



Seconde C.I.T

ETUDE de CAS n°2

*Enseignement d'exploration : CIT*

## ETUDE DE CAS n°2: Comment sécuriser sa chambre?

---



## Serrure Biométrique

---



Principe des activités :

Prendre connaissance des documents fournis. Se répartir les tâches dans l'équipe. A l'aide des éléments fournis (vidéo, texte, lien internet, etc.), rechercher les éléments de réponses concernant l'étude de cas.

**Activité 1 :**

Rechercher les noms des différents moyens, de sécuriser une entrée, présents sur cette page. Répondre dans les cadres bleus)

Rechercher les évolutions techniques qui ont permis de passer d'un type à l'autre. (Répondre dans les cadres orange)



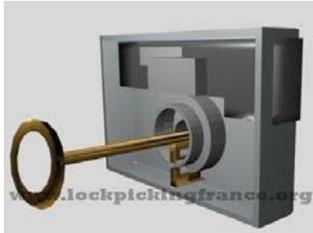









**Activité n°2** : rechercher le principe de fonctionnement de chacun de ces systèmes et le décrire en quelques mots ci-dessous :





### Activité 3 :

A partir du support didactique fourni (serrure digitale) ; Pourquoi a-t-on créé ce système ?

Donner le procédé technique utilisé dans ce système.

Quels autres types de biométrie auraient pu être utilisés ?

### Activité n° 4 :

Découverte du système didactique proposé (serrure biométrique). Enregistrer vos empreintes en suivant la démarche du dossier ressource. Choisir pour chaque élève un code différent et l'écrire ci-dessous :

---

---

---

Appeler l'enseignant pour les tests.

Faire la manipulation d'ouverture de la porte avec code. Quels sont les avantages / inconvénients de ces deux solutions.



Expliquer simplement ce qu'est la biométrie

Expliquer la qualité de la technique de lecture des empreintes digitales en termes de prix, d'efficacité, et d'efforts requis.

Quels autres types de biométrie auraient pu être utilisés ?

### **Activité 5 : Création d'une nouvelle serrure**

Le traitement des informations (code ou empreinte) se fait par programmation. L'activité consiste à découvrir ou revoir les bases de la programmation. Le choix du langage retenu pour cette étude de cas est le langage graphique de type scratch. Avant de programmer une nouvelle serrure, il faut s'assurer que vous avez les connaissances de programmation nécessaire.

Créer un compte en suivant le lien : <http://studio.code.org/join/QVRNJH>

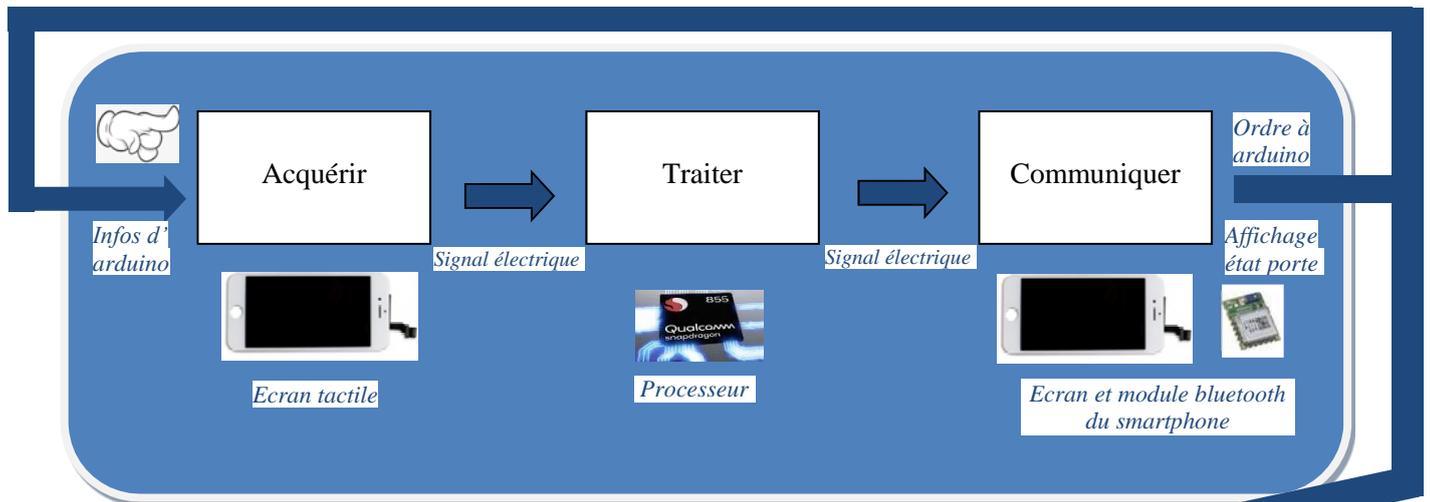
Faire complètement les étapes 2, 3, 6, 9, 10, 12, 14 et 16.



