# **App Inventor ressource Numéro 12**



Sources

http://ai2.appinventor.mit.edu/ https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3 http://appinventor.mit.edu/explore/ai2/setup-emulator.html

## **Tutoriels vidéos**

https://www.youtube.com/watch?v=0H9Y2OQohyE https://www.youtube.com/watch?v=I2DONpimg1M

## Nouvelle application, commande du robot Shield bot v1.1 via Bluetooth

Images que nous allons utiliser pour cette application



Voici les images de l'application terminée (un exemple de présentation ;)





Afin de ne pas avoir à reprendre toutes les saisies, vous pouvez **reprendre la ressource numéro 8** (De nombreuses informations y sont déjà décrites !) ...

Vous pouvez faire un « enregistrez sous » et renommer l'application en « robot\_version\_bluetooth\_version\_1 »

Il ne vous « reste » plus qu'à compléter l'application...



Nous allons décrire toutes les propriétés des composants...certains ne seront pas repris car ils sont déjà décrits dans la ressource numéro 8 (Arrangement horizontal 5). L'arrangement horizontal 7 est là

uniquement pour descendre les boutons au milieu du smartphone...Les 4 boutons ont la même structure, nous décrirons seulement le « bouton\_avant »



Le bouton « Robot » est ici pour espacer les boutons droite/gauche et pour rappeler l'icône de l'application

Composants	Propriétés	Composants	Propriétés	Composants	Propriétés
🕀 🔲 Screen 1	Label_titre_application	B Screen1	Bouton_avant	😑 🔲 Screen1	Robot
Arrangement_horizontal1     Arrangement_horizontal5     Arrangement_horizontal5     Arrangement_horizontal5     Bouton_deconnecte     Bouton_connecte     Arrangement_horizontal7	Couleur de fond Orange Gras V Italique Taille de police 20	Screen     Arrangement_horizontal1     Arangement_horizontal5     Arrangement_horizontal5     Marge_logo_bluetooth     Bouton_deconnecte     Bouton_connecte     Arrangement_horizontal7     Arrangement_horizontal7	Couleur de fond Par défaut Activé Gras Italique	Arrangement_horizontal1     Abel_titre_application     Marrangement_horizontal5     Marrangement_horizontal5     Marrangement_horizontal7     Bouton_connecte     Marrangement_horizontal7     Marrangement_horizontal2	Couleur de fond Aucun Activé Ø Gras Italique
Arrangement_horizontal2     Bouton_avant	Type de police Par défaut +	Bouton_avant Arrangement_horizontal3	Taille de police	Bouton_avant Arrangement_horizontal3	Taille de police
Arrangement_horizontal3     Bouton_gauche     Robot     Bouton_droit     Arrangement_horizontal6	HasMargins	Bouton_gauche Robot Bouton_droit Sarrangement_horizontal6 Bouton_arrière	Type de police Par défaut • Hauteur 70 pixels	Bouton_gauche Robot Bouton_droit Arrangement_horizontal6 Bouton_arrière	Type de police Par défaut • Hauteur 70 pixels
Bouton_arrière	Automatique Texte Commande Robot Bluetoo	Client_Bluetooth1	Largeur 70 pixels Image avant.jpg	Client_Bluetooth1	To pixels Image robot.jpg
Renommer Supprimer	gauche -	Renommer Supprimer	Forme par défaut •	Renommer Supprimer	par défaut •
arriere.jpg avant.jpg bluetooth.jpg droit.jpg gauche.jpg robot.jpg robot.bluetooth.jpg Charger fichier	Couleur texte Noir Visible	Média arriere jpg avant.jpg bluetooth.jpg droit.jpg gauche.jpg robot.jpg robot.jpg robot.bluetooth.jpg Charger fichier	Montrer réaction  Texte  Alignement texte centre •  Couleur texte Par défaut Visible	wetula arriere.jpg avant.jpg bluetooth.jpg droit.jpg gauche.jpg robot.jpg robot.jbluetooth.jpg Charger fichier	Montrer réaction  Texte  Alignement texte centre •  Couleur texte Par défaut Visible

Bouton connecte » va nous permettre de sélectionner la carte Bluetooth à utiliser (par exemple vous pouvez comme moi avoir dans votre liste les connexions Bluetooth de votre voiture...)

