

Projet « Yakindu Like »

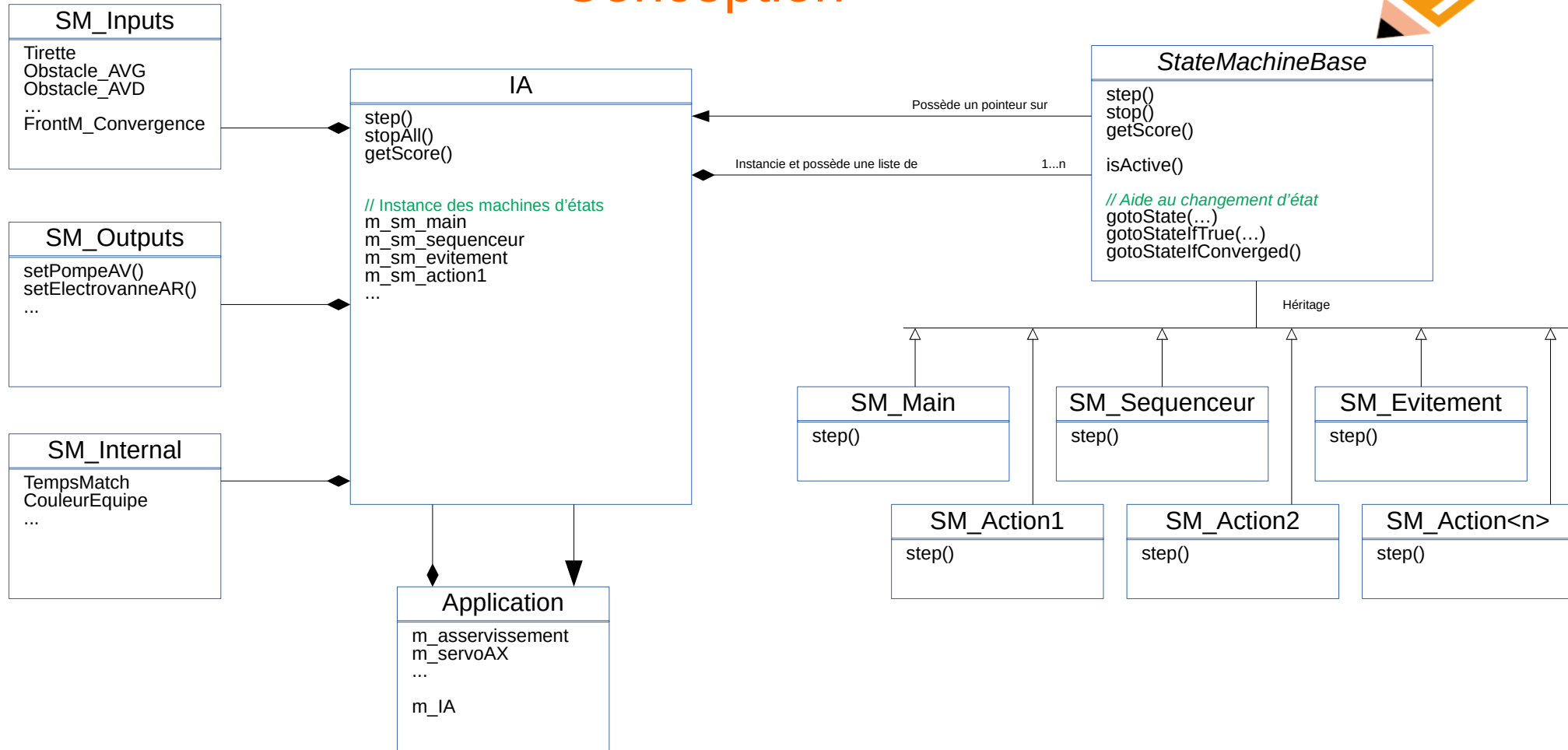


Cahier des charges



- Pouvoir se répartir facilement le développement de la stratégie entre plusieurs personnes
 - Éviter les merges à 3h du mat la tête dans le sac
 - Réutilisation du code commun entre GROSBOT et MINIBOT
 - Permettre le test de tout ou partie de la stratégie en simulation sous Labotbox
 - Permettre d'activer un bout de stratégie sans le reste
 - Mettre au point/tester un petit bout de stratégie indépendamment du reste en réel ou en simu
 - Fournir une API facilitant le développement
 - Avoir une ossature et API claire facilitant la génération automatique de code
-

Conception



Concept



- SM = State Machine
 - Une SM a accès à toute l'application (asservissement, servo, ...) réelle ou simulée sans faire la différence
 - Une SM calcule son score
 - Une SM peut activer ou arrêter une autre SM
 - Une SM indique si elle a effectué le job qui lui était confié
-

PROPOSITION TECHNIQUE



- Avoir une notion de « missions ». Une mission est une machine d'état particulière car elle possède :
 - Une coordonnée à atteindre sur le terrain (point d'intérêt)
 - Un score maximum atteignable si la mission réussit
 - Une priorité d'exécution
 - Exemple de mission 2020 : « activer le phare », « Récupérer les verres au distributeur », ...
 - Avoir des machines d'états qui peuvent s'exécuter en parallèle
 - Déployer le pavillon pendant une mission en cours d'exécution
 - Trier les objets à l'intérieur du robot pendant le chemin du retour
 - Avoir un séquenceur permettant de donner « l'intelligence », c'est à dire choisir la mission la plus pertinente à exécuter suivant certains critères (*priorité, heure du match, point d'intérêt le plus proche du robot, score possible, ...*)
 - Le choix de l'algo peut être fixe pour toute la durée du match ou être modulé pendant le match (la première minute, on donne la priorité aux missions qui rapportent le plus et ensuite aux missions qui sont dans notre camp).
 - Possibilité de debug pour savoir quelle SM a été activée/stoppée
-

PROPOSITION TECHNIQUE SEQUENCEUR



- Le séquenceur, c'est ce qui donne le côté « intelligence » au robot. En fonction des événements du match (détections d'obstacle), le séquençage des missions pourra être différent.
 - Le séquenceur peut donner un caractère aléatoire au match
 - Il permet de sélectionner automatiquement une mission qui n'est pas encore achevée lorsque la précédente est terminée.
 - Le critère de sélection de la prochaine mission est configurable avant le match par nous ou pendant le match par le séquenceur lui-même.
 - Possibilité de le configurer en mode basique :
 - Critère de sélection : la priorité de la plus élevée
 - Dans ce cas, la prochaine mission à exécuter est celle qui a la priorité la plus élevée et qui n'est pas encore exécutée.
 - Les priorités de chaque mission sont fixées par nous avant le match
-

PROPOSITION TECHNIQUE EVITEMENT



- Une mission peut être ou non interrompue par un évitement d'obstacle.
- En fin d'évitement, le comportement du séquenceur est configurable :
 - Reprendre la mission en cours juste avant la détection d'obstacle
 - Choisir la mission la plus pertinente en fonction d'un critère



APPLICATION COUPE 2020



- Les missions :
 - Activer le phare
 - Aller chercher les bouées au distributeur
 - Récupérer 2 bouées vers la zone de départ
 - Récupérer 4 bouées en chemin
 - Déposer les bouées dans le port
 - Arriver à bon port

- Les autres SM :
 - Autotest
 - Déployer le pavillon
 - Évitement d'obstacle
 - Rechercher point d'arrivée Nord / Sud
 -

