

# RACING AND WRESTLING



## STEMJAM Teaching Guide

Developing make spaces to promote creativity  
around STEM in schools

Acronym: STEMJAM

Project no. 2016-1-ES01-KA201-025470

[www.stemjam.eu](http://www.stemjam.eu)



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

# RACING AND WRESTLING

## ABSTRACT

The controlling mBots by Bluetooth and wireless technology. Preparing the rough surface for mBots wheels to prevent skidding. Preparing the fields for mBots racing and wrestling.

Through a game programmed in Scratch (mBlock Software), the students will have to answer different questions, if they succeed the mBot will advance, but if they fail, it will go back. But there is more, the two mBot will be linked through a rope, so the first one that manages to push the other mBot towards his field, will win the battle.

## DIDACTIC OBJECTIVES

### TECHNOLOGY

- ❖ Writing the necessary codes by keyboard control.
- ❖ Wireless technology knows how to control the mBot.
- ❖ Bluetooth and wireless technology controls the mBot on narrow road and bordered field.
- ❖ Student's interest increases with amusing learning.

Learn to program a game, know general culture questions and the strategy to answer before or after the rival team.

STEM Subject:    Science             Technology             Engineering             Mathematics

Education Level:            12-14 years             14-16 years

## PROBLEM STATEMENT

Create a video game with Scratch and at the same time be connected to the mBot.



## BOM (Bill of Materials Needed)

➤ (x2) mBots => Ref. 90054



❖ Different beams and structures:



- ❖ Cardboards, pens, painting, etc.
- ❖ Colourful tapes.
- ❖ Rough Surface.
- ❖ A Rope.

## ACTIVITY DESCRIPTION

### First version

Step 1: Rough surface is prepared for comfortable movement of mBot.

