

INTRODUCTION TO REPRESENTATION AND REASONING IN AI

Template description



DEVELOPING AN ARTIFICIAL INTELLIGENCE
CURRICULUM ADAPTED TO
EUROPEAN HIGH SCHOOLS

2019-1-ES01-KA201-065742

erasmus.aiplus@udc.es



UNIVERSIDADE DA CORUÑA



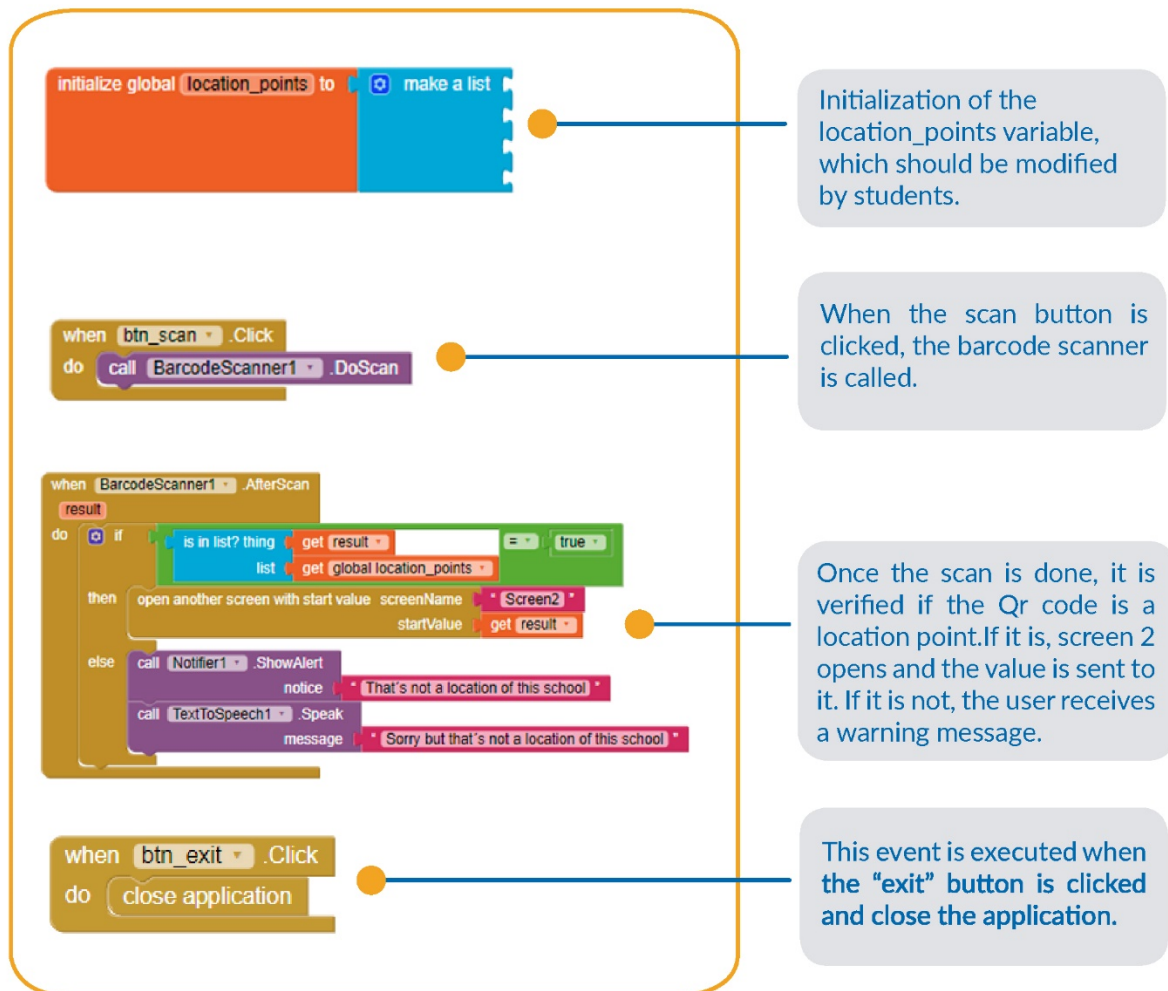
Co-funded by the
Erasmus+ Programme
of the European Union

Template description

To focus the attention in representation and reasoning, a template is provided to focus the attention in AI aspects. This template includes not only the graphic part of the app but also some pre-programmed blocks in order to make the programming easier for the students. This document aims to describe all the pre-programmed blocks of the template, organized in screen1 and screen2.

