

Compétition de robotique **FIRST®** 2025

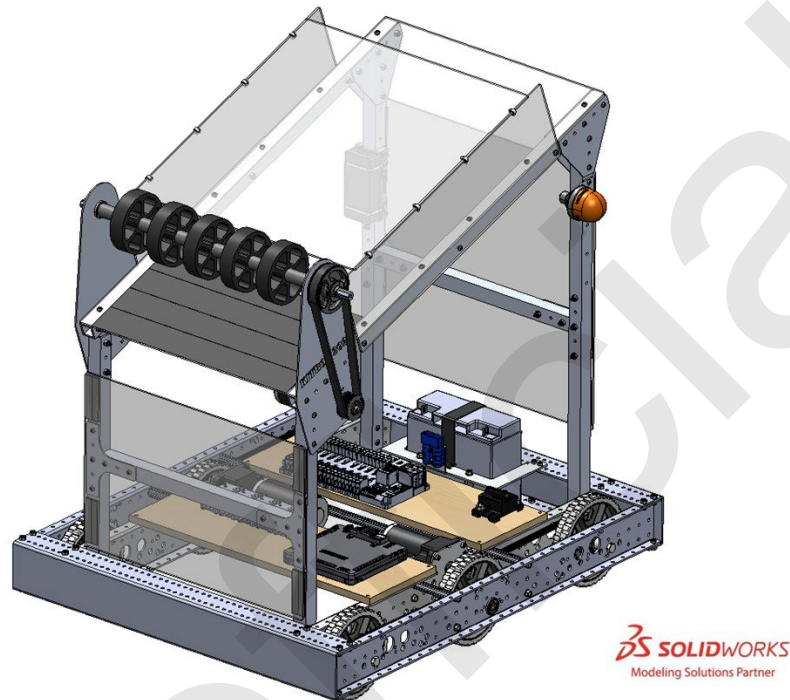
Guide d'amélioration/itération du KitBot

1 Table des matières

1	Table des matières.....	2
2	Introduction	3
3	Quand commencer à penser à des changements?.....	4
4	Amélioration – Ajout de nouvelles fonctionnalités.....	5
4.1	Analyse du jeu	5
4.2	Identifier les améliorations.....	7
4.3	Prochaines étapes	7
5	Itération – Amélioration des capacités existantes.....	8
5.1	Tests ciblés.....	8
5.2	Remue-méninges sur les améliorations.....	8
5.3	Prochaines étapes	14
6	Tri, filtre et sélection – Transformer les idées en plan.....	15
6.1	Prioriser	15
6.2	Analyse – Estimation des ressources et de l'ampleur	15
6.3	Analyse – Établissez votre « budget ».....	16
6.4	Planifier	17
6.5	Prochaines étapes	18
7	Annexe A – Exemple d'analyse de jeu de base, 2023.....	19
7.1	Exemple - Liste des tâches du robot.....	19
7.2	Exemple – Comment obtenir un bon classement.....	20
7.3	Exemple – Comment gagner des matchs.....	21
7.4	Exemple - Améliorations.....	23

2 Introduction

Figure1 : KitBot 2025



Le KitBot pour REEFSCAPESM présenté par Haas est capable de réaliser les actions suivantes. Pour certaines actions, l'équipe devra explicitement ajouter du code pour les rendre possibles (par exemple, le code du mode Auto) :

- Se déplacer sur le terrain à l'aide d'un système d'entraînement différentiel (également appelé "Tank") conçu pour atteindre une vitesse maximale d'environ ~ 15 pi/s ($\sim 4,5$ m/s). Le KitBot ne peut pas passer sous les cages profondes et peu profondes, mais il peut manœuvrer entre les cages ou les pousser hors de son chemin pendant qu'il se déplace.
- Précharger un Corail pour l'utiliser en mode Auto
- Marquer des points de Départ
- Marquer un Corail dans L1 du Récif
- Recueillir du Corail dans la Station de Coraux
- Jouer la défensive

Il s'agit d'un ensemble de capacités assez basiques par rapport à toutes les tâches possibles dans le jeu. En outre, le KitBot a été conçu avec un souci de simplicité, ce qui signifie qu'il peut y avoir des possibilités d'itération et d'amélioration des capacités existantes. Dans cette optique, les équipes peuvent choisir d'ajouter des composants supplémentaires pour permettre au robot de ramasser des pièces de jeu au sol, de grimper sur la Barge, etc. Ce document décrit un processus possible que vous pourriez suivre pour réfléchir et décider d'éventuelles améliorations à votre KitBot.

3 Quand commencer à penser à des changements?

Avant ou après avoir construit votre KitBot, vous souhaitez peut-être examiner le jeu et déterminer s'il existe des fonctionnalités supplémentaires que vous aimeriez essayer d'ajouter au robot de votre équipe. Le KitBot est conçu pour la possibilité d'ajouts futurs, et nous encourageons la plupart des équipes à l'assembler et à jouer un peu avec son robot avant de le modifier ou d'y ajouter des fonctionnalités, mais il existe des avantages potentiels à envisager des ajouts avant la construction. Voici quelques compromis qui peuvent vous aider à décider si vous devez effectuer cet exercice avant ou après avoir construit le KitBot.

Avant de construire le KitBot

- Avantages
 - Permet à l'équipe de design d'y travailler immédiatement au lieu de devoir attendre la fin de la construction
 - Il est plus facile d'apporter des modifications aux pièces de base avant leur assemblage
- Inconvénients
 - Vous pourriez concevoir plus de designs de robot que vous ne pouvez en construire
 - Vous pourriez modifier des parties du design de base que vous ne comprenez pas entièrement

Après avoir construit le KitBot

- Avantages
 - Certitude que vous disposez d'un robot capable de jouer certains aspects du jeu
 - Meilleure position pour prioriser l'itération (l'amélioration des capacités existantes) par rapport à l'amélioration (l'ajout de nouvelles capacités)
 - Meilleure compréhension de la fonction du design de base et de ce qui peut et ne peut pas changer
 - Les tests et l'entraînement peuvent avoir lieu simultanément au travail sur les améliorations
 - Les améliorations au développement du logiciel peuvent être testées immédiatement
- Inconvénients
 - Rien à concevoir tant que la construction n'est pas terminée
 - Les changements aux pièces de base peuvent nécessiter de démonter des éléments
 - Vous souhaitez peut-être utiliser des écrous et des boulons au lieu de rivets pour démonter les éléments plus facilement

4 Amélioration – Ajout de nouvelles fonctionnalités

Cet exercice peut être réalisé avant ou après la construction du KitBot (voir la section [2](#) pour les avantages et inconvénients de chaque option). Si vous effectuez cet exercice après avoir construit le KitBot, nous vous recommandons de terminer les remue-méninges de tous les exercices d'amélioration et d'itération (sections [3](#) et [4](#)), puis de les combiner à l'exercice de filtrage sélectif (section [5](#)) pour hiérarchiser les changements que votre équipe poursuivra.

