

# Projet informatique individuel 2017/2018

Projet du deuxième semestre

Sujet : Application mobile pour les cyclistes



*www.pexels.com*

Livrable :

**Rendu Final**

Quentin FUMEAU

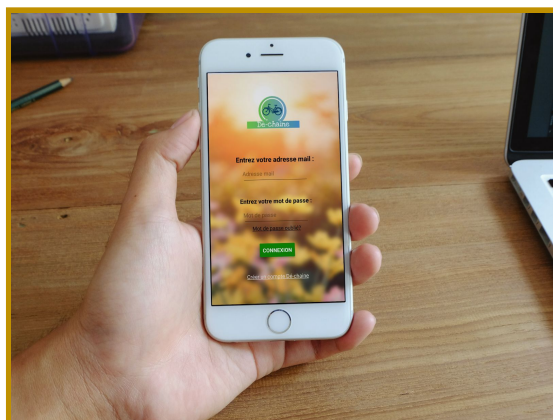
Groupe 3 – 2ème année

# Sommaire

---

<b>Sommaire</b>	<b>2</b>
A. Remise en contexte	3
B. Récapitulation générale	3
<b>Architectures</b>	<b>4</b>
A. Architecture logicielle	4
B. Architecture de l'application	4
<b>Technologie</b>	<b>5</b>
A. React-native	5
B. Firebase	7
C. Les outils utilisés	8
<b>L'application</b>	<b>9</b>
A. Ecrans finaux	9
B. Explications des écrans finaux	11
<b>Tests utilisateur</b>	<b>12</b>
<b>Versions</b>	<b>13</b>
<b>Planification</b>	<b>14</b>
B. Difficultés rencontrées	14
<b>Conclusion</b>	<b>15</b>
Points d'améliorations	15
Difficultés rencontrées	15
Apport du projet	15

Lien vers github pour le code : <https://github.com/qfumeau/yarnApp>



# Introduction

---

## A. Remise en contexte

Ce projet informatique est la continuité d'un projet réalisé lors du premier semestre dans le module d'IHS. Au cours de ce projet j'avais, avec mon groupe, créé les maquettes interactives d'une application dédiée aux cyclistes. J'ai choisi de prolonger ceci au cours du projet informatique individuel car je trouvais que c'était une bonne base pour commencer le développement mobile et que je gagnais du temps en ayant des maquettes déjà réalisées.

L'application que j'ai développée permet à un cycliste de pouvoir localiser son vélo, garder un historique de ses trajets et consulter des statistiques telles que la distance parcourue, les calories dépensées, les économies d'essence réalisées ou les émissions de Co2 évitées grâce à l'usage du vélo à la place de la voiture.

## B. Récapitulation générale

Au fur et à mesure du projet, j'ai dû enlever certaines fonctionnalités de l'application et en rajouter d'autres. Ne pouvant pas avoir une puce GPS, j'ai choisi de géolocaliser le portable de l'utilisateur et non son vélo. Je n'ai donc pas pu avoir l'historique d'un trajet de façon automatique. Pour pallier à cela j'ai créé une page permettant de rentrer à la main les données relatives à un trajet.