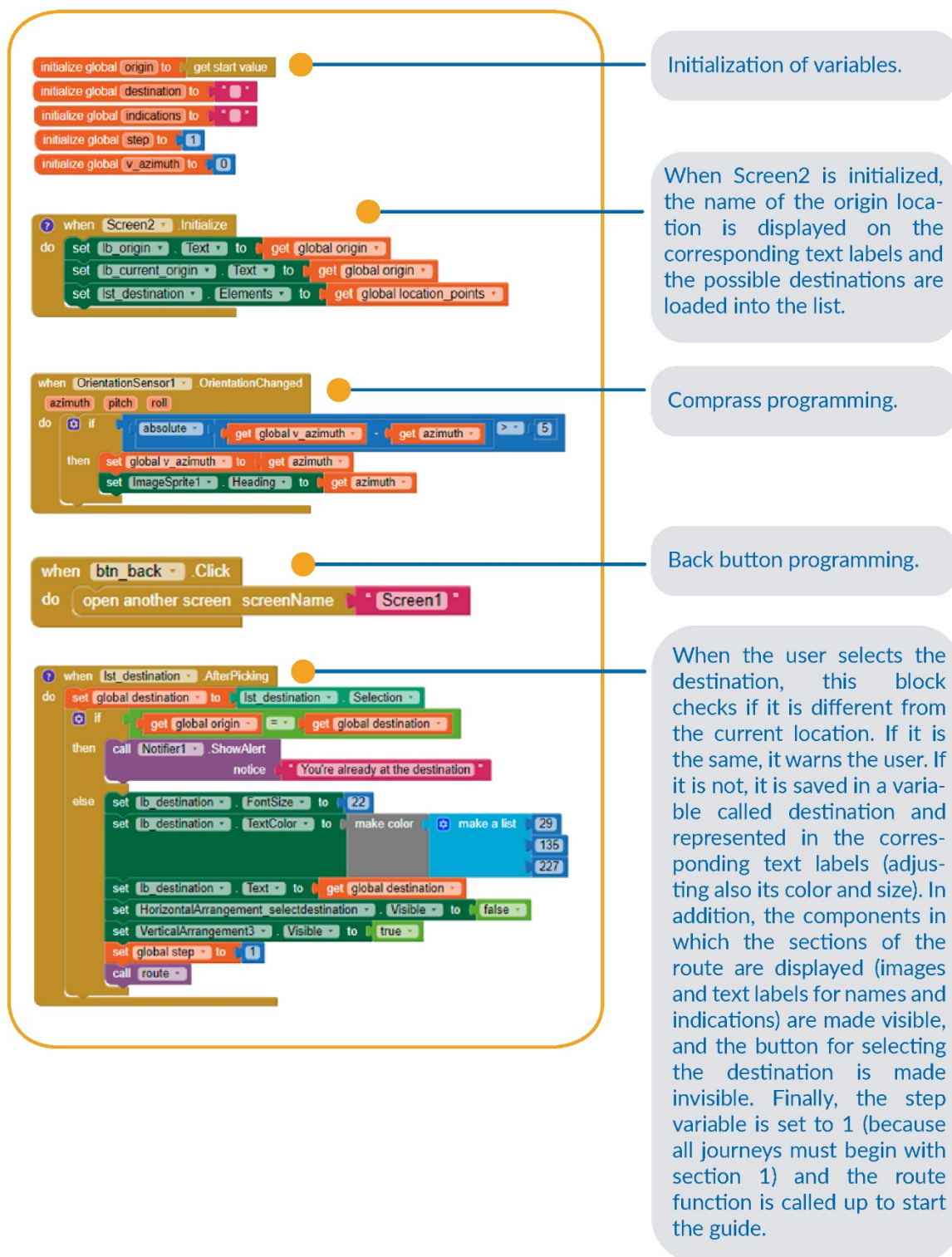
• Screen1

In this first window, all the blocks are completely programmed except for the first one, which the students must complete with the names of the location points of their school.



• Screen2

In this window, the blocks that should not be modified will be:



The screenshot displays the code for Screen2, organized into several functional blocks. The first block, 'Initialization of variables', sets up global variables for origin, destination, indications, step, and azimuth. The second block, 'When Screen2 is initialized', sets up text labels and a list of destinations. The third block, 'Compass programming', handles orientation changes and updates the azimuth and heading. The fourth block, 'Back button programming', opens Screen1 when the back button is clicked. The final block, 'When the user selects the destination', checks if the selected destination is the current location and either shows an alert or updates the destination display and starts the route.

