HEROKIDNA

interdisciplinary course of

Design and Robotics

8° edition, 2020

Project:
Herokidna

Professors:
Andrea Bonarini, DEIB department, Politecnico di Milano
Maximiliano Romero , Design department, IUAV, Venice University

Tutors:
Rohan Vangal

Students:
Matteo Velati, school of Engineering
Jalaladdin Abbasi, school of Engineering
Juliana Campos, school of Design
Stefano Agresti, school of Engineering
Sofia Wiener, school of Design

Abstract

This work aimed to design and manufacture a functional robot for a battle tournament. The work was developed merging the design and engineering knowledge in an iteration process where design and engineer had to keep a strong flux of communication during the process of building. Most of the work was done remotely due to external conditions apart from the final set of the robot.

Phase 1: Discover

The first part of the work was to seek an effective form of remote communication, due to the complicated circumstances caused by the coronavirus pandemic. We had to find tools that could provide us with an organization and a pace of work monitoring, eg: Miró; Drive; chronograms etc. This was necessary to have a fluid and common flow of language and communication.
On top of that, we used this first phase to choose which animals we would be taking into consideration as inspiration for our robot.

Team Organization

Our team is composed of five members: Stefano Agresti, the team leader and computer science engineer; Sofia Wiener, product designer doing integrated design master; Juliana Campos, product designer doing PSSD mater; Jalaladdin Abbasi, mechanical engineer; Matteo Velati, computer science engineer. We worked in multiple tasks, trying to help each other in problem solving with our expertises.

A general report of how we divided our tasks is the following:
Stefano Agresti: Stefano participated as team leader and in the development of the firmware for the Raspberry Pi board. In early stage, he took care of the sensors' selection, while in late stage he handled the organization of tests on the various components. He realized the first prototype and, throughout the development, he coordinated the team's efforts.

Sofia Wiener: Sofia participated in the study part of the concept between the mixture of animals, seeking to combine different techniques and studying how they could be applied to the robot. She was also responsible for its design and technical development, such as 3D modeling with technical specificity of size and details to accommodate the internal components. She also took charge of producing the robot through 3D printing and assembling it.

Juliana Campos: Juliana participated in the concept design of animals, looking for possible combinations between them and making the first sketches to understand these merges. She also participated in the communication material during all phases of the project, through the presentation of ideas; follow up of the work; video storyboard and design of the logo. Worked with Matteo in the development and study of emotional expressions in search of developing the interface that would work like the eyes of the robot.

Jalaladdin Abbasi: Jalaladdin participated at first stages in proposing ideas regarding the mechanisms that could be used in defence, attack and drinking from the pond status. Then he made some prototypes in collaboration with Stefano for checking the feasibility of the tongue strategy for attacking. Then he designed a 3D model as an initial estimation of the internal mechanisms of the robot. At the same stage he did some calculations regarding the torque needed for each part of the robot. According to the calculations he decided the type and the number of motors needed and he checked their availability in the market. He again with the collaboration of Stefano, did prototyping and testing of the neck mechanism and rotating target mechanism.

Matteo Velati: Matteo participated in the development of the firmware for the Raspberry Pi module inside the robot. From a high-level point of view, he determined the best strategies to adopt in each different situation the robot might find itself in.
He also participated in the selection of components and hardware design, cooperating with Stefano, and finally realizing the electrical wiring and assembling the robot.

Project Management

Our team followed GANTT as the main organization of the tasks, updated weekly.
This was our first planned GANTT:

Task	Sub task	resp.	Week 01	Week 02	Week 03	Week 04	Week 05	Week 06	Week 07	Week 08	Week 09	Week 10
			17/03	24/03	31/03	07/04	14/04	21/04	28/04	05/05	12/05	19/05
Discover	Animal 01 Behaviour and chracteristics	Designer 01		Juliana								
	Animal 02 Behaviour and characteristics	Designer 02		Sofia								
	Sensors	Eng01		Stefano								
	Actuators	Eng02		Jalal								
	Possible movements and strategies	Des/Eng		Matteo								
Define	Animal 01 Behaviour and chracteristics	Designer 01			Juliana							
	Animal 02 Behaviour and characteristics	Designer 02			Sofia							
	Sensors and SW architecture	Eng01			Stefano							
	Actuators and SW architecture	Eng02			Jalal							
	Possible movements and strategies	Des/Eng			Matteo							
Develop	Animal 01 Body	Designer 01				Juliana	Juliana	Juliana	Juliana	Juliana		
	Animal 02 Body	Designer 02				Sofia	Sofia	Sofia	Sofia	Sofia		
	HW and SW	Eng01				Stefano	Stefano	Stefano	Stefano	Stefano		
	HW and SW	Eng02				Matteo	Matteo	Matteo	Matteo	Matteo		
	HW (Body)	Des/Eng				Jalal	Jalal	Jalal	Jalal	Jalal		
Deliver	Integrated Body - Presentation	Designer 01									Juliana	Juliana
	Integrated Body - Presentation	Designer 02									Jalal	Jalal
	Electronics and SW - Manual	Eng01									Stefano	Stefano
	Electronics and SW - Manual	Eng02									Matteo	Matteo
	Integrated Body - Project (CAD)	Des/Eng									Sofia	Sofia
Lessons			17/03 09:15 (8hs) Lessons and brief	24/03 9.15 (8hs) Lessons. Discovery and Revision	31/03 09:15 (8hs) Definition and revision	07/04 09:15 (8hs) Development	14/04 Pasqua	21/04 9.15 (8hs) Development presentation and revisions		05/05 9.15 Development presentation and revisions		19/5 9.15 Final presentations
Revison			17/03 16:00 Brief definition and first ideas	24/03 16:00 Discovery presentation	31/03 16:00 Selection of the features and technology. Presentation	07/04 Revisions	(No lesson)	21/04 Revisions				

Due to the pandemic, together with the professors, the schedule was changed adding four weeks to the development process.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
Task	Sub task	resp.	Week 01	Week 02	Week 03	Week 04	Week 05	Week 06	Week 07	Week 08	Week 09	Week 10	Week 11	Week 12	Week 13	Week 14
			17/03	24/03	31/03	07/04	14/04	21/04	28/04	05/05	12/05	19/05	26/05	02/06	09/06	17/06
Discover	Animal 01 Behaviour and chracteristics	Designer 01		Juliana												
	Animal 02 Behaviour and characteristics	Designer 02		Sofia												
	Sensors	Eng01		Stefano												
	Actuators	Eng02		Jalal												
	Possible movements and strategies	Des/Eng		Matteo												
Define	Animal 01 Behaviour and chracteristics	Designer 01			Juliana											
	Animal 02 Behaviour and characteristics	Designer 02			Sofia											
	Sensors and SW architecture	Eng01			Stefano											
	Actuators and SW architecture	Eng02			Jalal											
	Possible movements and strategies	Des/Eng			Matteo											
Develop	Animal 01 Body	Designer 01				Juliana	Juliana	Juliana	Juliana	Juliana	Juliana	Juliana	Juliana	Juliana		
	Animal 02 Body	Designer 02				Sofia	Sofia	Sofia	Sofia	Sofia	Sofia	Sofia	Sofia	Sofia		
	HW and SW	Eng01				Stefano	Stefano	Stefano	Stefano	Stefano	Stefano	Stefano	Stefano	Stefano		
	HW and SW	Eng02				Matteo	Matteo	Matteo	Matteo	Matteo	Matteo	Matteo	Matteo	Matteo		
	HW (Body)	Des/Eng				Jalal	Jalal	Jalal	Jalal	Jalal	Jalal	Jalal	Jalal	Jalal		
Deliver	Integrated Body - Presentation	Designer 01													Juliana	Juliana
	Integrated Body - Presentation	Designer 02													Jalal	Jalal
	Electronics and SW - Manual	Eng01													Stefano	Stefano
	Electronics and SW - Manual	Eng02													Matteo	Matteo
	Integrated Body - Project (CAD)	Des/Eng													Sofia	Sofia
Lessons			17/03 09:15 (8hs) Lessons and brief	24/03 9.15 (8hs) Lessons. Discovery and Revision	31/03 09:15 (8hs) Definition and revision	07/04 09:15 (8hs) Development	14/04 Pasqua	21/04 9.15 (8hs) Development presentation and revisions		05/05 9.15 Development presentation and revisions		19/05 9.15 Development presentation and revisions		04/06 9.15 Development presentation and revisions		17/6 9.15 Final presentations
Revison			17/03 16:00 Brief definition and first ideas	24/03 16:00 Discovery presentation	31/03 16:00 Selection of the features and technology. Presentation	07/04 Revisions	(No lesson)	21/04 Revisions				19/05 Revisions		04/06 Revisions		

Research

We collected data on all prey and predatory animals of which we appreciated at least one physical or behavioral characteristic that we would've liked to exploit in the realization of our robot. We studied various animals and made multiple combinations in order to find the perfect match.

Animal strategies: PREDATORS

OWL

Owls are solitary and nocturnal birds and their hunting strategy is based on stealth and surprise. They are extremely good at stalking their prey, patiently waiting for the right time to swoop down and get them. They manage to be silent during the flight, thanks to the notched edges and the shape of their wings which attenuate the sound of the flapping of wings. Furthermore, the dull colouring of their wings can make them almost invisible under certain conditions and specific habitats.

Their main preys are weasels, bats, shrews, fishes and insect-eating birds, and they typically live in wooded areas.

The owl is being hunted by wild cats and foxes, but it’s able to twist the neck in 270 ° and fold in 180°, which allow it to control its surroundings, and can also see in complete darkness.

FEATURES: stealth, patient, very good hearing, rotating head, night-vision.

CHAMELEON

Chameleons are reptiles of the saurian family, famous above all for their ability to change the color of the skin for camouflage purposes.

Distinctive characteristics of the species are the large and protruding eyes that can rotate 360 degrees and are independent of each other. It is a patient animal that through mimicry waits for the right moment to attack its prey. It has a retractable tongue covered with sticky saliva: the chameleon extracts its extendable and sticky tongue and hurling it at incredible speed catches insects in a flash.

