

Cahier des charges

Extensions XBMC



Nicolas Afonso
Jean-François Bianco
Pierre Lartigue

XBMC_ricm5@googlegroups.com
Février 2014

Elisa Martinez
Rebecca Poustis

Table des matières

I.	Présentation du projet	3
	L'équipe projet	3
	Contexte et description.....	3
	Où	3
	Quand.....	3
	Comment.....	3
	Pourquoi.....	3
	Combien	3
II.	Tâches	4
	Liste des tâches avec ordre de priorité	4
	Etude de l'existant.....	4
III.	Spécifications fonctionnelles	7
	Par tâches.....	7
IV.	Gestion de projet	8
	Méthodologie utilisée	8
	Planning prévisionnel	8

Présentation du projet

L'équipe projet

Les membres de l'équipe sont :

- Pierre Lartigue (option Systèmes et réseau)
- Nicolas Afonso (option Systèmes et réseau)
- Jean-François Bianco (option Systèmes et réseau)
- Elisa Martinez (option Multimédia)
- Rebecca Poustis (option Multimédia)

Contexte et description

XBMC (abréviation de Xbox Media Center) est un media center open source (licence GPL). Le logiciel est disponible sur Linux, Windows, OSX, iOS et Android. L'utilisateur peut donc gérer ses médias, les consulter depuis n'importe quel device. L'expérience XBMC peut être améliorée avec des addons (intégration de Twitter, Facebook, Youtube, télécommande, services supplémentaires ...). Le but de ce projet sera de créer de nouveaux addons.

Où

L'utilisateur sera amené à utiliser ce médiacentre depuis son domicile. Par exemple, sur son canapé il annonce à voix haute une commande vocale pour lancer la musique (projet d'extension de commande vocale de XBMC), consulte une vidéo sur sa TV ainsi connectée qui possède une illumination « intelligente » (projet d'extension ampoules hue).

Quand

L'utilisateur peut utiliser XBMC à n'importe quel moment de la journée, lorsqu'il désire contrôler ou accéder à ses médias.

Comment

Pour l'instant, l'utilisation se fera sous la forme d'une interaction avec une souris/un clavier, un mobile, une tablette ou encore par commande vocale.

Pourquoi

Les évolutions technologiques font que l'on peut aujourd'hui avoir une meilleure interaction / gestion de ses médias et donc ne pas avoir à se déplacer pour faire une action rapide et fréquente (changer la musique en cours de lecture n'importe où dans sa maison par exemple).

Combien

Projet libre et gratuit (open source). Néanmoins un certain nombre de matériel est nécessaire pour mener à bien ce projet (développement et expérimentations).

Tâches

Liste des tâches avec ordre de priorité

Ordre de priorité	Nom de la tâche	Description
1	Hue	Récupérer l'image et régler les LEDs pour faire la couleur dominante
2	OpenHab	Description de l'habitat pour naviguer dans l'interface
3	XBMC Reflexive Remote Dynamic	Rajouter la navigation pour tous les plugins dans l'interface mobile
4	Intégration caméras de surveillance	Afficher les flux de la caméra
5	Intégration NFC	Associer un objet à un média
6	Commande vocale	Faire des recherches sur des mots clefs
7	Reconnaissance visage	Avoir plusieurs profils utilisateurs (Famille, parents, enfant)
8	Follow me	Suivre le déplacement/activité de l'utilisateur
9	Flux d'actualité twitter	Suivre le flux directement sur la télé
10	Intégration VLC	Pour la lecture d'un média diffusé par XBMC
11	UPnP low-power	Economie d'énergie
12	Friend To Friend	Partage de XBMC
13	Réservation de ressources	Préservation des ressources (CPU, réseau) pour XBMC malgré des applications qui tourneraient en parallèle

Etude de l'existant

- **XBMC Reflexive Remote Dynamic remote control for XBMC :**
 - Remote : existe déjà pour Android, Windows Phone, iOS (peut être facile à utiliser : http://wiki.xbmc.org/index.php?title=Main_Page)
Code source pour Android : <https://github.com/freezy/android-xbmcremote>
 - Module dans le noyau : conception modulaire - modules indépendants effectués par des bibliothèques de code localisés / isolées sans dépendance
Code source XBMC : <https://github.com/xbmc/xbmc>
 - Langages : Python, android
- **Hue :**
 - Script Ambient light control pad de xbmc
 - Philips Hue API
 - Code de Shadi pour communiquer directement avec le pont Philips. Il reste l'intégration avec XBMC.

- Présentation ampoules sur apple store :
<http://store.apple.com/fr/product/HA779ZM/A/ampoules-hue-de-philips-kit-de-d%C3%A9part>
- Présentation site de philips : <http://www.meethue.com/fr-US>
- Doc pour les dev : <http://developers.meethue.com/>

- Serveur DHCP :
 - <http://sourceforge.net/projects/dhcpserver/>
 - <http://binaire-life.fr/win/windows-xpvista7-installer-et-configurer-un-serveur-dhcp/>
- Algo couleur dominante :
 - JS : <http://jsfiddle.net/xLF38/>
 - Java :
<https://hg.mozilla.org/mozillacentral/file/afb7995ef276/mobile/android/base/gfx/BitmapUtils.java>

- **Commande vocale :**
 - L'utilisation de commandes vocales dans XBMC possible grâce à l'API Voice Recognition de Google qui marche sur navigateur + possibilité android. Egalement VoxCommando qui a l'air plus fiable, plus adapté à xbmc et facile à intégrer
 - <http://voxcommando.com/>
 - <http://www.javacodegeeks.com/2012/08/android-voice-recognition-tutorial.html>
 - <https://www.google.com/intl/fr/chrome/demos/speech.html>

