

Utilisation du logiciel **mBLOCK**

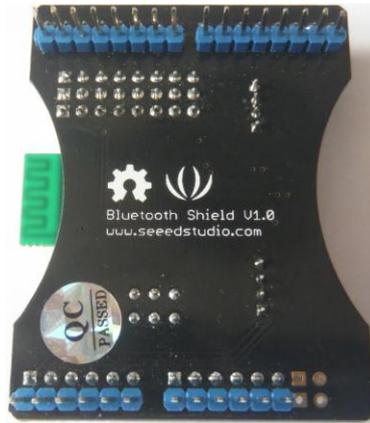


pour piloter une carte Arduino Uno



et de l'application **App Inventor**
pour communiquer avec un shield
Bluetooth

Utilisation du shield Bluetooth

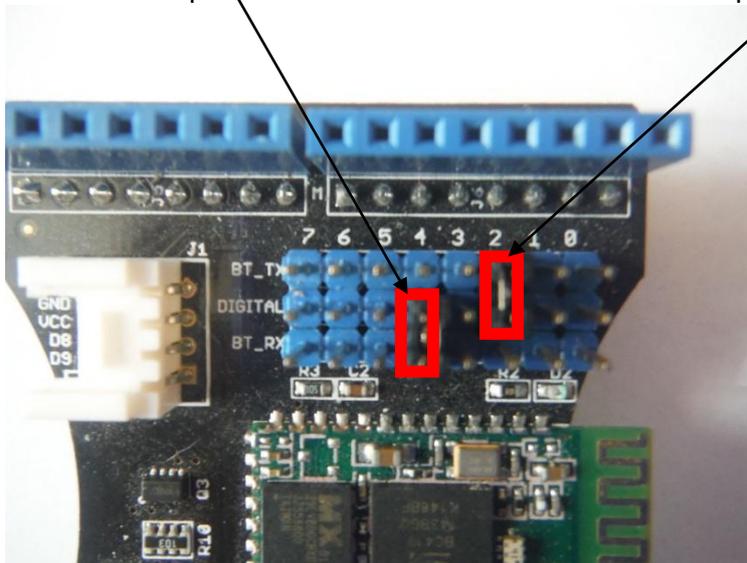


ATTENTION : il existe plusieurs sortes de shields Bluetooth.

L'extension proposée est conçue pour ce shield uniquement.

Configuration de la carte

Placer le cavalier **RX** sur l'entrée numérique 4 et le cavalier **TX** sur l'entrée numérique 2



Les broches 2 et 4 ont été choisies car elles ne sont pas des sorties PWM

Placer le shield ci-dessous sur le shield Bluetooth et le shield Bluetooth sur l'Arduino Uno



Tableau des entrées - sorties

Broche	Entrée/Sortie	Analog/Numér	Valeurs	Composant	
A0	Entrée	Analogique	0 à 1023 (2^{10})	Bouton poussoir	
A1	Entrée	Analogique	0 à 1023		
A2	Entrée	Analogique	0 à 1023		
A3	Entrée	Analogique	0 à 1023		
A4	Entrée	Analogique	0 à 1023		
D0	RX Arduino				
D1	TX Arduino				
D2					TX shield Bluetooth
D3	Entrée/Sortie	Numérique pwm	haut/bas ou 0 à 255 (2^8)	DEL verte	
D4					RX shield Bluetooth
D5	Entrée/Sortie	Numérique pwm	haut/bas ou 0 à 255 (2^8)	DEL jaune	
D6	Entrée/Sortie	Numérique pwm	haut/bas ou 0 à 255 (2^8)	DEL rouge	
D7	Entrée/Sortie	Numérique			
D8	Entrée/Sortie	Numérique			
D9	Entrée/Sortie	Numérique pwm	haut/bas ou 0 à 255 (2^8)	DEL RGB (RVB) - rouge	
D10	Entrée/Sortie	Numérique pwm	haut/bas ou 0 à 255 (2^8)	DEL RGB (RVB) - vert	
D11	Entrée/Sortie	Numérique pwm	haut/bas ou 0 à 255 (2^8)	DEL RGB (RVB) - bleu	
D12	Entrée/Sortie	Numérique			
D13	Entrée/Sortie	Numérique			

IMPORTANT :

Pour utiliser ce shield Bluetooth, il faut télécharger l'extension « bluetooth_extension » à l'adresse : https://github.com/paulcoiffier/bluetooth_extension