Dans la palette « interface utilisateur » il faut choisir « sélectionneur de liste »

E Sélectionneur de liste 📀

Palette		
Interface utilisateur		
Disposition		
Média		
Dessin et animation		
Capteurs		
Social		
Stockage		
Connectivité		
Déclencheuractivité 📀	,	
Client Bluetooth 🧿	,	
Serveur Bluetooth 🧿		
S Web 🤊		
LEGO® MINDSTORMS®		
Expérimental		
	Palette         Interface utilisateur         Disposition         Média         Dessin et animation         Capteurs         Social         Stockage         Connectivité         ©         Déclencheuractivité         ©         Serveur Bluetooth         ©         Web         ©         LEGO® MINDSTORMS®         Expérimental	

Nous allons maintenant rajouter un « service Bluetooth » à cette application...

Ce service se place en dessous de votre application (physiquement à l'écran)

Faire un glissé/déposé et le service se place tout seul en bas de l'application.



Pour la partie programmation par Blocs, il va falloir prendre en compte de nombreuses modifications...pas forcément très simples à comprendre...

Voici l'ensemble des blocs que vous avez utilisé...cette partie là ne change pas !!



Le « Bouton\_connecte », le sélectionneur de liste, va vous permettre de choisir la connexion Bluetooth (ici celle de la carte HC-06)

Une fois le choix de la connexion Bluetooth fait, la liaison va s'établir entre votre terminal (tablette ou smartphone) et le « Bouton\_connecte » va disparaître et laisser la place au « Bouton\_déconnecte »

Et pour le dernier ensemble, quand en clique sur le « Bouton\_déconnecte », la liaison Bluetooth et déactivée, le « Bouton\_déconnecte » disparaît et le « Bouton\_connecte » revient à l'écran....



Les boutons de déplacement vont maintenant pouvoir être utilisés



quand     Bouton_arrière     .Enfoncé       faire     appeler     Client_Bluetooth1     .Envoyer1Octet       nombre     5	quand       Bouton_arrière •       .Retiré         faire       appeler       Client_Bluetooth1 •       .Envoyer1Octet         nombre       / 4
quand       Bouton_droit • .Enfoncé         faire       appeler       Client_Bluetooth1 • .Envoyer1Octet         nombre       1	quand       Bouton_droit • .Retiré         faire       appeler       Client_Bluetooth1 • .Envoyer1Octet         nombre       1
quand     Bouton_gauche .Enfoncé       faire     appeler       Client_Bluetooth1 .Envoyer1Octet       nombre	quand       Bouton_gauche •       .Retiré         faire       appeler       Client_Bluetooth1 •       .Envoyer1Octet         nombre       [4]
quand       Bouton_avant • .Enfoncé         faire       appeler         Client_Bluetooth1 • .Envoyer1Octet         nombre	quand     Bouton_avant • .Retiré       faire     appeler     Client_Bluetooth1 • .Envoyer1Octet nombre

Quand on maintient le bouton arrière enfoncé, on envoie via le Bluetooth le nombre 5 Quand on maintient le bouton droit enfoncé, on envoie via le Bluetooth le nombre 3 Quand on maintient le bouton gauche enfoncé, on envoie via le Bluetooth le nombre 2 Quand on maintient le bouton avant enfoncé, on envoie via le Bluetooth le nombre 1 Pour les 4 boutons, lors du relâchement, on envoie via le Bluetooth le nombre 4 (qui provoquera l'arrêt!)

Nous allons maintenant passer à la partie « Arduino »

Matériel nécessaire pour réaliser cet exercice

Matériel	Matériel
Carte UNO + cordon USB	Cordon Grove
Robot Shield bot v1.1	Une carte Bluetooth HC-06

## Voici les blocs assemblés





# Code

```
#include <SoftwareSerial.h>
#include <Shieldbot.h>
SoftwareSerial mySerial(2,3);
int ABVAR 1 Blue = 0;
//libraries at http://duinoedu.com/dl/lib/grove/EDU_Shieldbot_Grove/
Shieldbot monBot;
void setup()
{
 mySerial.begin(9600);
 monBot.brancher();
 \_ABVAR_1_Blue = 4 ;
}
void loop()
{
 monBot.ecrireVitesse(1000,1000);
  _ABVAR_1_Blue = mySerial.read();
 if (( ( _ABVAR_1_Blue ) == ( 1 ) ))
 {
  monBot.avancer("tout_droit");
 }
 if (( ( _ABVAR_1_Blue ) == ( 2 ) ))
 {
  monBot.ecrireVitesse(300,1000);
  monBot.avancer("tout_droit");
 }
 if (( ( _ABVAR_1_Blue ) == ( 3 ) ))
 {
  monBot.ecrireVitesse(1000,300);
  monBot.avancer("tout_droit");
 }
 if (( ( _ABVAR_1_Blue ) == ( 5 ) ))
 {
  monBot.ecrireVitesse(500,500);
  monBot.reculer("tout_droit");
 if (( ( _ABVAR_1_Blue ) == ( 4 ) ))
 {
  monBot.stopper();
 }
}
```

Pour l'avance moteur « tout droit » on choisit une vitesse de 1000/1000

Pour tourner à droite et à gauche on choisit une vitesse 300/1000, rotation plus fluide

Pour reculer on choisit une vitesse de 500/500 et on n'oublie pas de changer le sens du déplacement dans le bloc « shieldbot 1.1 reculer »

En passant la souris sur cet élément, une flèche de déroulement apparaît et nous avons « avancer, reculer et pivoter »



Il faudra dans un premier temps appairer votre carte HC-06 avec votre terminal et remarque très importante pour le câblage, il faudra croiser RXD et TXD entre la carte HC-06 et la carte Arduino !!