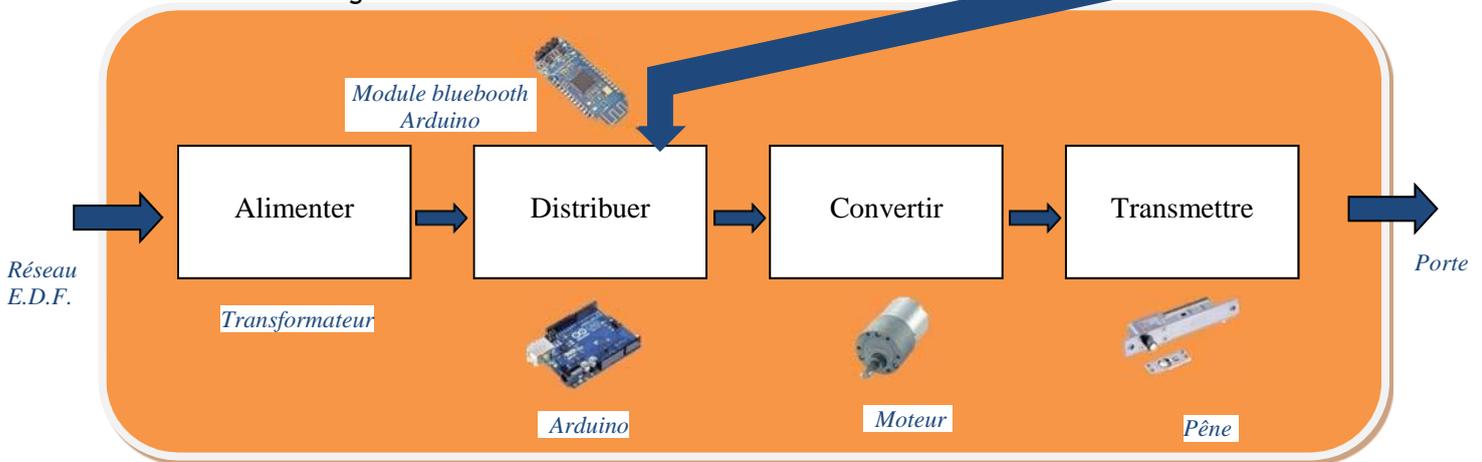
**Activité 6 :**

**Programmation d'une nouvelle serrure**

On se propose de réaliser une nouvelle serrure pilotée par smartphone. Cette nouvelle serrure sera composée de la chaîne d'information suivante :



Et de la chaîne d'énergie :



La programmation se fera avec l'univers APP INVENTOR, univers de programmation pour smartphone android créé par google et désormais entretenue par le Massachussets Institute of Technology (M.I.T.)



Tester l'univers d'app inventor à l'aide du mode opératoire [programmer un smartphone Android].



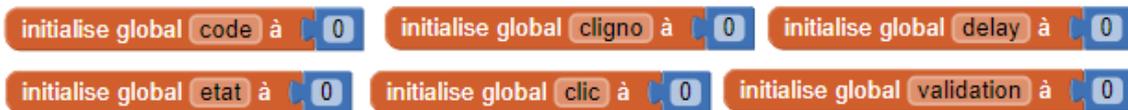
Pour notre serrure, nous allons faire deux écrans. Le premier sera un digicode (à gauche) et le second permettra de commander le pêne de la serrure (à droite).  
 Pour organiser les touches du digicode, il faudra placer par-dessus l'image des boutons transparents (sans couleur de fond) et les organiser avec [arrangement tableau] qui se trouve dans le menu « disposition » de la palette.