The screenshot shows the GitHub repository page for 'paulcoiffier / bluetooth_extension'. At the top, there are navigation links for 'Code', 'Issues', 'Pull requests', 'Pulse', and 'Graphs'. Below this, the repository name and statistics are displayed: '4 commits', '1 branch', '0 releases', and '0 contributors'. A 'Branch: master' dropdown and a 'New pull request' button are visible. A 'Find file' button and a 'Clone or download' dropdown are also present. The main content area shows a file tree with folders 'dist', 'js', and 'src', and files 'README.md' and 'bluetooth.s2e'. Below the file tree, the 'README.md' content is displayed, featuring a title 'Extension MBlock bluetooth', a description 'Extension pour la configuration et l'utilisation d'un module Bluetooth Arduino', and sections for 'Installation' and 'Utilisation'. The 'Installation' section instructs users to download the extension by clicking a link and then opening the MBlock extension manager. The 'Utilisation' section lists parameters for the 'Bluetooth init' block: TX, RX, Nom, and Code Pin.

Cliquer sur **ICI**

Téléchargez l'extension en cliquant **ICI**

The screenshot shows a file download dialog box with the text: 'Voulez-vous ouvrir ou enregistrer Bluetooth extension.zip (4,26 Ko) à partir de raw.githubusercontent.com ?'. There are three buttons: 'Ouvrir', 'Enregistrer' (with a dropdown arrow), and 'Annuler' (with a close 'x' icon).

Cliquer sur **Enregistrer sous**

The screenshot shows a file save dialog box with three buttons: 'Ouvrir', 'Enregistrer' (with a dropdown arrow), and 'Enregistrer et ouvrir'. The 'Enregistrer' dropdown menu is open, showing three options: 'Enregistrer', 'Enregistrer sous' (highlighted by a mouse cursor), and 'Enregistrer et ouvrir'.

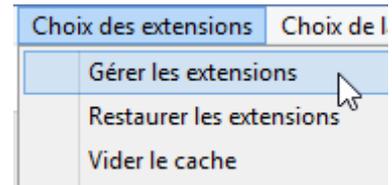
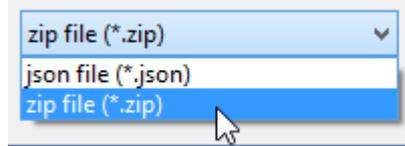
Sélectionner le dossier dans lequel le fichier sera enregistré

Laisser le fichier **Bluetooth extension** au format **.zip**

Dans mBlock, cliquer sur **Choix des extensions** puis **Gérer les extensions**

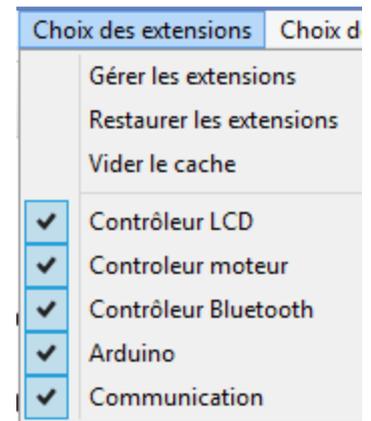
Cliquer sur **Ajouter** puis sélectionner le fichier **Bluetooth extension.zip**

Sélectionner le format **zip file (*.zip)**

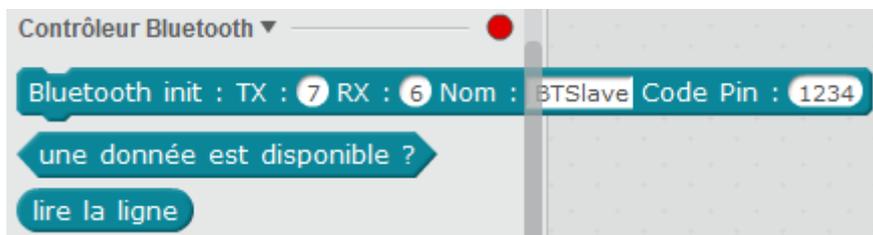


Cliquer sur **Choix des extensions** puis **Gérer les extensions**

Sélectionner **Contrôleur Bluetooth**



Ce menu apparaît dans le script **Pilotage**



Exercice 1

Un bouton pour allumer et un bouton pour éteindre la LED verte

1 - Travail dans mBlock

Travail à effectuer : répéter indéfiniment, si une donnée est disponible, lire la ligne reçue. Si le signal reçu est égal à 1 alors allumer la DEL verte (broche 3). Si le signal reçu est égal à 2 alors éteindre la DEL verte (broche 3).