FEATURES: camouflage, stealth, extendable tongue, sticky saliva, very fast.

HERON

The heron is a bird that belongs to the Ardeidae family and is characterized above all by a slender and very graceful physique.

The legs, very thin and very long, are yellow just like the beak, which is also long and robust. One of the most unmistakable characteristics of the heron is the neck: also very long but very peculiar as it can fold back on itself assuming an S shape.

Feeding is mainly based on fish, amphibians, insects, and small mammals. They also eat shellfish and water snakes. It catches and hunts its prey in shallow water by trapping the prey in its long sharp beak: it freezes up waiting for the passage of prey and then captures it with a rapid movement of the neck.

FEATURES: long neck, strong beak, attacks from above

KILLER WHALE

The killer whale is a giant marine mammal belonging to the Delphinidae family. It has a gigantic and heavy body but nevertheless, thanks to its powerful tail, it is the fastest animal in the marine world. She has a mouth full of sharp and strong teeth with which she mercilessly hunts

even huge prey: her voracity leads her to attack every animal she encounters along the way. It lives mainly in cold waters and is undoubtedly one of the most fearsome and dangerous marine animals for the entire marine fauna.
Killer whales typically hunt in small groups and are able to communicate through a sound signal called sonar.
FEATURES: top of food chain, extremely fast, huge, voracious, powerful tail, sonar

Animal strategies: PREY

HORSE

The horse is an animal that lives mainly in herds and that behaves differently according to the danger it is facing. In fact, it can kick a predator approaching from behind, ramp to frighten an enemy, or use its long legs to escape at full speed. It is able to notice the danger thanks to a wide peripheral vision, due to the lateral position of the eyes on the muzzle, which allows it to see everything that surrounds it with a single glance.
FEATURES: watchful, rear-kick, quick escaping

SQUID

Squid are cephalopods with elongated bodies, large eyes, eight arms, and two tentacles. They mainly live in salty waters and are hunted by sharks, seabirds, seals, so they need a strong defensive strategy to get rid of all of them.
Squids have a strong camouflage ability and can change their color, making use of different types of camouflage: active camouflage, blending into the background, and backlighting. This also helps them get closer to their prey.
When threatened, the squid distracts the attacks of predators by expelling a cloud of ink, giving itself the opportunity to escape. The ink quickly disperses to form a dark cloud that obscures the squid escape maneuvers.
FEATURES: camouflage, distracting ink, fast escaping

RHINOCEROS

Rhinoceroses are massive animals of very considerable size and weight; the facial region serves as a support for one or two mighty horns, placed one behind the other. It has a powerful body, covered by a very thick or armored skin, almost entirely naked. The tail is small, the legs appear short and robust. They inhabit places rich in water, swamps, river banks where there are periodic flooding. The animal manages to open gates even in bushes inaccessible to other animals, as its armor protects it from thorns.
It is a very irascible animal that, when annoyed, gets angry and loads its predator trying to hit it with its horns at full speed.
FEATURES: robust, huge, irascible, armored skin, mighty horns

ECHIDNA

Echidnas are medium-sized, solitary mammals covered with coarse hair and spines. It is one of the only living mammals that lay eggs.
Echidnas are very timid animals. When they feel endangered they attempt to bury themselves or if exposed they will curl into a ball similar to that of a hedgehog, both methods using their spines to shield them. Strong front arms allow echidnas to continue to dig themselves in whilst holding fast against a predator attempting to remove them from the hole.
They lay one single egg which they protect until the hatching, which takes place 10 days after the gestation; the young echidna, called "puggle", remains in the pouch to be protected and fed until it starts to develop spines.
FEATURES: coarse hair and spines, protecting eggs

Selection and studies

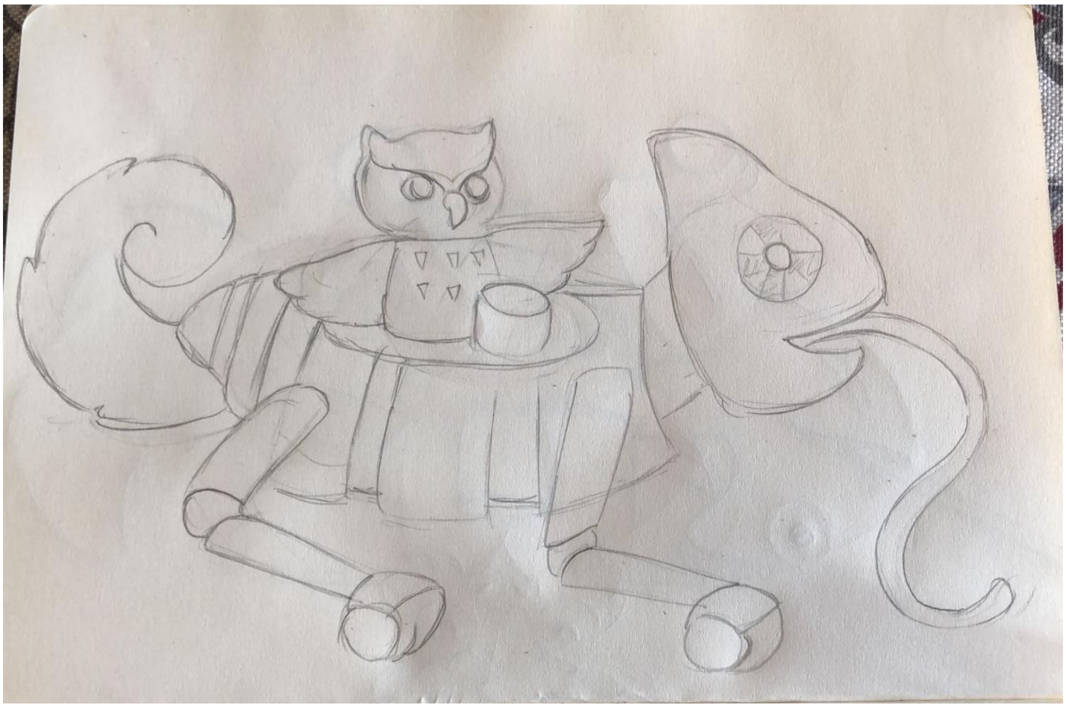
After that, we create a comparison table to help us with our final decision. Some raw sketches were done to start thinking about possible combinations.

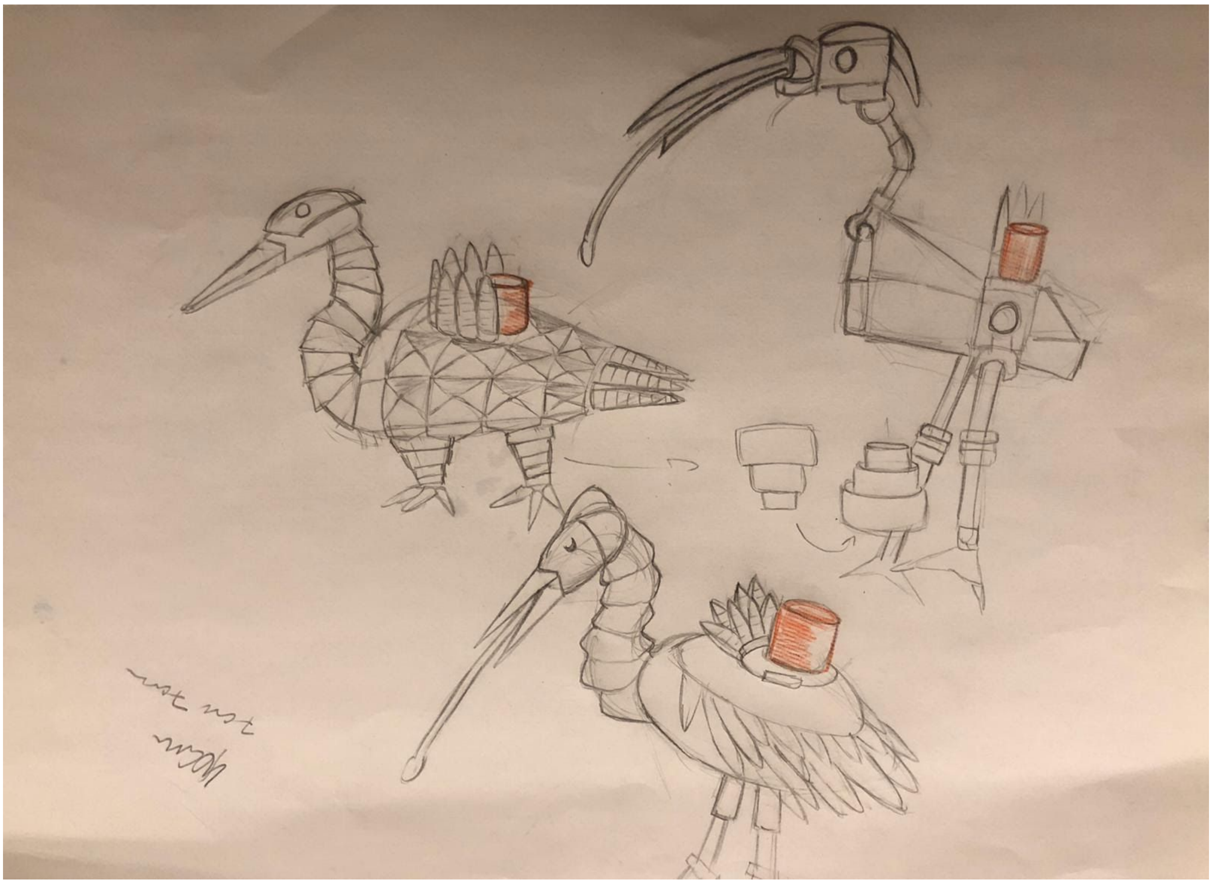
Animal	Physical features	Behavioural features
OWL	Good hearing, rotating head, night vision	Stealth, patient, silent
CHAMELEON	Camouflage, extendable tongue, sticky saliva, fast movements and attacks	Stealth
HERON	Long neck, long beak, attacks from above	Patient
KILLER WHALE	Powerful tail, fast movements, huge dimension	Killing instinct, voracious, collaborative
HORSE	Wide field of view, rear-kick, fast movements	Watchful
SQUID	Camouflage, distractive ink, quick escaping	Stealth