Bien que cela ne soit pas strictement nécessaire, avoir défini des objectifs d'équipe est très utile pour le filtrage sélectif de cet exercice. Pour plus d'informations sur la définition d'objectifs d'équipe, consultez la vidéo Goal Setting (voa) accessible sur la page [Team Management Resources](#) sous la section Team Organization.

Les exercices et techniques présentés ici constituent une version abrégée d'un processus complet d'analyse du jeu et de stratégie de robot. Pour une ressource plus complète sur ce processus, consultez la section Strategic Design de la présentation (voa) "Effective FIRST® Strategies for Design & Competition" de la page [Technical Resources](#) sous la section Other Technical Resources->Scouting/Strategy.

4.1 Analyse du jeu

4.1.1 Analyse du jeu – Que pourrait faire d'autre notre robot ?

La première étape du remue-méninges sur les améliorations potentielles consiste à analyser le jeu et à établir une liste de choses possibles que le robot pourrait faire. Vous trouverez ci-dessous une série de questions permettant de procéder à cette opération. Pour un processus plus complet, consultez le document « Kickoff Worksheet » (voa) sur la page [Technical Resources](#) sous la section Other Technical Resources->Kickoff. C'est différent d'une simple liste de tâches dans le jeu (même si c'est un bon point de départ!) Ne commencez pas encore à réfléchir à la manière dont le robot effectuera ces tâches. Vous trouverez ci-dessous une liste de questions qui peuvent aider votre équipe à élaborer cette liste:

1. Quelles sont les façons de marquer des points dans le jeu ?
2. Est-il possible de décomposer davantage certains de ces éléments (par exemple, marquer à différentes distances peut être une capacité distincte)?
3. Comment le robot acquiert-il des pièces de jeu ? Il existe souvent plusieurs façons de prendre possession qui nécessitent des capacités distinctes.
4. Y a-t-il d'autres tâches qui pourraient aider votre alliance (par exemple, des éléments du terrain à manipuler, des moyens d'aider les partenaires, etc.) ?
5. Existe-t-il des moyens de ralentir le score des adversaires qui nécessitent des capacités de robot distinctes (c'est-à-dire pas seulement défendre en utilisant la moto-propulsion du robot) ?

Vous cherchez un exemple ? Consultez la liste [Annexe A – Exemple d'analyse de jeu](#) de base, 2023 de tâches possibles pour le jeu 2023.

4.1.2 Analyse du jeu – Comment se classer haut ?

Si les objectifs de votre équipe consistent à essayer d'atteindre un classement aussi élevé que possible (par exemple, essayer d'être capitaine d'alliance, essayer de maximiser vos points de district pour se qualifier pour le championnat de district, etc.), il est important de revoir les critères avec lesquels les équipes sont classées. Si les objectifs de votre équipe n'impliquent pas un classement élevé (par exemple, être sélectionné dans une alliance, concevoir un mécanisme intéressant, gagner un prix, etc.), vous pouvez ignorer cette section. Les critères de classement se trouvent généralement dans la section Tournoi du [Manuel du jeu](#) et sont généralement basés sur les points de classement (RP), gagnés pour les victoires ou les égalités et pour l'accomplissement d'objectifs de jeu spécifiques. En supposant un jeu de Compétition de robotique **FIRST®** typique avec un classement dans ce style, posez-vous ces questions :

1. Outre les matchs gagnés, comment les alliances peuvent-elles gagner des points de classement ?
2. Pour chacun de ces RP, quelles sont les capacités du robot nécessaires :
 - a. pour maximiser votre capacité à permettre à vos alliances de gagner des RP ?
 - b. pour obtenir les RP en jouant avec 2 autres robots comme le vôtre ?
 - c. pour faire une contribution minimale au RP de votre alliance ?
3. Quels sont les 1^{er} et 2^e critères de bris d'égalité après les RP (il est généralement peu probable qu'il y ait une égalité au-delà de cela et très peu probable qu'une telle égalité décide de plus d'un rang ou deux au maximum) ? Ces éléments ne devraient pas avoir une influence majeure dans le choix des capacités à ajouter, mais peuvent fournir un coup de pouce en cas de décision difficile entre deux capacités lors de la définition des priorités.

4.1.3 Analyser le jeu – Gagner des matchs ?

Si les objectifs de votre équipe impliquent de gagner des matchs, vous devrez passer du temps à examiner le jeu et à réfléchir à ce à quoi ressemble une alliance gagnante. Selon vos objectifs, certaines de ces questions peuvent être plus ou moins pertinentes.

1. Dressez la liste de toutes les tâches de pointage dans le jeu avec des valeurs en points.
 - a. Combien de fois une alliance peut-elle accomplir chacune de ces tâches? Il n'existe peut-être pas de réponse unique et claire, certains jeux peuvent avoir des tâches qui s'excluent mutuellement.
2. Pour chaque tâche, estimez la vitesse à laquelle vous pensez que la tâche peut être accomplie.
 - a. Les réponses à cette question varieront fortement en fonction de la force du robot. Vous souhaitez probablement choisir une « classe » particulière de robot et répondre systématiquement en fonction de ce niveau. Pour les marquages nécessitant une pièce de jeu, n'oubliez pas d'inclure le temps nécessaire pour aller chercher cette pièce de jeu et revenir au lieu de marquage.
3. Pour certaines tâches, vous souhaitez peut-être également prendre en compte un « taux de réussite ». Il s'agit de la fréquence (%) à laquelle vous pensez que le robot accomplira la tâche dans le temps que vous avez estimé.

- a. Pour une tâche de tir, cela peut être évident : à quelle fréquence la pièce de jeu entrera-t-elle dans le but ? Pour d'autres tâches, cela peut être moins évident, mais essayez de réfléchir à la fréquence à laquelle vous devrez peut-être réaligner et réessayer par exemple. Exprimez la réponse sous forme décimale (par exemple, taux de réussite de 80 % = 0,8)
4. Utilisez les chiffres des trois étapes précédentes pour calculer les points/seconde pour chaque tâche de marquage (succès*points/secondes) et triez du plus efficace au moins efficace.

