

Lundi 17 Sept.

8h30 : Arrivée en cours, présentation du projet.

10h : Choix du sujet, réflexion sur idées

11h : Revue des idées, choix du sujet : Détecteur de nid de poules intégré à la Twizy.

11h30 : Réflexion sur la BOM et composants à choisir.

12h : Ecriture du schéma par blocs.

12h20 : Ecriture du schéma FAST

Mardi 18 Sept.

13h45 : Arrivée en cours, commencement du cahier des charges

15h30 : Réflexions sur la BOM et choix des composants.

16h45h : Commencement des calculs pour choix de batterie.

17h20 : Tout le monde du groupe est présent.

Lundi 24 Sept.

8h30 : Arrivée en cours.

9h30 : Fin de l'écriture de la BOM, reprise du cahier des charges.

Mardi 25 Sept.

13h45 : Arrivée en cours

16h : Fin de la BOM, commencement du travail sur le module GPS.

16h30 : Reprise du cahier des charges.

17h30 : GPS fonctionnel.

Lundi 1er Oct.

8h30 : Arrivée en cours

8:40 : Passage en revue de CDCF.

9h : Répartition des tâches, récupération du matériel oublié à la maison.

9h30 : revue du GPS sur carte.

10h : Configuration de GitHub pour l'équipe entière.

11h : Ecriture d'un parser pour le traitement des données du GPS.

12h : Firmware qui bloque des pins sur la carte : Voir Mr. Viateur.

Mardi 2 Oct.

13h40 : Arrivée en cours.

13h45 : Mise en place de la caméra pour faire un timelapse.

14h : Travail sur l'accéléromètre et le télémètre.

14h45 : Module GPS : Tentative de faire en I2C.

16h : Travail sur le module sigfox.

17h : Télémètre infrarouge terminé et fonctionnel, accéléromètre fini mais non testé.

Lundi 8 Oct.

8h30 : Arrivée en cours.

8h50 : Arrivage des produits commandés.

9h : Reprise travail télémètre

Mardi 9 Oct.

13h45 : Arrivée en cours.

14h : Récapitulatif des tâches à faire.

14h10 : Travail sur le Sigfox, accéléromètre, télémètre.

14h30 : Début rédaction de la documentation technique.

Lundi 15 Oct.

8h30 : Arrivée en cours.

Tâches à finir : Fusion SD + GPS, Accéléromètre.

10h : Problèmes de compatibilité entre SD et GPS.

11h : Accéléromètre enfin débloqué (changement de matériel, on a compris que le MMA7660FC était brûlé).

12h : Accéléromètre fonctionnel.

22h : Création du programme de démo

1h : Fin du programme de démo (avec Sigfox, Accel, GPS, Télémètre).

Mardi 16 Oct.

13h45 : Arrivée en cours.

13h50 : Réunion sur le développement des algorithmes et du workflow du programme.

14h10 : Review du code avec l'équipe afin de rattraper le retard de tout le monde.

Lundi 22 Oct.

Victor Verbeke :

8h59 : Arrivée en retard, début du travail.

9h05 : Mon PC étant actuellement pas en bon état, j'ai dû le formater. Mais github est cassé pile quand je reconfigure mes repositories

9h30 : Fin de la configuration du PC (Atom, github, Slack)

9h30 - 10h45 : Opti du flow d'exécution.

Pause.

11h - 12h15 : PCB

Dimitri Kokkonis :

8h30 : Arrivé en salle de projet, début du travail.

9h00 : Réparation d'un mauvais callback Sigfox qui marchait pas.

9h00 - 11h00 : Configuration du serveur Ubidots pour que les positions sont traités dans des variables synthétiques ("lat_modified" et "lng_modified"). J'ai également essayé de formater la position de façon qu'elle puisse être affichée sur un Map Widget, mais on ne peut pas créer des objets JSON dans des variables synthétiques sur Ubidots (le Map Widget impose un format d'objet JSON très particulier afin de pouvoir afficher la position sur la carte).

11h00 - 11h15 : Pause.

11h15 - 12h45 : J'ai réessayé d'intégrer les positions du module Sigfox dans un Map Widget de Ubidots, mais j'ai pas réussi à le faire. Un serveur custom couplé avec une API écrite par nous serait mieux pour traiter ces données et les afficher comme on veut.

Valentin Renard :

8h30 : Arrivée en projet et répartition du travail.

9h00 : Réflexion sur les idées de vidéo pour le concours.

9h30 : Recherche de logiciels pour faire de l'animation 3D et de documentation pour argumenter.

Elodie :

8h30: Arrivée en cours

8h45 : Recherches sur la façon de créer le PCB

9h45 : Prise en main du logiciel Altium Designer avec les tutos

11h30 : Début de création du modèle PCB du projet sur Altium Designer

Mardi 23 Oct.