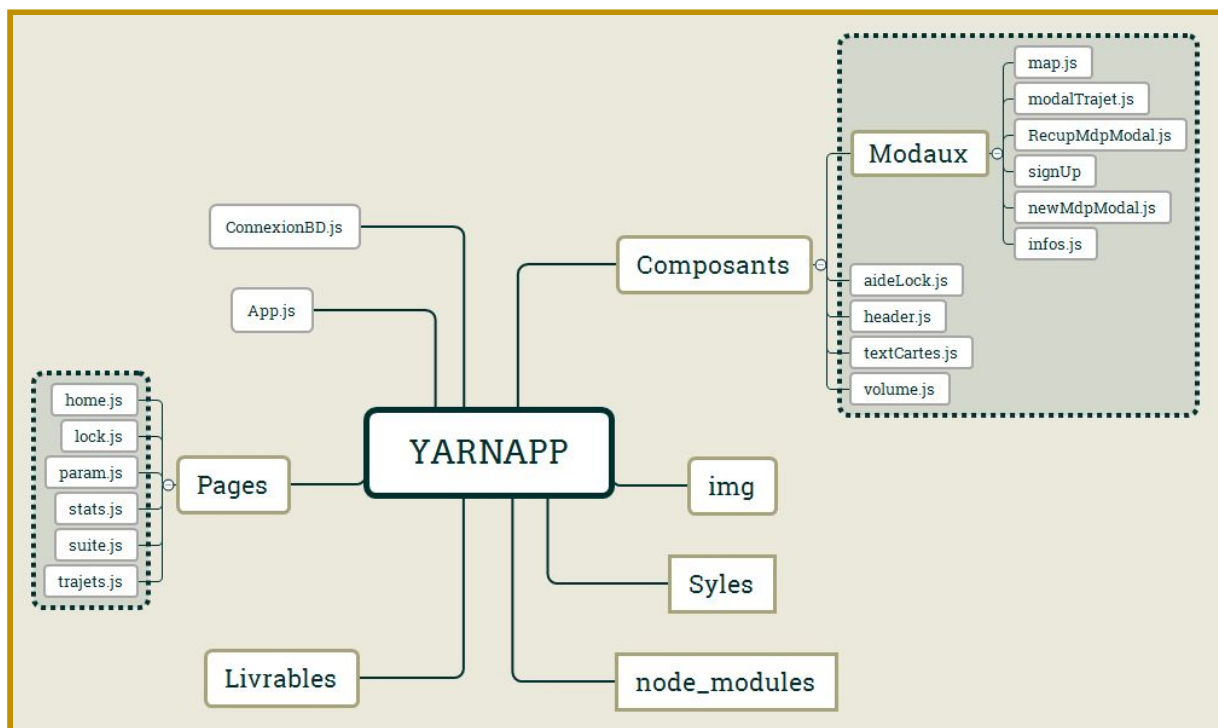
J'ai créé les composants d'activation ou non et de réglage de volume des notifications mais je n'ai pas réussi à interagir avec les paramètres du mobile.

J'ai ajouté une page donnant des informations sur la manière dont les statistiques sont calculées.

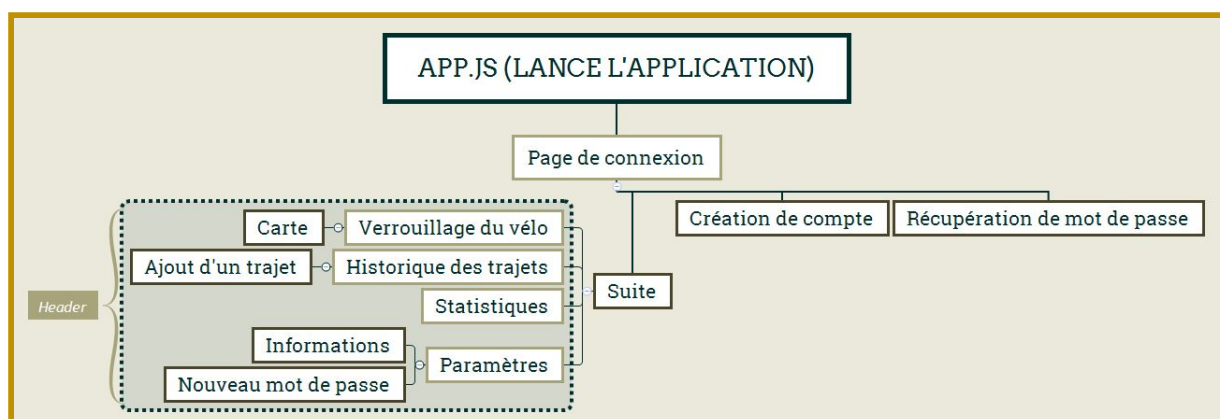
J'ai ajouté la possibilité de supprimer un compte et de modifier le mot de passe d'un compte.

# Architectures

## A. Architecture logicielle



## B. Architecture de l'application



# Technologie

---

## A. React-native

React-Native est un framework javascript créé par Facebook. Il dérive du React, une bibliothèque destinée à la création d'application web monopage créé par Facebook également et souvent utilisée avec AngularJS.

React-Native est un framework basé sur le principe de créer une application utilisant javascript pour les fonctions et traitements algorithmique, React pour les balises et plus globalement la vue. Une fois une page créée, celle-ci est compilée et le bundle javascript est construit soit pour un IOS soit pour Android. Les applications créées sont donc des applications multi-plateformes et offrent l'avantage de ne nécessiter qu'une équipe de développement ; les différences entre IOS et Android étant pour la plupart assez vite adaptées.