- Arrangement horizontal
- HorizontalScrollArrangement
- Arrangement tableau
- Arrangement vertical
- VerticalScrollArrangement

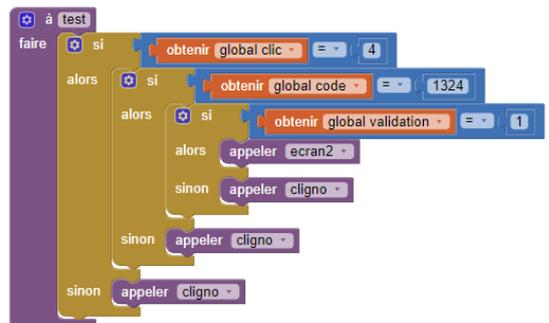
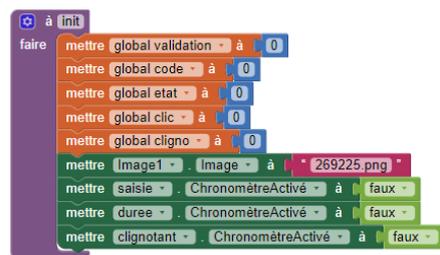
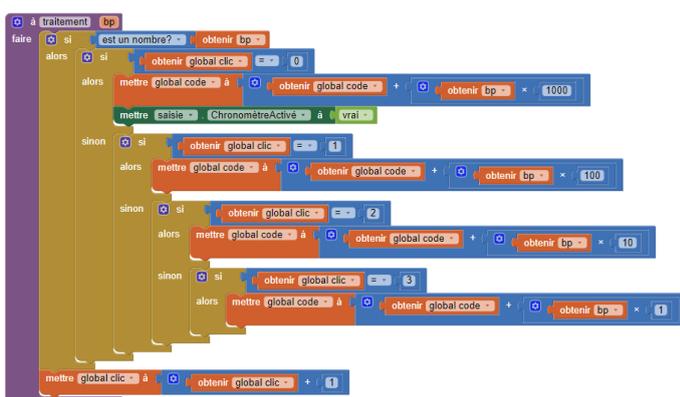


Les images utiles pour ces deux écrans sont dans le fichier compressé : [images app inventor].

Pour l'écran 1 (digicode), on commence par déclarer les variables :



On déclare ensuite trois procédures :





# Seconde C.I.T ETUDE de CAS n°2

On déclare maintenant les événements lorsque l'on appuie sur les boutons du digicode :

```

quand Bouton_1 - Clic
faire appeler traitement - bp 1

quand Bouton_2 - Clic
faire appeler traitement - bp 2

quand Bouton_3 - Clic
faire appeler traitement - bp 3

quand Bouton_4 - Clic
faire appeler traitement - bp 4

quand Bouton_5 - Clic
faire appeler traitement - bp 5

quand Bouton_6 - Clic
faire appeler traitement - bp 6

quand Bouton_7 - Clic
faire appeler traitement - bp 7

quand Bouton_8 - Clic
faire appeler traitement - bp 8

quand Bouton_9 - Clic
faire appeler traitement - bp 9

quand Bouton_0 - Clic
faire appeler traitement - bp 0

quand Bouton_diese - Clic
faire mettre global validation - à 0

quand Bouton_etoile - Clic
faire mettre global validation - à 1
  
```

Pour saisir le code, il y a un temps maximum. Il y a aussi un temps de clignotement pour la led rouge et la durée de chaque changement d'état de la led :

```

à bascule
faire si obtenir global etat = 0
alors mettre Image1 - Image - à "269240.png"
mettre global etat - à 1
sinon mettre Image1 - Image - à "269225.png"
mettre global etat - à 0

à ecran2
faire appeler init -
mettre Image1 - Image - à "cyan-off-buttons-vector-10969885-ConvertImage(1...)"
ouvre un autre écran Nom écran "Screen2"

à cligno
faire appeler bascule -
mettre clignotant - ChronomètreActivé - à vrai -
mettre duree - ChronomètreActivé - à vrai -

quand saisie - Chronomètre
faire appeler test -
si obtenir global cligno = 1
alors appeler init -

quand duree - Chronomètre
faire appeler bascule -

quand clignotant - Chronomètre
faire mettre global cligno - à 1
  
```

Enfin, il faut déclarer ce qui se passe à l'ouverture de l'écran :

```

quand Screen1 - Initialise
faire appeler init -
  
```

Pour l'écran 2 (commande de la porte), on va afficher un message porte ouverte ou porte fermée sur l'écran selon que l'on appuiera sur le bouton vert (ouverture) ou le bouton orange (fermeture).