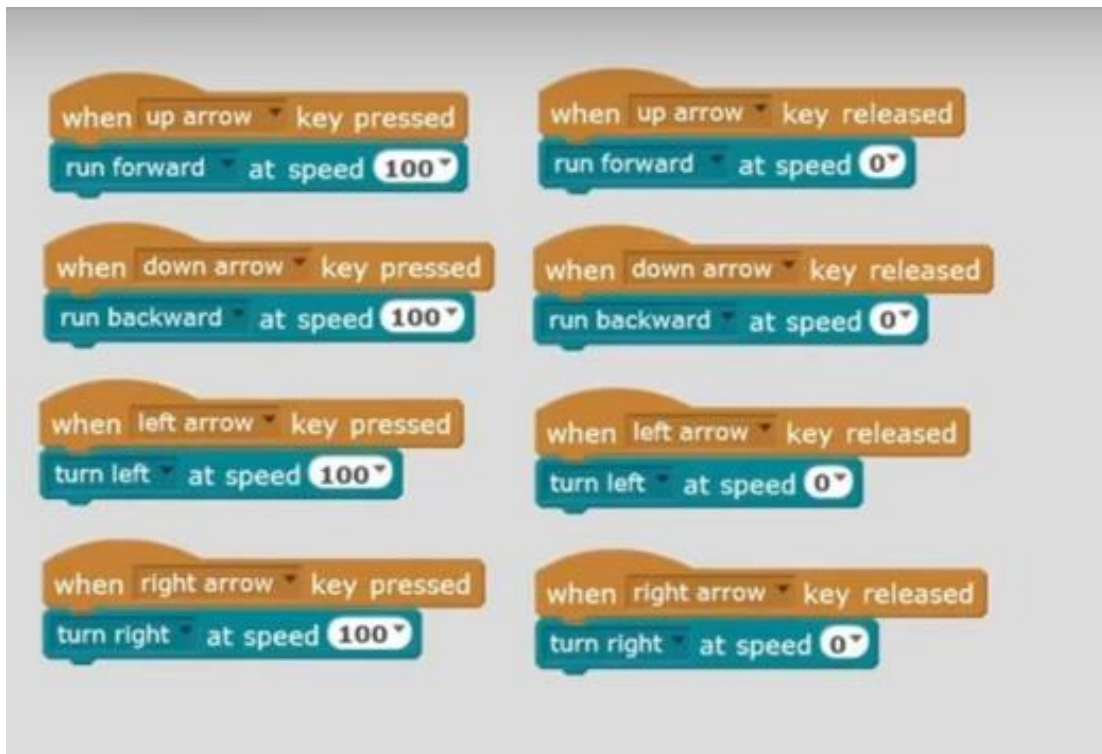




Step 2: Information about areas of racing and wrestling



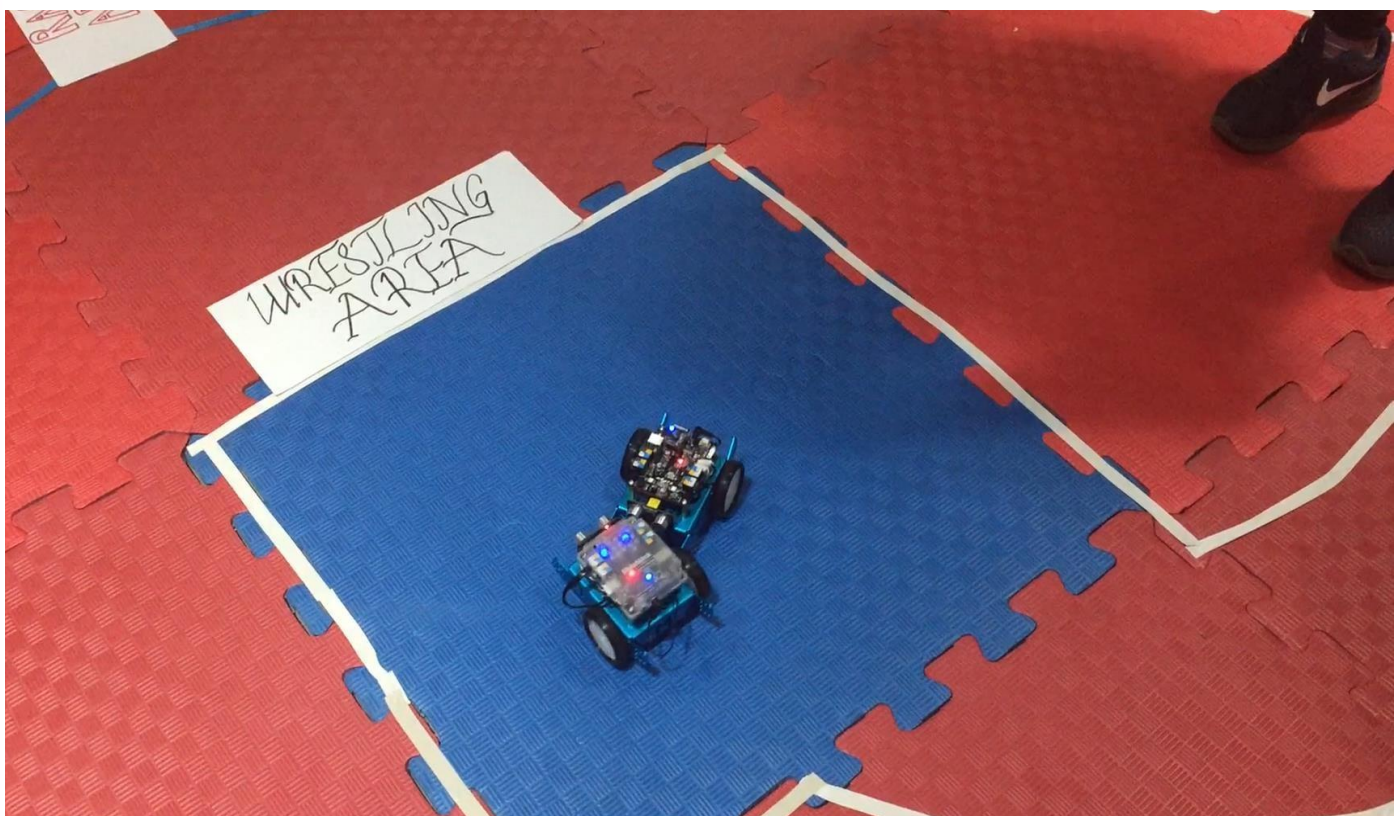
Step 3: To Code for Wireless connection and to set Bluetooth connection



The image shows a collection of Scratch code blocks arranged in two columns. The left column contains four blocks: 'when up arrow key pressed' followed by 'run forward at speed 100'; 'when down arrow key pressed' followed by 'run backward at speed 100'; 'when left arrow key pressed' followed by 'turn left at speed 100'; and 'when right arrow key pressed' followed by 'turn right at speed 100'. The right column contains four corresponding blocks: 'when up arrow key released' followed by 'run forward at speed 0'; 'when down arrow key released' followed by 'run backward at speed 0'; 'when left arrow key released' followed by 'turn left at speed 0'; and 'when right arrow key released' followed by 'turn right at speed 0'.