React-Native fonctionne essentiellement par Composants, c'est-à-dire une classe comportant un traitement et une vue. Certains composants sont disponibles sur GitHub suivant ce que l'on recherche. Mais il faut parfois les créer soit-même si on veut utiliser un composant spécial ou simplement si on ne trouve pas un composant répondant parfaitement à notre besoin.

Structure d'un composant :

```
export default class MonComposant extends React.Component{
  //fonctions relatives au composant
  render(){
    return(
      //Vue du composant
      <View style={styles.maVue}>
        <Text>Hello World</Text>
      </View>
    )
  }

  const styles = StyleSheet.create({
    //styles appliqués à une vue
    maVue:{
      //styles
    }
  })
}
```

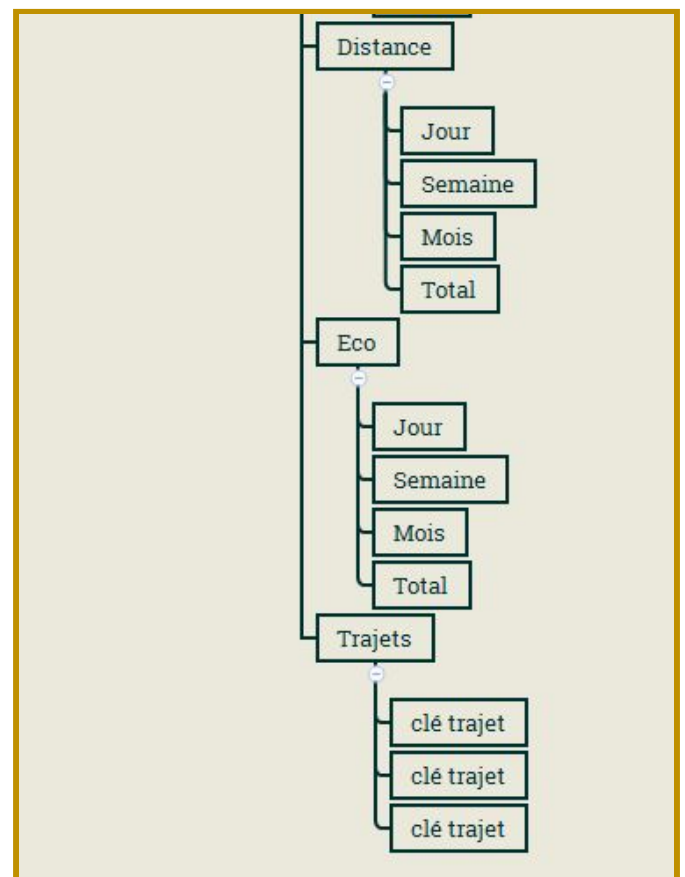
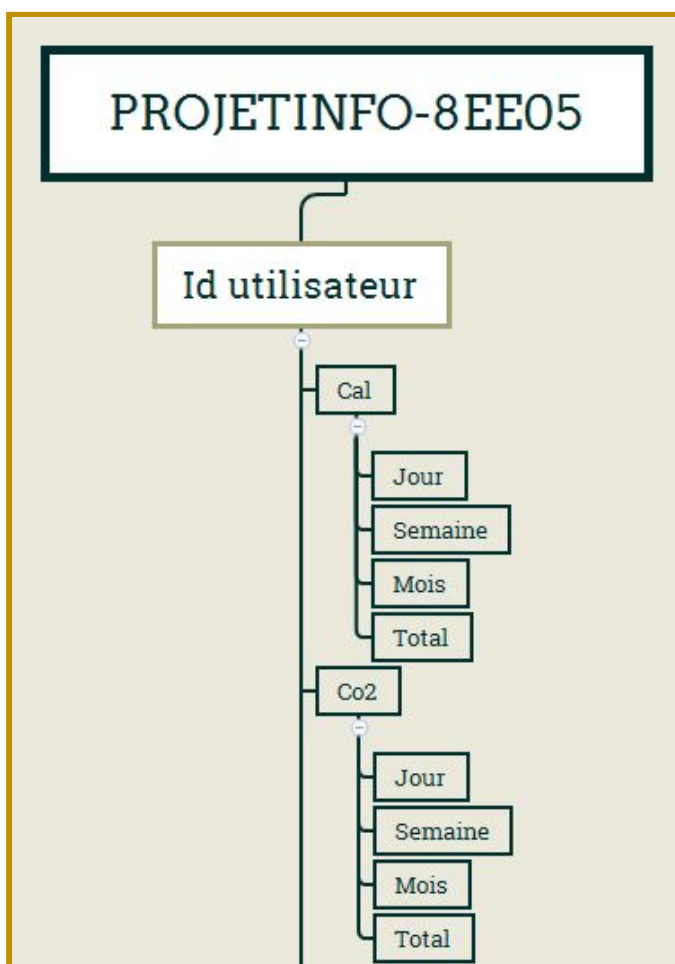
Points forts	Points faibles
Des composants graphiques existants (bibliothèque react-native)	Le développement utilisant Expo qui engendre des complications lors de la compilation
Même langage et même technologie pour android et IOS	Une documentation assez peu pratique à utiliser
Un fonctionnement proche de celui de HTML et CSS pour la vue	Une technologie jeune et donc des questions sans réponses
Beaucoup de composants à dispositions	Le design parfois compliqué à réaliser

