
Projet Robot d'Accueil

rapport final



Réalisé par :

Groupe d'étudiants de l'ENSTA Paris

Encadré par :

Thibault Toralba

Classe : 2A Techniques Avancées

Année universitaire : 2020/2021

Table des matières

1	Présentation du projet	2
1.1	Naissance du projet	2
1.2	Enjeux et objectifs	2
1.3	Parties prenantes	3
2	Organisation du projet :	4
2.1	Organisation de l'équipe-projet	4
2.1.1	Répartition des rôles	4
2.2	Répartition des tâches	4
2.3	Forces et faiblesses du projet	4
2.4	Les choix qu'on a fait	5
2.5	Analyse des risques	6
3	Livrable prévus	7
3.1	Livrable logiciels	7
3.2	Livrable documentaires	8
4	Présentation de la solution technique	9
4.1	Livrables réalisés	9
4.1.1	Simulation ROS	9
4.1.2	L'interface graphique	10
4.1.3	Le site Web	10
4.1.4	Les livrables non réalisés	12
5	La maîtrise du projet	13
5.1	Communication	13
5.2	Planning initiale	13
5.3	Planning actualisé	15
6	Cahier des charges	16

Chapitre 1

Présentation du projet

1.1 Naissance du projet

L'idée du projet est né suite à l'analyse d'une problématique d'orientation et d'accueil des visiteurs au sein des locaux de l'ENSTA faite par Messieurs Gandiol et Chapoutot. L'ENSTA possède en effet de nombreuses salles et couloirs numérotés d'une manière spécifique et il peut être parfois difficile de comprendre cette organisation rapidement. Dans le cas de l'accueil d'un visiteur sans qu'un membre de l'administration ne vienne le chercher au niveau de la porte d'entrée principale de l'ENSTA, il est souvent difficile de lui indiquer l'endroit où il est attendu. Il fallait donc créer un dispositif interactif, innovant et à l'image de l'ENSTA pour répondre à ce problème et accompagner les visiteurs. C'est ainsi qu'est né le projet Robot d'Accueil.

1.2 Enjeux et objectifs

Ce PIE constitue un projet conséquent et son accomplissement complet sera atteint dans plusieurs années. Il consiste en la création, à partir de zéro, d'un robot mobile capable de réaliser la fonction follow-me pour guider toute personne utilisant ses services afin de trouver un endroit précis au sein de l'ENSTA. Cette fonction follow-me est déjà utilisée dans de nombreux domaines comme l'aviation (voitures guidant les avions atterrissant sur les pistes des aéroports), mais les nombreux paramètres à prendre en compte de façon automatique rendent le projet complexe et chronophage. Cette année, la promotion 2022 avait pour objectifs d'introduire le sujet ainsi que d'effectuer une analyse complète, précise et réaliste du projet. Vous trouverez dans la suite de ce rapport les éléments qui découlent de ce premier état des lieux et les livrables associés que nous avons choisi de travailler en priorité cette année. De plus, nous avons également pour objectif de sortir un prototype avant notre départ en PRe (stage de recherche) en mai. Ce premier robot n'avait pas à réaliser la totalité des tâches prévues mais au minimum pouvoir accompagner une personne de la porte d'entrée principale de l'ENSTA au Poste de Contrôle Sécurité (PCS) afin que ses agents puissent remplir la fonction d'orientation des visiteurs comme ils le font déjà.

1.3 Parties prenantes

Dans la réalisation du projet Robot d'Accueil, on peut distinguer 3 type d'acteurs différents. Ils participent tous au bon déroulement du PIE et sont complémentaires dans l'aide qu'ils nous fournissent.

Nos encadrants :

M. TARUFFI gère l'ensemble des PIE de l'ENSTA Paris et s'assure que ces derniers avancent correctement. Il joue également de rôle de Product Owner en exigeant un rapport initial du projet à son commencement et un rapport final de ce dernier avant notre départ en PRe (stage de recherche).

MME. BOURNAUD vient environ une fois par mois à notre rencontre afin de s'informer sur notre avancement et nous conseiller sur notre démarche Ingénierie Système. Elle est également disponible par mail et répond au plus vite à toutes nos interrogations sur les méthodes de gestion de projet utilisées.