Commencer par mettre **TX à 2, RX à 4, Nom BTSlave** (par défaut) ou un autre nom et Code PIN **1234** (par défaut) ou un autre nombre



```
Arduino - générer le code
Bluetooth init : TX : 2 RX : 4 Nom : BTSlave Code Pin : 1234
```

Répéter indéfiniment



```
Arduino - générer le code
Bluetooth init : TX : 2 RX : 4 Nom : BTSlave Code Pin : 1234
répéter indéfiniment
```

Si une donnée est disponible, alors

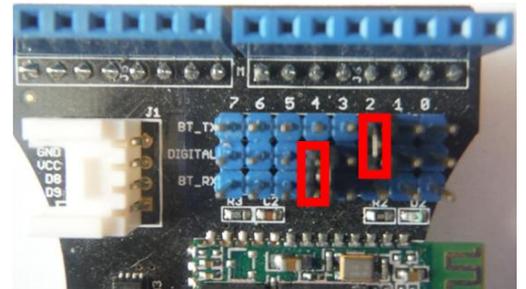


```
Arduino - générer le code
Bluetooth init : TX : 2 RX : 4 Nom : BTSlave Code Pin : 1234
répéter indéfiniment
  si une donnée est disponible ? alors
```

Si le signal reçu est égal à 1 alors allumer la DEL verte (broche 3)



```
Arduino - générer le code
Bluetooth init : TX : 2 RX : 4 Nom : BTSlave Code Pin : 1234
répéter indéfiniment
  si une donnée est disponible ? alors
    si lire la ligne = 1 alors
      mettre l'état logique de la broche 3 à haut
```



Si le signal reçu est égal à 2 alors éteindre la DEL verte (broche 3)



Sauvegarder le fichier, par exemple **Bluetooth LED verte.sb2**

Téléverser le programme dans l'Arduino Uno

2 - Travail dans l'application App Inventor

Travail à effectuer : envoyer le code « 1 » si le bouton **Allumer LED verte** est sélectionné et envoyer le code « 2 » si le bouton **Eteindre LED verte** est sélectionné.

Lancer le navigateur **Mozilla Firefox**

Se rendre sur le site <http://appinventor.mit.edu/explore>

Cliquer sur



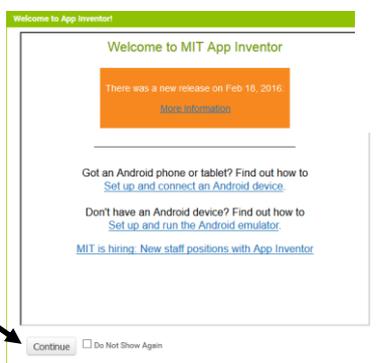
Indiquer l'adresse e-mail @gmail.com

Indiquer le mot de passe

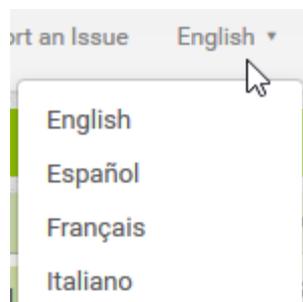
Cliquer sur **Connexion**



Cliquer sur **Continue**



Cliquer sur **English** et sélectionner **Français**



Cliquer sur

Projets ▾

Cliquer sur

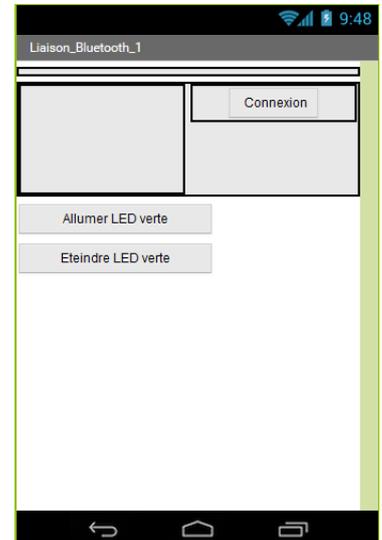
Importer le projet (.aia) de mon ordinateur ...