Victor Verbeke :

13h45 - 17h45 : PCB (design schematic)

Dimitri Kokkonis :

13h45 : Arrivé en salle de projet, reprise du travail.

13h45 - 16h00 : Prise en main et testing des nouveaux capteurs IR (construction des connecteurs à base des headers car il n'y avait plus des connecteurs 3-pin de style Grove, édition du code de base des capteurs IR pour ajouter tous les nouveaux capteurs, vérification du bon fonctionnement des capteurs).

16h00 - 16h15 : Pause.

16h15 - 17h00 : Examinations d'autres solutions backend (Microsoft Azure).

17h00 - 18h00 : Test Sigfox + 6 capteurs IR. Tout marche correctement.

Valentin Renard :

13h45 : Arrivée en projet et répartition du travail.

Toute la séance : Recherche de solution de remplacement à la librairie SDFFileSystem qui n'est pas compatible avec la librairie mbed-os 5.10. J'ai trouvé SDBlockDevice mais cette fois c'est la librairie mbed-os qui ne fonctionne plus.

Elodie

13h45 : Arrivée en cours

14h : Rédaction de la doc technique

15h : Travail sur Altium Designer pour terminer le modèle PCB (schéma)

Lundi 5 Novembre

Victor Verbeke :

8h30 - 12h45 : PCB (Design carte imprimée)

Elodie :

Altium Designer pour faire le modèle PCB à partir du schéma déjà réalisé

Valentin Renard :

8h30 : Arrivée en salle de projet. Recherche d'idées pour la vidéo.

Dimitri Kokkonis :

8h30 - 12h45 : Recherche d'une solution au problème de compatibilité de SDFFileSystem - mbed-os.

Mardi 6 Novembre

Victor Verbeke :

13h45 - 17h45 : PCB (Fin design carte imprimée et impression)

Elodie :

-Altium Designer pour terminer le PCB

Valentin Renard :

13h45 : Arrivée en projet et répartition du travail.

Toute la séance : Test de la carte SD avec Dimitris.

Dimitri Kokkonis :

13h45 - 18h00 : Solution trouvée - Modification du code & développement d'un algorithme pour enregistrer les données sur la carte en prenant compte du fait qu'il faut fermer le fichier pour que l'écriture s'effectue.

Lundi 12 Novembre

Victor Verbeke :

9h - 12h45 : PCB (redesign carte imprimée, carte précédente non fonctionnelle)

Dimitri Kokkonis :

8h30 - 12h45 : Développement et test de l'algorithme d'écriture sur la carte SD (écritures multiples sur des fichiers différents afin d'éliminer les pertes de données). J'ai soudé les header et les capacités de découplage sur le PCB qu'on a fabriqué.

Valentin Renard :

8h30 : Arrivée en projet et répartition du travail.

Elodie : Travail sur le PCB

Mardi 13 Novembre

Victor Verbeke :

13h45 - 17h45 : PCB (fin redesign et impression)

Valentin Renard :

13h45 - 18h00 : Configuration du cloud sous Azure avec Power BI pour analyser les données.

Dimitri Kokkonis :

13h45 - 18h00 : Recherche de composants pour l'optimisation de l'étage d'alimentation. Conception de deux optimisations afin de baisser la consommation d'énergie.

Elodie : Travail sur le script de la vidéo + prise en main du logiciel Solidworks pour créer le boîtier

Lundi 19 Novembre

Victor Verbeke :

8h30 - 10h : Finalisation PCB avec Dimitris.

10h - 10h30 : Ecriture scénario de la vidéo pour le Twizy Contest

10h30 - 12h45 : Modification des documents (cahier des charges, création de la fiche technique)

Valentin Renard :

8h30 - 12h45 : Fin de la configuration d'Azure, le cloud marche.

Dimitri Kokkonis :

8h30 - 12h45 : J'ai soudé les headers & les capacités de découplage sur le PCB.
Vérification du bon fonctionnement du PCB. Mesures de consommation d'énergie.

Elodie : Création du boîtier sur Solidworks

Mardi 20 Novembre

Victor Verbeke :

13h45 - 16h : Travail sur le boîtier, étude des optimisations avec transistors.

16h - 16h40 : Enregistrement audio pour vidéo Twizy Contest

16h40 - 18h : Etude des optimisations avec transistors.

Elodie : Travail sur le boîtier + enregistrement de la bande son pour la vidéo

Dimitri Kokkonis :

8h30 - 12h45 : Enregistrement de la partie audio nécessaire pour la vidéo. Vérification PCB
+ Cloud, mesures de consommation d'énergie.

Valentin Renard :

8h30 - 12h45 : Configuration de Power BI pour analyser les données et les cartographier.
Test de la réception des données avec le système complet.