M. TORALBA est notre tuteur pour ce PIE. Il est membre de l'U2IS à l'ENSTA Paris et nous aiguille sur les décisions à prendre au niveau technique. Il nous a déjà fortement aidé sur le choix de l'environnement pour coder les fonctions du Robot d'Accueil ainsi que sur l'achat du matériel le plus adapté à notre sujet. M. TORALBA fait aussi régulièrement des points avec toute notre équipe sur l'avancement du projet et reste disponible pour répondre à toutes nos questions techniques.

Notre client :

L'ENSTA Paris constitue notre unique client. Le Robot d'Accueil est en effet développé pour une utilisation au sein des locaux de l'école et participera donc à son image. Les membres de l'administration initiateurs de ce sujet de PIE tels que M. GANDIOL et M. CHAPOUTOT, sont les principaux éléments permettant de situer ce projet dans la stratégie de l'ENSTA Paris et de définir les exigences que ce robot doit satisfaire.

Notre équipe :

Nous réalisons de A à Z le Robot d'Accueil ce qui implique différents degrés de réflexion allant de l'analyse de la problématique à la présentation finale d'un prototype fonctionnel. Ainsi notre équipe se nourrit des conseils et de l'expérience des 2 autres types de parties prenantes afin de mener à bien le projet d'ingénierie que nous avons choisi.

Chapitre 2

Organisation du projet :

2.1 Organisation de l'équipe-projet

2.1.1 Répartition des rôles

Chef de projet : Mahdi Cheikrouhou
Secrétaire : Mariem Maazoun
Porte parole : Aurélien Mondinot
Responsables Data : Marwen Bahri
Responsable matériel : Jianzhou Ma
Responsable sécurité : Mouin Ben Ammar
Responsable des achats : Dhimoïla Gabriel

2.2 Répartition des tâches

Equipe qui travaille sur la simulation ROS :
— Mahdi Cheikhrouhou
— Jianzhu Ma
— Gabriel Dhimoïla
Equipe qui travaille sur l'interface graphique
— Marwen Bahri
Equipe qui travaille sur le site web
— Aurélien Mandinot
Equipe qui travaille sur la documentation
— Mohamed Issa
— Mouin Ben Ammar
— Mariem Maazoun

2.3 Forces et faiblesses du projet

La matrice swot nous permet de mettre en perspective toutes les données externes et les données internes.



FIGURE 2.1 – Matrice SWOT

2.4 Les choix qu'on a fait

On a décidé cette année de s'occuper par la partie software du robot , c'est à dire tout ce qui est programmation de code soit pour le fonctionnement du robot soit pour d'autre choses. De cette façon , l'équipe de l'année prochaine va s'occuper seulement par la partie hardware. Ces choix ont été pris en accord avec notre encadrant monsieur Toralba. On a réussi à terminer la plupart des livrables prévues qu'on détaillera plus tard dans le rapport.

2.5 Analyse des risques

Description du risque	Causes	Mesures
Prise en main du logiciel ROS trop longue	Nouveau langage informatique	S'y mettre dès que possible et être assidu dans l'apprentissage
Bugs dans le téléversement des codes	Codes ou machines	Ne pas téléverser au dernier moment
Ne pas pouvoir tester les codes sur un des robots déjà présents à l'ENSTA	Confinement	Les rendre suffisamment clairs pour que l'équipe prochaine puisse rapidement commencer par cela
Mauvaise communication au sein de l'équipe	Difficultés liées à l'expression du français/timidité	Communiquer le plus souvent possible sur les avancées de chacun
Démotivation	L'équipe risque de ne jamais se rencontrer	Bonne gestion d'équipe
Mauvaise gestion des obstacles	Plusieurs obstacles qui bougent simultanément	Code de détection des obstacles efficace