En fonction de la liste de temps et d'efficacité que vous avez établie, évaluez les éléments suivants (ignorez ceux qui ne sont pas adaptés aux objectifs de votre équipe):

1. À quoi ressemble le match pour une alliance gagnante moyenne en qualifications ?
2. À quoi ressemble le match pour une alliance qui gagnerait environ 80 % des matchs de qualification ?
3. À quoi ressemble le match pour une alliance qui peut gagner un match éliminatoire ?
4. À quoi ressemble le match pour une alliance qui remporte le tournoi ?

En vous basant sur les objectifs de votre équipe et sur une évaluation de vos capacités, estimez votre rôle dans chacune de ces alliances :

1. Combien de points essayez-vous de contribuer pour l'alliance ?
2. Y a-t-il des tâches spécifiques dans ces alliances auxquelles vous pensez devoir contribuer ?
3. Y a-t-il des tâches spécifiques que vous souhaitez confier à vos partenaires d'alliance ?

4.2 Identifier les améliorations

Comparez maintenant les capacités du KitBot à la liste des tâches et aux pointages que vous venez de générer en fonction des objectifs de votre équipe. Quelles fonctionnalités devez-vous ajouter pour répondre aux points et aux capacités identifiés? Lequel de ces éléments est le plus important ?

4.3 Prochaines étapes

Si vous envisagez des améliorations avant la construction de votre KitBot, passez à la section [6](#) pour quelques idées sur la façon de transformer ces idées en un plan. Si vous avez déjà construit votre KitBot, passez à la section [5](#) pour obtenir des informations sur la manière de tester votre robot et d'identifier d'autres améliorations potentielles à inclure dans votre processus de planification.

5 Itération – Amélioration des capacités existantes

L'itération est un élément essentiel du processus de conception technique et la conception d'un robot en Compétition de robotique *FIRST* n'est pas différente. Une fois que vous avez créé votre KitBot, vous souhaitez utiliser le temps restant avant ou entre vos tournois pour faire deux choses: vous entraîner et itérer. Pour plus d'informations sur la sélection et l'entraînement des pilotes, consultez « Guide to Selecting Drivers » et « Improving Driver Performance » sur la page [Technical Resources](#) dans la section Other Technical Resources.

Un processus simple d'itération est décrit ci-dessous, comprenant les étapes suivantes:

- Test
- Idéation
- Planification
- Mise en œuvre

5.1 Tests ciblés

La première étape d'un processus d'itération consiste à identifier les domaines à améliorer. Pour votre KitBot, nous vous recommandons de le faire via des tests ciblés. La majorité de ces tests consisteront à utiliser votre robot comme vous le prévoyez pendant un match, ce qui servira également d'entraînement aux pilotes. La seule exception concerne les tests limites. À condition que vous ayez suffisamment de temps (nous recommandons au moins une semaine) avant votre événement, vous pouvez faire les tests ciblés de votre robot dans des situations risquées telles que heurter des objets de manière agressive, acquérir trop de pièces de jeu, appuyer sur des boutons dans un ordre incorrect, etc. pour voir ce qui casse. Pendant que vous testez le robot, notez les réponses aux questions suivantes :

1. Qu'est-ce qui s'est cassé ? Comment ça s'est cassé ?
2. Que fait le robot de manière incohérente (par exemple, ne pas saisir des pièces de jeu, laisser tomber des pièces de jeu, rater des occasions de marquer, etc.) ?
3. Quelles tâches prennent beaucoup de temps aux pilotes à accomplir (par exemple, saisir des pièces de jeu, traverser le terrain, s'aligner pour marquer, relâcher la pièce de jeu, etc.) ?

Si vous disposez de plusieurs pièces de jeu, vous souhaitez peut-être effectuer une comparaison minutieuse des performances du robot avec chaque pièce de jeu pour voir s'il existe une variation. Ensuite, prenez une ou deux pièces de jeu et modifiez-les de la façon que vous pensez que ça pourrait se produire pendant le jeu (sur-gonflées ou sous-gonflées, déformées, éraflées ou déchirées, etc.). Répétez les tests avec le robot et documentez tout changement de comportement.

5.2 Remue-méninges sur les améliorations

Maintenant que vous avez identifié les domaines potentiels d'amélioration, l'étape suivante consiste à générer des idées pour améliorer ces faiblesses. Ce document divise le remue-méninges en deux parties, mécanique et logicielle/électrique. C'est pour rendre le processus un peu plus gérable, et parce

que ces éléments nécessitent souvent des ensembles de ressources différents, ils peuvent souvent être poursuivis en parallèle.

5.2.1 Remue-ménages sur les améliorations – Mécanique

Pour chacun des sujets d'amélioration potentiels identifiés lors de vos tests, réfléchissez à quelques modifications mécaniques potentielles qui pourraient améliorer les performances. Vous trouverez ci-dessous quelques questions pour vous mettre sur la piste dans chacune des trois catégories :

5.2.1.1 *Qu'est-ce qui s'est cassé ?*

1. Existe-t-il des moyens de modifier la géométrie de la pièce cassée pour la rendre plus solide ?
2. Existe-t-il des matériaux alternatifs que nous pourrions utiliser qui renforceraient la pièce avec la même géométrie ?
 - a. N'oubliez pas que plus fort ne signifie pas toujours plus rigide. Parfois, fabriquer une pièce à partir d'un matériau plus flexible (par exemple, changer une pièce en aluminium pour du polycarbonate) peut lui permettre d'absorber de l'énergie sans déformation permanente.
3. Voulons-nous réellement que la pièce soit plus solide ou voulons-nous fabriquer des pièces de rechange ?
 - a. Si la cause de l'échec semble anormale et peut se produire rarement dans un tournoi, voire jamais, vous pouvez laisser la pièce telle quelle, ou parfois même la rendre légèrement plus faible. Même si cela peut sembler contre-intuitif au premier abord, la conception d'un maillon faible connu et la préparation à son remplacement en cas de dommage peuvent aider à prévenir des dommages inattendus sur des pièces plus coûteuses ou plus difficiles à remplacer. Les goupilles de cisaillement mécaniques et les fusibles électriques sont quelques exemples de telles [pièces sacrificables](#).