Initialization of variables.

```

initialize global (origin) to (get start value)
initialize global (destination) to ( )
initialize global (indications) to ( )
initialize global (step) to (1)
initialize global (v_azimuth) to (0)
  
```

When Screen2 is initialized, the name of the origin location is displayed on the corresponding text labels and the possible destinations are loaded into the list.

```

when Screen2.Initialize
do
  set lb_origin.Text to (get global origin)
  set lb_current_origin.Text to (get global origin)
  set lst_destination.Elements to (get global location_points)
  
```

Compass programming.

```

when OrientationSensor1.OrientationChanged
  azimuth pitch roll
do
  if (absolute (get global v_azimuth) - (get azimuth) > 5)
  then
    set global v_azimuth to (get azimuth)
    set ImageSprite1.Heading to (get azimuth)
  
```

Back button programming.

```

when btn_back.Click
do
  open another screen screenName (Screen1)
  
```

When the user selects the destination, this block checks if it is different from the current location. If it is the same, it warns the user. If it is not, it is saved in a variable called destination and represented in the corresponding text labels (adjusting also its color and size). In addition, the components in which the sections of the route are displayed (images and text labels for names and indications) are made visible, and the button for selecting the destination is made invisible. Finally, the step variable is set to 1 (because all journeys must begin with section 1) and the route function is called up to start the guide.

```

when lst_destination.AfterPicking
do
  set global destination to (lst_destination.Selection)
  if (get global origin) = (get global destination)
  then
    call Notifier1.ShowAlert
    notice (You're already at the destination)
  else
    set lb_destination.FontSize to (22)
    set lb_destination.TextColor to (make color (make a list (29 (135 (227)
    set lb_destination.Text to (get global destination)
    set HorizontalArrangement_selectdestination.Visible to (false)
    set VerticalArrangement3.Visible to (true)
    set global step to (1)
    call route
  
```

```

when btn_nextsection .Click
do
  set global step to get global step + 1
  set btn_previoussection .Visible to true
  call route

```

When the user clicks on the next section button:

- The event adds 1 to the step variable.
- A button is activated to go to the previous step.
- The route function is called to show the following section.

```

when btn_previoussection .Click
do
  set global step to get global step - 1
  set btn_nextsection .Visible to true
  set btn_end .Visible to false
  call route

```

When the user clicks on the previous step button:

- The event subtracts 1 from the step variable.
- A button to go to the next step turns on, the end of route button turns off, and the route function is called.
- The route function is called to show the previous section.

```

when btn_speak .Click
do
  call TextToSpeech1 .Speak
  message get global indications

```

When the speak button is clicked, the directions are provided by voice through the speaker.

```

when btn_end .Click
do
  call TextToSpeech2 .Speak
  message "You have arrived at your destination, would you ..."

```

When this button is clicked, the user is asked if he/she wants to scan a new location.

```

when TextToSpeech2 .AfterSpeaking
result
do
  call SpeechRecognizer1 .GetText

```

When the user clicks on the arrival button, he is asked if he wants to scan his new location. This event handler runs when the sentence has been completed and calls the voice recogniser.

```

when SpeechRecognizer1 .AfterGettingText
result partial
do
  if
    upcase get result = "YES"
  then
    open another screen screenName Screen1

```

Check the captured word and execute the appropriate action. If the result is YES, screen 1 opens.

```

to first_step
do
  set btn_nextsection . Visible to true
  set btn_previoussection . Visible to false
  set btn_end . Visible to false

```

At the first step of each route it is necessary:

- To make visible the button to go to the next step
- Hide the button to go to the previous step
- Turn off the end button

```

to last_step
do
  set btn_nextsection . Visible to false
  set btn_end . Visible to true
  if get global step ≠ 1
  then
    set btn_previoussection . Visible to true
  else
    set btn_previoussection . Visible to false

```

At the last step of each route it is necessary:

- Make visible the button through which the user passes to the previous step (if there is more than one step).
- Hide the button that takes the user to the next step.
- Activate the end of route button.

In addition, if the total number of steps is one, the previous section button must be turned off.

```

to Print_origin_and_destination_images_names
  image_origin
  name_origin
  image_destination
  name_destination
  indications
do
  set global indications to get indications
  set Image_origin . Picture to get image_origin
  set lb_current_origin . Text to get name_origin
  set image_destination . Picture to get image_destination
  set lb_current_destination . Text to get name_destination
  set lb_indications . Text to get indications

```

This function represents the instructions to be followed in each section, as well as the source and destination names and images. This function serves to simplify the code.

The blocks that should be modified will be:

```

initialize global location_points to make a list

```

Initialization of the location_points variable, which should be modified by students.

