

# Journal de bord

Gabin Serruot

4 mars 2024



# Table des matières

<b>I</b>	<b>Journal de bord</b>	<b>5</b>
<b>1</b>	<b>Janvier</b>	<b>7</b>
<b>2</b>	<b>Février</b>	<b>9</b>
<b>3</b>	<b>Mars</b>	<b>11</b>
<b>4</b>	<b>Avril</b>	<b>13</b>
<b>5</b>	<b>Mai</b>	<b>15</b>
<b>6</b>	<b>Juin</b>	<b>17</b>
<b>II</b>	<b>Notes pour plus tard</b>	<b>19</b>
<b>III</b>	<b>Annexes</b>	<b>23</b>



Première partie

**Journal de bord**



# Chapitre 1

## Janvier

### 1. 08/01/2024 :

- J'ai lu le dossier de présentation du projet
- J'ai essayé de créer une organisation sur Github afin d'avoir tous les codes au même endroit et afin d'éviter d'être dépend les uns des autres

### 2. 09/01/2024 :

- J'ai finalement décidé de ne pas faire une organisation, juste de faire un dossier dans mon compte Github et accorder l'accès à Yahya et Andréa afin de tout simplifier. De cette manière, je peux utiliser Github Desktop
- J'ai commencé le diagramme des cas d'utilisation "Aller à la recherche d'un actif dans l'entrepôt" avec le site visual-paradigm : figure 6.1a
- J'ai commencé le diagramme des cas d'utilisation "Contrôler le chargement du camion" avec le site visual-paradigm : figure 6.1b
- Il y aura aussi les diagrammes d'exigences et de déploiement à faire, même s'ils ne sont pas notés dans le dossier de présentation du projet

### 3. 10/01/2024 :

- J'ai d'abord perdu une bonne heure à essayer de debugger mon journal de bord, l'inclusion d'images ne fonctionnait pas alors qu'il manquait juste "usepackagegraphicx" pour que tout fonctionne.
- J'ai repris les diagrammes que j'ai fais la veille avec M. Hacquard : figures 6.2a et 6.2b

### 4. 11/01/2024 :

- Je me suis lancé dans les diagrammes de déploiement mais Pierre m'a dit que ces diagrammes sont à faire par équipe car ils représentent tout le système, pas chacun le sien
- J'ai donc décidé de faire la reformulation du cahier des charges

### 5. 12/01/2024 :

- J'ai vu pour la première fois un beacon avec M. Lejoncour
- J'ai commencé le cahier de recettes

### 6. 15/01/2024 :

- J'ai fais l'IHM de l'application (la veille mais faut pas le dire, cf image 6.3)
- J'ai continué le cahier de recettes
- J'ai aidé Andréa à installer **JMerise**
- "Définir le plan de numérotation des tags" devient "Donner un tableau exemple d'association entre les beacons et les actifs" dans **Planification des tâches du projet**

### 7. 16/01/2024 :

- Avec Yahya et Andréa nous avons fait le diagramme de déploiement (cf image 6.4a) et le diagramme d'exigences (cf image 6.4b)
- Le diagramme de déploiement convient bien mais par le diagramme d'exigences
- Pour l'IHM, je ne ferai probablement pas de système de connexion car ça rajouterait des contraintes trop importantes en terme de charge de travail, je me contenterai probablement plus d'un bouton de type "switch" pour basculer entre le mode de contrôle du chargement et le mode de recherche d'actif dans l'entrepôt

### 8. 17/01/2024 :

- Nous allons reprendre le diagramme d'exigences. En parlant avec M. Hacquard, nous avons fait deux diagrammes d'exigences, un pour le camion (cf image 6.5a) et un pour l'entrepôt (cf image 6.5b)
- Nous avons également légèrement modifié le diagramme de déploiement (cf image 6.6)

## 9. 18/01/2024 :

- Je vais commencer mes fiches de test car M. Hacquard a dit que les exigences en périphérie du diagramme d'exigences représentent les tests unitaires et les exigences plus hautes dans la hiérarchie du diagramme représentent les tests d'intégration
- J'ai réagencer le diagramme d'exigences de l'entrepôt pour améliorer la lisibilité
- J'ai perdu énormément de temps car **Github Desktop** a fait des siennes, il ne m'affichait plus les commits donc je ne pouvais pas faire de push sur mon Github. Je suis sorti de la salle sans que le problème soit résolu, espérons que ça ira mieux demain