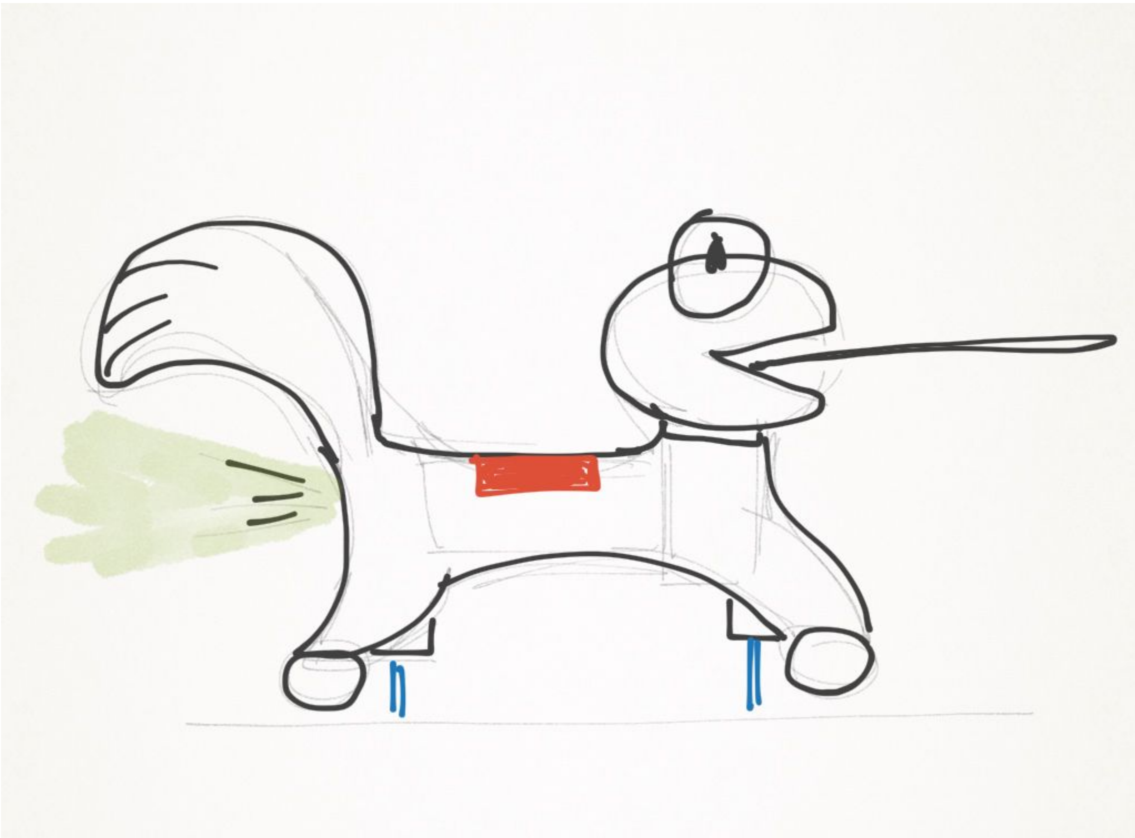
RHINOCEROS	Robust, huge body, armoured skin, mighty horns	Irascible
ECHIDNA	Coarse hair and spines, covers weak spot	Timid, protective, digger

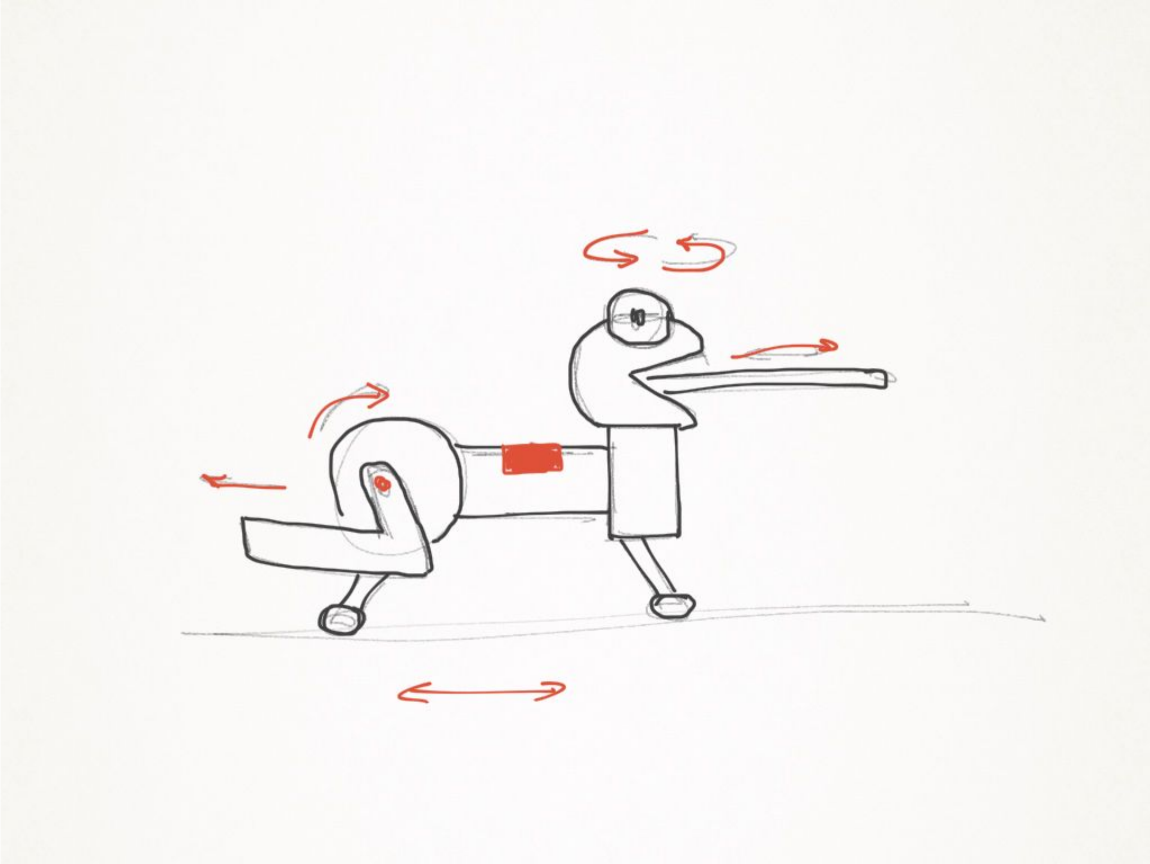
Animals combination	Offensive features	Defensive features	Attacking strategy	Defending strategy
CHAMELEON + HORSE	Sticky saliva, fast attacks, extendable tongue	Rear-kick, fast movements	Aim for red target and throw the tongue to it	Kick enemies approaching, run away
CHAMELEON + SQUID	Sticky saliva, fast attacks, extendable tongue	Distractive ink, quick escaping	Aim for red target and throw the tongue to it	Emit a cloud of smoke and escape
CHAMELEON + ECHIDNA	Sticky saliva, fast attacks, extendable tongue	Spines, covers weak spots	Aim for red target and throw the tongue to it	Protect the red target from lateral attacks
OWL + RHINOCEROS	Rotating head, good hearing, mighty horns	Robust, huge body	Rush into targets hitting with horns	Red target hard to reach, can flip enemies
HERON + HORSE	Long neck, attacks from above	Rear-kick, fast movements	Attack other targets from above	Kick enemies approaching, run away
HERON + SQUID	Long neck, attacks from above	Distractive ink, quick escaping	Attack other targets from above	Emit a cloud of smoke and escape
HERON + ECHIDNA	Long neck, attacks from above	Spines cover weak spots	Attack other targets from above	Protect the red target from lateral attacks
KILLER WHALE + RHINOCEROS	Powerful long-tail, fast movements, mighty horns	Rear-kick, fast movements	Attack the target with the tail	Red target hard to reach, can flip enemies

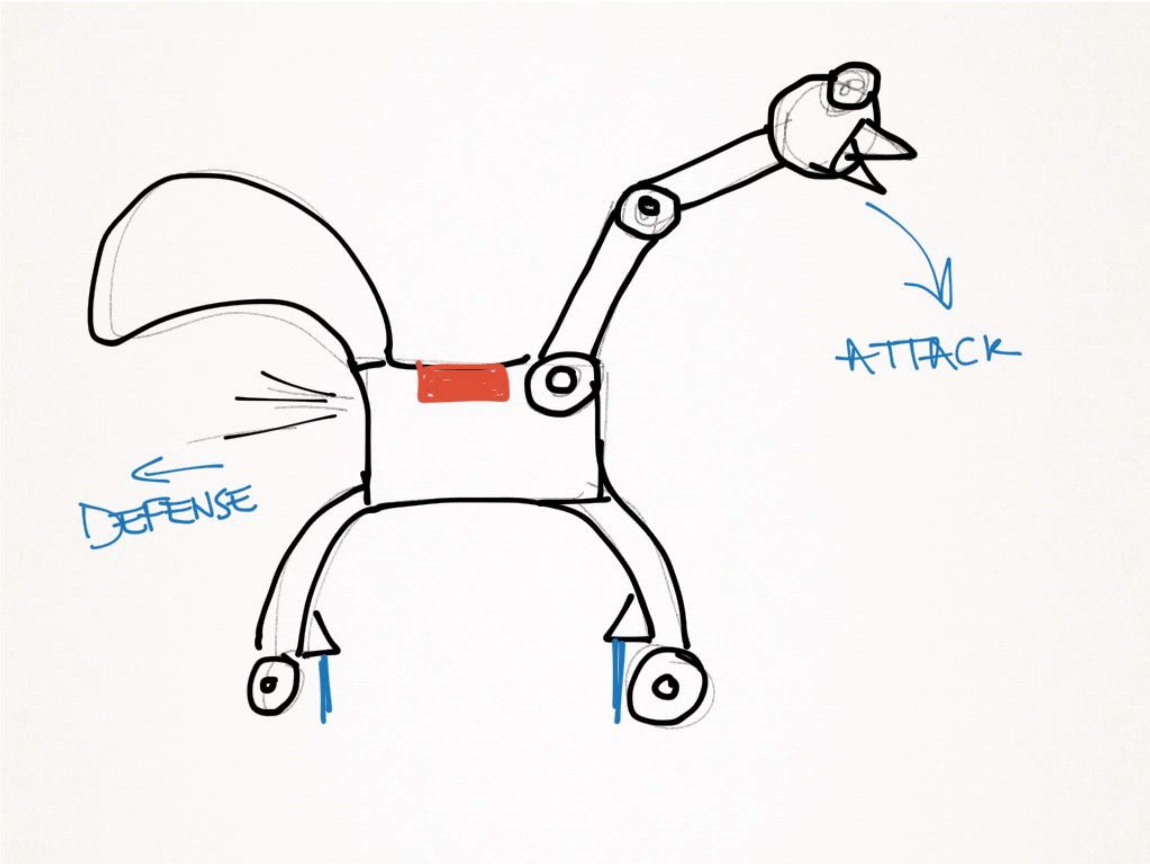
These are the first sketches we did to get an idea of what our robot might look like:










ATTACK
DEFENSE

Premises: Robot strategies

Our robot premises were to be able to implement attack and defence mechanisms and strategies to be able to make the most out of its position on the field, and thus earn points in the final competition.
Taking into account the rules, the size of the playing field and the method of scoring, we have compiled a ranking of importance for the actions that our robot will have to perform:

- The first thing it needs to be able to do is drinking water from the pond, which is worth a 20 point gain every 5 seconds;
- Secondly, it must be able to defend itself from opposing attacks, not being eaten so as not to lose 30 points every 5 seconds;
- Being able to attack opposing enemies, gaining 10 points every 5 seconds;
- Finally, do not lose 1 point for inactivity penalty, intended as not losing or earning points for five seconds consecutively;

Functionalities

Location detection

Since the field's shape and size are fixed, our robot will be equipped with distance sensing sensors, such as ultrasound and infrared sensors. Through software algorithms it will be able to recognize firm obstacles from moving obstacles, thus being able to correctly identify the walls and consequently know its position in the field.
The main objective is to identify and reach the center of the field, where the puddle resides, as soon as possible: the robot will analyze the underlying flooring using color sensors, placed under the body and on an ejection device that will allow it to drink from the puddle even without being physically on it.
To recognize and target the red targets of our opponents, the robot will use a camera placed on its head and an advanced image recognition system.

Predation and defense mechanisms

Taking advantage of the combined characteristics of the animals we have selected, we have come up with several alternatives to develop our predation mechanism:

- A possible attack method could be the use of a language that can be launched and rewound, consisting of a rope and a weight at the end of it. We have various ideas of what to attach to the end of the rope: we could use a hook, sticky material, magnetic lock rings, or even a net.

- Alternatively, we thought of building a long robotic arm, capable of precisely touching the enemy target and staying in touch easily for the time necessary to score points.
- Another different approach could be to have horns placed horizontally on the head of our robot and go towards other enemies trying to hit the opponent's target.

We also had several ideas on how to protect it from enemy attacks:

- An always working tactic is to run away once the danger of an approaching enemy is perceived: by building the robot in a sufficiently light and reactive way, we would be able to escape every enemy.
- Alternatively, we could reach a good compromise between weight and speed, in order to try to overturn our opponents without damaging them permanently.
- One idea is to take a cue from horse rear kicking when it is approached from behind. We could create a mechanism capable of extracting legs from the back of the body which can loosen the enemy, without damaging it.
- Another active mechanism is to spread a cloud of coloured smoke behind us when an approaching enemy is sighted, in order to mislead the visual identifying methods of our enemies and allow us to flee in time.
- A completely different idea is to add a passive counter-offensive: since the target must always be visible from an angle of at least 270°, we thought of building a quarter cylinder (which would, therefore, cover a 90° angle) and rotate it around our red target, with sufficient strength and speed to not allow our enemies to remain hooked for the 5 seconds necessary to score the points.
- We also may consider rotating the target itself such that it will be hard for our opponents to hit it.

-

Actuators

The available kinds of actuators typically used in the small size robots:

- DC Motors
- Brushless Motors
- Servo Motors
- Stepper Motors
- Solenoids
- Pneumatic Actuator

Having this in mind we analyzed our options:

1- DC Motors:

- They rotate fully 360 degree
- They can track a given rotation, coupling them with an encoder
- Lightweight
- Powerful
- Cheap (1/3 of the brushless Motors)

- Robust
- Heating problems

(comparing to the brushless Motors)

- Use less current
- Lower speed
- Less torque
- Less control over speed

2- Brushless Motors:

- Expensive
- Less robust
- Lasts longer

(comparing to the DC Motors)

- Less maintenance
- Higher speed achievable
- Better control over speed
- More current usage
- Higher torque

3- Servo Motors

- Lightweight
- Not very powerful
- Easy to code
- Take more current (comparing to the stepper motor)
- Fast
- Limited rotation
- Needs accurate PWM signals

4- Stepper Motors

- Moves in steps
- Less current
- More holding torque
- Unlimited rotation
- Slow in moving but precise
- Needs on the pulse for each step
- More expensive than a servo
- No feedback
- Have noise
- Loses torque in high speeds

5- Solenoids

- High torque/force
- Compact size
- Simplified control

- High reliability
- Linear motion output

6- Pneumatic actuators
    - Light (they use air as working fluid)
    - No heat generation
    - High reliability
    - Clean
    - Not good at speed control

- And made some decisions around what would fit better for us.

Phase 2: Define

Final Decision

With this information in mind, we decided to combine the characteristics of Heron and Echidna, which lead us to develop the robot strategies according to those two animal behaviors. After broad research we start to define specific goals, making rough prototypes to test mechanisms, aesthetics, and functionalities.

The two chosen animals, heron and echidna

Strategy

Among all the alternatives we explored, we finally decided to use the combination between Heron and Echidna, with the possibility of integrating the Chameleon’s tongue.
In particular, our robot will use its long neck and beak to attack its prey and passively defend itself through spines present on the whole body, in addition to the use of a rotating element around the target to counter enemy attacks.
Initially, the idea was to exploit a mechanism similar to that of the chameleon's tongue to hook our prey, however, by making an in-depth analysis and developing the first prototypes, we realized the many problems that this attack method would entail which brought us to another road. At this stage of work, we had not yet ruled out the use of such a mechanism, though we were studying how to use it to be able to drink from the pond, which would’ve been a much simpler task.
Once positioned on the field, the main strategy of our robot would be to first position itself in a space within the pond and then be able to attack its opponents from there.

Sensors

The sensors we chose to use are ones that are commonly used with platforms like Arduino for simple robots. Specifically, our idea is to use the following sensors:

- 4x Ultrasonic sensors HC-SR04 used to locate approaching obstacles (both walls and enemies);
- 2x Color sensors TCS230 used to recognize the pond (placed under the robot) and the target on the other robots' backs.

Functionalities

Wheels

We have 4 wheels, so we need four motors for this application. We need to make our device heavy to prevent it from being flipped by attacker robots. So, as an estimation, for a 4-5 kilograms robots, motors with the following specification are suitable:

Type: DC motor
Torque: 1.5-2 kg.cm (unload)
Voltage: 12 volts
Rpm: 300-400 (no load)

There are several options to buy. Since the DC motors are the cheapest among all, we choose them after choosing the other needed motors (servo, solenoids, etc).

Defender's head

We need to rotate a quarter cylinder rapidly with a weight less than one kilograms, so not a high torque is needed:

Type: DC Motor
Torque: 1kg.cm (unload)
Voltage: 12 volts
Rpm: 100-200 (no load)

Attacking head

We need to throw the disk projectile as precise as it can be to the target. So the attacking head will be combined of three parts: pointing mechanism, the shooting mechanism, and retrieving the shooting ball.

For the pointing mechanism, we need:

- one servo motor, rotating the whole head in the x-y plane. Since all the head part is mounted on this part, we need a powerful servo motor.
  RoboticServo SBRS-5314HIG
  GWS S125 1T/2BB

Hitec HS-525MG
would be reasonable for the application

- One servo motor is also needed to rotate the head in the z-direction. Since it only holds the shooting ball and the pointing gun, it possible to use a low power servo.
the “MG 995 – 12V” would be a good choice.

For the shooting mechanism, we also have two options:

- One is two, throw the shooting ball by a solenoid. Assuming that our shooting ball weighs 50 grams, solenoids with at least 0.1N of exerting force would be suitable. Out choices can be:
F04155-12v
Or
SKU-20041507

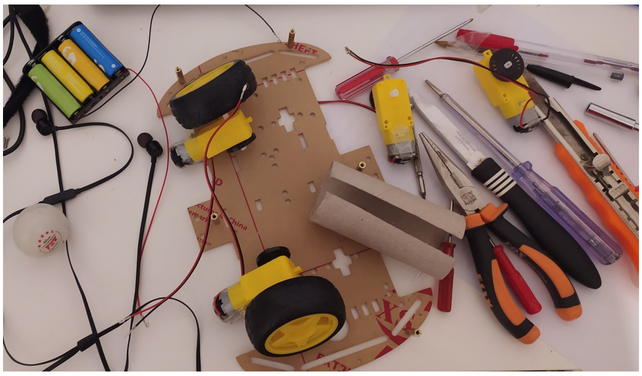
- The other option is to use a hand-made pneumatic actuator, using a sealed reservoir, compressor, pipe ,and an opening valve.

