

Prise en main AppInventor





Organisation logiciels / matériels



Logiciels : Picaxe Editor avec Blockly / Scratch mBlock



AppInventor – Façade « Designer »





AppInventor – Façade « Blocs »





Prérequis

- Créer un compte Gmail pour avoir accès à AppInventor (lien: http://ai2.appinventor.mit.edu/).
- > Accès internet tout au long de l'utilisation d'AppInventor.
- > Connaissances de base programmation par blocs sur Scratch mBlock ou Picaxe Editor (Blockly).
- > Passer AppInventor en français avant chaque utilisation.





Etape 1 : Charger le modèle de base

Télécharger le modèle de base ('Bluetooth_base.aia') disponible sur <u>www.a4.fr/wiki</u>.

Ce fichier modèle contient la base du code (trop compliqué pour un élève) permettant de se connecter à un module Bluetooth.

- Renommer le modèle de base par le nom de votre programme (exemple: test.aia).
- Importer le modèle de base renommé dans AppInventor et l'ouvrir dans la façade 'Designer'.

Mes projets
Commencer nouveau projet
Importer le projet (.aia) de mon ordinateur
Importer le projet (.aia) de mon dépôt Supprimer projet
Enregistrer le projet Enregistrer le projet sous Point de contrôle
Exporter projet sélectionné (.aia) sur mon ordinateur Exporter tous les projets
Importer keystore Exporter keystore

Fichier.aia

Les fichiers aia permettent de modifier le design et le code de l'application. Ils peuvent être importés et exportés à partir d'AppInventor.



Etape 2: Paramétrer l'application de base



Façade 'Designer': Dans la liste des composants, sélectionner 'Screen1' (représente l'application de manière



Etape 3: Afficher les composants cachés

Façade 'Designer': Au dessus de l'interface, cliquer sur 'Afficher les composants cachés dans l'interface'.

Cela fera apparaitre la section cachée nommée 'Programme_conteneur'.

Note: Cette section est cachée pour empêcher l'utilisateur d'avoir accès au programme tant que la connexion Bluetooth n'est pas établie.

Rendre cette section visible (momentanément) va permettre d'ajouter des éléments à l'intérieur.

Remarque: Tous les éléments que vous ajoutez doivent se trouver dans cette section.





Etape 4: Placer un bouton

Façade 'Designer': Placer dans la section 'Programme_conteneur' un bouton en le faisant glisser dans celle-ci. Note: Le bouton créé apparait maintenant dans la liste de composants. Vous pouvez cliquer dessus afin de modifier ses paramètres comme sa couleur, son nom, ou bien son texte. Ici, le bouton s'appellera 'Bouton1' et le texte sera 'Avancer une seconde'.





Etape 5: Relié le bouton à un évènement

Façade 'Blocs' : Dans les bibliothèques de blocs, se trouve la même liste des composants de la façade 'Designer'. Dans cette liste, cliquer sur le composant 'Bouton1' créé précédemment pour avoir accès aux blocs spécifiques à ce composant. Pour les blocs spécifiques au Bluetooth, rechercher dans cette même liste le composant 'Bluetooth'. L'exemple de code ci-dessous envoie par Bluetooth la donnée '1' quand le 'bouton1' est cliqué.





Etape 6: Compiler l'application

- > Une fois que le code de la façade 'Designer' est terminé, on peut compiler l'application afin de tester celle-ci sur un smartphone.
- > Dans le menu 'Construire', cliquer sur 'App (enregistrer .apk sur mon ordinateur)'.
- Note: la création et l'enregistrement du fichier .apk sur l'ordinateur peut prendre plusieurs minutes selon votre connexion internet et la disponibilité des serveurs.



Placer le fichier .apk dans la mémoire du smartphone Android.

Fichier.apk

Un fichier apk permet d'installer l'application sur le smartphone. Il est créé à partir d'Applnventor et doit être placé dans la mémoire du smartphone pour être lancé.



Etape 7: Installer l'application



A partir du smartphone, cliquer sur le fichier .apk enregistré précédemment (1).

Autoriser les sources inconnues (2). Cliquer sur 'Installer' (3) puis, lorsque le processus est fini, cliquer sur 'OK'.

Une application avec l'icone d'A4 Technologie doit maintenant apparaitre dans vos applications.

Avant de lancer l'application, il est nécessaire d'appairer le smartphone au module Bluetooth une première fois et de programmer le matériel pour traiter les données reçues.



Etape 8: Appairage au module Bluetooth



Mettre le matériel sous tension afin d'alimenter le module Bluetooth.

Celui-ci possède une LED témoin sur sa face inférieure près de la prise de connexion : si le module est alimenté, cette LED doit clignoter.

Aller dans les paramètres Bluetooth du smartphone pour appairer le module Bluetooth. Lancer une recherche d'appareil (1).

Le module se nomme « HMSoft ». Cliquer dessus et rentrer le code pin '1234' (2).

