PROJET DE SCIENCES DE L'INGENIEUR

SYSTEME D'APPRENTISSAGE AUTONOME DU VELO

Présenté par Paul Gaigne

PREMIÈRE PARTIE

- -Mise en place du projet
 - -Recherches multiples

Recherches

- -Établi comment on souhaitait que notre système fonctionne
 - -Niveau etc
- -Système pignon crémaillère pour effectuer le mouvement de translation
 - -Irréversibilité → recherche moteur

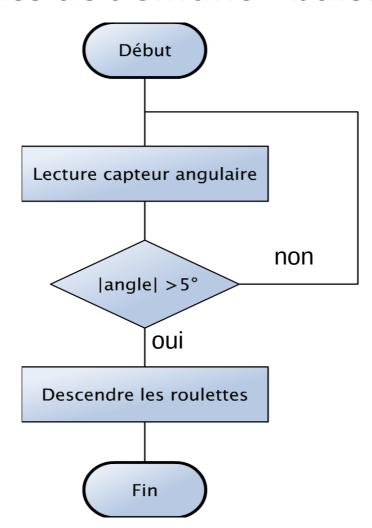
DEUXIÈME PARTIE

-Programmation sur Arduino

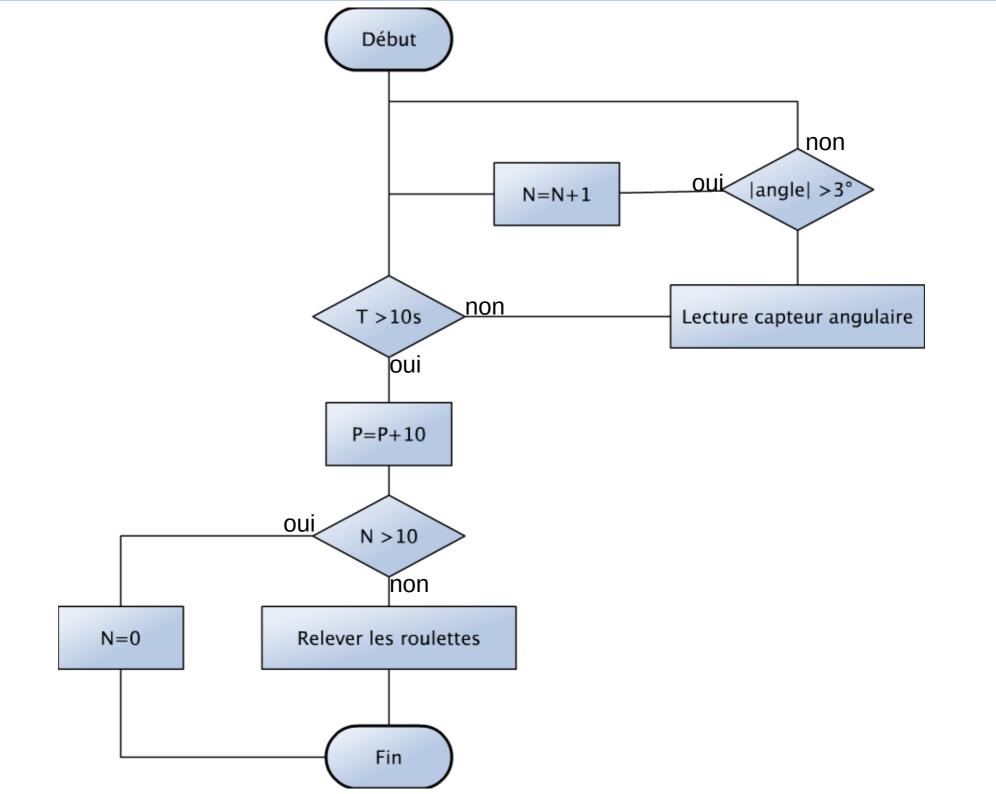
Programmation sur Arduino

- -Simplicité
- -Quelques base grâce au cours
- -Réponse à nos besoins: -Moteurs
 - -Capteur angulaire