- **Reconnaissance visage :**
 - SDK (demander de l'argent à Didier pour l'acheter !) :
<http://www.neurotechnology.com/verilook.html>
 - Liste d'API et autres librairies : <http://blog.mashape.com/post/53379410412/list-of-50-face-detection-recognition-apis>
 - <http://api.lambdal.com/docs/start> api de détection de visage qu'on peut utiliser pour identifier une personne parmi un album de personnes pré-enregistrées

- **Intégration avec OpenHAB :**
 - Réaliser le plugin XBMC
 - Doc/Tuto pour envoyer des ordres au serveur XBMC (reco faciale/son/requete HTTP) :
<http://encausse.wordpress.com/2013/02/16/s-a-r-a-h-tutorial-plugin-xbmc/>
 - Doc XBMC pour réaliser des plugins :
<http://forum.xbmc.org/showthread.php?tid=160975>
http://wiki.xbmc.org/index.php?title=Python_development
http://wiki.xbmc.org/index.php?title=Add-on_development
 - Langage Python
 - Communication vers XBMC : JSON-RPC
 - Communication vers OpenHab : MQTT ou RESTFULL

- **UPnP low-power :**
 - Protocole existe (<http://upnp.org/specs/lp/lp1/>, <http://upnp.org/specs/lp/UPnP-Ip-LPArchitecture-v1.pdf>), la communication UPnP/XBMC aussi (<http://wiki.xbmc.org/?title=UPnP>)

- **Intégration de caméras de surveillance :**
 - Les caméras wifi se comportent comme des médias distants dans XBMC. Utiliser les caméras comme les Hue en mode Ambient Light (sans OpenHab)
 - Script existant : <http://windowsmediacenter.blogspot.fr/2014/01/ip-security-cam-image-in-xbmc-using.html>
 - <http://homeawesomation.wordpress.com/2013/02/18/doorbell-ipcam-xbmc-update/>

- **Intégration VLC :**
 - Apparemment déjà intégré car dispose déjà d'un serveur UPnP qui renvoie une adresse, lisible directement depuis VLC.

- **Follow me :**
 - <http://www.howtogeek.com/75535/how-to-sync-your-media-across-your-entire-house-with-xbmc/>
 - <http://lifehacker.com/5634515/how-to-synchronize-your-xbmc-media-center-across-every-room-in-the-house>
 - Idée : utiliser une librairie / config centralisée sur mysql server for xbmc.

- **NFC :**
 - Doc officielle : <http://developer.android.com/guide/topics/connectivity/nfc/index.html>
 - Tutorial : <http://fr.openclassrooms.com/informatique/cours/aller-plus-loin-dans-le-developpement-android/le-partage-par-nfc>
 - Idée : utiliser une application NFC (ou ajouter la fonction à la remote xbmc) pour effectuer des actions sur XBMC, par exemple lancer une série ou un film.

- **Suivre le flux d'actualité twitter :**
 - existe déjà (<https://code.google.com/p/twitxbmc/>), afficher des notifications (<http://www.pyrrah.info/2013/12/21/xbmc-afficher-des-notifications/>)

Par tâches

- **XBMC Reflexive Remote Dynamic remote control for XBMC :**
Affichage sous forme de liste au lieu d'une interface de télécommande.
- **Hue :**
Affichage de la couleur dominante.
- **Commande vocale :**
L'application répond à des commandes vocales spécifiques.
- **Reconnaissance visage :**
Identifier l'utilisateur pour afficher son profil associé (contrôle parental)
- **Intégration avec OpenHAB :**
Récupérer et commander les informations des capteurs connectés à OpenHAB
Gérer les notifications
- **Intégration de caméras de surveillance :**
Dans le plugin OpenHAB, choisir une caméra et affichage de la caméra en réaction à un événement.
- **NFC :**
Faire des objets actifs sur des actions de XBMC.

Gestion de projet

Méthodologie utilisée

L'équipe suivra une méthodologie de gestion de projet agile, et plus particulièrement les principes de Scrum. Nous opterons pour un scrum master tournant entre les quatre membres du groupe (un scrum master ne peut en aucun cas occuper la place de chef de projet, Nicolas ne sera donc jamais responsable de la bonne application de la méthode scrum). Étant relativement novice dans la gestion de projets utilisant la méthode scrum, l'équipe a choisi cette technique afin de permettre à tous ses membres de s'y former. La durée des sprints sera d'une semaine, car cela nous a paru être la période la plus adaptée. Nous avons également prévu une séance de poker planning en fin de phase de conception.

Planning prévisionnel

Nom de la tâche	Durée	Début	Fin
Cahier des charges	1 jour	Mar 04/02/14	Mar 04/02/14
Ambilight	5 jours	Mer 05/02/14	Mar 11/02/14
OpenHAB	7 jours	Ven 07/02/14	Lun 17/02/14
Créer plugin	1 jour	Ven 07/02/14	Ven 07/02/14
Récupérer info du sitemap	3 jours	Mar 11/02/14	Jeu 13/02/14
Modifier info	2 jours	Ven 14/02/14	Lun 17/02/14
XBMC reflexive remote	14 jours	Mer 12/02/14	Lun 03/03/14
Intégration caméras de surveillance	2 jours	Mar 11/03/14	Mer 12/03/14
Intégration NFC	2 jours	Jeu 13/03/14	Ven 14/03/14
Commande vocale	2 jours	Mar 18/03/14	Mer 19/03/14
Reconnaissance visage	4 jours	Jeu 20/03/14	Mar 25/03/14