

# LO52 Projet Automne 2016 : Android et les "objets connectés" : KODI-like

## 1 Enoncé

Ces derniers mois, vous avez pu assister à différentes conférences dans le monde des téléphones mobiles, tablettes et autres périphériques embarqués mettant en scène des accessoires, objets apportant plus que de simples fonctionnalités de communication voix/data. Ainsi, on est capable maintenant de contrôler son contenu multimédia, son téléviseur, son électroménager ou bien encore ses propres constantes personnelles.

Le projet de l'unité de valeur LO52 pour ce semestre va s'articuler autour de ce sujet et va avoir pour objectif de vous faire mettre en place votre premier "système Android" permettant de piloter un objet connecté (un lance missiles USB) mais aussi de servir de serveur multimédia. Vous aurez à votre disposition :

- Une carte Pandaboard représentant votre système embarqué et démarrant sur un Android que vous aurez au préalable personnalisé
- Un lance-missiles USB représentant l'objet connecté que vous serez amené à contrôler de différentes manières à l'aide de votre système
- Un dongle WiFi afin de mettre le support de cette fonctionnalité
- Un switch permettant de brancher votre Pandaboard en Ethernet

Dans la suite du sujet, chaque partie va être détaillée afin de vous expliquer le travail à faire et d'explicitier les résultats attendus. Une section sur la mise en place de l'environnement de travail sera aussi présente. En plus du code source que vous fournirez à l'enseignant, vous aurez aussi à réaliser une présentation (Powerpoint, Libre Office, LaTeX) et une démonstration en fin de semestre.

A la fin du sujet, vous trouverez aussi quelques recommandations/directives spécifiques pour certains aspects du projet.

## 2 Configuration de l'environnement de travail

1. Les identifiants de connexion à la machine virtuelle sont : lo52tp, lo52tp
2. Petit rappel concernant les "Shared Folders" sous Virtual Box : une fois configuré depuis l'interface, pour les monter dans votre machine virtuelle de TP, utilisez la commande (demander à l'enseignant en cas de doute) :

```
sudo mount -t vboxsf -o uid=$UID,gid=$(id -g) id_rep_virtualbox pt_montage_vm
```

3. A l'aide de la commande `repo` adapté, changez les fichiers `.gitconfig` de tous les projets AOSP pour matcher votre login/email. Demandez à l'enseignant avant de lancer la commande.
4. Créez une branche locale à l'aide de la commande `repo`. Cette branche aura pour nom **[nom-groupe]-gr[a ou b]-lo52**.
5. Pour la compilation, vous utiliserez la **ligne de commande Linaro** que vous devrez trouver.

## 3 La carte Pandaboard

### 3.1 Détails de l'élément

Le **périphérique Android** sera représenté dans le cadre de ce projet par la **carte Pandaboard** avec écran. Cette carte est équipée à la fois du **WiFi** et de l'**Ethernet** pour communiquer en

réseau. Pour le projet, nous utiliserons l'**Ethernet** qui aujourd'hui fonctionne parfaitement. Cependant, comme vous aurez aussi à faire le support du Dongle WiFi, vous aurez aussi la possibilité de passer par cette connexion.

Dans le cadre du projet, vous devrez faire un ensemble d'adaptations à l'**image système Android** utilisée afin de la rendre unique. La procédure pour comprendre et apprendre comment réaliser cette image sera vue plus en détail en cours et en TDs et vous mettrez en application ces connaissances durant le projet. Voici la liste des modifications attendues pour l'image système Android qui tournera sur ce "**HTPC**" :

- La **Pandaboard** tournera sur un **android 4.0.4 Linaro** personnalisé par vos soins avec pour nom **arrow**.
- Ce système disposera d'un **noyau personnalisé** avec pour suffixe le nom cité précédemment en minuscule
- Il contiendra aussi l'intégration d'une **bootanimation personnalisée** représentant votre nom de groupe de projet (si difficulté de trouver comment faire, demander à l'enseignant)
- Le **fond d'écran par défaut** sera changé pour avoir quelque chose en rapport avec votre nom de votre projet
- Vous intégrerez la **libusb** à la version 1.0.9 (fournie par l'enseignant) afin de piloter le lance-missiles USB
- En plus de ce composant, vous ferez du programme exemple fourni lors du TP2 un **binaire natif** de votre système et aussi une **interface JNI** exploitable par une **extension de framework** que vous créerez par la même occasion. Le choix dans le nom des classes et des packages Java est laissé libre
- Votre image intégrera aussi une application de type Media Center décrite un peu plus loin
- Le support du Dongle WiFi qui sera prêté pendant le semestre par l'enseignant

### 3.2 Consignes particulières

Quelques consignes :

- Toutes les modifications seront incluses dans une seule et même image qui correspondra à la définition d'une cible spécifique au sein des sources Android
- Les différents groupes de projet seront établis lors du premier TD/TP et les binômes se feront au sein des mêmes groupes de TP

### 3.3 Résultats attendus

Votre système devra :

- Démarrer avec la nouvelle **bootanimation**
- Avoir un **fond d'écran par défaut personnalisé** dès la fin du démarrage
- Lorsque l'utilisateur ira consulter les **informations de son périphérique**, afficher les bonnes informations
- Donner la possibilité d'utiliser la **libusb** en utilisant le **client natif** ou l'**extension de framework** et la **JNI** à travers **ADB**
- Permettre d'utiliser l'application Media Center qui permettra ou de parcourir une liste prédéfinie ou bien de se connecter à un serveur pour récupérer des metadata et les fichiers multimédia afin de les jouer
- Se connecter au réseau WiFi interne ou bien celui fourni par l'enseignant

## 4 L'application Media Center

### 4.1 Détails de l'élément

Les chapitres précédents vous ont présenté l'initialisation de votre environnement de développement ainsi que les attentes vis à vis de la plateforme Android. Il reste un élément important à expliciter : l'application Media Center.

Cette application est une application Android complète permettant de :

- Parser et afficher un explorateur de fichier par type de média (Vidéos, Audio)
- A travers la présence d'un fichier de metadata dont vous choisirez le format, de donner des informations à la fois sur le format de fichiers (codecs,...) mais aussi sur le contenu (série, épisode, résumé...)
- Lire le média une fois celui-ci sélectionné

### 4.2 Consignes particulières

Deux consignes :

- Le look&feel de l'interface graphique est laissé au choix des étudiants mais sera aussi pris en compte dans l'évaluation
- Les étudiants peuvent utiliser une liste en dur de médias ou bien monter un serveur NFS ou tout autre type de partage de fichiers

## 5 Modalités d'évaluation

Votre projet sera évalué sur les critères suivants :

- Périphérique démarrant avec bootanimation personnalisé
- Périphérique avec nouveau fond d'écran
- Vérification de la version du Kernel
- Connexion ADB au système, vérification de la présence de la libusb et du fonctionnement du client natif, de la JNI et du framework
- Test de l'application Media Center sur la GUI, la stabilité...
- Vérification des différents contenus additionnels
- Revue du code avec les étudiants
- Soutenance de projet par les étudiants

## 6 Consignes générales

- Le lance-missiles et la carte Android seront reliés directement par USB
- Les étudiants ont carte blanche pour l'interface graphique de l'application
- Il ne sera pas nécessaire d'imprimer les supports de présentation
- Une simple archive de l'ensemble des fichiers que vous aurez créés/modifiés vous sera demandé à la fin du projet afin de permettre à l'enseignant de valider votre travail