Sélectionner **ARDUINO_LIAISON_BLUETOOTH.aia**

Interface utilisateur sur le Smartphone ou la tablette

Le bouton **Connexion** permet d'établir la connexion entre le Smartphone ou la tablette et le module Bluetooth.

Le bouton **Déconnexion**, caché derrière le bouton **Connexion** permet d'arrêter la connexion entre le Smartphone ou la tablette et le module Bluetooth. Pour le faire apparaître, cocher « **Afficher les composants cachés dans l'interface** »



Le bouton **Allumer LED verte** permettra de mettre la broche 3 de l'Arduino à l'état **haut** et d'allumer la LED verte par l'envoi du code « 1 » vers le module Bluetooth.

Le bouton **Eteindre LED verte** permettra de mettre la broche 3 de l'Arduino à l'état **bas** et d'éteindre la LED verte par l'envoi du code « 2 » vers le module Bluetooth.

IDE App Inventor2 (fenêtre Designer)

Designer

Titre de l'application

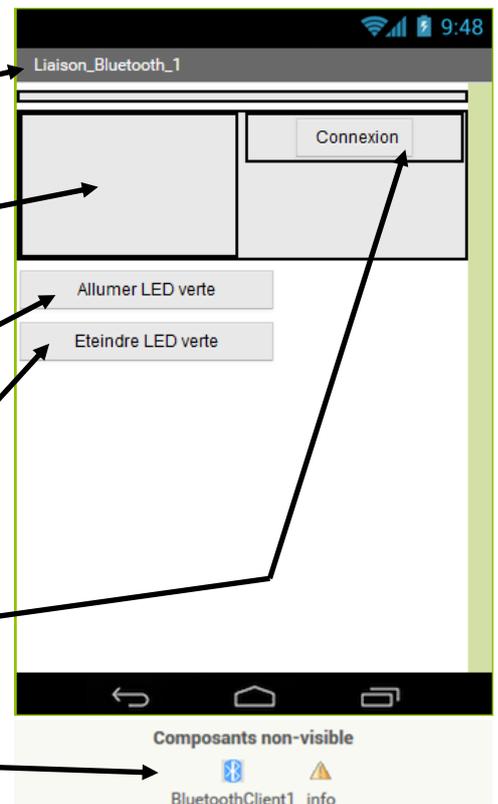
Possibilité d'insérer une image

Déclenchement de l'allumage de la LED verte en envoyant le code « 1 »

Déclenchement de l'extinction de la LED verte en envoyant le code « 2 »

Gestion de la communication en Bluetooth

Éléments non visibles de l'application

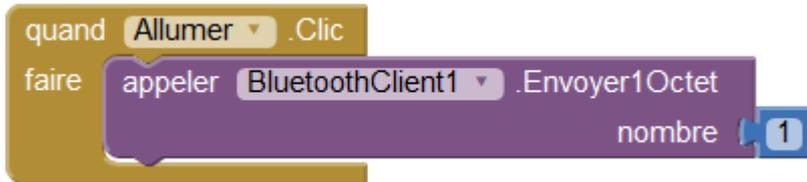


Dans la fenêtre Blocks

Blocks

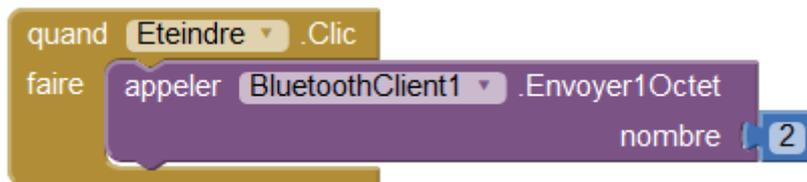
Allumage de la LED verte

Lorsque vous cliquez sur le bouton nommé **Allumer LED verte**, la valeur « 1 » est envoyée par liaison Bluetooth



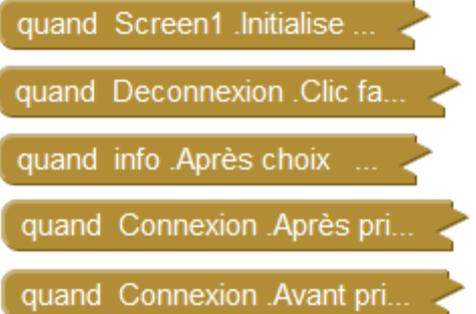
Extinction de la LED verte

Lorsque vous cliquez sur le bouton nommé **Eteindre LED verte**, la valeur « 2 » est envoyée par liaison Bluetooth



Gestion de la communication en Bluetooth

Ces blocs sont nécessaires pour gérer la communication en Bluetooth



On peut déployer la visualisation de ces blocs en effectuant un clic droit sur le bloc souhaité et en sélectionnant « **Développer le bloc** ».