For retrieving the ball, we need

- A DC motor, coupled with an encoder. Since we are just rolling the rope, we need the motor to rotate both clockwise (while shooting) and counterclockwise (while rolling back the rope).
The properties for this kind of motor would be:
Torque= 0.5 kg.cm (no load)
Voltage= 12 volts
Rpm= 300-400 (no load)

Tongue

One of the main things we needed to prototype was the tongue mechanism. This clearly represented the most challenging task among the ones we were facing, so it was the first one whose feasibility we wanted to test.
The first prototype was really raw, made from scratch with materials we were able to find home (due to the coronavirus outbreak). We built two of them in parallel while video calling on Skype and the results, though still very elementary, were pretty convincing.
The way we decided to implement this feature came from an idea of our mechanical engineer (Jalal). The design is made of two wheels rotating in the opposite direction; a projectile in the shape of a rectangle (made with cardboard) is placed within them so that it is thrown away by their movement.
The resulting speed and direction of the projectile weren’t exceptional, but we had a large room for improvements considering the materials we used. However, after producing this prototype, even though we were sure of the potential behind this idea, we decided to scale back on its importance, as we weren’t sure we would be able to hit a target with this mechanism. We, therefore, recycled it as a back-up idea for drinking from a safe distance from the pond.



The materials used to prototype the tongue

In the end, however, we decided to drop the idea completely, as it would have made the design process too complicated to be worth the effort to realize it.

Autonomous movement

This task, though easier than the former one, was definitely as fundamental for the correct functioning of the robot. To implement it, we used a car that Stefano had previously built with Arduino and applied an ultrasonic sensor on top of it. Then it was just a matter of writing some code to have the first prototype. In this case, too, the result was a bit raw, with the car detecting only obstacles right in front of it and steering right when facing one of them.
Clearly a more complicated code would be needed in the end, combined with at least two sensors, but the results looked promising.

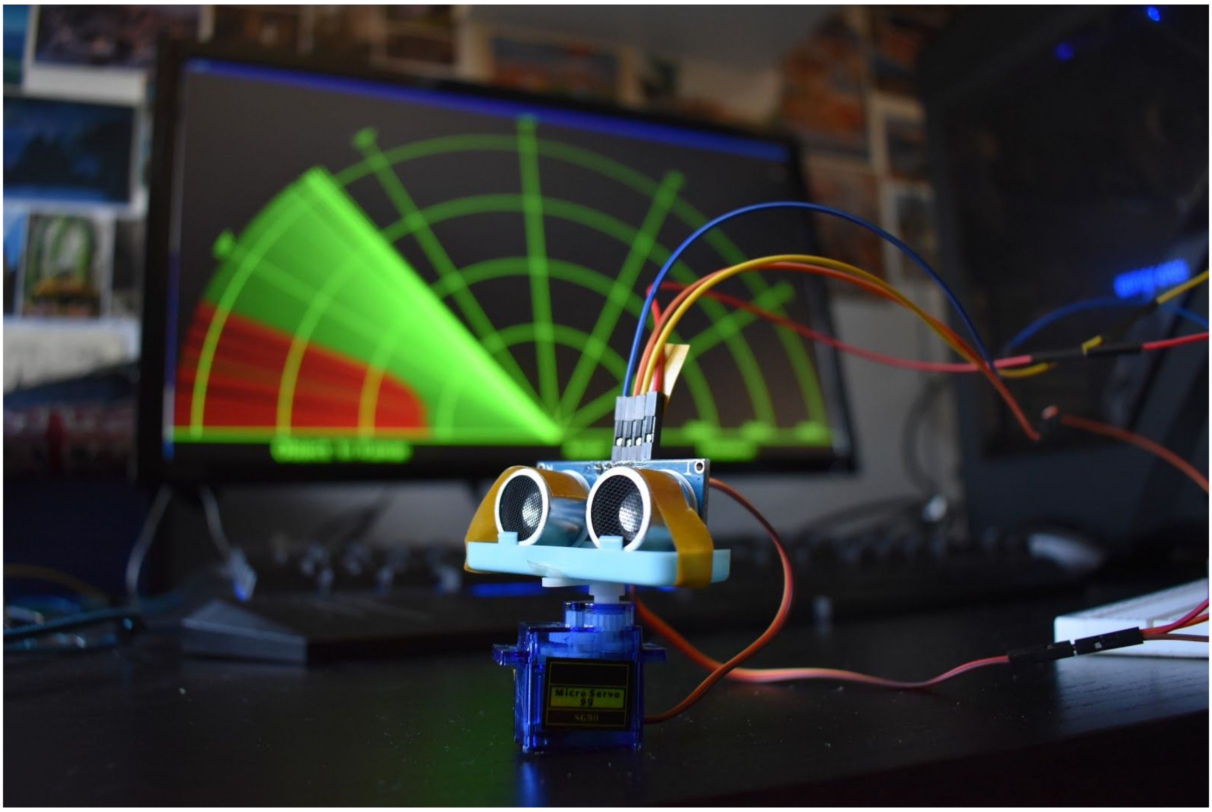
The prototype of autonomous driving vehicle

Electronics

Unfortunately, given the problematic situation related to the coronavirus outbreak, we couldn't test the electronics as extensively as we would've liked. However, we were capable of retrieving an ultrasonic sensor, as well as an Arduino Nano and an ESP32 board.
Stefano built a few small prototypes to gain familiarity with the sensor (one is the car we talked about in the previous section, the other is a small radar), while Matteo studied performance differences between the two boards.
Studying the advantages and differences between the main Arduino and ESP32 boards, we decided that the most suitable for the realization of our robot is the ESP32.
ESP32 board has Bluetooth and Wi-Fi functionality unlike Arduino Uno or Nano, which only has GPIO pins. Also, it has 10 internal capacitive touch sensors. These can sense variations in anything that holds an electrical charge, like the human skin. So they can detect variations induced when touching the GPIOs with a finger.

Compared to Arduino, it is faster (ESP32 has a frequency of 160-240MHz, most of the Arduino boards have a frequency ranging from 16MHz to 84 MHz), has a larger memory and supports real-time operating systems.

Last but not least, ESP32 works with Arduino IDE with the installation of ESP32-Arduino Core: we can treat the ESP32 as a supercharged Arduino Uno, having also the possibilities of using Arduino libraries.



The prototype of the radar

Coding

The code we developed in this early stage wasn't particularly complicated, as the prototypes themselves were quite simple. However, we still took a couple of steps that would be necessary for the later parts of the project, such as creating a Github repository to make the group work easier to coordinate (especially since we have to work from home).

We also took some time to gain some deeper insights in the Arduino programming language by building the prototypes we talked about in the previous sections.

Structure

During these early stages, we spent some time considering which mechanical choices would serve best for our purposes.

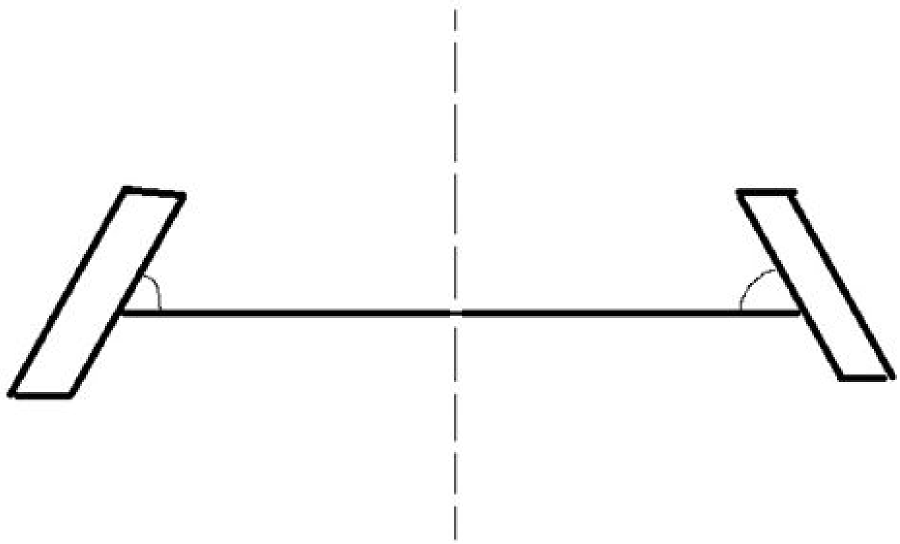
Wheels

The bigger the wheels the faster our robot, since we can cover more distance given a certain rotation. But having a bigger wheel means needing a higher torque. Thus, we need a more powerful motor for driving them, which means more expense and usually a bigger and heavier motor.

Big wheels also reduce the agility of the robot, since the bigger the wheels the harder it is to change the direction of the robot.

From the stability point of view, since the wheels are located at the bottom part of the robot, they play an important role in the movement stability of the robot. The lower the size of the robot with respect to the wheels the higher the stability of the movement. As a consequence, given the Geometry of our robot and the probable torque sources from the upper part of the body, we can use 4 wheels in the sides, and two auxiliary wheels in the back and in the front to stabilize the machine.

Also, in order to stabilize the Robot, we can also locate the wheels with a small angle with respect to the vertical line, so that we prevent the robot from being flipped, especially from the sides.