Pour appairer votre carte Bluetooth à votre périphérique Android, le code est 1234



Ne pas oublier de mettre à jour votre logiciel Ardublock <u>http://duinoedu.com/telecharger.html</u>

Rappel de la ressource numéro 4

Activer le débogage USB sous Android,

#### Étape 1 : Passer en mode administrateur

Pour commencer, rendez-vous dans les « Paramètres » , puis cherchez l'onglet « À propos » ou « À propos du téléphone » . Il est généralement situé tout en bas de la liste.

Cliquez dessus. Ici, on peut voir la catégorie « Options pour les développeurs » . Il n'apparaîtra pas tant que vous ne serez pas en mode administrateur.

Cliquez ensuite sur « Informations Logicielles » . Le nom peut varier d'un modèle à un autre, mais le dossier est forcément présent. Il recense toutes les informations liées aux logiciels de votre mobile. Il vous faut trouver le « Numéro de build » . Si vous ne savez pas ce que c'est, pas de panique ça n'a aucune importance pour arriver à bout de la manœuvre.

Une fois le numéro de build visible, cliquez dessus plusieurs fois de suite. Six, sept ou plus si nécessaire. Au bout d'un moment, une notification apparaîtra à l'écran.

Elle vous signale que vous êtes désormais en mode administrateur / développeur. C'est parfait, c'est exactement ce que l'on veut. Retournez dans les paramètres généraux.

# Étape 2 : activer / désactiver le débogage USB

App Inventor ressource numéro 12

L'onglet « Options pour les développeurs » est maintenant disponible dans les « Paramètres » . Cliquez dessus.

Un message d'alerte apparaît. En effet, modifier des paramètres à tort et à travers en mode administrateur peut avoir un impact néfaste sur votre téléphone.

Cliquez sur « OK », puis tenez-vous en à cocher la case « Débogage USB ».

Voilà, votre mobile peut maintenant être connecté à votre PC et il sera considéré comme un périphérique de stockage. À l'image d'une clé USB classique ou d'un disque dur externe.



Les raccordements sur la carte Arduino





Pour information sur le câblage réalisé ici, j'ai utilisé un câble Grove éclaté sur 4 picots (TX et RX croisés! Avec la partie blocs ci-dessus)

Carte HC-06	Câble Grove	Carte Arduino	
RXD	Fil jaune	Pin 3	
TXD	Fil blanc	Pin 2	
GND	Fil rouge	GND	
VCC	Fil noir	+5V	





#### **Remarque**

Lors de votre connexion Bluetooth, il s'avère que nous ne savons pas si les appareils sont bien connectés entre-eux.

Pour cela, nous allons mettre en place un test de connexion et nous allons l'afficher sur l'écran du smartphone (ou tablette)

Quelques modifications dans la partie design de votre application...Nous allons créer un label « état de connexion » qui sera modifié lors des connexions Bluetooth

Par la même occasion, on renomme les boutons « connecté et déconnecté » par « Bouton de connexion » et « Bouton de déconnexion »