FIGURE 2.2 – Analyse des risques

Chapitre 3

Livrable prévus

3.1 Livrable logiciels

- Simulation ROS : Ce livrable constitue le but principal de notre équipe cette année. En effet, dans cette simulation on va créer un robot ayant toutes les fonctionnalités exigées dans le robot d'accueil et va servir comme un modèle de vérification de nos codes et stratégies. En outre, une fois que cette simulation sera prête, il sera possible de téléverser rapidement les codes sur l'un des robots déjà présents à l'ENSTA.
- Code déplacement et asservissement : Cet ensemble de code permettra de transformer les consignes de trajectoire en des commandes ROS pour faire déplacer le robot en y appliquant un couple sur les deux moteurs du robots.
- Code détection des obstacles : La détection des obstacles peut être l'une des fonctionnalités les plus compliquées à réaliser à cause du milieu de travail du robot (perturbation fréquentes, plusieurs obstacles en mouvement). Ceci est dit, le code de détection des obstacles doit être fiable et efficace afin de garantir la sécurité et le bon fonctionnement du robot.
- Code choix de trajet et odométrie : Le code de choix de trajet permet au robot de choisir le chemin qui le ramènera à sa destination à partir de sa position initiale. Afin que cet algorithme fonctionne correctement, le robot doit être capable de se repérer dans son milieu de travail (le hall de l'ENSTA).
- Code reconnaissance vocale : A travers le code de reconnaissance vocale, le robot doit être capable de déterminer le besoin de l'utilisateur, voire lui donner des instructions d'utilisation, et de prendre les bonnes décisions en fonction de son besoin. Dans le cas où le code n'arrive pas à comprendre la parole (qui est probable dans le milieu du travail du robot) il fait recours à l'interface graphique.
- Interface graphique (et son code) : L'interface graphique donne la possibilité à l'utilisateur de communiquer avec le robot en utilisant une tablette qui affiche

les instructions possibles qu'il peut donner au robot. Grâce à cette interface, nous nous assurons que l'utilisateur arrive à utiliser le robot dans toutes les conditions.

- Site Web : Le groupe de l'année dernière a déjà commencé à développer le site web mais il reste des rubriques à compléter par notre équipe. Ce support digital sera hébergé sur un serveur, disponible à tous et utilisé comme moyen de communication pour promouvoir notre robot et les initiatives étudiantes de la grande école d'ingénieur qu'est l'ENSTA.

3.2 Livrable documentaires

- Rapport initiale d'ingénierie système : Ce rapport est l'équivalent de la première partie de la méthode ingénierie système qui correspond à établir les enjeux , les parties prenantes , composition de l'équipe , diagramme de Gantt , la matrice swot , les livrables et l'analyse de risque.
- Documentation des codes (ROS, communication et interface graphique) : La documentation des codes est primordiale pour assurer que les prochaines équipes seront capables de comprendre, modifier et améliorer les codes qu'on va développer cette année. Elle sera composée de deux parties, l'une sera des commentaires sur les codes et l'autre sera un document séparé dédié pour l'explication des algorithmes et des stratégies.

Chapitre 4

Présentation de la solution technique

4.1 Livrables réalisés

Dans cette section on va détailler un peu les livrables réalisés ainsi que les raisons pour lesquelles nous n'avons pas réalisés quelques livrables.

4.1.1 Simulation ROS

La simulation ROS est la partie la plus complexe de nos livrables. Elle est divisée en 3 parties.

Le calcul du plus court chemin : développé par Gabriel Dhimoila

La simulation ROS a besoin de calculer la trajectoire du robot sur une carte. La carte a été simplifiée pour n'être qu'une grille de booléens en deux dimensions dont les cases sont des obstacles ou des cases vides. (false = case vide, true = obstacle) La simulation ROS se sert donc d'un programme en C++ pour calculer le plus court chemin, nous appellerons ce programme le PathFinding. La solution qui a été adoptée est de compiler le PathFinding en un exécutable qui sera appelé depuis le programme principal ROS. Pour aider à générer la grille, une interface graphique du programme est disponible à cette adresse :

<https://github.com/EugioSynthesisThirty/PIE/tree/master/PathFinding%20-%20Graphique>

Le code dont se sert la simulation est lui disponible à cette adresse :

<https://github.com/EugioSynthesisThirty/PIE/tree/master/PathFinding%20-%20Minimal> Pour plus d'informations je vous invite à lire les readmes des deux liens.

La carte de l'ENSTA dans l'environnement ROS : développé par Jianzhu Ma

Cette partie consiste à établir une carte précise de l'ENSTA dans l'environnement Gazebo, qui est basé sur ROS. En employant un plan du rez-de-chaussée de l'ENSTA et en mesurant l'échelle réelle, le groupe arrive à simuler un environnement correspondant aux conditions d'emploi du robot. Pour réaliser une navigation en temps réel du robot, une carte en forme xml a été développée en utilisant l'algorithme

cartographier et un modèle virtuel de Turtlebot3.