Vérifier que le module se trouve bien dans les 'périphériques associés ' pour valider l'appairage(3).



Etape 9: Programmation matériel

Dans cet exemple, le matériel est un robot Loupiot programmé à partir de Picaxe Editor. Le modèle de base pour rédiger le programme est le fichier 'Loupiot_Base.xml' téléchargeable sur <u>www.a4.fr/wiki</u> Ouvrir le logiciel Picaxe Editor pour réaliser ce code en ouvrant le modèle de base. Note: les fichiers de correction pour cet exemple sont contenus dans le dossier 'Smartphone envoi'.



Ce programme fait avancer le robot pendant une seconde quand la commande '1' est reçue par Bluetooth.

Téléverser ce programme dans le robot et lancer l'application préalablement installée pour tester l'envoi d'une donnée du smartphone vers le robot.



Etape 10: Lancement de l'application



Mettre le matériel sous tension afin d'alimenter le module Bluetooth.

Dans vos applications, lancer l'application 'Test Applnventor' précédemment installée (1).

Cliquer sur le bouton 'Connexion' (2) pour avoir accès au module Bluetooth appairé et cliquer dessus (3).

Si la connexion est réussie, la LED du module Bluetooth reste allumée en continu et le bouton 'avancer une seconde' apparait (4).

Si vous cliquer dessus, le robot avance pendant une seconde.



Autre exemple : Recevoir des données

Le robot Loupiot envoie une variable à chaque fois que le bouton-poussoir est appuyé. Cette variable correspond au nombre d'appuis sur son bouton-poussoir depuis son allumage. Note: les fichiers de correction pour cet exemple sont contenus dans le dossier 'Smartphone réception'.





Autre exemple : Recevoir des données

Dans le code AppInventor de la diapositive précédente, le bloc 'Quand horloge 1 .chronomètre' est relié à un composant de type horloge. Celui-ci représente un capteur qui active un évènement toutes les x millisecondes. Où x se paramètre dans la façade 'Designer' de l'application dans les paramètres du composant.

	Projets 🔹	Connecte * Construire * Aide * Mes Projets Gale	erie Guide	Reporter un problème França	is ▼ Louis.bernot91@gmail.com ▼		
test Screent Ajouter écran Supprimer écran							
Palette	Interf	ace		Composants	Propriétés		
Interface utilisateur		Afficher les composants cachés dans l'interface		😑 📃 Screen1	Horloge1		
Disposition		Cochez pour voir un aperçu sur un appareil de taille tablette.		😑 🛄 Entête	TimerAlwaysFires		
Média		Test Applinventor		A4_logo	Chronomètre Activé		
Dessin et animation				BLT_bouton			
Capteurs		Déconnexion Connexion Notice		Connexion	IntervalleChronomètre		
Accéléromètre 🧿		TECHNOLODIE		Notice			
Scanneur code à barre 🧿				🔤 Séparateur_1	•X		
😚 Horloge 🔹 📀		Avancer une seconde		Programme_conteneur			
H GyroscopeSensor 🤊							
Capteur position				A Erreur_syst			
Champ proche 🤊				🔁 Horloge1			
Capteur orientation 🤊							
Pedometer 🤊							
🌯 ProximitySensor 🤊							
Social							
Stockage				Renommer Supprimer			
Connectivité				Média			
LEGO® MINDSTORMS®		Composants non-visible		IconeA4.png			
Expérimental		Bluetooth Info_blth Erreur_synt Horloge1		Sonnette.mp3			
Extension				Charger fichier			



RAPPELS : Principaux blocs AppInventor Bluetooth

Les trois blocs présentés ci-dessous sont accessibles à partir des bibliothèques de blocs en cliquant sur le composant nommé 'Bluetooth' dans la liste de composants.

appeler Bluetooth 🔽 .Octets disponibles pour le réception

appeler Bluetooth .RecevoirOctetNonSignéNuméro1



Donne le nombre de données reçues qui n'ont pas encore étés lues.

Lit la première donnée reçue qui n'a pas encore été lue. Note: si on utilise ce bloc et qu'aucune donnée n'a été envoyée, cela créée une erreur.

Il faut toujours interroger le nombre de données disponibles avant d'utiliser celui-ci.

Envoie la valeur contenue dans le bloc (nombre ou variable) accroché à celui-ci.



Autres matériels compatibles





Autre moyen d'installation pour une application

Il existe un deuxième moyen pour installer une application AppInventor sur un smartphone sans avoir à générer un fichier. Pour cela, il suffit d'installer l'application 'MIT Al2 Companion' disponible sur 'PlayStore' (1). Dans le menu 'Construire' d'AppInventor, cliquer sur 'App (Donnez le code QR pour fichier .apk)' (2). AppInventor va générer un QR code qu'il faudra lire à partir de l'application 'MIT Al2 Companion' (3).

Cette méthode nécessite que le smartphone soit connecté à un réseau internet.