## B. Firebase

Firebase est un service de base de données NoSql en temps réel créé par Google. Ce service permet d'éviter les requêtes SQL et la gestion de tables.

Points forts	Points faibles
Pas de requête SQL	NoSql
Gestion simple des utilisateurs	Prise en main compliquée (récupération des données )
Temps réel	Obligation de la gestion du temps réel (fonctions asynchrones)
Des requêtes existantes pour différents traitements récurrents	Obligation d'avoir une connexion internet

Architecture de la base de données :



## C. Les outils utilisés



Le framework utilisé, permettant de créer une application android et/ou IOS.



Service de gestion de base de données en temps réel développé par Google.



Service qui à mis à disposition son API dérivée de Google Map.



Permet de lancer l'application en react-native sur un android ou sur un IOS et ainsi de voir en temps réel les modifications apportées.



Editeur de code utilisé. Permet l'utilisation de fonctionnalités telles que des compilateurs ou correcteurs pour React-Native.



Prettier

Permet l'auto indentation du code.





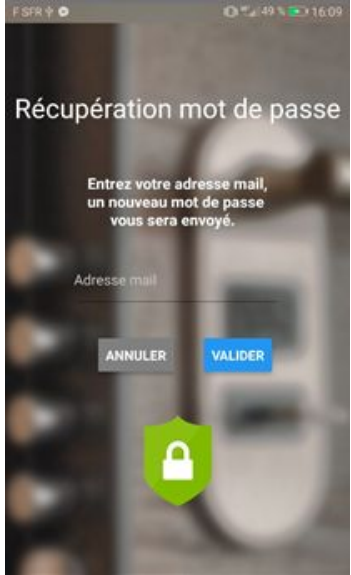
GitHub

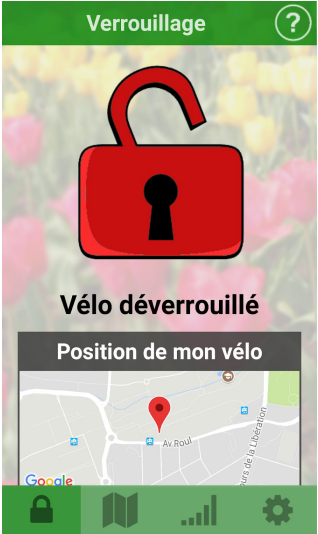
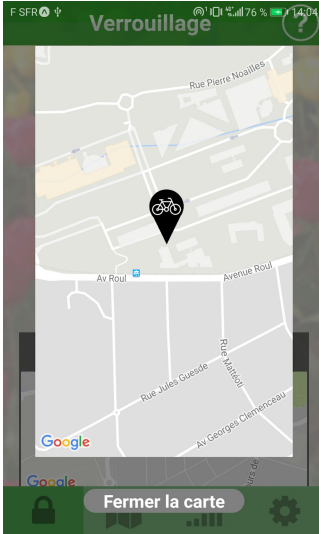

En plus du versionning, j'ai beaucoup utilisé GitHub pour récupérer des composants ou chercher des solutions donnés par le communauté.