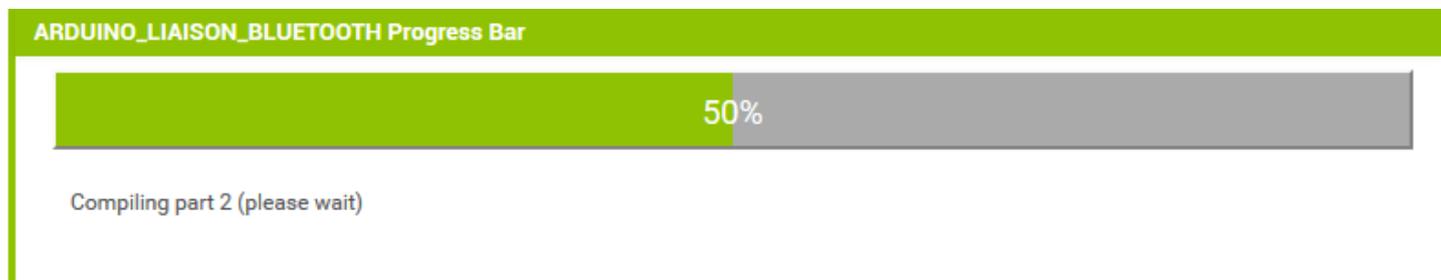
Transfert de l'application vers la tablette

Cliquer sur

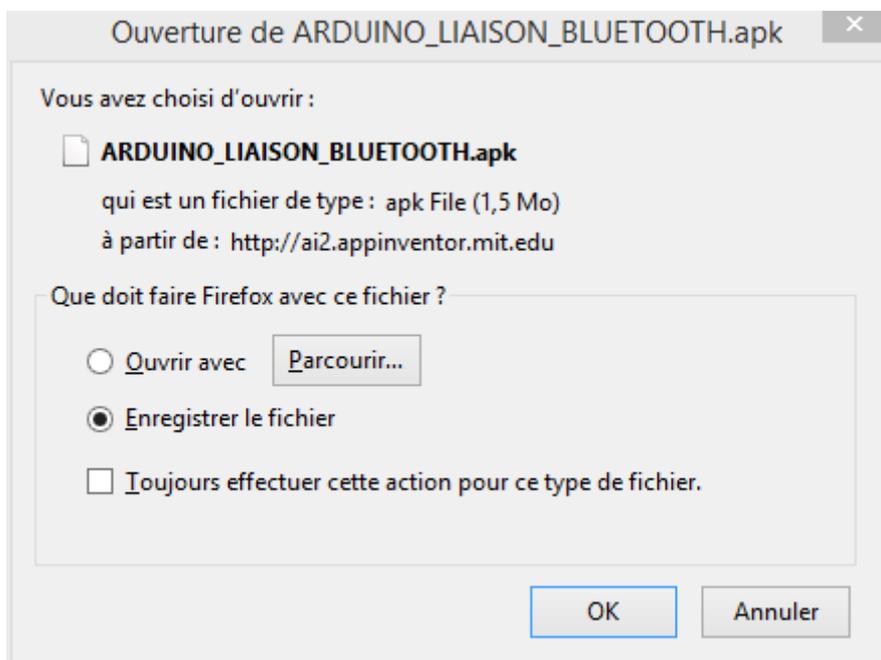
Construire ▾

Cliquer sur **App (enregistrer .apk sur mon ordinateur)** pour sauvegarder l'application

La barre suivante apparaît



Puis la fenêtre



Cliquer sur **OK**

Le fichier **ARDUINO_LIAISON_BLUETOOTH.apk** est sauvegardé dans le répertoire **Téléchargements**

Brancher la tablette sur l'ordinateur, à l'aide du cordon

Copier, à l'aide de l'explorateur Windows, le fichier **ARDUINO_LIAISON_BLUETOOTH.apk** sur la tablette (de préférence dans le répertoire **Document**)