## 10. 19/01/2024 :

- A priori Github Desktop fonctionne de nouveau, j'ai quand même pris du temps pour reprendre là où j'en étais hier
- On a convenu avec M. Le Joncour que nous feront notre première revue non formelle vendredi prochain, le 26 janvier donc il faudra que nous fassions notre diapo lundi ou mardi prochain afin d'être prêt en avance. Il s'agit d'une revue non formelle donc il n'y aura normalement pas de partie individuelle, il y a juste la présentation en groupe à faire. Pour cette revue nous attendons que Andréa et Yahya aient finis le modèle logique de données
- J'ai réalisé le cahier plan de numérotation des tags
- J'ai poursuivi les fiches de tests

## 11. 20/01/2024 :

- J'ai terminé (sur mon temps libre oui) les fiches de tests

## 12. 22/01/2024 :

- J'étais en CCF, je n'ai rien pu faire du tout je suis sorti 10 minutes avant la fin

## 13. 23/01/2024 :

- Nous avons d'abord décidé de ce que nous allions mettre dans nos diapos

## 14. 24/01/2024 :

- J'ai poursuivi les diapos
- Nous avons perdu environ 1h15 car nous avions une présentation sur la poursuite d'études au Québec

## 15. 25/01/2024 :

- J'ai poursuivi les diapos mais pendant peu de temps, j'ai terminé à la maison
- Nous avons passé beaucoup de temps avec M. Hacquard car nous avons essayé une nouvelle alternative à notre MCD : notre tentative (image 6.7a) et la proposition (image 6.7b)

## 16. 26/01/2024 :

- Nous sommes passé pour la première revue, non notée et informelle

## 17. 29/01/2024 :

- Les débuts ont été très compliqués, la connexion a été infernale et mes données mobiles également
- J'ai créé le projet sur **Android-studio** comme étant un **navigation drawer activity**
- J'ai commencé à créer le Figma

## 18. 30/01/2024 :

- J'ai fini Figma : image 6.8
- J'ai essayé de lancer le projet sur Android studio avant de réellement commencer à coder mais je n'ai pas réussi, la première compilation a pris plus de 20 minutes sans aboutir

## 19. 31/01/2024 :

- J'ai essayé d'installer **JetBrains Toolbox** afin d'installer **Android Studio Giraffe** et non **Android studio Hérisson**, l'objectif étant de voir si la version trop récente aurait pu expliquer les problèmes de création de projet
- A priori la solution ne venait pas uniquement de là, j'ai créé un dossier dans mes documents et fais un lien avec github
- J'en ai également profité pour refaire mon architecture dossier du projet. J'ai tout mis dans mes documents avec cette architecture :

```
Documents/
  Tracking-BLE/
    Tracking-BLE-Code/
      Le reste
    Tracking-BLE-documents/
      Le reste
```