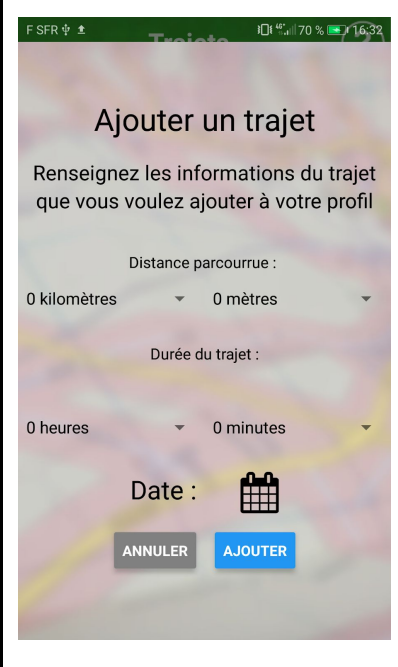

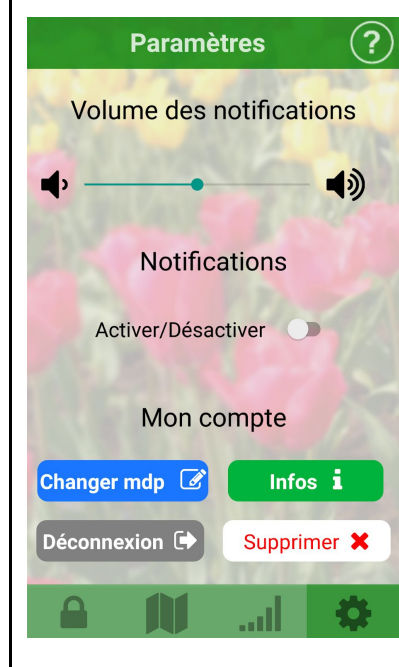


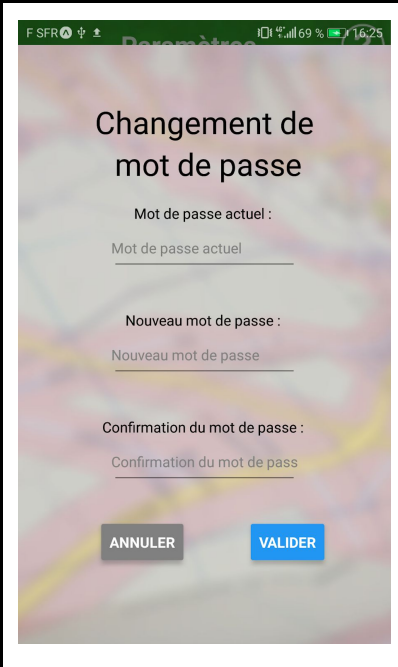

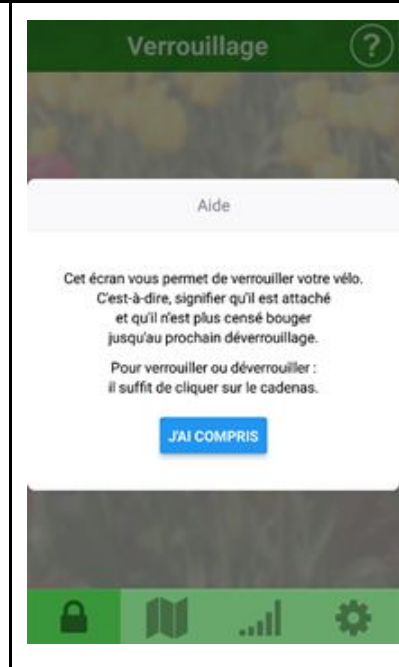
# L'application

## A. Ecrans finaux

		
<p><b>Ecran de connexion (1)</b></p>	<p><b>Ecran de création de compte (2)</b></p>	<p><b>Ecran de récupération de mot de passe (3)</b></p>

		
<p><b>Verrouillage de la position du vélo (4)</b></p>	<p><b>Position du vélo (5)</b></p>	<p><b>Historique des trajets (6)</b></p>

		
<p><b>Ecran d'ajout d'un trajet (7)</b></p>	<p><b>Ecran de statistiques (8)</b></p>	<p><b>Ecran de paramètres (9)</b></p>

		
<p><b>Ecran de changement de mot de passe (10)</b></p>	<p><b>Ecran d'informations sur les calculs des statistiques (11)</b></p>	<p><b>Popup d'aide (12)</b></p>


## B. Explications des écrans finaux

- (1) : Cet écran permet de s'authentifier sur l'application, la base de données utilisée étant Firebase, l'application nécessite une connexion à internet pour fonctionner. Les erreurs ne sont pas traduites par gain de temps et car cela ne représente pas une gêne pour l'utilisation de l'application ; ceci pour toutes les pages de l'application.
  