Le code de déplacement du robot : développé par Mahdi cheikhrouhou : Cette partie consiste à exécuter le code de calcul du chemin en lui donnant une position initiale et une position finale et une partie du code pour faire déplacer le robot selon le chemin généré. Le code est implémenté en langage python. Toute la simulation est faite sur le site [TheConstructism](#) Grâce à la licence que nous a donné monsieur Toralba. Ce code est testé directement dans le simulateur ROS en ligne. Nous avons essayé plusieurs positions initiales et finales différentes et ils ont donné de bons résultat.

4.1.2 L'interface graphique

Développée par Marwen Bahri. Le robot d'accueil aura une interface graphique avec laquelle l'utilisateur peut interagir. Cette interface consiste en un site web qui sera connecté sur le localhost d'une tablette. On a décidé que ça soit un site web pour faciliter le développement : c'est beaucoup plus simple et rapide que de développer une application mobile. Un site web répond à tous nos besoins. De plus, les éventuelles améliorations et modifications par le groupe de l'année prochaine seront plus simples pour un site que pour une application mobile. Le site web se compose de trois pages principales. Une page d'accueil avec laquelle l'utilisateur peut faire sa recherche, une page du résultat de la recherche qui comprendra un tableau dont les lignes sont le personnel et/ou les salles présentes dans la base des données et qui répondent au critère de la recherche. Un clique sur une ligne donnée affichera un panneau qui contient des informations supplémentaires sur cette instance avec deux boutons : un pour retourner au résultat de la recherche et l'autre pour demander au robot de nous guider vers la destination cherchée. La dernière page est simplement une page dans laquelle la phrase « suivez-moi » est écrite et qui reste affichée jusqu'au le robot arrive à la destination. La tâche de reconnaissance vocale n'était pas développée cette année, elle reste à faire pour l'année prochaine au plus d'autre améliorations potentielles de l'interface utilisateur que nous avons proposé dans le rapport de documentation du site web.

4.1.3 Le site Web

Cette partie est développée par Aurélien Mondinot, porte-parole. L'objectif est notamment de faire la publicité de l'ENSTA à travers un projet qui allie programmation et mécanique. Ce projet met donc en valeur différents points : aussi bien les connaissances des élèves déjà acquises mais également leur capacité à apprendre et utiliser de nouveaux outils rapidement. La réalisation d'un site internet faisant office de vitrine du projet était donc nécessaire. En plus de rappeler nos différents objectifs et de présenter l'équipe, le site présente une page « avancement » et qui expose les deux parties -ROS et interface graphique- sur lesquels nous nous sommes particulièrement penchés et avons essayé au mieux de poursuivre le projet. Précisons, la page d'accueil du site comprend différentes ancres qui mène plus bas dans la page. Ces ancres mènent notamment à un rappel sur le cadre de notre projet, à la présentation de l'équipe ainsi que de nos objectifs, mais également aux divers moyens de nous

contacter. Ensuite, la seconde page ("Avancement" sur le menu) entre davantage dans le sujet et traite du travail effectué, à savoir des codes C++ et de la simulation ROS pour que le robot puisse aller d'un point A à un point B. Enfin, cette page présente également quelques points de l'interface graphique du robot qui permettra à un usager de renseigner sa destination souhaitée (numéro de salle, bureau...). Cette page présente également un carousel d'images comprenant notamment une photo du campus de l'ENSTA Paris, les écoles membres de l'institut Polytechnique de Paris.. Voici quelques photos du site.