5.2.1.2 *Qu'est-ce qui est incohérent/irrégulier ?*

1. Existe-t-il des moyens de modifier la géométrie des pièces pour tenter d'améliorer la régularité ?
 - a. Par exemple, l'ajout ou la suppression de contraintes ou de guides, l'augmentation ou la diminution de la compression sur une pièce de jeu, la suppression des espaces ou des fentes avec lesquels une pièce de jeu peut interagir par inadvertance, ou l'ajout de roues ou de rouleaux supplémentaires connectés à un système existant.
2. Y a-t-il des moyens de modifier les matériaux pour essayer d'améliorer la régularité ?
 - a. Les idées pour cela visent généralement à augmenter ou à réduire la flexibilité ou la friction.
3. Est-ce que des actionneurs supplémentaires pourraient augmenter la régularité ?
 - a. Par exemple, l'extension active d'un mécanisme pour saisir des pièces de jeu, l'ajout de loquets ou de portes supplémentaires dans le trajet d'une pièce de jeu, l'ajout d'un actionneur supplémentaire pour aider à garantir l'orientation des pièces de jeu dans un système, etc.

5.2.1.3 Qu'est-ce qui prend du temps ?

1. Existe-t-il des guides mécaniques qui peuvent être ajoutés pour accélérer la tâche des pilotes ?
 - a. Par exemple, des cales ou des entonnoirs qui interagissent avec des éléments de terrain, ou un marqueur physique sur un robot qui aide les pilotes à mieux s'aligner visuellement avec un élément du terrain.
2. Existe-t-il des moyens de rendre le robot plus résistant au désalignement ?
 - a. Par exemple, l'augmentation de la taille de l'ouverture pour l'acquisition de pièces de jeu ou la modification de la géométrie pour marquer à partir de différents alignements.
3. Le problème est-il le temps que prend le mécanisme pour terminer l'action ? Peut-on accélérer en changeant de ratio de vitesse ?
 - a. Si vous envisagez d'accélérer un mécanisme, assurez-vous de vérifier la quantité de courant requis dans sa configuration actuelle. Le courant évoluera de manière approximativement linéaire avec le ratio de vitesse, donc si l'accélération du mécanisme pousse le courant à des niveaux préoccupants, vous devrez peut-être également ajouter un autre moteur.

5.2.1.4 Des idées concrètes pour le KitBot 2025

Vous trouverez ci-dessous quelques suggestions spécifiques d'améliorations mécaniques potentielles pour le KitBot 2025. Sautez cette section si vous préférez vous en tenir aux idées que l'équipe développera.

1. **Ajouter un blindage pour empêcher la possession accidentelle de Corail** – Bien que le KitBot dispose de blindage pour empêcher les manières les plus probables pour le corail de rester coincé (lors de la collecte ou de la livraison), il a toujours des endroits où le corail pourrait rester coincé par inadvertance (échappé accidentellement d'un partenaire ou d'un adversaire, etc.) et qui compteraient comme votre seul corail autorisé à contrôler. Un blindage pourrait être ajouté à des endroits pour réduire cette probabilité.
2. **Comblent les espaces d'acquisition/livraison de Corail** - Pour simplifier les choses, le KitBot acquiert les Coraux par une rampe (piste) entièrement passive. Cela laisse un espace entre la rampe et le mur à cause de la contrainte pour la rampe de s'arrêter au bord du périmètre du châssis. Le KitBot livre également les Coraux depuis l'intérieur du périmètre du châssis, laissant un espace entre l'extrémité de la rampe (piste) et le Récif. Un mécanisme actionné pourrait être utilisé pour combler cet espace après le début du match, en prolongeant la rampe au-dessus des pare-chocs. Vous souhaiterez probablement vous arrêter au bord du pare-chocs ou légèrement avant pour permettre à celui-ci de continuer à absorber les impacts avec le mur ou le Récif.
3. **Alignement du Corail** – Avec la conception initiale du KitBot, il est possible que le Corail ne tombe pas dans le robot et ne repose pas complètement droit contre le rouleau, ce qui entraînera parfois une éjection irrégulière. Y a-t-il des moyens d'ajouter des éléments mécaniques supplémentaires (motorisés ou non) qui pourraient améliorer cela ?

5.2.2 Remue-méninges sur les améliorations – Électrique/Logiciel

Les améliorations électriques et logicielles sont regroupées ici car elles vont souvent de pair. L'amélioration du code d'un robot nécessite souvent l'intervention de capteurs pour permettre au logiciel de réagir à ce qui se passe. Tout comme le brainstorming mécanique, considérez chacun de vos sujets d'amélioration potentielle et réfléchissez à la manière dont les données du programme ou l'automatisation pourraient apporter une amélioration.

5.2.2.1 *Qu'est-ce qui s'est cassé ?*

1. La panne aurait-elle pu être évitée par des restrictions logicielles pour un mécanisme ?
 - a. Ces mesures sont le plus souvent mises en œuvre à l'aide d'interrupteurs de fin de course pour indiquer les limites de déplacement autorisées. Une alternative consiste à utiliser des « limites souples » pour les mécanismes qui ont déjà l'information de position absolue (c'est-à-dire que vous savez que votre mécanisme est à 30 degrés, vous pouvez donc l'empêcher par logiciel de se déplacer plus loin dans cette direction).
2. La panne aurait-elle pu être évitée avec des limites de courant ?
 - a. Les limites de courant peuvent aider à réduire la force maximale qu'un mécanisme peut produire et peuvent limiter la surchauffe du moteur si celui-ci fige en raison d'un blocage inattendu du mécanisme.
3. La panne aurait-elle pu être évitée par un verrouillage logiciel ?
 - a. Parfois, les robots ont des mécanismes qui ne doivent être actionnés que lorsque le robot est dans un certain état, sinon quelque chose pourrait se casser. Ces exclusions peuvent être appliquées dans le logiciel pour aider à empêcher les pilotes de faire une erreur et d'endommager le robot.