# Chapitre 2

## Février

1. **01/02/2024 :**
  - J'ai enfin commencé le développement
  - J'ai essayé de mettre l'application provenant de github sur mon téléphone afin d'avoir une balise fonctionnelle en permanence
  - J'ai rempli la base de données avec des valeurs random pour soulager Yahya
2. **02/02/2024 :**
  - J'ai modifié les fichiers du projet de l'application afin d'avoir les noms de fichier corrects, les bons logos dans le menu et les bonnes redirections entre chaque page. Désormais, l'application ressemble visuellement à ce que je voulais
3. **05/02/2024 :**
  - J'ai ajouté un séparateur entre les éléments du menu afin d'isoler les paramètres
  - J'ai ajouté une table **parametres** afin d'ajouter en base de données certains paramètres de l'application :
    - major pour trier quels beacons essayer de détecter
  - J'ai modifié l'image du logo de l'application pour avoir le logo de l'entreprise
  - J'ai essayé de commencer à remplir la page des paramètres mais le problème est la difficulté apparente à créer des champs de saisie ainsi que du texte à afficher
4. **06/02/2024 :**
  - J'ai essayé de commencer à remplir la page des paramètres mais le problème est la difficulté apparente à créer des champs de saisie ainsi que du texte à afficher
  - J'ai donc essayé d'utiliser le code de l'application afin de commencer à remplir mon application mais je me suis rendu compte que l'utilisation de ces codes allait être vraiment compliquée, par exemple le code semble appeler une classe ou un fichier **beacon** qui semble ne pas exister. M. Le Joncour ne semble pas plus comprendre que moi et nous avons galéré à comprendre ce qui se passe, en vain
5. **07/02/2024 :**
  - Je vais essayer de nouveau de comprendre comment incorporer le code de l'appli dans le miens en espérant avoir un meilleur succès que hier
6. **08/02/2024 :**
  - Je n'ai toujours pas plus réussi hier donc je vais essayer de retrouver moi-même comment détecter les beacons environnants afin de l'intégrer au mieux dans mon application
7. **09/02/2024 :**
  - Finalement M. Le Joncour a réussi à me montrer comment utiliser la recherche de beacons grâce à ce lien donc je vais normalement pouvoir garder mon application de base
  - Le seul problème va être de découvrir comment écrire dedans, comment passer de 2 fichiers par page à un seul
8. **12/02/2024 :**
  - Grâce à mon application perso, je sais maintenant passer de deux fichiers à un seul donc je vais essayer finalement de remplir en premier lieu la page du camion
  - J'ai obtenu l'application de M. Le Joncour permettant de détecter les beacons environnants : **app\_basic\_view**
  - J'ai finalement réussi à afficher les beacons environnants dans la page camion ce qui me permet d'avoir une liste à jour en permanence
9. **13/02/2024 :**
  - Je vais essayer aujourd'hui de configurer un beacon en eddystone afin d'avoir une valeur de température
  - Finalement ça s'annonce trop compliqué, j'ai beaucoup essayé mais M. Le Joncour m'a conseillé de faire le reste avant de regarder le eddystone

10. **14/02/2024 :**
  - Je vais essayer d'encapsuler la listview des beacons dans un genre de frame (si ça existe) afin d'avoir quelque chose de plus agréable à regarder et à utiliser
  - En fait il n'y a pas besoin du tout, il suffit de paramétrer les dimensions de la listview
  - Finalement je vais plutôt commencer à préparer le visuel de l'application maintenant que je me suis bien mis dans le XML
  - J'ai aujourd'hui fais toute la page du camion, la seule chose qui manquera ce sera l'affichage de la température
11. **15/02/2024 :**
  - Il va maintenant falloir que je fasse la page des paramètres de l'application
  - Il faut regarder comment afficher un **EditText**
  - J'ai résolu ce problème, il fallait passer l'argument **android :layout\_height="match\_parent"** à **android :layout\_height="wrap\_content"**, j'étais persuadé de l'avoir fais hier mais apparemment pas
  - Il faudra que j'utilise l'expression régulière REGEX `^[A-Z]{2}-[0-9]{3}-[A-Z]{2}$` SANS L'ESPACE APRES LE CIRCONFLEXE pour la saisie de la plaque d'immatriculation
12. **16/02/2024 :**
  - J'ai fais la validation du formulaire de saisie de l'immatriculation
  - J'ai également fais pour que lorsque l'utilisateur valide sa saisie, l'application affiche le succès ou l'échec de la saisie
13. **04/03/2024 :**
  - J'ai commencé par formater l'affichage de la liste des couples major-minor sur la page du camion
  - J'ai ensuite fait les éléments de la page de l'entrepôt
14. **05/03/2024 :**
  - Il faut faire pour enregistrer la valeur de la plaque d'immatriculation localement au téléphone (JSON ou datastore) et faire pour que au démarrage de l'application, si localement on a une plaque on la saisie et sinon on demande de la saisir

## Chapitre 3

# Mars

1.



## Chapitre 4

### Avril

1.



## Chapitre 5

### Mai

1.





## Chapitre 6

### Juin

1.