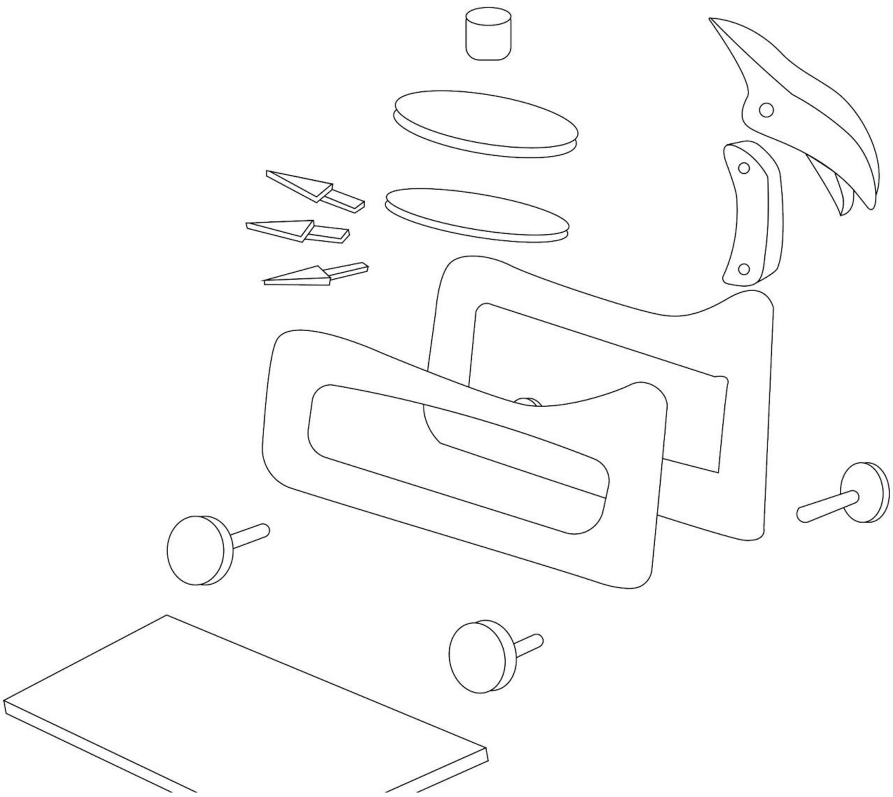
Materials and development tools

To make a shape that could combine structure and aesthetics we analyzed and compared Thermoforming, 3D printing, and Laser cutting. The echidna has spines and this means a shape too complex for thermoforming and a lot of material for 3D printing, so laser cutting presented itself as the fastest, simplest, and more economic way of solving all these issues together. We

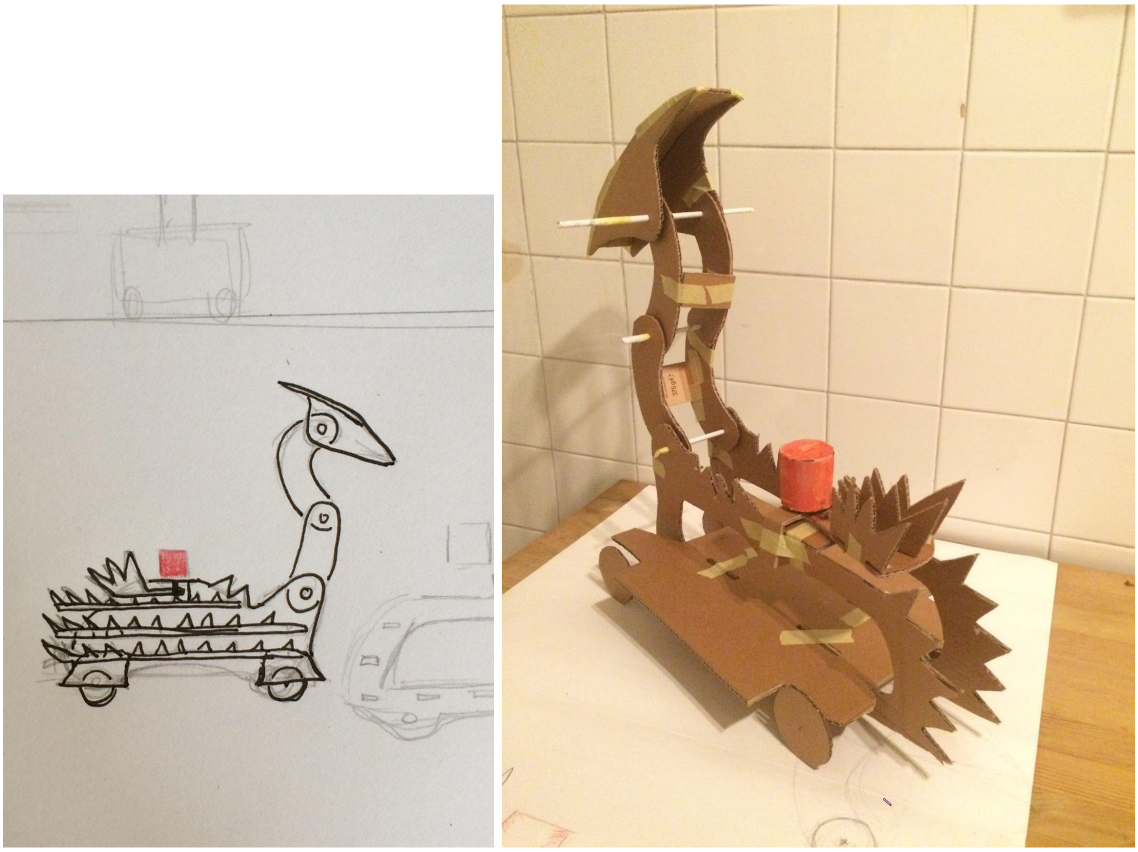
might consider using thermoforming for the head, in order to make it lighter and with a softer shape.
The material wood is MDF, 3mm, or 5mm.
With further development, we decided to roll back on these decisions, opting for 3D printing instead, as Sofia decided to buy a 3D printer for herself, thus making this a cheaper and more flexible alternative with respect to the others, which would've required going to external shops.

Shape

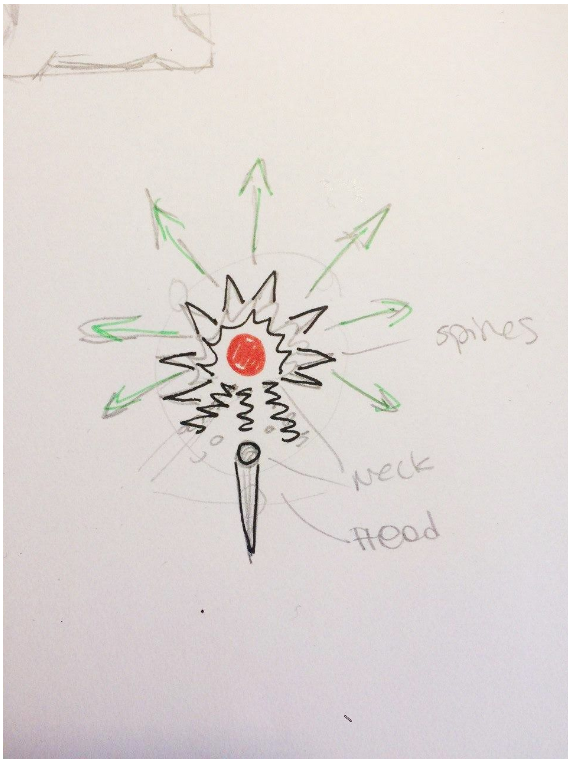
During the second week of work, we had to build a cardboard model of the final robot. After a discussion among ourselves, we agreed on a rough shape and on the dimensions and started building the model (one for each of us).
Though the final shape would be slightly modified to allow a better implementation of electronics and mechanical movements, these cardboards gave us a good idea of what the final result would be like.