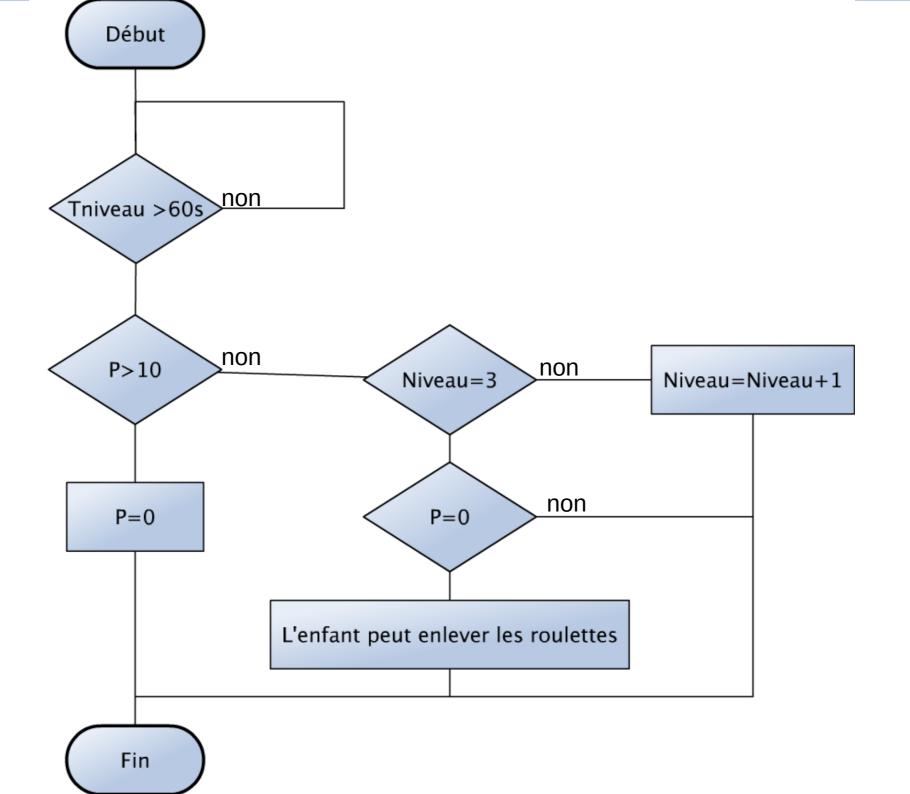
Detecter chute de l'enfant + baisser roulettes



Relever les roulettes



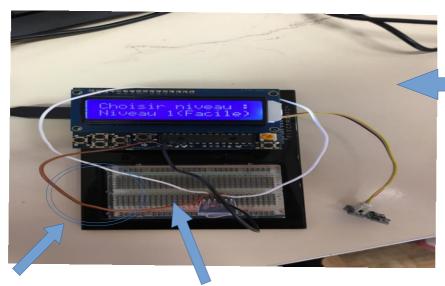
Adapter la hauteur des roulettes au niveau de l'enfant



Indiquer quand l'enfant est prêt a faire du vélo sans roulettes

Afficher les statistiques de la session

Écran:



Zone d'affichage (2x16)

Boutons : droite, gauche, haut et bas

Bouton select

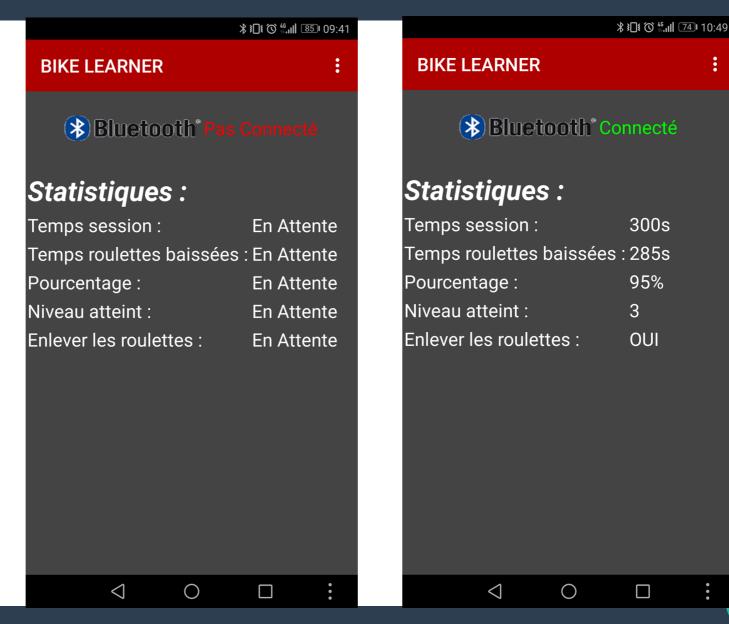
Transmettre les statistiques vers un smartphone

```
when Screen1 .BackPressed
    close application
     ListPicker1 *
                   .BeforePicking
    set ListPicker1 *
                      Elements
                                        BluetoothClient1 *
                                                           AddressesAndNames
                                   to
when ListPicker1
                   .AfterPicking
               call BluetoothClient1 *
                                     .Connect
                                                ListPicker1 *
                                                              Selection *
                                      address
           set ListPicker1 . Elements to
                                              BluetoothClient1 *
                                                                  AddressesAndNames
                                        IsConnected *
                     BluetoothClient1 *
                                               Connecté 1
                 set Label1 . Text
           then
                 set Label1 *
                               TextColor ▼
                                           Pas Connecté
           else
                     Label1 *
                                Text *
                                TextColor *
                 set Label1 *
```

```
initialize global input to
initialize global (ist) to ( create empty list)
 when Clock1 . Timer
 do
     (if
                  BluetoothClient1 *
                                    . IsConnected •
                                                    and *
                                                                 call BluetoothClient1 . BytesAvailableToReceive
            procedure2 •
      then
                                   call BluetoothClient1 . ReceiveText
            set global input . to
                                                       numberOfBytes
                                                                        call BluetoothClient1 . BytesAvailableToReceive
            set global list . to
                                split * text
                                               get global input *
                                         at [ " [] "
            set Label2 . Text . to I
                                        select list item list | get global list *
                                                   index
                                                           2
            set Label3 . Text to
                                        select list item list
                                                           get global list *
                                                    index
                                        select list item list
            set Label4 . Text to
                                                            get global list .
                                                           4
                                                   index
                                        select list item list | get global list *
            set Label5 . Text to
                                                           (5)
                                                   index
                           Text • to
            set Label6 •
                                        select list item list
                                                            get global list *
                                                           6
                                                    index
            if 🗑
                        Label6 -
                                  . Text - = - OUI
                   call Notifier1 . ShowMessageDialog
            then
                                                         Félicitations, votre enfant est maintenant prêt ...
                                             message
                                                        Félicitations |
                                                  title
                                            buttonText
                                                        - (OK) -
            (if
                        Label6 *
                                  . Text
                                                  . . . .
                  set Label6 •
                                             NON '
                                 Text *
            then
            set global list . to . create empty list
```

