

Playbot4all project



Ariel Ratzonel



Matias Salaris

Computer Science and Engineering students at Politecnico di Milano

Project's goal

The purpose of this project was to design, develop and build a robot puppet with which every child could play, in particular keeping in mind the characteristics that such a game should have to entertain and stimulate the children's imagination with learning difficulties

We have therefore developed a soft and aesthetically pleasing toy, able to emit sounds and move in space, combined with an easily understandable and scalable game logic to various levels of difficulty.

The robot puppet

Main characteristics:

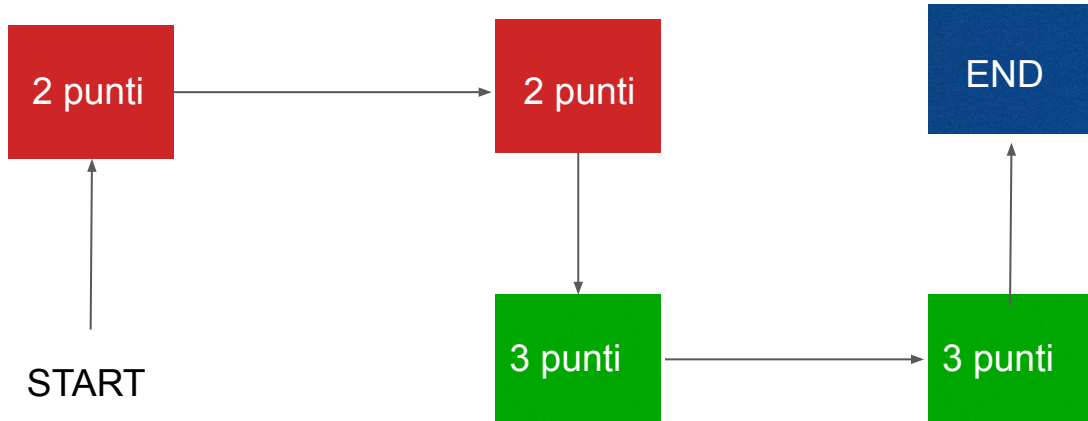
- Softness
- Impact resistance
- Simple and pleasing aesthetics



Game's rules

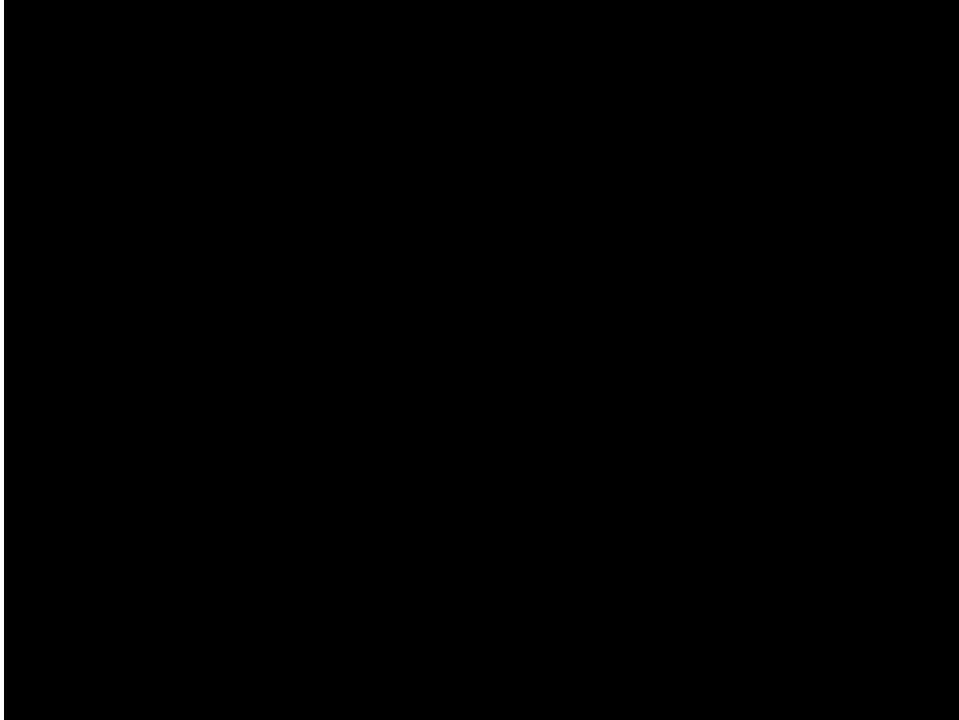
Goal: Place the red and green signs to make the robot reach the destination and score more points

- Red sign: Makes the robot turn right and guarantees 2 points
- Green sign: Makes the robot turn right and guarantees 3 points
- Blue sign: Destination



Example route

Example of functioning (Video)

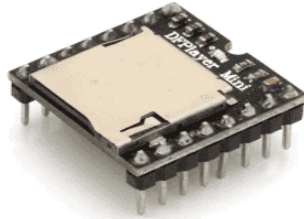


Material: Pixy Cam



Pixy allows you to photograph a color and assign it a label, every time pixy is running and "sees" the color it has saved, recognizes it and makes the association with the given label, this allows the robot to do different behaviors based on the color seen.

Material: DFPlayer Mini



The robot, in addition to moving, emits sounds. The sounds announce the turns (right / left), the beginning and the end of the game. For the management of the sounds we used a Df player mini, which is a device that allows you to insert an SD card inside. We recorded the sounds to be emitted by the robot and saved them in the SD card in a order, since in the code to access the sounds you put the index of the sound saved in the card.

Material: Motors e Encoder



The robot moves thanks to two wheels which are powered by two motors. The speed of the wheels is regulated by the encoders, which have the task of counting the revolutions of the wheels in order to know when to increase / decrease the revolutions when needed.

Implementation choices

Every hardware component led us to face software problems to be solved.

The main implementation choices were:

- Development of an Adjust() function which increases the speed of a wheel, temporarily, when the robot tends to go to one side (a situation that can be due to numerous factors: Low batteries, imperfections in the ground, etc.)
- Making sure that the camera picks up color blocks every 100 frames (the best our specific camera could do before starting to run into bugs).
- Managing the audio through the DFRobotDFPlayerMini.h library