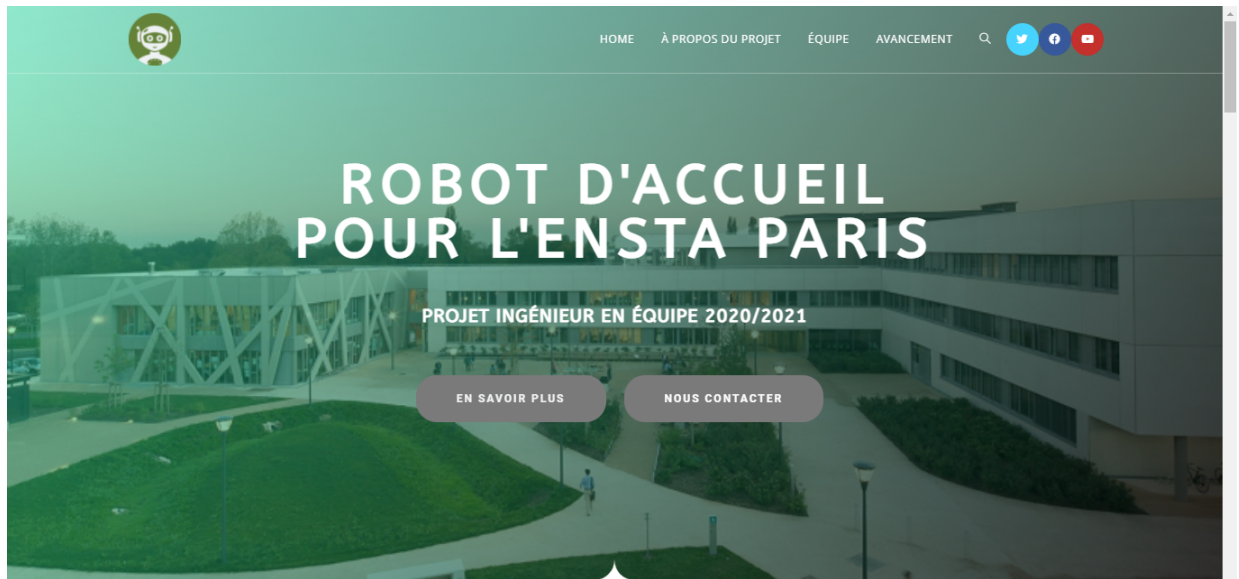


FIGURE 4.1 – Page d'accueil, menus et ancrés



FIGURE 4.2 – Les objectifs suivies

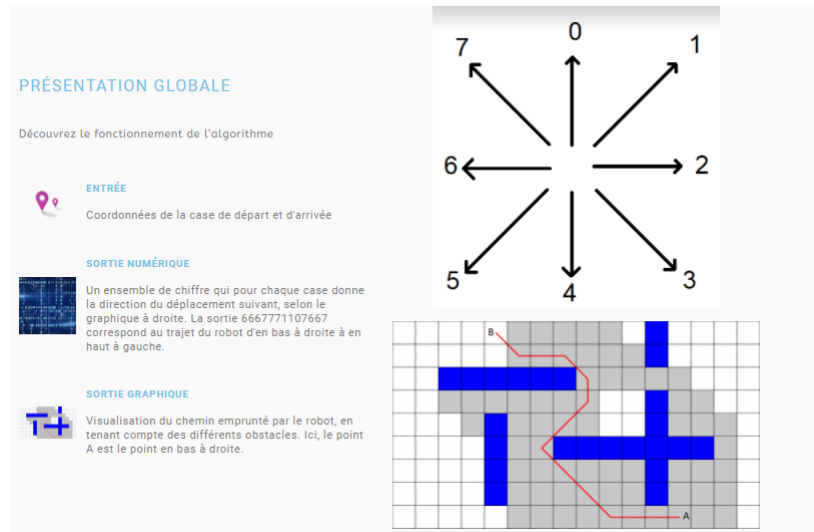


FIGURE 4.3 – Présentation de codes C++

Ce site sera hébergé sur un des serveurs de l'ENSTA Paris, démarche dans laquelle des membres de l'association DATA ont aidé.

4.1.4 Les livrables non réalisés

Le code de détection des obstacles

Ce code a été déjà développé par l'équipe de l'année dernière qui en fait utilise une caméra pour calculer la distance minimale entre le robot et un éventuel obstacle.