Composants	Propriétés	Composants	Propriétés	Composants	Propriétés
Creart	Bouton deconnecte	B Screen1	Bouton_connecte	B Screen1	Bouton_connecte
				B Amonement horizontal	
Arrangement_horizontal1	Couleur de fond		Couleur de fond		Couleur de fond
Label_titre_application	Aucun	In Label_titre_application	Aucun	Initabel_titre_application	Aucun
8 Arrangement_horizontal5	August .	Arrangement_horizontal5	Élémente de la chaîne	Arrangement_horizontal5	Déments de la chaîne
Imana Inno bhuateoth	ACOVE	Image logo bluetooth	ciencina de la criane	Image_logo_bluetooth	
- Indge_roge_ordenous					
Bouton_deconnecte	Gras	Bouton_deconnecte	Lotinia.	Bouton_deconnecte	Activity
Bouton_connecte	8	Bouton_connecte	2	Bouton_connecte	8
8 Americante horizontal7	Italique	8 Ananoement horizontal7		8 Arrangement horizontal7	
	0		Gras	A	Gras
Interat_connexion	Taile de poirs	illieiar_connexion	*		
Arrangement_horizontal2	Tarre de porce	Arrangement_horizontal2	Italique	Arrangement_horizontal2	Italique
Bouton_avant	20	Bouton_avant	0	Bouton_avant	0
	Tune de políce	B American Indiana	Taile de coice	B Arrangement horizontal3	Taile de police
	Our different a		20		20
Bouton_gauche	Par default +	Bouton_gauche	40	Bouton_gauche	
Robot	Hauteur	Robot	Type de police	Robot	Type de police
Danton david	Atomaticue	Same dani	Durdifiert	Bouten desit	Par défaut -
	( contraction of the second se		- al aciava -		Far delaux -
Arrangement_horizontal6	Largeur	<ul> <li>Marangement_horizontal6</li> </ul>	Hauteur	<ul> <li>Ell Arrangement_horizontal6</li> </ul>	Hauteur
Bouton_amière	250 chels	Bouton_amière	Asomatique	Bouton_arrière	Automatique
Client Rivetryth1		Client Bluetroth1		Client Bluetpoth1	
	Image		Largeur		Largeur
	Augun	1	200 paols	4	200 pixels
Renommer Supprimer	Forme	Renommer Supprimer	Image	Renommer Supprimer	Image
	par défaut +		Auton		Augun
láfis		Midia		Mirlin	
incura .	Montrer reaction	income.	ItemBackgroundColor	тема	ItemBackgroundColor
arriere.jpg	8	arriere.jpg	Noir	arriere.jpg	Noir
avant.jpg	Texte	avant.jpg	temTextColor	avant.jpg	BernTextColor
bluetooth.jpg	Bouton déconnexion	bluetooth.ipg	Blanc	bluetooth.ipg	Ranc
denition		Aniting		And inc.	
enoughy ins	Alignement texte	analysis in a	Selection	a on gog	Sélection
gaure.pg	centre -	gaucie.jpg		gaucne.jpg	
rabat.jpg		robot.jpg		robot.jpg	
robot_bluetooth.jpg	Couleur texte	sobot_bluetooth.jpg	Forme	robot_bluetpoth.jpg	Forme
Charger fichier	Par défaut	Charger fichier	par défaut +	Charger fichier	par défaut +
1	Visible		Montres alastica		
	0		Wonder reaction		Montrer reaction
			Affichage bar filtrage		Affichage bar filtrage
			0		0
			Texte		Texte
			Bouton connexion		Bouton donnering
			Alignement texte		Alignement texte
			centre +		centre .
					and a
			Couleur texte		Couleur texte
			Par défaut		Par défaut
			Tes		-
			102		litre
			16.31		
			visibe		Visible
					8





Voici une capture d'écran sur App Inventor et ci-dessous les 3 états successifs de l'application,

avant la connexion, « État connexion »

puis connecté en Bluetooth « Vous êtes connecté »

et pour finir, après la déconnexion «Vous êtes déconnecté »



quano faire quano	Bouton_connecte  Avant prise  Mettre Bouton_connecte   Client_Bluetooth1  Adresses et noms  Bouton_connecte  Après prise	quand       Bouton_deconnecte        Clic         faire       appeler       Client_Bluetooth1        .Déconnecter         mettre       Bouton_connecte        . Visible        à ( vrai          mettre       Bouton_deconnecte        . Visible        à ( faux          mettre       etat_connexion        . Texte        à ( faux          mettre       etat_connexion        . Texte        à ( faux			
faire	mettre Bouton_connecte • . Sélection • à (appeler Client_Bluetooth1 • .Se connecter				
	adresse	Bouton_connecte • . Sélection •			
	Image: Simple Simpl				
	alors mettre etat_connexion · . Texte · à / * Vous êtes connecté *				
	mettre [etat_connexion v]. Couleur texte v] à [				
	mettre Bouton_connecte • ) . (Visible • ) à ( faux • )				
	mettre Bouton_deconnecter . Visibler à la vrair				

On remarque sur ces captures d'écrans

Une fois la sélection réalisée, on vérifie l'état de la connexion Bluetooth (« Client\_Bluetooth1 » « Est connecté »)

On effectue un test de « Si / Alors »

Si la connexion Bluetooth est effective alors je mets « Etat\_connexion » à « Vous êtes connecté » et je le mets en vert

Pour la partie déconnexion, même principe, si la connexion Bluetooth n'est plus active, je mets « Etat\_connexion » à « Vous êtes déconnecté » et le mets en rouge

Cette modification sera nommée robot\_version\_bluetooth\_version\_2