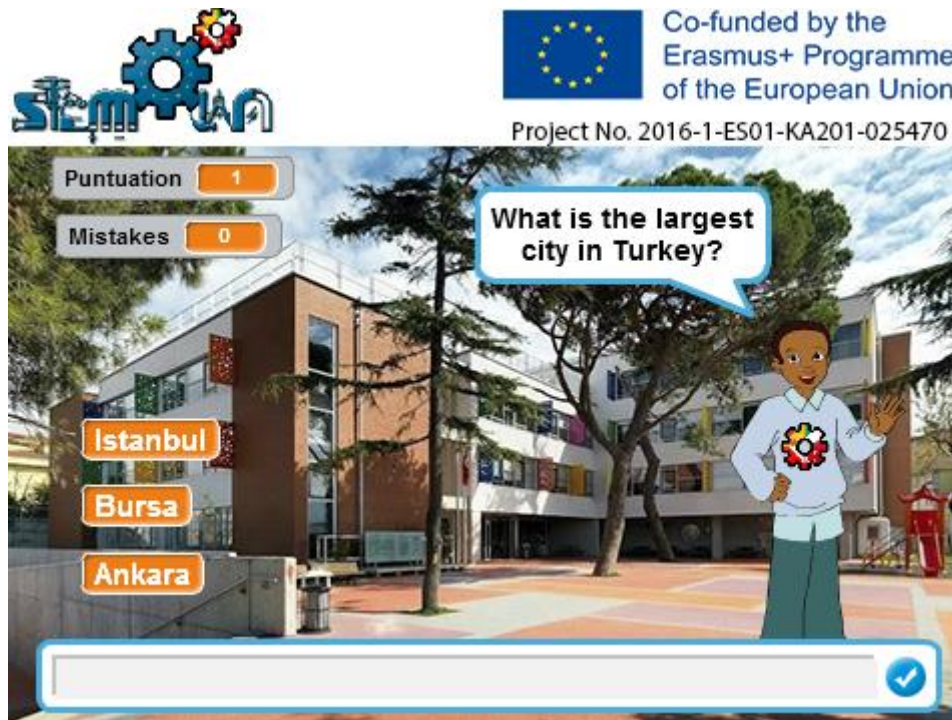
Step 4: Improvement of mBot control ability.



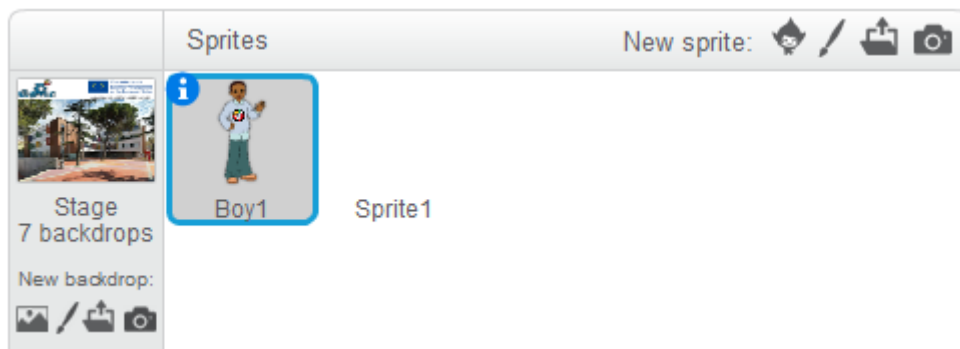


## Second version

First, we will decide what kind of game we want to play. In this case, we have opted for a game of questions and answers (quiz game).



Now, we start to program the game code. First, select the background and the sprite that we want to display in the game:



Scripts

Costumes

Sounds



New costume:



boy1-a



Clear

Import



STEMJAM L...  
108x54



200%

Vector Mode  
Convert to bitmap

Then, we initialize the variables:

The image shows the Scratch script editor with the 'Scripts' tab selected. The left sidebar shows the 'Data & Blocks' category, with a 'Make a Variable' button and a list of variables: Box1, Box2, Box3, Mistakes, Puntuation, mBot\_SpeedBack, mBot\_SpeedGo, time\_correct, and time\_mistake. The main workspace contains a script starting with 'when green flag clicked', followed by several 'set' blocks for Puntuation, Mistakes, Box1, Box2, Box3, mBot\_SpeedGo, mBot\_SpeedBack, time\_correct, and time\_mistake. It also includes 'hide variable' blocks for Box1, Box2, and Box3, and 'set led on board' and 'show drawing' blocks. The script ends with a 'broadcast Game Start' block.

```
when green flag clicked
  set Puntuation to 0
  set Mistakes to 0
  set Box1 to 
  set Box2 to 
  set Box3 to 
  set mBot_SpeedGo to 255
  set mBot_SpeedBack to -100
  set time_correct to 2
  set time_mistake to 0.5
  hide variable Box1
  hide variable Box2
  hide variable Box3
  set led on board all red 0 green 0 blue 150
  show drawing Port4 x: 0 y: 0 draw: 
  broadcast Game Start
```

The next step is program the welcome to the game:

```
when I receive Game Start
say Hello!! for 2 secs
say Welcome to the MBOT great contest for 3 secs
ask What is the name of your team? and wait
repeat until not answer = 
  ask Please, insert the name of your team and wait
say join Great answer for 2 secs
say Let's GO!! for 2 secs
show variable Box1
show variable Box2
show variable Box3
broadcast Question1
```



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

Project No. 2016-1-ES01-KA201-025470



We establish the questions, the different options and what will happen if it is correct or wrong:

```
when I receive Question1
  set Box1 to Paella
  set Box2 to Pizza
  set Box3 to Chicken Soup
  ask What is the favourite food in Valencia (Spain)? and wait
  if answer = Paella or answer = paella then
    change Puntuation by 1
    play tone on note C5 beat Half
    set led on board all red 0 green 150 blue 0
    say Good!! for 2 secs
    set motor M1 speed mBot_SpeedGo
    set motor M2 speed mBot_SpeedGo
    wait time_correct secs
    set motor M1 speed 0
    set motor M2 speed 0
  else
    change Mistakes by 1
    play tone on note C3 beat Half
    set led on board all red 150 green 0 blue 0
    say No :( for 2 secs
    set motor M1 speed mBot_SpeedBack
    set motor M2 speed mBot_SpeedBack
    wait time_mistake secs
    set motor M1 speed 0
    set motor M2 speed 0
  broadcast Question2
```

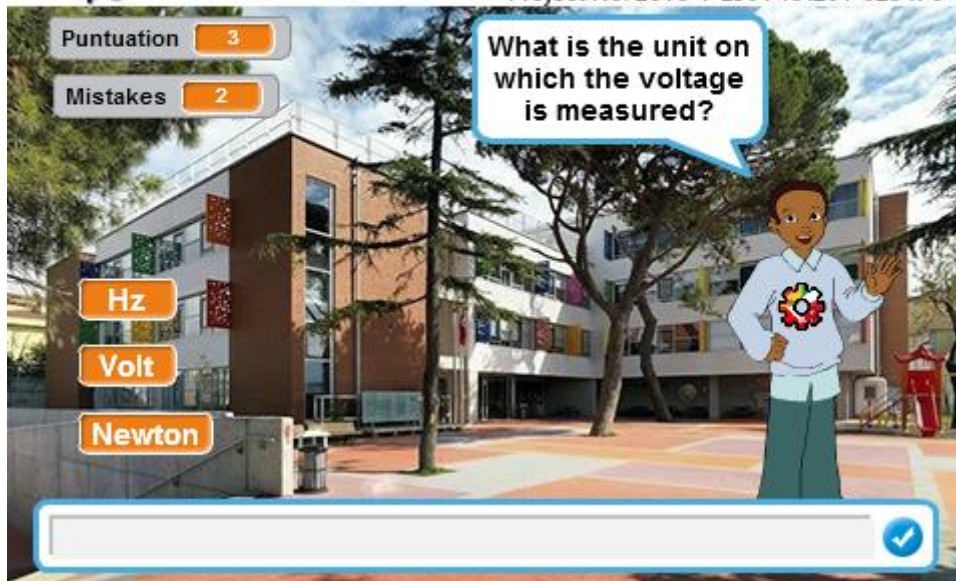
The answers options will display in option's boxes and our sprite ask the question.

If user answer correct, it acquires one point, and mBot run forward during 3 seconds. In the other case, if user answer wrong, it acquires one mistake and mBot run backward during 0.5 seconds.



