

Rapport de réunion

SASIAE

6 novembre 2013

Résumé

Compte rendu de la réunion du mercredi 6 novembre 2013 faite de 11h20 à 12h20, un peu dans l'urgence car la semaine suivante M. Rollet sera absent.

Antoine Rollet : Le cahier des charges doit contenir les besoins mais pas forcément les aspects réalisations internes. Mais c'est bien d'avoir l'architecture générale pour se poser les bonnes questions.

Loïc : Abstraction : le robot capte des valeurs et les renvoie. C'est l'API qui définit les possibilités des capteurs. Définition des différents capteurs : GP2, Lidar, Balises, IL EN MANQUE : Incrémentaux, accéléromètres, gyroscopes...) Il est préférable que tout les capteurs que Eirbot peut être amené à utiliser soient implémentés. *Rqe (Hugo) : Est-ce que ce serait pertinent d'écrire un petit document sur les différents capteurs ?*

Antoine Rollet : Un capteur est un capteur. Bien les spécifier et considérer leurs rôles. Attention, il faut bien **distinguer** les deux entités Client et Equipe de développeurs! **A faire :**

- API : Bilan de l'existant et modifications attendues.
- Capteurs : trouver les points communs, il faut les abstraire!

Benoît : Classer les capteurs en fonction du type de renvoi : Tout ou rien, Analogique, Numérique,...

problème : du mal à envisager des solution globales, pas spé à la partie (API, sim, Gui,...)

Antoine Rollet : : Bien imaginer, envisager les solutions globales

problème : qui est l'utilisateur ?

Antoine Rollet : : Le codeur est-il réellement différent de l'utilisateur du simulateur ?

Loïc : : Oui les deux sont différent, car il se peut très bien que celui qui utilise le simulateur ne fasse pas parti des développeurs du code robot. Comme il est possible qu'un développeur n'utilise pas le simulateur. De toute façon pour écrire le code robot, sans tests de simulation, il n'aura besoin que de l'API.

discussion générale : Différences entre utilisateurs :

- le codeur utilise seulement l'API;
- le testeur utilise l'interface + l'API, scénarios.

Les capteurs sont positionnés dans le robot et configurés dans l'API.

Hugo : Il est donc nécessaire de faire deux diagrammes fonctionnels, un par type d'utilisateur

Antoine Rollet : : Effectivement, l'API est à part du simulateur car elle définit l'implémentation du code en fonction du besoin (robot)

Les fonctionnalités de très haut niveau.

C'est GUI qui les donne. La création du code est séparée. On charge l'exécutable

Loïc : : On charge et on compile le code.

Encore les capteurs et le rôle du simulateur et de l'API.

Antoine Rollet : Il nous encourage à aller voir Aymeric Vincent pour séparer le type capteur et son instantiation.

Pour l'instant les capteurs sont pas super bien définis dans l'aversive.

En ce qui concerne le fichier de description physique du robot. Le simulateur devra comprendre au préalable différents types de robot. Ou alors on chargera dans le simulateur un type de Robot. On peut signifier si tout les éléments sont branchés ou pas ! Il faut réfléchir à la manière d'afficher le truc.

Discussion sur l'initialisation intéressante mais trop longue 12 :03 ... On abrège.

Antoine Rollet : aime bien le schéma fait par J-R.

Il faut faire des schémas, avoir les bonnes discussions et VITE, ECRIRE LA DOCUMENTATION ! Pas de démarches bidon, juste de l'utilitaire. Diagramme de classe pas fonctionnel Diagramme fast, Séquence, cas d'utilisation, pieuvre, SADT-A0 ?

Dans les fonctionnalités qu'on avait écrites pour l'API « Scinder le code en trois parties », ce n'est pas fonctionnel. C'est important de le décrire, bien sur, mais non fonctionnel. Changer la couleur des diagrammes pour afficher les besoins non fonctionnel ce n'est pas la meilleure manière.

Eviter d'annoncer des trucs infaisables. Il faut donc faire une rapide étude de faisabilité. Il faudra en faire une plus poussée une fois les besoins fonctionnels et non fonctionnels parfaitement établis.

Penser au petit trucs (table,...) les informations sont très importantes.

Guider un robot à la souris ?

Scénarios plus génériques ?

Scénarios scriptés ?

Possibilités ?

Fin de séance, reste à dégrossir

Conclusion

Il reste beaucoup à faire avant d'attaquer à proprement parler le développement. Il semble que nous avons choisi un cycle en V, il faut donc s'y tenir. Ce serait bien d'avoir une ébauche du cahier des charges en y mettant aussi les diagrammes fonctionnels et non fonctionnels, avant le rendez-vous pris avec Aymeric Vincent le mercredi 13 novembre 2013 à 13H30 dans son bureau (213).