Il faut donc mettre un label transparent (sans texte) sous les deux boutons de couleur. Enfin, le bouton blanc permettra si on fait un appui long dessus de fermer l'écran et protéger à nouveau l'application avec le digicode.

```

initialise global bascule à 0

quand Bouton3 - Clic long
faire ouvre un autre écran Nom écran "Screen1"

quand Screen2 - Initialise
faire mettre Label1 - Texte - à ""

quand Bouton1 - Clic
faire si obtenir global bascule = 0
alors mettre Label1 - Texte - à "Porte ouverte"
mettre Bouton1 - Image - à "boutonvert2.jpg"
mettre Bouton2 - Image - à "boutonorange.jpg"
mettre global bascule - à 1
sinon mettre Label1 - Texte - à "Porte fermée"
mettre Bouton1 - Image - à "boutonvert.jpg"
mettre Bouton2 - Image - à "boutonorange2.jpg"
mettre global bascule - à 0

quand Bouton2 - Clic
faire si obtenir global bascule = 1
alors mettre Label1 - Texte - à "Porte fermée"
mettre Bouton2 - Image - à "boutonorange2.jpg"
mettre Bouton1 - Image - à "boutonvert.jpg"
mettre global bascule - à 0
sinon mettre Label1 - Texte - à "Porte ouverte"
mettre Bouton2 - Image - à "boutonorange.jpg"
mettre Bouton1 - Image - à "boutonvert2.jpg"
mettre global bascule - à 1
  
```

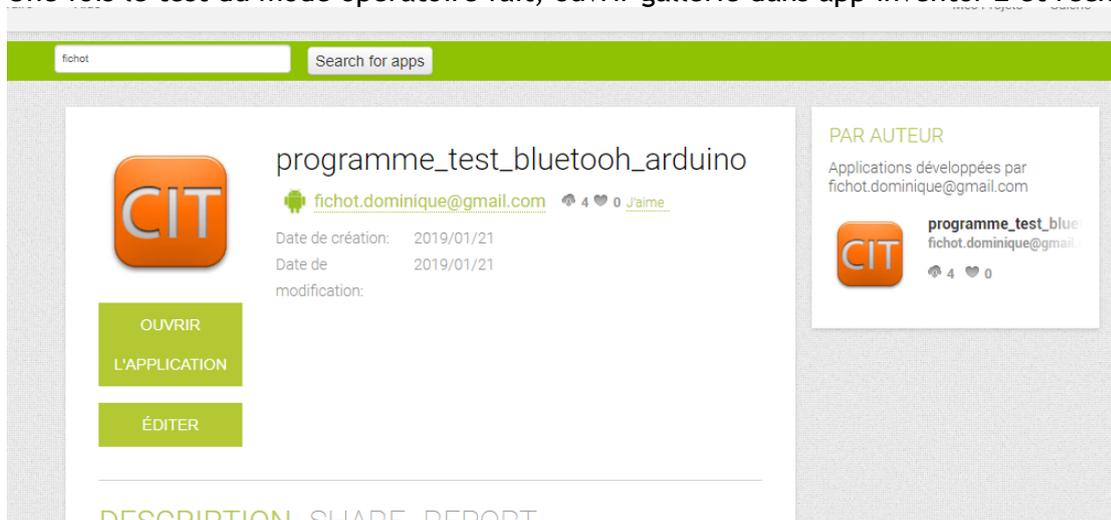


A l'aide du mode opératoire [tester un programme Android] vérifier :

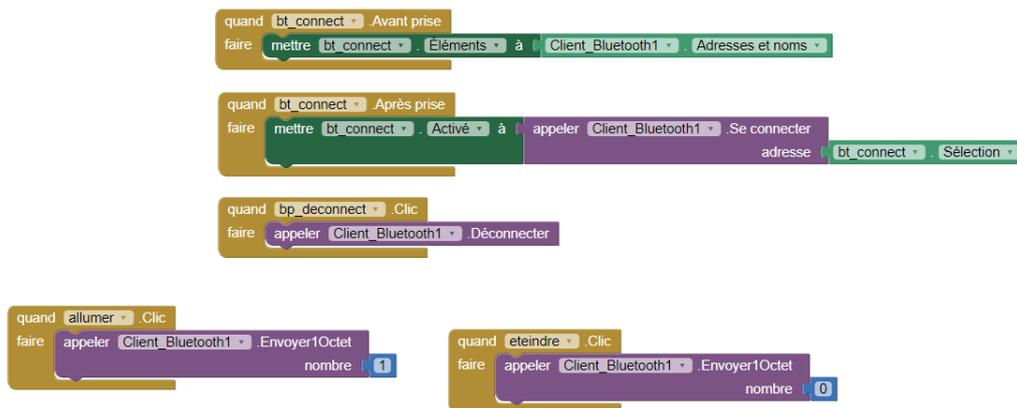
- Que la saisie du code + \* ouvre l'écran 2 sur le smartphone,
- Que l'appui long sur le bouton blanc fasse revenir à l'écran digicode.
- Que l'appui sur le bouton vert fasse changer son éclairage et fasse afficher « porte ouverte »
- Que l'appui sur le bouton orange fasse changer son éclairage, change l'état du bouton vert et affiche « porte fermée ».