Installer l'application

Débrancher la tablette

Lancer l'application **ARDUINO_LIAISON_BLUETOOTH** sur la tablette

Appairer la tablette et le module Bluetooth en tapant le code Pin choisi dans le fichier mBlock (**1234** par défaut)

Faire fonctionner l'application

Exercice 2

Un seul bouton visible pour allumer et éteindre la LED verte

1 - Travail dans mBlock (identique à l'exercice 1)

Travail à effectuer : répéter indéfiniment, si une donnée est disponible, lire la ligne reçue. Si le signal reçu est égal à 1 alors allumer la DEL verte (broche 3). Si le signal reçu est égal à 2 alors éteindre la DEL verte (broche 3).

Plus rapide, ouvrir le fichier **Bluetooth LED verte.sb2**

Téléverser le programme dans l'Arduino Uno

2 - Travail dans l'application App Inventor

Travail à effectuer :

- Le bouton **Allumer LED verte** permettra de mettre la broche 3 à l'état **haut** et d'allumer la LED verte par l'envoi du code « 1 » vers le module Bluetooth. Le bouton prendra alors le nom **Eteindre LED verte**.
- Le bouton **Eteindre LED verte** permettra de mettre la broche 3 à l'état **bas** et d'éteindre la LED verte par l'envoi du code « 2 » vers le module Bluetooth. Le bouton prendra alors le nom **Allumer LED verte**.
- Ajouter l'image **led verte.png**.
- Ajouter l'icône de l'application **led verte.png**.

Cliquer sur

Projets ▾

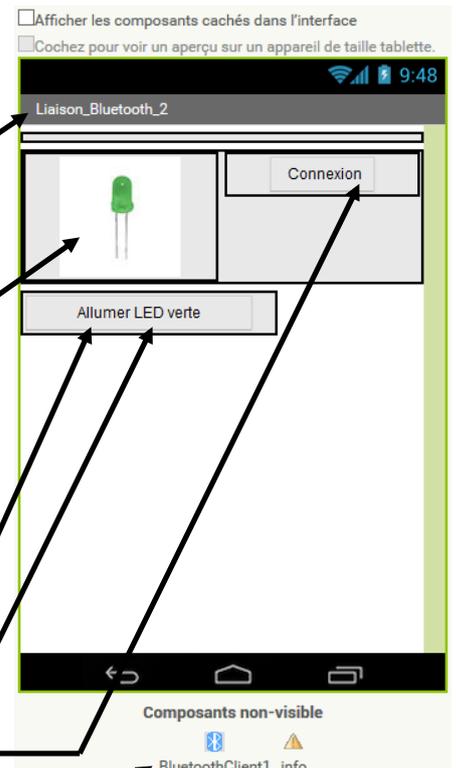
Cliquer sur

Importer le projet (.aia) de mon ordinateur ...

Sélectionner **ARDUINO_LIAISON_BLUETOOTH_2.aia**

Interface utilisateur sur le Smartphone ou la tablette

Cocher « Afficher les composants cachés dans l'interface » pour le faire apparaître



IDE App Inventor2 (fenêtre Designer)

Designer

Titre de l'application

Image à insérer **led verte.png**

Déclenchement de l'allumage de la DEL verte en envoyant le code « 1 »

Déclenchement de l'extinction de la DEL verte en envoyant le code « 2 »

Gestion de la communication en Bluetooth

Éléments non visibles de l'application

Les boutons **Connexion**, **Déconnexion**, **Allumer LED verte**, **Eteindre LED verte** sont déjà présents.

Ajouter l'image **led verte.png**

Ajouter l'icône de l'application **led verte.png**



Dans la fenêtre Blocks



Allumage de la LED verte

Lorsque vous cliquez sur le bouton nommé **LED_VERTE** :

- Le bouton prendra alors le nom **Eteindre LED verte**
- La valeur « 1 » est envoyée par liaison Bluetooth

```
quand LED_VERTE .Clic
faire
  si
    LED_VERTE . Texte = " Allumer LED verte "
  alors
    mettre LED_VERTE . Texte à " Eteindre LED verte "
    appeler BluetoothClient1 .Envoyer1Octet
      nombre 1
```

Extinction de la LED verte

Lorsque vous ne cliquez pas sur le bouton nommé **LED_VERTE** :