- (2) : Cet écran permet de se créer un compte. Pour qu'un compte puisse être créé, il faut une adresse mail valide ainsi qu'un mot de passe d'au moins 6 caractères. Une vérification de saisie permet d'éviter que l'utilisateur saisisse un mot de passe contenant une faute de frappe.
  
- (3) : Cet écran permet à l'utilisateur de récupérer son compte s'il a perdu son mot de passe. Il est demandé à l'utilisateur de rentrer son adresse mail, il recevra alors un e-mail de réinitialisation de mot de passe.
  
- (4) : Cet écran permet de signifier à l'application que son vélo est verrouillé en cliquant sur le cadenas qui change de couleur et de texte suivant si le vélo est verrouillé ou non. De plus, cet écran permet à l'utilisateur de visualiser rapidement l'emplacement de son vélo. L'utilisateur peut en cliquant sur la carte ouvrir celle-ci dans une fenêtre afin d'avoir une meilleure visualisation.
  
- (5) : Cet écran est la carte agrandie contenant la position du vélo de l'utilisateur.
  
- (6) : Cette page permet à l'utilisateur de conserver un historique des trajets qu'il a réalisés. Chaque trajet contient la date, le temps et la distance associés. En cliquant sur "Ajouter un trajet ?" l'utilisateur aura la possibilité d'ajouter un trajet qu'il a réalisé dans la journée.
  
- (7) : Cette page permet à l'utilisateur d'ajouter manuellement un trajet qu'il a réalisé en saisissant la distance qu'il a parcourue, le temps que lui à pris ce trajet et la date à laquelle il a été réalisé. (la date est modifiable pour des raisons de simplicité et de tests mais avec une balise le trajet serait ajouté immédiatement et donc la date serait rentrée automatiquement).
  
- (8) : Cet écran liste les statistiques de l'utilisateur. Ces statistiques sont calculées à chaque ajout de trajet.
  
- (9) : L'écran de paramètres contient 2 composants (volume et notification activées ou non) n'effectuant pas de traitement. Malgré mon suivi de la documentation de React-Native quant à l'interaction avec des fonctions telles que le volume ou les notifications je n'ai pas réussi à manipuler ces fonctionnalités. J'ai donc laissé les composants, ne laissant ainsi que les fonctions associées à implémenter.

- (10) : Ecran permettant la modification du mot de passe de l'utilisateur, cette modification est impossible si le mot de passe actuel est faux.
- (11) : Cet écran est un écran informatif expliquant à l'utilisateur comment sont calculées ses statistiques. Un bouton permet également de réinitialiser les données de l'utilisateur.
- (12) : Cette popup est présente sur chaque page de l'application, permettant à l'utilisateur de trouver de l'aide ou une explication relatifs à la page sur laquelle il se trouve.

## Tests utilisateur

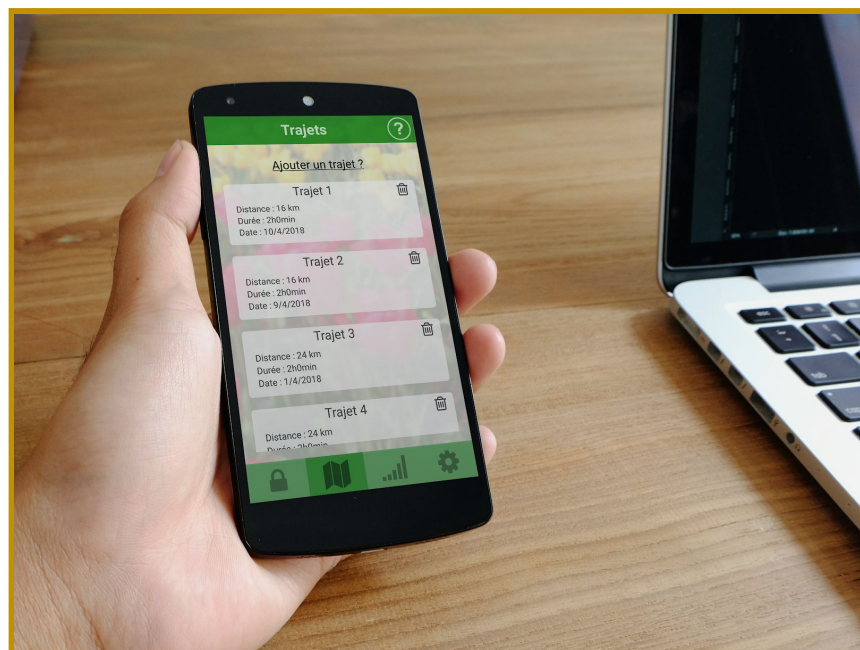
J'ai choisi de ne pas me pencher sur l'utilisation de l'API de Facebook afin de pouvoir réaliser des tests utilisateurs afin de corriger mon application. Je n'ai pas formalisé ces tests par manque de temps et parce que des tests avaient déjà été menés sur les maquettes de l'application. J'ai simplement fait tester mon application à 5 personnes et leur ai demandé au fur et à mesure leurs avis et commentaires sur les différentes fonctionnalités et pages. Ces tests m'ont permis de corriger certaines erreurs que je n'avais pas vues, et de changer certaines erreurs de design surtout sur la page de paramètres.