Tester le mode opératoire [test d'acceptation Arduino]

Une fois le test du mode opératoire fait, ouvrir galerie dans app inventor 2 et rechercher :



Le programme contient deux écrans et le programme principal du programme de l'écran 2 contient :



- Avant prise montre la liste des connexion bluetooth visible depuis le smartphone.
- Après prise montre le rangement du nom de la carte bluetooth dans « client\_bluetooth1 »
- Allumer clic montre l'évènement lorsque l'on appuie sur le bouton allumer : cela envoie [1] depuis le smartphone vers l'arduino via le bluetooth.
- Eteindre clic montre l'évènement lorsque l'on appuie sur le bouton éteindre : cela envoie [0] depuis le smartphone vers l'arduino via le bluetooth.



Créer le programme suivant sur blockly :

```

initialisation (setup)
  mettre la variable echange à 0
  initialisation : bluetooth
    broche RX 6
    TX 7
    vitesse 9600

si bluetooth une donnée est disponible ?
alors mettre la variable echange à bluetooth donnée lue sur le port de communication
sinon
  si echange = 1
  alors mettre la DEL sur la broche Numérique 4 à l'état 1 (état haut)
  si echange = 0
  alors mettre la DEL sur la broche Numérique 4 à l'état 0 (état bas)
  
```

Téléverser ce programme sur Arduino.

Brancher le module Bluetooth sur votre grove sur D6. Le module clignote en vert.

Appairer votre smartphone avec la carte Bluetooth (HMSOFT), le code est 1234.

Ouvrir l'application app inventor selon la méthode choisie dans le mode opératoire [tester un programme android].

Passer à l'écran 2, cliquer sur connecter et choisir HMSOFT. La led du module Bluetooth doit cesser de clignoter et rester allumée en vert.

Cliquer alors sur allumer ou éteindre pour vérifier le bon fonctionnement du programme.

### A vous de faire:

Intégrer les éléments de programme nécessaire pour que l'application du digicode fonctionne pour allumer la led et l'éteindre avec les boutons vert et orange.



## Activité 7 : les principes d'innovation

Illustrer les principes d'innovations suivants :

### 4) Asymétrie

- Remplacer la forme symétrique d'un objet en une forme asymétrique
- Si l'objet est déjà asymétrique, renforcer son asymétrie

### 5) Fusion

- Grouper ou fusionner les objets identiques ou similaires (homogènes), assembler les parties identiques, destinées à des opérations parallèles ou contiguës
- Combiner, regrouper dans le temps les opérations homogènes ou contiguës

### 6) Multifonctions

- Rendre apte une partie de l'objet à réaliser plusieurs fonctions pour remplacer les fonctions des autres parties de l'objet

### 13) Inversion

- Inverser l'action utilisée normalement pour résoudre le problème
- Rendre fixes les pièces mobiles (ou l'environnement externe) et mobiles les parties fixes
- Retourner l'objet ou inverser le processus

### 14) Courbe

- Remplacer les droites par des courbes, les plans par des hémisphères, les cubes par des sphères, ...
- Utiliser des rouleaux, sphères, spirales, voûtes
- Remplacer les translations par des rotations, utiliser les forces centrifuges
- ...

### 15) Dynamisme

- Permettre ou prévoir l'ajustement des caractéristiques d'un objet (d'un processus, ou de l'environnement) pour rendre son action optimale ou pour se placer dans les meilleures conditions opératoires
- Diviser un objet en éléments pouvant se déplacer les uns par rapport aux autres
- Rendre flexible ou adaptable l'objet (ou le process) rigide ou non-flexible

### 16) Excessif ou partiel

- S'il est difficile d'obtenir le résultat à 100% d'une manière donnée, réaliser partiellement ou à l'excès l'action pourra simplifier considérablement le problème