Le code de reconnaissance vocale

Ce livrable a été annulé pour cette année pour plusieurs raisons :

- Le bruit : le robot va être dans un environnement où il y a plusieurs bruits donc cela nécessite plusieurs tests et plus de précision.
- on a démarré l'interface graphique un peu plus tard donc on n'a pas trouvé le temps nécessaire pour la développer et la tester

La liaison entre l'interface graphique et le code de déplacement

La liaison entre ces 2 codes peut se faire par une seule ligne de code mais on l'a pas testé à cause des environnements différents pour les 2 codes. Le code de déplacement est dans la simulation ROS et l'interface graphique est sous la forme d'un projet java JEE qui est un peu difficile d'intégrer dans le simulateur ROS en ligne dans le site [theconstructism](http://theconstructism.com)

Chapitre 5

La maîtrise du projet

5.1 Communication

La communication au sein de l'équipe était un problème et elle a ralenti l'avancement du projet. En effet, nous n'avons pas systématiquement fait de réunion hebdomadaire. C'est généralement à cause des examens et de la charge de travail. De plus, il y a eu quelques absences dans les réunions, en présentiel ou en ligne . Cependant, à chaque réunion on a fait un compte rendu et on précise les tâches suivantes à faire pour chacun .

Nous avons utilisés des plate-formes de collaboration et de communication en ligne comme le Slack , microsoft teams et facebook messenger. Ce dernier a été le plus utilisé dans notre communication car c'est le site qu'on visite pratiquement chaque jour , donc il est plus facile de transmettre les informations.

Les livrables logiciels sont tous regroupé dans un github avec ses documentations qui va servir pour les équipes prochaines . Pour réaliser les rapports de documentation , on utilisé le langage LATEX dans [overleaf](#) qui est un éditeur en ligne , collaboratif , en temps réel.

Pour conclure, je pense que la communication au sein de l'équipe était un problème majeure pour la réalisation du projet, accentué par un contexte difficile de pandémie mondiale. Ce qu'on peut conseiller l'équipe de l'année prochaine c'est que le chef de projet essayera de motiver les membres de l'équipe et d'essayer de suivre une calendrier bien déterminé avec des deadlines fixées.

5.2 Planning initiale

Le planning initiale est représenté par le diagramme de gantt suivant :

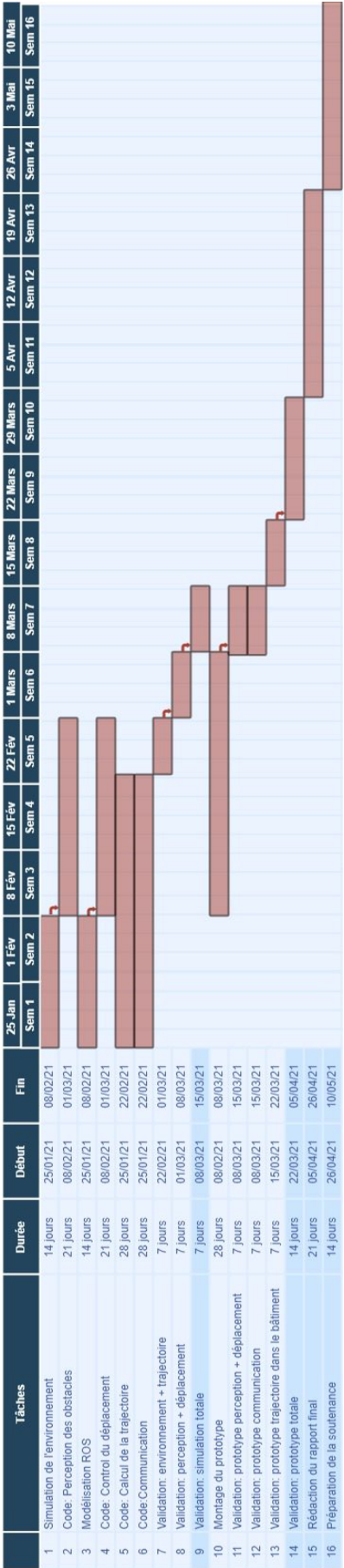


FIGURE 5.1 – Diagramme de Gantt Initiale

5.3 Planning actualisé

Le planning initial est représenté par le tableau suivant :

Activity	Status	Début	Fin	Jours
		25-01-21	11-05-21	262.0
code de calcul de trajectoire	finis	25-01-21	09-04-21	55.0
code de déplacement	finis	25-01-21	23-04-21	65.0
Carte de l'ensta	fini	25-01-21	09-04-21	55.0
Site Web	fini	25-01-21	23-04-21	65.0
interface graphique	fini	12-04-21	23-04-21	10.0
rapport finale et video	fini	26-04-21	11-05-21	12.0

FIGURE 5.2 – Tableau de Gantt actualisé

Chaque activité du tableau implique le développement et le test de chaque code. En fait, le code est toujours testé au fur et à mesure de son développement.