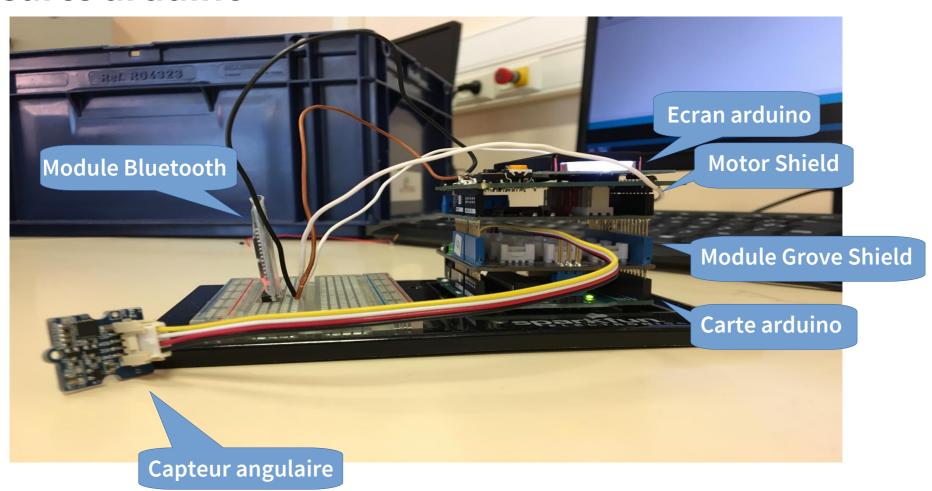
```
initialize global input to
initialize global (ist) to ( create empty list)
 when Clock1 . Timer
 do
     (F)
                                  . IsConnected and call BluetoothClient1 .BytesAvailableToReceive
                 BluetoothClient1 *
          gall procedure2 •
     then
                                             initialize global delaytime to 0
           set global input * to
                                                to procedure2
           set global list . to
                               split
                                                 set global delaytime v to call Clock1 v .SystemTime
                                                 set global delaytime v to ( 0)
                                                                          get global delaytime *
                                                                                               1000
           set Label2 . Text to
                                                 while test
                                                            do
           set Label3 * . Text * to
            set Label4 . Text . to
                                                 index 1
                                                        4
                                       select list item list | get global list *
           set Label5 . Text to
                                                        [5]
                                                 index
            set Label6 •
                         Text • to
                                       select list item list
                                                         get global list *
                                                 index
           if 🗑
                       Label6 . Text . = . OUI
                  call Notifier ... ShowMessageDialog
            then
                                                      Félicitations, votre enfant est maintenant prêt ...
                                           message
                                                      Félicitations
                                               title
                                          button Text
                                                      - (OK) -
                       Label6 • Text •
                  set Label6 •
                                Text *
            then
           set global list to [ ] create empty list
```

Application mobile qui permet de consulter directement sur son téléphone les statistiques de l'enfant



Résultat final

Carte arduino



Résultat final