5.2.2.2 *Qu'est-ce qui est incohérent/irrégulier ?*

1. Les commandes sont-elles intuitives et évidentes ?
 - a. Si les pilotes font des erreurs, cela peut signifier que plus d'entraînement est nécessaire, mais cela peut également signifier que les commandes ne sont pas intuitives. Une fois que vous avez sélectionné vos pilotes, assurez-vous que les commandes ont du sens pour elles et eux. Ce qui peut être évident pour une personne peut être étrange pour une autre. Une mauvaise conception de commande peut également inclure l'utilisation d'un seul bouton pour basculer un état que les pilotes ne peuvent pas constater. Par exemple, si appuyer sur un bouton active la roue du lanceur et appuyer à nouveau dessus l'arrête, comment savoir si vous avez appuyé dessus et comment penser à l'éteindre ?
2. Existe-t-il des problèmes de timing que l'automatisation logicielle pourrait aider à prévenir ?
 - a. Parfois, l'incohérence/irrégularité vient du fait que les pilotes font quelque chose au mauvais moment. L'automatisation de ce timing par le logiciel peut aider en supprimant l'incohérence humaine du processus. L'éjection de Corail en est un exemple. Si le bouton est maintenu trop brièvement (de sorte que le rouleau cesse d'être alimenté

- avant que le Corail ne quitte complètement le robot), se passe-t-il quelque chose de différent si le bouton est maintenu plus longtemps ?
3. Existe-t-il des séquences qui pourraient être gérées automatiquement par le code ?
 - a. Parfois, pour effectuer une action, un robot doit actionner plusieurs mécanismes dans un ordre spécifique, avec ou sans timing spécifique. L'automatisation logicielle peut-elle être utilisée pour faire respecter cet ordre ou automatiser toute la séquence afin d'éviter les erreurs ?
 4. Existe-t-il des verrouillages logiciels qui pourraient aider à prévenir les problèmes ?
 - a. L'incohérence peut parfois provenir de l'exécution d'une action alors que le robot n'était pas dans le bon état (non aligné avec le terrain, mécanisme d'interaction mal préparé, etc.). Le code peut-il être utilisé pour détecter l'un de ces états incorrects et empêcher les pilotes de lancer l'action ?
 5. Le comportement du robot semble-t-il irrégulier même avec ce qui semble être des actions correctes des pilotes ?
 - a. Parfois, cette irrégularité est due à des changements dans le robot ou dans l'environnement, comme une tension de batterie différente ou un peu plus de friction lorsqu'une pièce s'use. L'ajout de capteurs et de contrôles logiciels peut aider à garantir que le robot fait la même chose à chaque fois.

5.2.2.3 *Qu'est-ce qui prend du temps ?*

1. L'alignement peut-il être automatisé ?
 - a. Le moyen le plus courant consiste à utiliser la détection visuelle pour détecter l'emplacement du robot sur le terrain et à l'utiliser pour diriger le robot vers l'emplacement souhaité, mais selon la tâche, d'autres capteurs tels que des télémètres ou des brise-faisceau peuvent également être utiles.
2. Pouvez-vous acheminer plus d'informations aux pilotes ?
 - a. Les pilotes peuvent faire de leur mieux avec leurs yeux pour voir ce qui se passe, mais le robot peut également aider en envoyant des signaux supplémentaires. La diffusion d'une caméra vers le poste de pilotage, l'affichage des états du robot sur le tableau de bord, l'utilisation de lumières sur le robot pour signaler l'état et l'ajout d'infos du contrôleur (le cas échéant) sont des exemples de moyens de fournir des informations sur l'état du robot aux pilotes.

5.2.2.4 *Des idées concrètes pour le KitBot 2025*

Vous trouverez ci-dessous quelques suggestions spécifiques pour une automatisation logicielle potentielle du KitBot 2025. Sautez cette section si vous préférez vous en tenir aux idées que l'équipe proposera.

1. **Boucle fermée de contrôle du roulement** – Soit en changeant le moteur alimentant la roue du lanceur avant pour un moteur avec un encodeur intégré (Venom, NEO/Vortex, Falcon, Kraken), ou en ajoutant un encodeur directement ou en démultipliant le moteur existant, vous pouvez passer du contrôle logiciel en boucle ouverte actuel (estimation de la vitesse de la roue en

fonction de la tension fournie) au contrôle en boucle fermée (contrôle précis de la vitesse de la roue en fonction du retour d'information de l'encodeur). Pour obtenir des informations sur la vitesse du capteur choisi, vous devrez consulter la documentation du moteur ou du capteur approprié. Pour utiliser ces informations afin de contrôler la vitesse des roues, consultez la section [Advanced Controls Introduction section of the WPILib documentation](#), en particulier l'article sur « Tuning a Flywheel Velocity Controller ».

2. **Détection de Corail** – Un interrupteur, un capteur de proximité ou un capteur de rupture de faisceau pourrait être ajouté au robot pour détecter quand un Corail est dans le robot. Vous pouvez utiliser ces informations pour fournir un retour d'information aux pilotes via des lumières, le tableau de bord ou la vibration du contrôleur (si vous utilisez un contrôleur avec vibration tel qu'un contrôleur Xbox).
3. **Alignement visuel** – Le KitBot peut être facilement aligné avec le Récif en raison de la grande ouverture, de la proximité de vos pilotes et de la possibilité de faire d'accoter les pare-chocs contre le Récif pour ajuster l'espacement. L'alignement avec la Station des Coraux peut être plus difficile en raison de l'angle avec vos pilotes, mais le Joueur humain a probablement beaucoup de place pour se déplacer et s'ajuster. L'ajout d'une caméra pour traiter les [AprilTags](#) peut être utilisé pour s'aligner sur l'un ou l'autre des emplacements plus rapidement et plus précisément. Vous pouvez le faire en ajoutant une webcam branchée sur le roboRIO ou en ajoutant un système de vision externe comme un [Limelight](#), un [RaspberryPi avec WPILibPi](#) ou une [carte coprocesseur avec PhotonVision](#). Une fois que vous disposez des données de l'AprilTag, vous pouvez les utiliser pour aligner le robot en :
 - a. Demandez aux pilotes de positionner le robot près de la cible et de l'orienter approximativement dans la bonne direction de manière à être sûr que l'AprilTag est bien en vue.
 - b. Demander aux pilotes d'appuyer sur un bouton et de le maintenir enfoncé pour lancer le ciblage optique.
 - c. Pendant que le bouton est maintenu, contrôlez la rotation de la base motrice à l'aide de l'emplacement de l'AprilTag. Vous pouvez probablement utiliser un contrôle proportionnel simple pour vous rapprocher suffisamment. Comparez l'emplacement de l'AprilTag au centre de l'image, multipliez le résultat par une constante que vous réglez et appliquez le résultat à la rotation de la base pilotable (appliquez le résultat comme valeur de rotation de la fonction de déplacement en mode arcade ou la moitié de la valeur en signes opposés de chaque côté d'une fonction motrice en mode tank).
 - d. Permettez aux pilotes de garder le contrôle du système de propulsion de la base pilotable soit en continuant à appliquer la valeur du joystick au mouvement avant/arrière d'une fonction de conduite d'arcade, soit en ajoutant la valeur du joystick à la valeur AprilTag pour une conduite Tank (pour la conduite Tank, vous devrez également détecter si l'une des valeurs dépasse 1 et réduire les deux côtés si c'est le cas).
 - e. Les pilotes peuvent alors conduire le robot vers la cible tandis que le robot reste pointé vers l'AprilTag. Lorsque les pare-chocs atteignent la cible, le centre du robot doit être approximativement aligné avec le centre de la cible. Il se peut que le robot ne soit pas

encore aligné contre le mur. Continuez à pousser le robot vers la cible et les pare-chocs devraient s'adosser contre le mur. Vous pouvez également potentiellement relâcher le bouton d'alignement optique et tourner légèrement dans la direction appropriée pour faciliter cette partie du processus.