	
<p>Ecran avant les tests, mauvais choix de délimitation, "changer mot de passe" ressemble à un titre et non un bouton</p>	<p>Plus de délimitation, tous les boutons ont le même style</p>

# Versions

Version 1	Version 2
<ul style="list-style-type: none"> <li>Page de connexion ✓</li> </ul>	<ul style="list-style-type: none"> <li>Utilisation API google map ✓</li> </ul>
<ul style="list-style-type: none"> <li>Page de verrouillage du vélo ✓</li> </ul>	<ul style="list-style-type: none"> <li>Page historique des trajets ✓</li> </ul>
<ul style="list-style-type: none"> <li>Page de statistiques ✓</li> </ul>	<ul style="list-style-type: none"> <li>Utilisation API Facebook ✗</li> </ul>
<ul style="list-style-type: none"> <li>Liaison avec un BD ✓</li> </ul>	<ul style="list-style-type: none"> <li>Page de paramètres ✓</li> </ul>

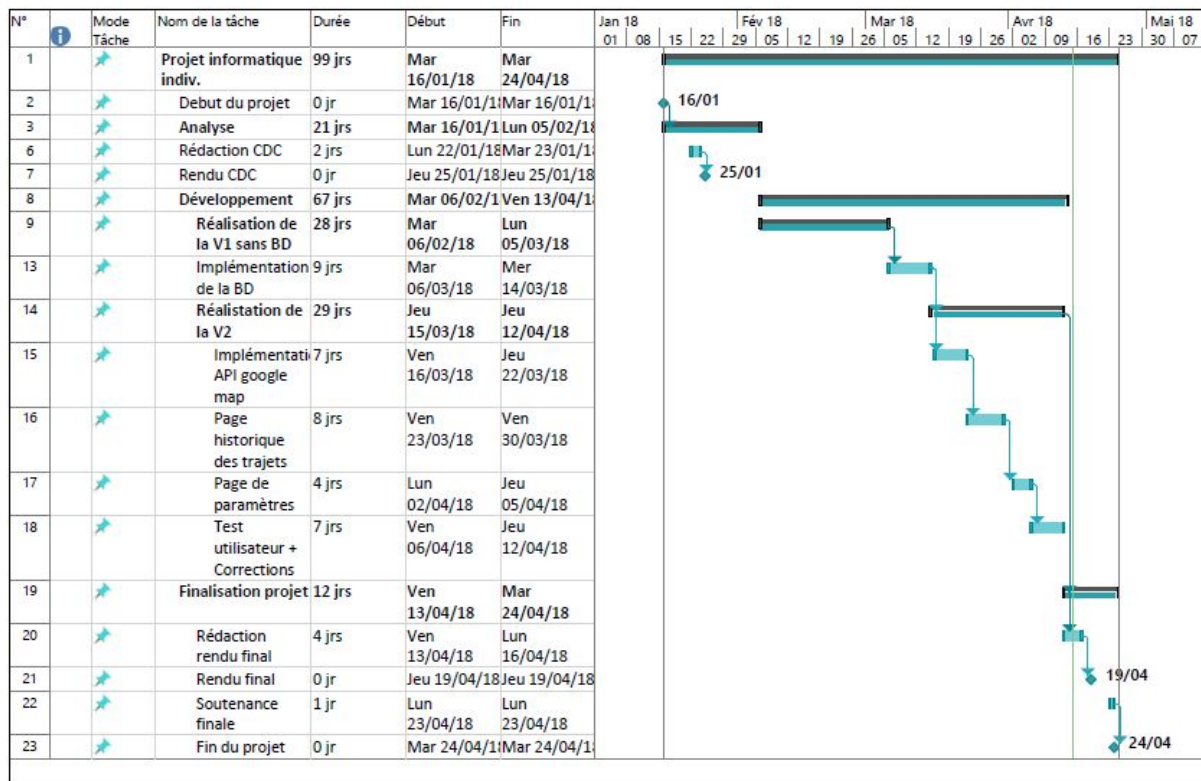
Je voulais en version 2 utiliser l'API de facebook afin de permettre un partage des statistiques. Mais en plus de contraintes de temps, cette fonctionnalité ne me paraissant pas indispensable, j'ai préféré utiliser le temps que je lui avais associé à la réalisation de tests utilisateurs et corrections de l'application en fonction des retours.