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

Project No. 2016-1-ES01-KA201-025470



If the user has answered all the questions in the correct way, a message of “CONGRATULATIONS!!! YOU WIN!!!”. If the puntuation is higher than mistakes, will display “Nice!! Your points are higher than your mistakes”. In other case, “You need to learn more” message will appear.

```
when I receive Final
if Puntuation = 10 then
  say CONGRATULATIONS!!! YOU WIN!!! for 20 secs
if Puntuation > Mistakes then
  say Nice!! Your points are higher than your mistakes for 20 secs
else
  say You need to learn more for 20 secs
```



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

Project No. 2016-1-ES01-KA201-025470

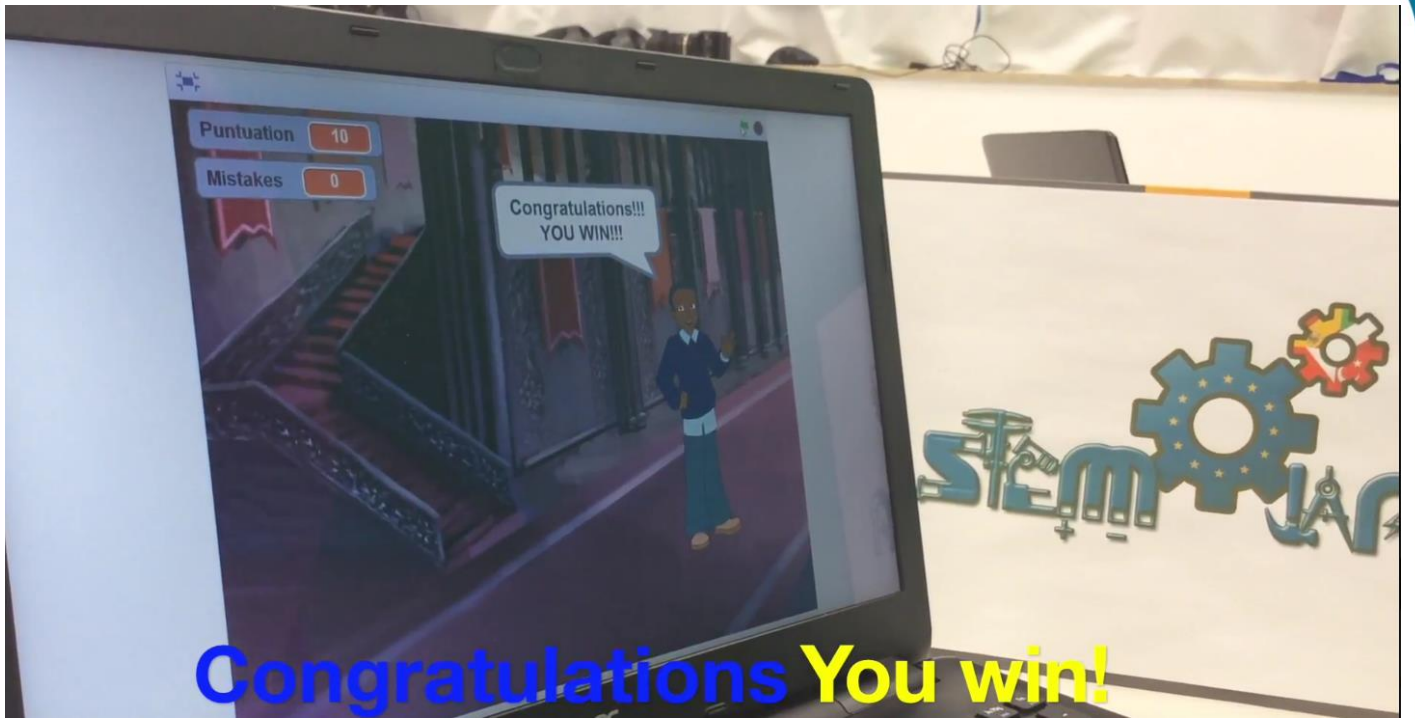


Once we have programmed the game, we must match a mBot to a computer, and the other mBot to the other computer.

Also, with a rope, match one mBot with the other.