- f. Si vous souhaitez cibler autre chose que le centre des AprilTags (par exemple, pour s'aligner de telle sorte que 2 Coraux reposent côte à côte sur le Récif), un peu plus de complexité sera nécessaire.

Ou, pour une utilisation plus avancée, vous pouvez utiliser les AprilTags pour obtenir des données de position complètes pour votre robot et l'aligner ou planifier un chemin n'importe où sur le terrain !

5.3 Prochaines étapes

Si vous avez envisagé des améliorations avant même de construire votre robot, décidez si vous souhaitez réfléchir à nouveau sur ce sujet pour l'ajouter à votre pool d'itérations, ou si vous avez une liste d'idées restantes à reconsidérer. Si vous n'avez pas encore envisagé d'améliorations, rassemblez votre liste complète d'idées issues des deux exercices et passez à la section [6](#) pour obtenir des conseils sur la façon de les prioriser.

6 Tri, filtre et sélection – Transformer les idées en plan

Si vous envisagez des améliorations avant de construire votre KitBot, vous n'avez peut-être pas encore d'idées d'itération. Procédez à cet exercice avec uniquement vos améliorations. Vous pourrez toujours revenir et le refaire avec les idées d'amélioration restantes ainsi que de nouvelles idées d'itération après avoir construit et testé votre robot. Si vous avez déjà construit votre robot, nous vous recommandons de combiner les idées de nouvelles capacités avec les idées d'itération dans un seul exercice de tri sélectif.

Transformer votre liste d'améliorations possibles en une liste de priorités sur lesquelles agir est certes un exercice difficile qui implique de considérer de nombreux facteurs. Il existe de nombreuses façons d'aborder l'équilibre entre l'avantage perçu de chaque changement et la difficulté de sa mise en œuvre et les ressources d'équipe qui peuvent être nécessaires pour le mettre en œuvre. Une méthode possible est détaillée ci-dessous.

6.1 Prioriser

En vous appuyant sur votre analyse du jeu et votre vision du rôle de votre robot au sein des alliances, priorisez vos améliorations et perfectionnements potentiels du plus impactant au moins impactant. Ne vous souciez pas encore de ce qui est facile et de ce qui est difficile, ou des ressources requises, créez simplement une « liste de rêve » ordonnée d'améliorations du robot. Un exemple est fourni ci-dessous :

1. Nouvelle capacité innovante
2. Un truc automatique qui aide à obtenir des RP
3. Un truc de fin de partie qui aide à obtenir des RP
4. Ajustement pour améliorer la précision du marquage
5. Fabriquer des pièces de rechange du composant X

6.2 Analyse – Estimation des ressources et de l'ampleur

Dans cette section, vous allez examiner votre liste et évaluer les ressources nécessaires pour réaliser chaque amélioration. Plutôt que d'attribuer simplement un concept ou un numéro à la difficulté d'une idée, essayez de créer quelques catégories différentes qui représentent les différentes ressources qui peuvent être nécessaires. Certaines de ces catégories peuvent représenter des contraintes de ressources physiques telles que le matériel disponible, la disponibilité des machines ou le budget pour acheter de nouveaux composants/matériaux et certaines peuvent représenter des ressources plus abstraites comme le temps total/la capacité des étudiants.es et des mentors dans un aspect spécifique de l'équipe.

Des catégories possibles sont: conception, fabrication, logiciel, matériaux/coût. Pour chaque idée, évaluez le coût en ressources pour chacune de ces catégories et attribuez une valeur de 0 à 5 (n'hésitez pas à remplacer par l'échelle de votre choix), 0 correspondant à aucune ressource requise et 5 à un besoin en ressources très important. Pour prioriser, utilisez le rang que vous avez créé à partir des [étapes 6.1](#), 1 à N (ce n'est pas grave si c'est une échelle différente des valeurs de catégorie).

Pour un nouveau mécanisme, l'idéal serait de proposer des chiffres approximatifs sans avoir besoin de réfléchir et de sélectionner le mécanisme spécifique que vous utiliseriez pour accomplir la tâche, mais pour certaines équipes, cela peut être plus difficile. Vous devrez peut-être faire un détour à ce stade pour réfléchir ou prototyper des idées de mécanismes spécifiques afin d'évaluer leurs besoins en ressources, voyez la section [6.5](#) pour des ressources supplémentaires.

Un exemple sans lien au jeu et au robot est fourni à Tableau 1.

Tableau 1: Exemple de ressources d'amélioration

Amélioration	Priorité	Conception	Fabrication	Code	Matériaux/Cout
Nouvelle capacité innovante	1	5	3	3	3
Un truc automatique qui aide à obtenir des RP	2	0	1	3	0
Un truc de fin de partie qui aide à obtenir des RP	3	3	2	1	1
Ajustement pour améliorer la précision du marquage	4	1	3	0	1
Fabriquer des pièces de rechange du composant X	5	0	2	0	2

6.3 Analyse – Établissez votre « budget »

Maintenant que vous avez défini les coûts relatifs des ressources pour vos idées, vous devez établir un « budget » (au sens plus large que monétaire) décrivant la capacité de votre équipe dans chaque catégorie. Pour chacune de vos catégories, en utilisant les coûts qui vous sont attribués comme référence d'échelle, tenez compte des ressources de votre équipe dans le temps entre maintenant et le tournoi. Assurez-vous de prendre en compte le temps dont vous aurez besoin juste avant l'événement pour tester et pratiquer les améliorations, puis préparer le transport de votre équipement. Attribuez un budget global à chaque catégorie qui représente vos ressources disponibles dans cette catégorie entre maintenant et votre prochain tournoi.