Il y a un retard dans la réalisation des livrables est c'est à cause de la mal organisation. En effet , les livrables sont peu liées donc les deadlines étaient flexibles ce qui a posé un problème dans quelques fonctionnalités (liaison interface , code dép) que nous n'avons pas attendu.

Chapitre 6

Cahier des charges

Catégorie	Fonction	Critères
Sécurité	Le robot doit éviter ou s'arrêter face à d'éventuels obstacles	Garder une distance d'au moins 10cm des obstacles.
	Le robot doit présenter une sécurité électrique	-Sécurité des composantes -Sécurité de l'utilisateur
	Le robot ne doit pas s'engager dans des escaliers	NaN
	Le robot doit présenter une sécurité informatique	Ne pas pouvoir être piraté
	La base de données (Enseignants, Administration, Location) doit être protégée	Ne pas pouvoir être piraté
	Le robot doit avoir un arrêt d'urgence	Sur le robot et à distance
	Le robot ne doit pas présenter de surface pouvant blesser l'utilisateur	Angles arrondies /Surface lisse.
	Le robot doit pouvoir être arrêtable avec les mains	Force des moteurs maximale (à définir)
Alimentation	Le robot doit pouvoir se déplacer en autonomie	Environ 30 minutes d'autonomie (80% de la capacité des batteries)
	Le robot doit pouvoir rentrer à sa borne de chargement	Lorsque le niveau des batteries descend en dessous de 20%
	Le robot doit pouvoir se recharger en un temps donnée	Revenir à sa place après avoir été chargé.
Se déplacer	Le robot ne doit pas entamer un trajet s'il n'a pas assez de batterie	seuil: 20% des batteries
	Le robot doit pouvoir se déplacer à une vitesse proche de la vitesse de marche	Entre 4 et 6 km/h

	Le robot doit pouvoir se déplacer dans toutes les directions	Roues adéquates.
	Le robot doit pouvoir aller à l'endroit spécifié par l'utilisateur	Le robot ne doit pas s'éloigner plus d'1m de la place d'arrivée.
	Le robot ne doit pas sortir de la zone prévue de déplacements	Zone à delimiter dans le bâtiment de l'Ensta.
Se repérer	Le robot doit pouvoir détecter des obstacles	Détecter les obstacles d'une distance de 2m.
	Le robot doit pouvoir connaître sa position	Triangulation ? (Implique la pose de balises dans l'ENSTA)
	Le robot doit connaître et reconnaître son environnement	capteurs et algorithmes de déplacement.
Agir	Accès aux zones qui nécessitent l'ascenseur ou de passer une porte (U2IS, Administration)	Accompagner le visiteur à l'ascenseur.
Données	Le robot doit avoir accès à une base de données	Salle, Nom, Fonction, Horaires de travail
	Le robot doit avoir accès à l'heure	Connaître les horaires de travail
Communiquer <i>Interface</i>	L'utilisateur doit pouvoir indiquer la salle ou une personne	A voix haute ou en écrivant sur la tablette du robot.
<i>Interface</i>	Le robot doit poser une question de confirmation (confirmation horaire, accompagnement) et acquérir la réponse	A voix haute et en affichant le message sur la tablette.
<i>Interface</i>	Signaler une situation critique	Batterie, Réajustement de trajectoire (Allumer un LED rouge)
<i>Interface</i>	Le robot doit pouvoir raconter une histoire sur l'ENSTA.	Lorsqu'il est en repos.
	Le dock doit pouvoir préparer un autre trajet pendant que le robot	

<i>Connectivité</i>	accompagne un utilisateur	
<i>Connectivité</i>	Le dock doit pouvoir indiquer la salle sans le robot	
<i>Connectivité</i>	Le dock doit pouvoir obtenir la position du robot et son état (qui, quoi, comment, batterie)	
<i>Connectivité</i>	Le robot doit pouvoir consulter la disponibilité de quelqu'un	
<i>Connectivité</i>	Un QR code doit être disponible pour "s'interfacer" avec le robot	
Esthétique	Taille ? Poids ?	
	Forme ?	
	Logo des sponsors ?	
	Matière ?	
	Processus de validation du design	