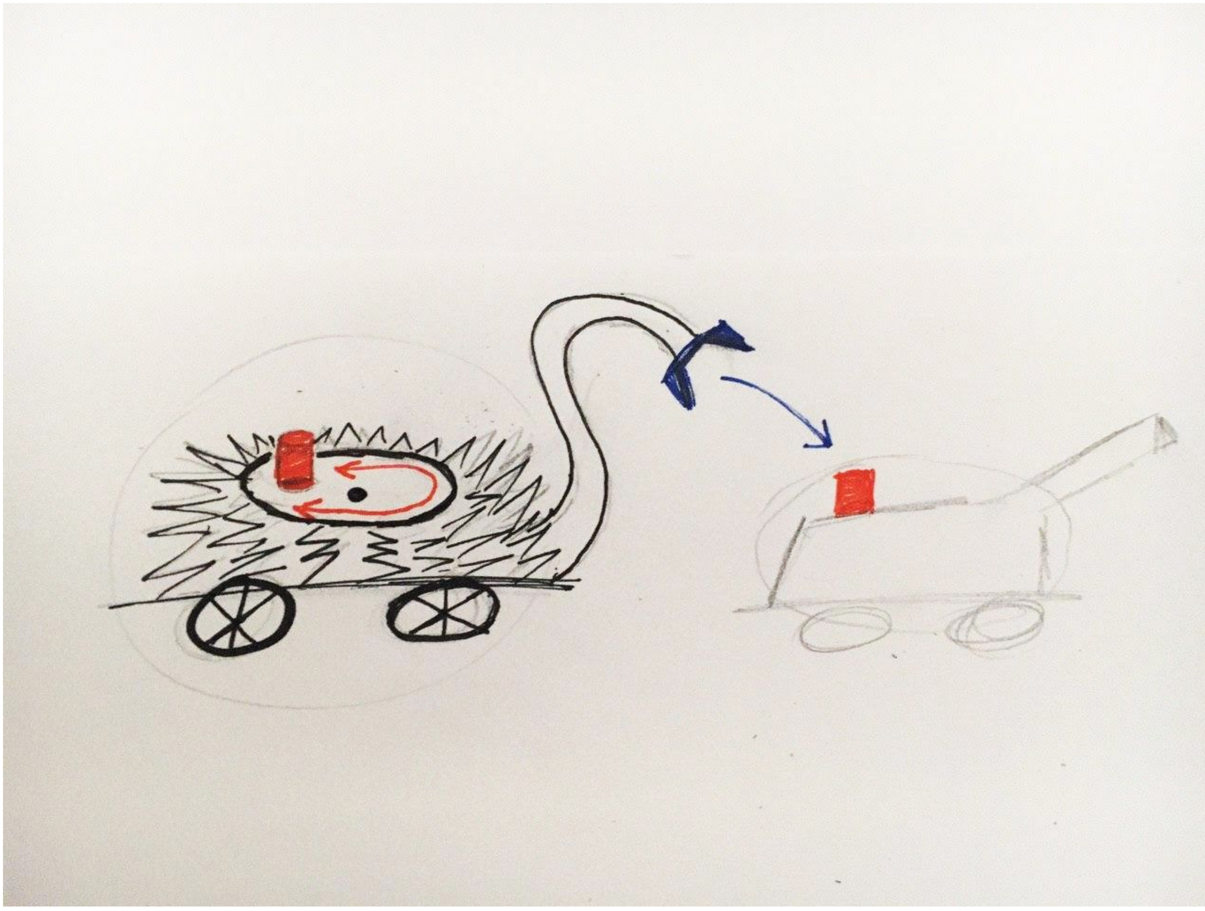


First exploded view of the robot



The cardboard model made by Sofia

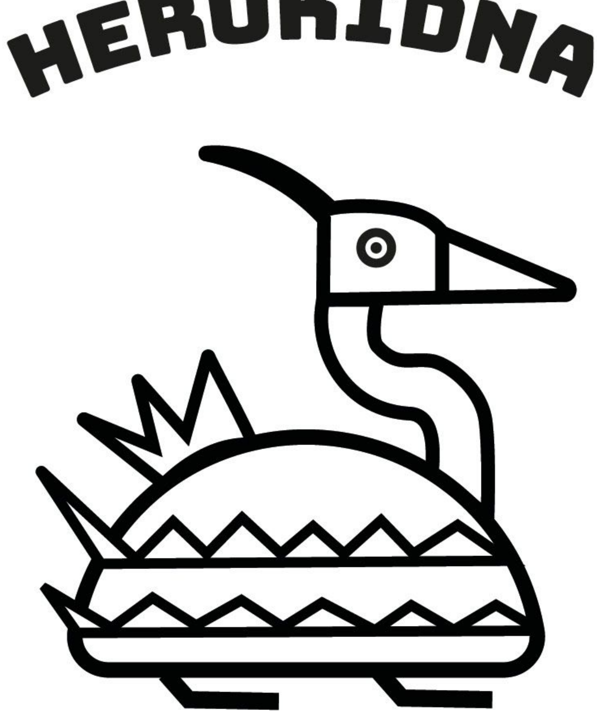




Some of the initial sketches for the robot's strategies

Logo & Concept

By combining the two animals we came up with a hybrid similar to a sort of dragon. The most unique characteristics of the animals were highlighted, the neck from the heron, and spikes from the echidna. Trying to find a harmonic appearance, we drew it with a particularly aggressive appearance, but we tried to balance it with a cartoon expression to bring humor and personality to the robot which suits the behaviors of our chosen animals.



The logo for our project

Phase 3: Develop

Here, we describe the development process, starting from the first prototypes to the final adjustments. This is what we did regarding the different aspects of the robot.

Strategy

WEEK 3

Most of the work in this field had already been done in the previous weeks. Our only change was in the defensive strategy. Given the rule that was imposed to cover the target at most from one side, we had to slightly modify our robot, choosing a new strategy consisting in moving the target itself, rather than moving spines around it.
Always in this week, we started working on the flowcharts to translate our strategies into commands for our robot. More about it in the “Coding” section.

WEEK 4-5

At this point in the work, our strategy had already been decided and no more subject to major changes. The only small things that needed to be defined were the behaviors in some specific situations: for example, if the robot is looking for the pond and an enemy approaches it, should we prioritize the pond research and ignore the threat, or should we flip around to face the other robot?
At the time of writing all these small details were being solved through the flow charts.

Mechanics

WEEK 3

This week, we also made some calculations to choose our motors. Calculations consist of the estimation of the torques needed by each motor, plus their required number. We also did a rough estimation of the weight of the robot. in the following you can see all the calculations:

The estimation of the weight of the Robot:

Weight of the motors= 330 grams
Weight of the foundation: 500 grams
Weight of the joints= 50 grams
Weight of the cover= 200 grams
Weight of the wheels= 200 grams
Weight of the batteries= 500 grams
Weight of the neck, disk, auxiliaries= 500 grams
Weight of the electric devices=250 grams
Error coefficient= 250 grams

Total weight of the Robot = 25280 grams
As a rough estimation= 2.5 kilograms

Total number of actuators needed: 8

1. The motors for the wheels

The pros and cons of the size of the wheels suitable for the robot size and its movements must come into in-detail evaluations. So here we do our calculations based on typical wheel diameters.

- CASE 1: 6.5 cm or 2.75 inches

Given the speed 2m/s and wheel diameter, the required speed(load) of the motor would be:
Motor speed=2/(3.14*0.065)= 9.8 RPS (Round Per Second)=587.65 RPM

If the speed is 3m/s:
Motor speed=3/(3.14*0.065)= 14.7 RPS (Round Per Second)=881 RPM

If the speed is 4m/s:
Motor speed=4/(3.14*0.065)= 19.6 RPS (Round Per Second)=1175 RPM

Calculating the torque:
T = 8 x C x W x D

where:
T is the torque in oz-in
C is the coefficient of friction
W is the weight in lbs
D is the wheel diameter in inches

Assuming the friction coeff to be at max (0.03):
T=8*0.03*5.5*2.75=3.63 oz.inch=0.27 kg.cm

This is the minimum torque that the motor must provide, which is not very high, so we can work more on the speed.

- CASE 2: D= 10 cm or 3.94 inches

Given the speed 2m/s and wheel diameter, the required speed(load) of the motor would be:
Motor speed=2/(3.14*0.065)= 6.36 RPS (Round Per Second)=381.97 RPM

If the speed is 3m/s:
Motor speed=3/(3.14*0.065)= 9.55RPS (Round Per Second)=573 RPM

If the speed is 4m/s:
Motor speed=4/(3.14*0.065)= 12.73 RPS (Round Per Second)=764 RPM

Calculating the torque:
T = 8 x C x W x D

where:

T is the torque in oz-in
C is the coefficient of friction
W is the weight in lbs
D is the wheel diameter in inches

Assuming the friction coeff to be at max (0.03):

T=8*0.03*5.5*3.94=5.2 oz.inch=0.37 kg.cm

This is the minimum torque that the motor must provide, which is not very high, so we could work more on the speed.

Then we evaluated the feasibility of these options, using RMF toolbox:
https://www.societyofrobots.com/RMF_calculator.shtml

Giving these desired specifications to the robot:

Mass	2.5	kg ▼
Desired Velocity	3	m/s ▼
Desired Acceleration	1	m/s^2 ▼
Expected Efficiency	80	%
Incline Angle	0	degrees
Wheel Diameter	0.065	meters ▼
# of Powered Wheels	4	wheels

The RMF (the index number of the robot) would be:

RMF	1.22	kg * m * rps ▼
RMF (zero incline)	1.22	lb * ft * rps ▼
Momentum	7.50	kg * m / s ▼
Motor Rotation Speed	14.7	rps ▼
Energy/second	0.154	Watts

According to this RMF, the Motor RMF must be higher than 1.22. so,

Torque	0.02	kg*m ▼
Speed	4000	rpm ▼
Motor RMF	1.33	(as above)

Changing the robot speed to 2m/s:

RMF	0.816	kg * m * rps ▼
RMF (zero incline)	0.816	lb * ft * rps ▼
Momentum	5.00	kg * m / s ▼
Motor Rotation Speed	9.79	rps ▼
Energy/second	0.103	Watts

We would be needing motors with following specification:

Torque	0.01	kg*m ▼
Speed	5000	rpm ▼
Motor RMF	0.833	(as above)

Considering the expenses observations and limitations for finding the pond, we finally decided that the speed of the robot should not be too high, so choosing the speed of 0.5 meters/second seemed a more reasonable choice:

Mass	2.5	kg ▼
Desired Velocity	0.5	m/s ▼
Desired Acceleration	0.5	m/s^2 ▼
Expected Efficiency	80	%
Incline Angle	0	degrees
Wheel Diameter	0.065	meters ▼
# of Powered Wheels	4	wheels

RMF	0.102	kg * m * rps ▼
RMF (zero incline)	0.102	kg * m * rps ▼
Momentum	1.25	kg * m / s ▼
Motor Rotation Speed	147	rpm ▼
Energy/second	0.0129	Watts

Torque	0.01	kg*m ▼
Speed	700	rpm ▼
Motor RMF	0.117	(as above)

As a general rule if you have 2 motors on your robot, make sure the stall torque on each is enough to lift the weight of your entire robot times your wheel radius. So the stall torque for each motor would be:

$$2.5*0.5*3.25= 4 \text{ kg.cm}$$

After that we considered the rotating arm:

2. Rotating Arm

arm length= 30 cm
arm weight= 250 grams

Calculating the torque:

$$T = 8 \times C \times W \times D$$

where:

T is the torque in oz-in
C is the coefficient of friction
W is the weight in lbs
D is the wheel diameter in inches

Assuming the friction coeff to be at max (0.03):

$$T=8*0.03*0.55*11.8=1.55 \text{ oz.inch}=0.11 \text{ kg.cm}$$

Mass	0.25	kg ▼
Desired Velocity	1	m/s ▼
Desired Acceleration	1	m/s^2 ▼
Expected Efficiency	80	%
Incline Angle	0	degrees
Wheel Diameter	0.3	meters ▼
# of Powered Wheels	1	wheels

The RMF of the Arm will be:

RMF	0.163	kg * m * rps ▼
RMF (zero incline)	0.163	lb * ft * rps ▼
Momentum	0.250	kg * m / s ▼
Motor Rotation Speed	1.06	rps ▼
Energy/second	0.0206	Watts

So the motor specification we will need is:

Torque	0.01	kg*m ▾
Speed	1000	rpm ▾
Motor RMF	0.167	(as above)

3.Rotating redpoint

Disk weight= 50 grams

Disk diameter= 10 cm

The weight of the redpoint = 20 grams

Number of gears= 2

Gear diameter: 6 cm

Number spines= 4

The weight of each spine= 100 grams

The force needed to overcome the movement of the spines

f=µ.n

n=m.g

n=0.1*9.8m/s^2

n=0.98 N

f=0.03*0.98=0.294 N

So the total force needed is:

F=4*f=1.176 N

The Equivalent mass is:

M_1=F/9,8=0.12 kg

The force needed to overcome the movement of the disk

$f=\mu.n$

$n=m.g$

$n=0.07*9.8ms2$

$n=0.686\ N$

$f=0.03*0.686=0.21\ N$

the equivalent mass is:

$$M2 = F9{,}8 = 0.021\ kg$$

so the minimum torque needed is:

$$T = 0.03 * 1.176 + 0.05 * 0.021 = 0.036\ N.m = 0.36\ kg.cm$$

the total equivalent mass is:

$$M = M1 + M2 = 0.14\ kg$$

The system specification:

Mass	0.14	kg ▼
Desired Velocity	0.5	m/s ▼
Desired Acceleration	0.5	m/s^2 ▼
Expected Efficiency	80	%
Incline Angle	0	degrees
Wheel Diameter	0.1	meters ▼
# of Powered Wheels	1	wheels

The disk RMF would be:

RMF	0.0228	kg * m * rps ▼
RMF (zero incline)	0.0228	kg * m * rps ▼
Momentum	0.0700	kg * m / s ▼
Motor Rotation Speed	95.5	rpm ▼
Energy/second	0.00288	Watts

So the motor specification would be:

Torque	0.005	kg*m ▼
Speed	300	rpm ▼
Motor RMF	0.0250	(as above)

Since we want to repel the attacking robots, we must use a higher torque low-speed motor: 3kg.cm.