À titre d'exemple, considérons une équipe composée de 2 élèves en CAO et d'un mentor en CAO, de seulement 1 ou 2 élèves en programmation sans mentor associé, d'une entreprise qui aide à la fabrication de pièces en plus de l'équipement de l'atelier de l'équipe et d'un stock substantiel de matières premières et de pièces COTS. Cette équipe pourrait décrire son budget pour ces catégories ainsi :

- Conception/design – 5
- Programmation – 3
- Fabrication – 10

- Matériaux - 10

6.4 Planifier

Maintenant que vous avez un budget et quelques coûts, il est temps de les rassembler dans une planification des idées à poursuivre. Une approche consiste simplement à parcourir la liste des priorités jusqu'à ce que le budget d'une catégorie soit épuisé, puis à continuer en ajoutant tous des éléments qui affectent uniquement les catégories avec un budget restant.

Une approche plus complexe consiste à demander à des individus ou à de petits groupes d'élaborer des planifications partielles, puis d'évaluer ces planifications en équipe et de sélectionner ce qui, selon le groupe, donnera le meilleur robot. Cela peut parfois conduire à une utilisation plus optimale des ressources.

Reprenons l'exemple du tableau ci-dessus avec notre équipe théorique qui a défini son budget de ressources de conception à 5 points (nous allons ignorer leur budget dans les autres catégories pour plus de simplicité). En utilisant la première approche, cette équipe choisirait d'ajouter une « Nouvelle capacité innovante », en épuisant l'intégralité de son budget de conception, puis « Fabriquer des pièces de rechange du composant X ».

Tableau 2: Exemple de planification d'amélioration – Option 1

Amélioration	Priorité	Conception	Fabrication	Code	Matériaux/Coût
Nouvelle capacité innovante	1	5	3	3	3
Fabriquer des pièces de rechange du composant X	5	0	2	0	2

Ce plan apporte deux améliorations et utilise 5 points de ressources de fabrication, 3 de programmation et 5 de matériaux.

En utilisant la deuxième approche, quelqu'un pourrait plutôt suggérer à l'équipe d'ignorer cette première amélioration et de répartir ses points de budget de conception sur les trois améliorations suivantes.

Tableau 3: Exemple de planification d'amélioration – Option 2

Amélioration	Priorité	Conception	Fabrication	Code	Matériaux/Coût
Un truc automatique qui aide à obtenir des RP	2	1	1	3	0
Un truc de fin de partie qui aide à obtenir des RP	3	3	2	1	1
Ajustement pour améliorer la précision du marquage	4	1	3	0	1

Fabriquer des pièces de rechange du composant X	5	0	2	0	2
---	---	---	---	---	---

Bien que cette planification ne prenne pas en compte l'élément ayant la priorité individuelle la plus élevée, il apporte 4 améliorations au lieu de deux et peut mieux utiliser les ressources de fabrication (8 contre 5) et logicielles (4 contre 1) disponibles. L'équipe devra ensuite évaluer la valeur de ces 4 améliorations par rapport aux 2 de l'autre plan et décider quelle approche produira le meilleur robot pour atteindre les objectifs de l'équipe.

6.5 Prochaines étapes

6.5.1 Améliorations – Nouveaux mécanismes

Pour ajouter de nouveaux mécanismes, l'étape suivante consiste à commencer à réfléchir aux implémentations possibles, puis à créer des prototypes pour déterminer une direction. Les ressources suivantes (voa) de la page [Technical Resources](#), sous Mechanical Resources->General, peuvent servir de point de départ utile :

- "Prototyping 101"
- "Prototyping Worksheet"
- "Mechanisms Worksheet"
- "NASA RAP Robotics Design Guide"

6.5.2 Améliorations – Capacités logicielles

Pour ajouter de nouvelles fonctionnalités logicielles, consultez "Control System and Programming Documentation" ainsi que "Programming 101" dans la section Software/Electrical Resources->Software de la page [Technical Resources](#).

6.5.3 Plan du projet

La dernière chose que vous souhaitez probablement faire avant de vous lancer et de commencer le travail est d'esquisser un plan de projet approximatif. Cela peut être aussi simple ou aussi détaillé que vous le souhaitez, mais vous souhaitez probablement au moins quelques étapes importantes entre maintenant et votre prochain tournoi. Ce plan vous aidera à vérifier si vous êtes sur la bonne voie et vous permettra d'ajouter des ressources (comme du temps de rencontre) ou de réduire les fonctionnalités avant de risquer de vous présenter en tournoi avec un robot inachevé ou non testé. Abandonner une ou deux fonctionnalités pour vous assurer d'avoir le temps de tester et de vous entraîner sera probablement payant à la fin.

7 Annexe A – Exemple d'analyse de jeu de base, 2023

7.1 Exemple - Liste des tâches du robot

Vous trouverez ci-dessous un exemple de liste de tâches du robot (voir la section [4.1.1](#)) pour le jeu de Compétition de robotique **FIRST 2023 CHARGED UPSM** présenté par Haas. Cette liste a été établie avec le recul, les équipes individuelles peuvent avoir ajouté plus ou moins de tâches sur leurs propres listes.

Façons pour un robot de marquer :

- Cône supérieur
- Cône du milieu
- Cône inférieur
- Cube supérieur
- Cube du milieu
- Cube inférieur
- Pont d'équilibre

Détails supplémentaires du pointage :

- Pont d'équilibre
 - Équilibre en roulant sur le pont depuis le bord large
 - Équilibre du côté étroit d'un pont déjà équilibré

Façons d'acquérir des pièces de jeu :

- Sous-station double
 - Cubes
 - Cônes renversés
 - Cônes debout
- Sous-station simple directe
 - Cubes
 - Cônes renversés
 - Cônes debout
- Sol
 - Cubes
 - Cônes renversés
 - Pointe d'abord
 - Bride d'abord
 - Cônes debout

L'acquisition de pièces de jeu du côté opposé du robot par rapport à celui où elles sont marquées peut accélérer les trajets, ce qui est particulièrement important en mode Auto (cela peut être moins important pour les bases pilotables qui peuvent tourner tout en se déplaçant, comme le mecanum ou le swerve).

Autres tâches :

- Donner des pièces de jeu aux partenaires ?

Tâches défensives :

- Acquérir des pièces de jeu adverses ? (Possibilité de saisir sans extension pour saisir légalement des pièces dans la ZONE DE CHARGEMENT des adversaires)

7.2 Exemple – Comment obtenir un bon classement

Vous trouverez ci-dessous un exemple de la manière de bien se classer dans le jeu de la Compétition de robotique *FIRST 2023 CHARGED UP* présenté par Haas.