- Le bouton prendra alors le nom **Allumer LED verte**
- La valeur « 2 » est envoyée par liaison Bluetooth

```
sinon si
  LED_VERTE . Texte = " Eteindre LED verte "
alors
  mettre LED_VERTE . Texte à " Allumer LED verte "
  appeler BluetoothClient1 .Envoyer1Octet
    nombre 2
```

Résultat :

```
quand Allumer_LED_verte .Clic
faire
  si
    Allumer_LED_verte . Texte = " Allumer LED verte "
  alors
    mettre Allumer_LED_verte . Texte à " Eteindre LED verte "
    appeler BluetoothClient1 .Envoyer1Octet
      nombre 1
  sinon si
    Allumer_LED_verte . Texte = " Eteindre LED verte "
  alors
    mettre Allumer_LED_verte . Texte à " Allumer LED verte "
    appeler BluetoothClient1 .Envoyer1Octet
      nombre 2
```

Gestion de la communication en Bluetooth

Ces blocs sont nécessaires pour gérer la communication en Bluetooth

quand Screen1 .Initialise ...

quand Deconnexion .Clic fa...

quand info .Après choix ...

quand Connexion .Après pri...

quand Connexion .Avant pri...

On peut déployer la visualisation de ces blocs en effectuant un clic droit sur le bloc souhaité et en sélectionnant « **Développer le bloc** ».

Transfert de l'application vers la tablette

Identique que sur la page 10

Lancer l'application **ARDUINO_LIAISON_BLUETOOTH_2** sur la tablette

Appairer la tablette et le module Bluetooth en tapant le code Pin choisi dans le fichier mBlock (**1234** par défaut)

Faire fonctionner l'application

Exercice 3

Un seul bouton visible pour allumer et éteindre la LED verte

Un seul bouton visible pour allumer et éteindre la LED jaune

Un seul bouton visible pour allumer et éteindre la LED rouge

1 - Travail dans mBlock

Travail à effectuer : répéter indéfiniment, si une donnée est disponible, lire la ligne reçue.

Si le signal reçu est égal à 1 alors allumer la DEL verte (broche 3).

Si le signal reçu est égal à 2 alors éteindre la DEL verte (broche 3).

Si le signal reçu est égal à 3 alors allumer la DEL jaune (broche 5).

Si le signal reçu est égal à 4 alors éteindre la DEL jaune (broche 5).

Si le signal reçu est égal à 5 alors allumer la DEL rouge (broche 6).

Si le signal reçu est égal à 6 alors éteindre la DEL rouge (broche 6).

2 - Travail dans l'application App Inventor

Travail à effectuer :

Envoyer le code « 1 » si le bouton **Allumer LED verte** est sélectionné

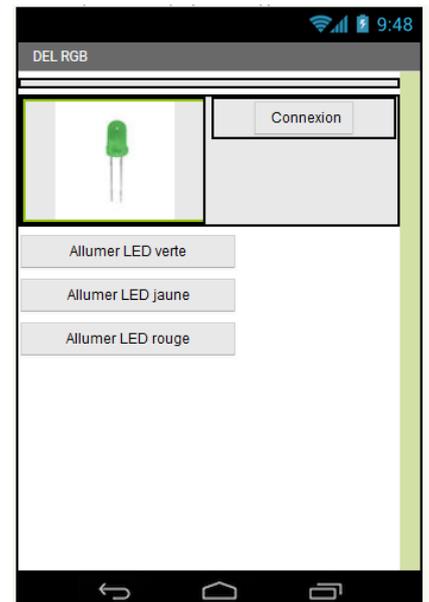
Envoyer le code « 2 » si le bouton **Eteindre LED verte** est sélectionné.

Envoyer le code « 3 » si le bouton **Allumer LED jaune** est sélectionné

Envoyer le code « 4 » si le bouton **Eteindre LED jaune** est sélectionné.

Envoyer le code « 5 » si le bouton **Allumer LED rouge** est sélectionné

Envoyer le code « 6 » si le bouton **Eteindre LED rouge** est sélectionné.



Transfert de l'application vers la tablette

Identique que sur la page 10

Lancer l'application **ARDUINO_LIAISON_BLUETOOTH_3** sur la tablette

Appairer la tablette et le module Bluetooth en tapant le code Pin choisi dans le fichier mBlock (**1234** par défaut)

Faire fonctionner l'application