Once we have it .....**May the best win!**



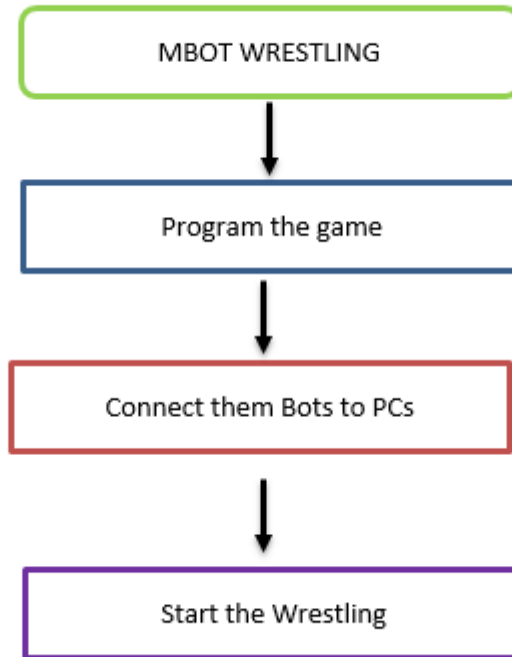
**Congratulations You win!**



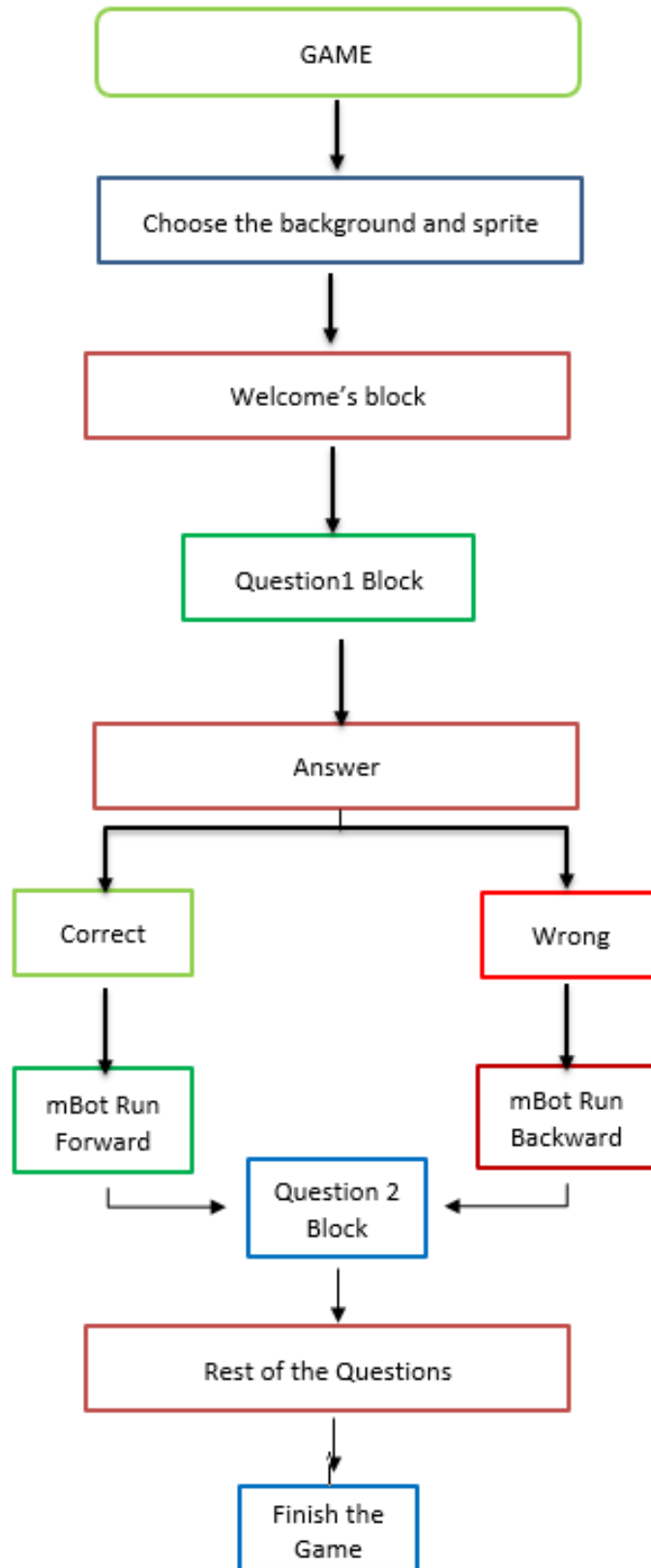
## FLOW CHART

### Second version

Schema of this activity:



Schema of game:



## STUDENT'S EVALUATION

20 Stemjam team members participated in this project.

Project was completed in one week.

All students in the project are willing and excited.

## BIBLIOGRAPHY

<https://scratch.mit.edu/explore/projects/all>

## MORE INFORMATION

The game interface and the code is explain more detailed in “Developing videogames with Scratch” presentation.