# Planification

## A. Récapitulatif



Jusqu'à l'implémentation de la base de données, j'ai réussi à garder de l'avance sur ma planification. L'utilisation de Firebase m'a ralenti et donc je me suis retrouvé à jour sans avance avec ma planification. Une fois cette difficulté passée, j'ai repris de l'avance. J'ai malgré tout choisi de ne pas implémenter l'API de Facebook afin de consacrer un peu de temps aux tests utilisateurs et à la correction de mon application en fonction de leurs retours.

## B. Difficultés de planification rencontrées

La principale difficulté de ce projet par rapport à la planification a été la nouveauté de la technologie. Je n'avais jamais réalisé d'application mobile et donc ne connaissais que peu la technologie associée, de plus React-Native étant un framework Javascript il m'a également fallu me former à ce langage. Enfin l'utilisation de Firebase m'a posé quelques problèmes à cause du noSql et du real-time impliquant l'utilisation de fonctions asynchrones.

La planification a donc été difficile car je l'ai basée sur une estimation personnelle des possibilités et difficultés de technologies que je ne maîtrisais pas.

# Conclusion

---

## A. Points d'améliorations

N'ayant pas eu de puce GPS à ma disposition j'ai utilisé la position du téléphone de l'utilisateur et non celle de son vélo. Il faudrait donc prendre en compte la position de la puce et non celle du vélo.

Je n'ai pas réussi à accéder aux notifications et paramètres du téléphone. Je pense que c'est dû à l'utilisation d'Expo. J'ai malgré tout créé les composants nécessaires à ces actions. Il faudrait donc compléter les fonctions associées à ceux-ci.

L'API google map a un problème lorsque je compile l'application mais fonctionne normalement lorsque je suis sur mon ordinateur. Je n'ai pas réussi à trouver d'où venait le problème. De plus, je n'ai pas réussi à représenter un trajet à l'aide de cette application par manque de temps.

## B. Difficultés rencontrées

Les difficultés que j'ai pu rencontrées sont en majorité dû à mon manque de pratique sur la technologie que j'ai choisi mais aussi à la jeunesse de celle-ci. La documentation liée à react-Native est parfois peu précise voir fausse à cause de mise à jour. Il y a des questions sans réponse malgré la visite de plusieurs forums.

L'utilisation d'Expo pour compiler m'a ralenti et a généré des complexifications par rapport au développement sans cet outil.

J'ai voulu mettre à profit ce projet pour apprendre à utiliser Firebase, ce service de base de données fonctionnant en NoSql a été déroutant et m'a fait perdre du temps pour essayer de comprendre le fonctionnement de cet outil.

Enfin, l'utilisation de Javascript m'a parfois fait perdre du temps car ce langage étant non typé plusieurs calculs effectués avec des string convertis en int me conduisaient à des erreur car au lieu de s'additionner, les valeurs se concaténaient.

## C. Apports du projet

Malgré les difficultés de planification et de développement ce projet a été très intéressant. Il m'a permis de me confronter à une nouvelle technologie, un nouveau langage et de nouvelles problématiques liées au développement mobile. De plus, React-Native est un framework jeune mais de plus en plus utilisé. Le fait qu'il soit basé sur le React, la réalisation de ce projet me permet de savoir utiliser le React et le React-Native. Enfin, j'ai trouvé ce projet très intéressant car il m'a appris à coder seul en n'ayant que pour appui la documentation et la communauté (Anglophone) de react-native.