1. Outre les matchs gagnés, comment les alliances peuvent-elles gagner des points de classement ?
 - a. RP de lien – Marquer au moins 5 liens ou au moins 4 liens avec coopération (cela fait référence à la valeur au lancement, pas à la valeur modifiée du championnat)
 - b. RP de la station de recharge – Obtenez au moins 26 points de station de recharge
2. Pour chacun de ces RP, quelles sont les capacités du robot nécessaires...
 - a. pour maximiser votre capacité à permettre à vos alliances de gagner des RP ?
 - i. Marquer un total de 15 pièces de jeu mélangées entre les deux types dans au moins la rangée inférieure et médiane pour créer 5 liens
 - ii. Équilibrer la station de recharge en mode Auto afin qu'un seul équilibre de 2 robots soit nécessaire en fin de partie pour sécuriser le RP
 - b. pour obtenir les RP en jouant avec 2 autres robots comme le vôtre ?
 - i. Marquez environ 5 pièces de jeu mélangées au total des deux types dans au moins la rangée inférieure et médiane pour aider à créer 5 liens
 - ii. Équilibrer la station de recharge en mode Auto afin qu'un seul équilibre de 2 robots soit nécessaire en fin de partie pour sécuriser le RP
 - c. pour faire une contribution minimale au RP de votre alliance ?
 - i. Marquer environ 3 à 5 pièces de jeu au total dans au moins la rangée inférieure pour contribuer aux liens.
 - ii. Aider à équilibrer la station de recharge dans le jeu final pour contribuer aux points de la station de recharge.
3. Quels sont les 1^{er} et 2^e critères de bris d'égalité après les RP (il est généralement peu probable qu'il y ait une égalité au-delà de cela et très peu probable qu'une telle égalité décide de plus d'un rang ou deux au maximum) ? Ces éléments ne devraient pas avoir une influence majeure dans le choix des capacités, mais peuvent fournir un coup de pouce en cas de décision difficile entre deux capacités lors de la définition des priorités.
 - a. Points de Station de recharge
 - b. Points Auto

7.3 Exemple – Comment gagner des matchs

Vous trouverez ci-dessous un exemple d'analyse points/temps pour le jeu 2023. Pour simplifier, cette analyse ne porte que sur la période Teleop et ignore la période Auto. On suppose également que les pièces de jeu de pointage contribuent aux liens, ce qui peut ne pas être une hypothèse valable si vous choisissez une stratégie Cube uniquement. Les estimations de temps et le taux de réussite utilisées sont des approximations et représentent un robot essayant de se qualifier pour les éliminatoires (~50^e percentile) lors d'un événement régional ou de district moyen.

Tableau 4: Exemple d'analyse des points vs temps

Actions de marquage	Points	Temps	Taux de réussite	Pts/sec
Cube inférieur	2 + 5/3 (Lien)	22	95	.158
Cône inférieur	2 + 5/3 (Lien)	30	95	.116
Cube médian	3 + 5/3 (Lien)	27	90	.122
Cône médian	3 + 5/3 (Lien)	35	85	.088
Cube supérieur	5 + 5/3 (Lien)	30	85	.160
Cône supérieur	5 + 5/3 (Lien)	38	80	.119
Équilibre à 1 robot	10	15	90	.6
Équilibre à 2 robots	20	20	85	.85
Équilibre à 3 robots	30	25	80	.96

Ensuite, quelques réponses approximatives aux questions sur les alliances gagnantes.

1. À quoi ressemble le match pour une alliance gagnante moyenne en qualifications ?

3x Mobilité = 9 points

Connecté et engagé Auto = 10 points

3x pièces de jeu Auto = +3 points

2x Liens Bas = $2*5 + 6*2 = 22$ Points

2x Liens Hauts = $2*5 + 6*5 = 40$ Points

Connecté et engagé x2 robots = 20 points

Stationnement = 2 points

Total = 106 points

2. À quoi ressemble le match pour une alliance qui gagnerait environ 80 % des matchs de qualification ?

3x Mobilité = 9 points
Connecté et engagé Auto = 10 points
4x Pièces de jeu Auto = +4 points
3x Liens bas = $3*5 + 9*2 = 33$ points
2xLiens hauts = $2*5 + 6*5 = 40$ points
Connecté et engagé x2 Robots = 20 points
Stationnement = 2 points
Total = 118 points

3. À quoi ressemble le match pour une alliance qui peut gagner un match éliminatoire ?

3x Mobilité = 9 points
Connecté et engagé Auto = 10 points
4x Pièces de jeu Auto = +4 points
3x Liens bas = $3*5 + 9*2 = 33$ points
2xLiens hauts = $2*5 + 6*5 = 40$ points
Connecté et engagé x2 Robots = 20 points
Stationnement = 2 points
Total = 118 points

4. À quoi ressemble le match pour une alliance qui remporte le tournoi ?

3x Mobilité = 9 points
Connecté et engagé Auto = 10 points
5x pièces de jeu Auto = +5 points
3x Liens Bas = $3*5 + 9*2 = 33$ Points
1x Lien médian = $5 + 3*3 = 14$ points
3x Liens Hauts = $3*5 + 9*5 = 60$ Points
Connectés et engagés x3 robots = 30 points
Stationnement = 2 points
Total = 162 points

Enfin, en prenant comme objectif une équipe qui essaie de se qualifier pour les séries éliminatoires, voici quelques réponses approximatives sur la contribution possible de l'équipe:

1. Combien de points essayez-vous de contribuer pour l'alliance ?
~30-40 points
2. Y a-t-il des tâches spécifiques dans ces alliances auxquelles vous pensez devoir contribuer ?
Connecté et engagé en mode Auto
3. Y a-t-il des tâches spécifiques que vous souhaitez confier à vos partenaires d'alliance ?
Non

7.4 Exemple - Améliorations

Supposons que le KitBot 2023 a été conçu pour livrer uniquement des Cubes bas. En considérant le total de points et les tâches identifiées ci-dessus, l'ajout d'une routine autonome "Dock and Engage" permettrait au robot de réaliser les tâches et les totaux de points identifiés. En regardant le tableau des points/temps, la prochaine capacité à envisager d'ajouter serait probablement les Cubes médians. C'est presque aussi efficace que les cubes bas (et pourrait être meilleur si cela pouvait être fait aussi rapidement que les cubes bas) et donne à l'équipe 3 opportunités de marquer supplémentaires.

Unofficial