Deuxième partie

Notes pour plus tard



- Pour faire des requêtes HTTP en Kotlin, on peut utiliser "volley" de google
- <https://developer.android.com/topic/libraries/architecture/datastore?hl=fr#kts>
- <https://medium.com/androidmood/stockage-de-données-avec-jetpack-datastore-5da427bd7323>
- <https://medium.com/@hasperong/get-current-location-with-latitude-and-longtitude-using-kotlin-2ef6c94c7b76>
- [https://altbeacon.github.io/android-beacon-library/distance\\_vs\\_time.html](https://altbeacon.github.io/android-beacon-library/distance_vs_time.html)



Troisième partie

*Annexes*







FIGURE 6.1



FIGURE 6.2

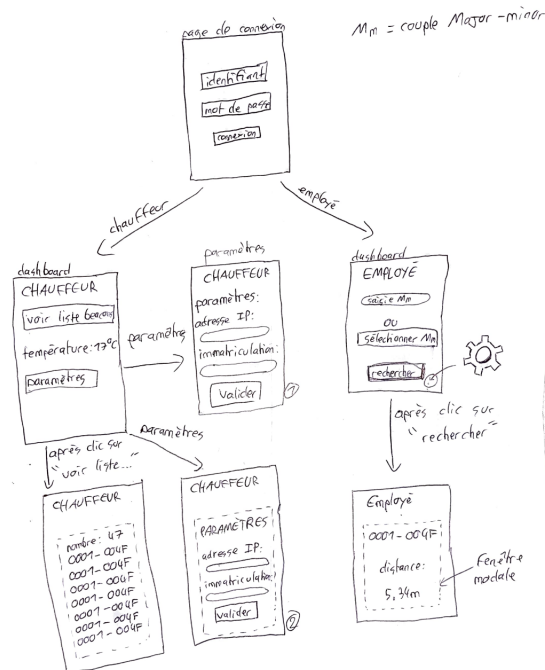
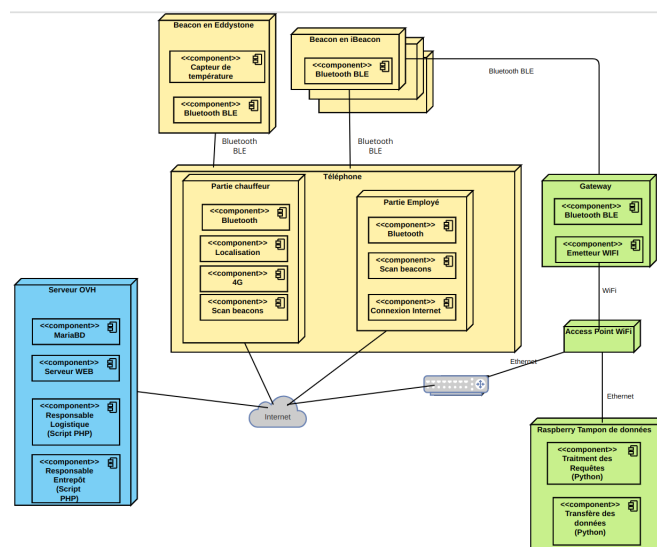
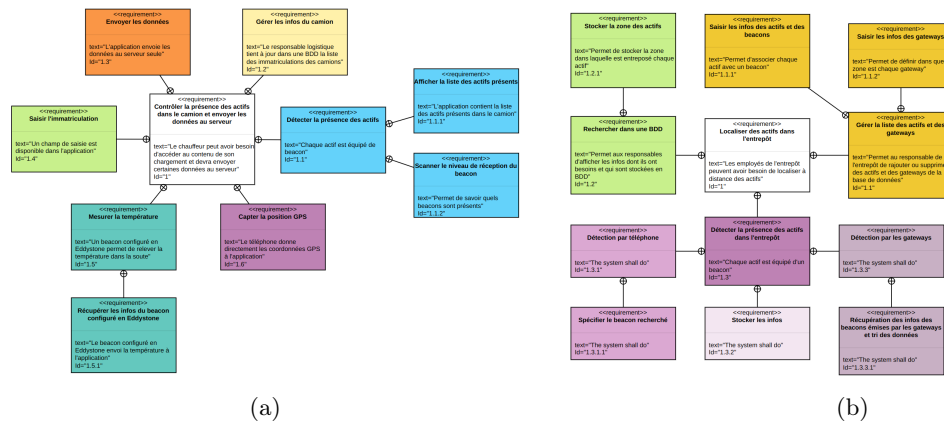


FIGURE 6.3





(a)



(b)

FIGURE 6.7



FIGURE 6.8