeu est une partie avec un nombre de coups limités pour rejoindre le lieux fixé au début de la partie.

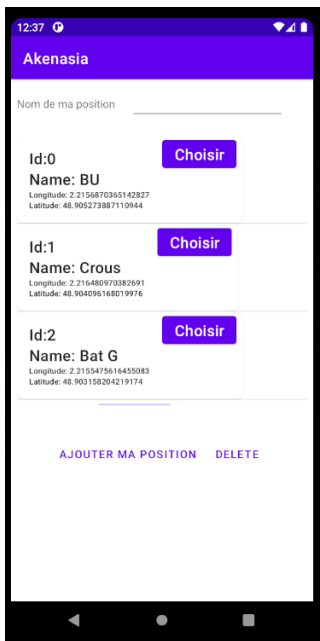


Figure 2. Liste des lieux

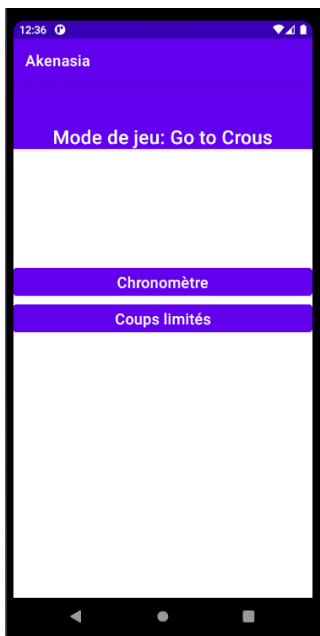


Figure 3. Choix du mode de jeu

## Chronomètre

- Dans un mode de jeu contre la montre, un chronomètre sera activé au début de la partie. Dans ce mode la partie s'arrête soit lorsque le compte à rebours tombe à 0, soit lorsque l'utilisateur arrive près de sa destination. Il pourra ensuite observer combien de temps il lui restait pour se rendre à son objectif.

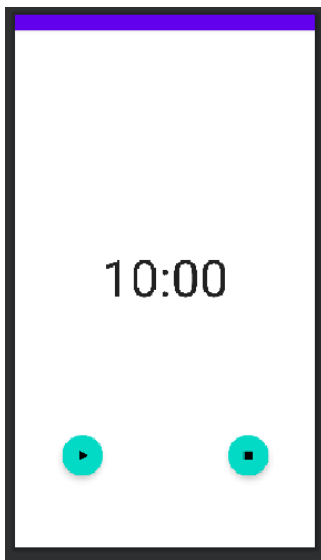


Figure 4. Blueprint du chronomètre



Figure 5. Mode chronomètre

## Coups limités

- Dans un mode de jeux à coups limités, un coup correspond à un rafraichissement de la position du joueur en appuyant sur le bouton "Refresh". Dans ce mode, la partie s'arrête lorsque l'utilisateur est suffisamment proche de la destination choisie. A chaque actualisation, le jeu indique si le joueur se rapproche ou s'éloigne de son objectif [1]. Le nombre de tentatives se met à jour lors de chaque récupération de la position [2] et une fois le compteur à 0, la partie est perdue.

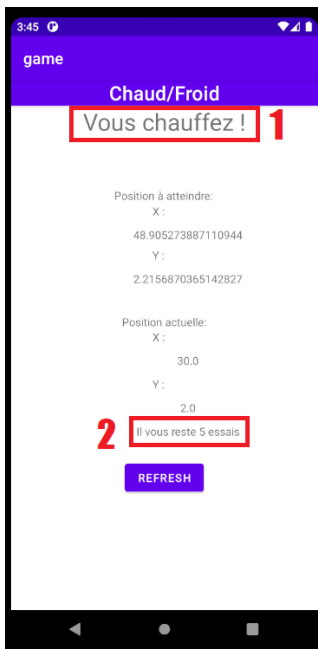


Figure 6. Mode coups limités

## Personnaliser sa liste

- L'utilisateur peut créer son parcours personnalisé. Lorsque l'on arrive sur la page contenant la liste des lieux proposés, on remarque la présence d'une entrée de texte. Pour ajouter un lieu, l'utilisateur doit se déplacer jusqu'au lieu (les coordonnées enregistrées seront celles de la position actuelle du joueur). Il faut ensuite entrer un nom [1] et cliquer sur "ajouter ma position" [2]. Il est possible de supprimer un lieu avec "delete" en entrant l'id du lieu que l'on veut supprimer [3].

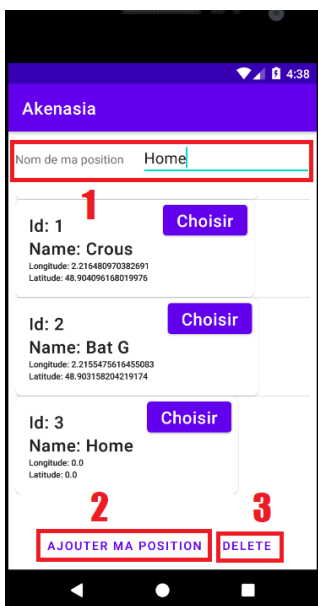


Figure 7. Ajout de lieu

## Map

- Au cours de la partie, le joueur peut voir sa position centrée sur une map. Lorsqu'il rafraichit sa position, l'affichage de la map s'actualise.

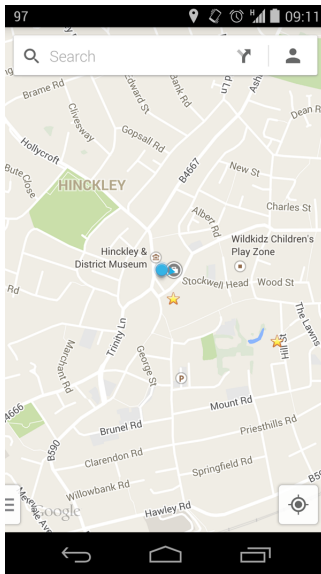


Figure 8. Position de l'utilisateur sur une map

## Historique des positions

- Il est possible d'accéder à l'historique de ses positions lors de ses refresh effectués au cours d'une partie à coups limités.

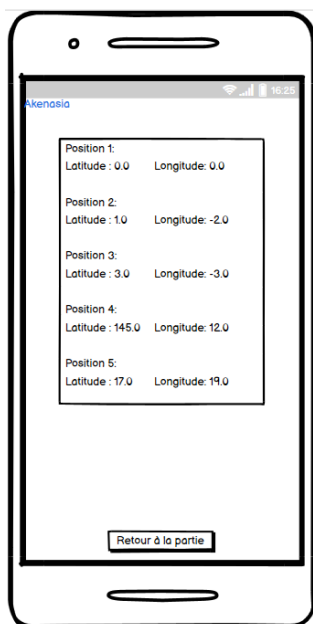


Figure 9. Blueprint de traçage des positions

- A la fin d'une partie, une map s'affiche avec le lieu de départ de l'utilisateur, ainsi que les lieux où il a rafraîchi sa position.





Figure 10. Blueprint des marqueurs sur map