Frog tongue strategy and removable beak: Motor torque= 0.5 kg.cm, Motor speed= 200 – 300 rpm.

To sum up, these are the specifications of the needed motor:

Application	Speed	Operating Torque (kg.cm)	Stall torque (kg.cm)	quantity
Wheel	700-800 rpm	1	4	4
Rotation of the redpoint disk	150-200 rpm	2	5	1
Neck	0.2 sec/60	0.5	2	1
Collecting the released rope	100-200	0.1	-	1

Electronics

WEEK 3

In the third week, we continued the work started in the previous ones, that is testing the sensors to get a better understanding of what their real-world performances look like.
The ones we analyzed were the infrared sensor and the color sensor.
Luckily, Matteo had a spare infrared sensor at home, so we could run some tests with it. Results however were disappointing, as its measurements proved to be much less precise than we had hoped. This led us to rethink our sensors' strategy by eliminating the infrared one, substituting it with an OV2640 camera.
Unfortunately, after further research, we had to replace this camera as well as the ESP-32 board. We figured out that, in order to perform image detection, we needed a much higher computation power, so we were forced to move to a Raspberry PI, which can run things like OpenCV with ease. Given this change of plans, we opted to use a camera built specifically for Raspberry PI instead of the OV2640.
On the other hand, the color sensor turned out to be much more reliable. Though not extremely precise, our experiments proved that it could be able to detect blue and red (the main two colors that our robot needs to look for) with minimal error. The main issue lies with false positives: sometimes the sensor perceived random colors (such as brown) as red, but this is just a problem of calibration that can be solved via software after the further experiment.

Following last week's research, we made our definitive decisions about the motors we will be using: those will be 5x DC motor (GB37RG model) for wheels and target platform, plus a servomotor (model PS-1109MG jx) for the neck.

WEEK 4-5

During the Easter break, we took care of checking that all the various components were working as expected.

After dropping the previously decided motors, due to the unreliability in shipping time of the website where we found them, we moved on to another DC motor. Luckily, we were able to find not only better motors (with better shipping times too!), but they also turned out to be greatly cheaper than the previous ones. We ended up buying 8 of them (5 to be actually used, plus 3 as replacements). To make sure their performances lived up to our expectations, we came up with a small experiment that showed how the motors behaved when moving with a weight of 3.7kg (well beyond our expected final weight of 2.5kg). Results were actually beyond our expectations: the motors were able to reach a speed of more or less 25cm/s, and this was obtained with the motors electronically limited at half of their maximum power (this was necessary due to the low stability of the structure of the experiment).

Our experiment to test the performances of the motors

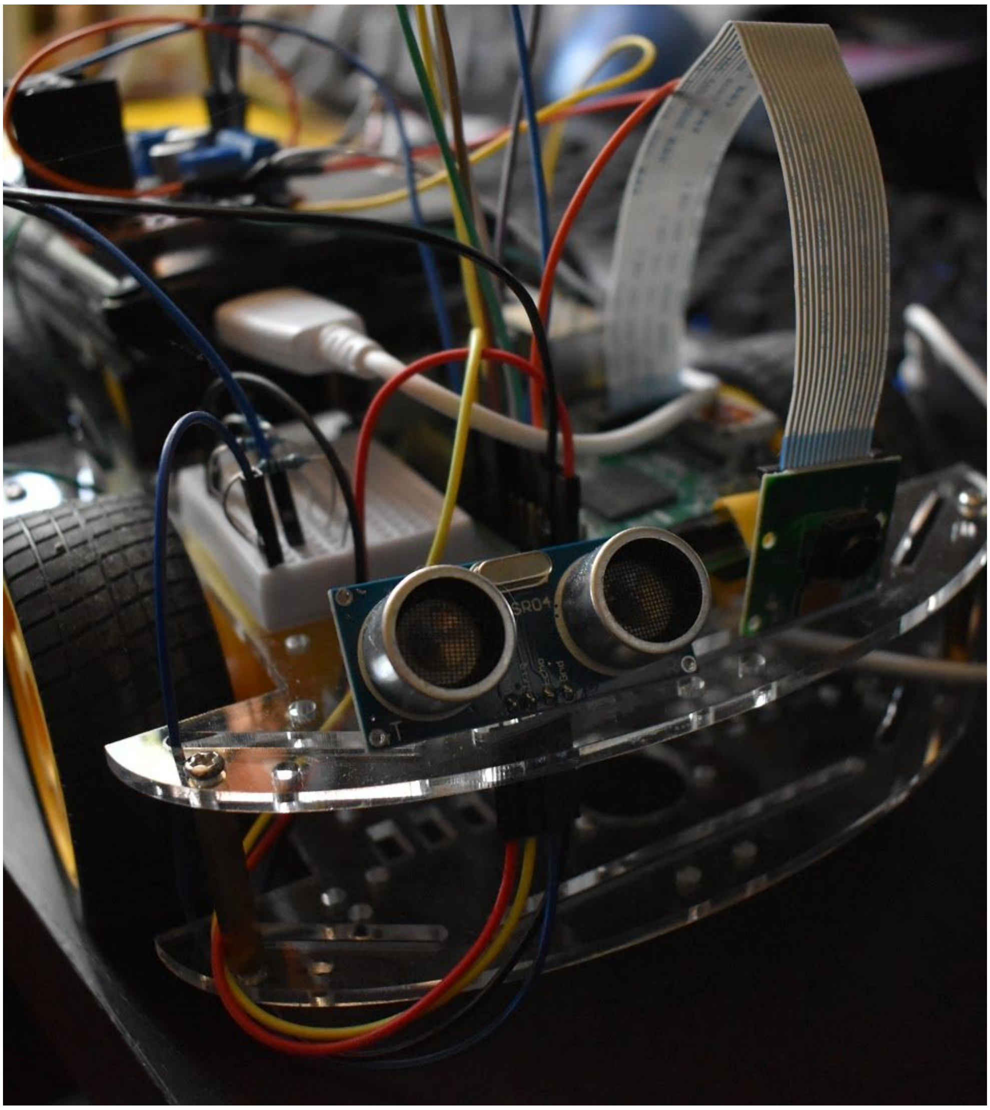
On top of that, we draw the first idea of which batteries we would be using to power the entire robot. Our choices fell on a classic power bank for the Raspberry Pi and the connected sensors, as the presence of a USB cable wouldn't require any extra steps to connect them, while we ordered 4 LiPo batteries (3.7V each) to power the DC motors.

Finally, we tested the camera for the Raspberry Pi. Results were again satisfying and are described in detail in the coding section.

To sum up, combining the tests made during these two weeks with those made in the previous ones, we learned how to work with these sensors, getting a good idea of their performances.

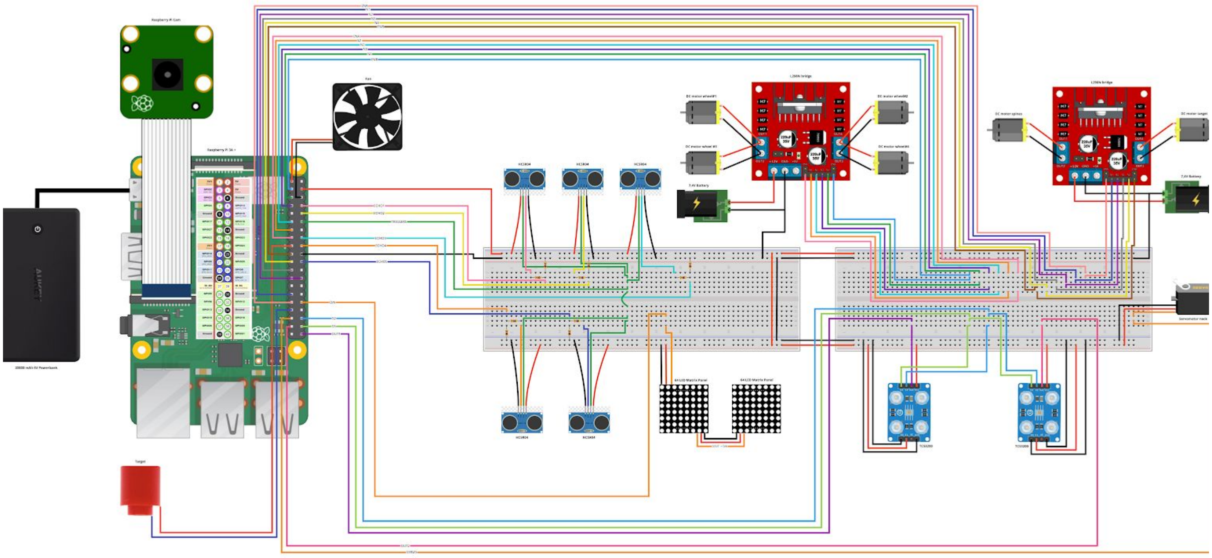
WEEK 6-7

This period marked the passage from tests based on Arduino to ones based on Raspberry Pi. We started with easy circuits, to gain confidence, working our way up towards a more complete system (like the one shown in picture) that included motors, ultrasonic sensors and camera.



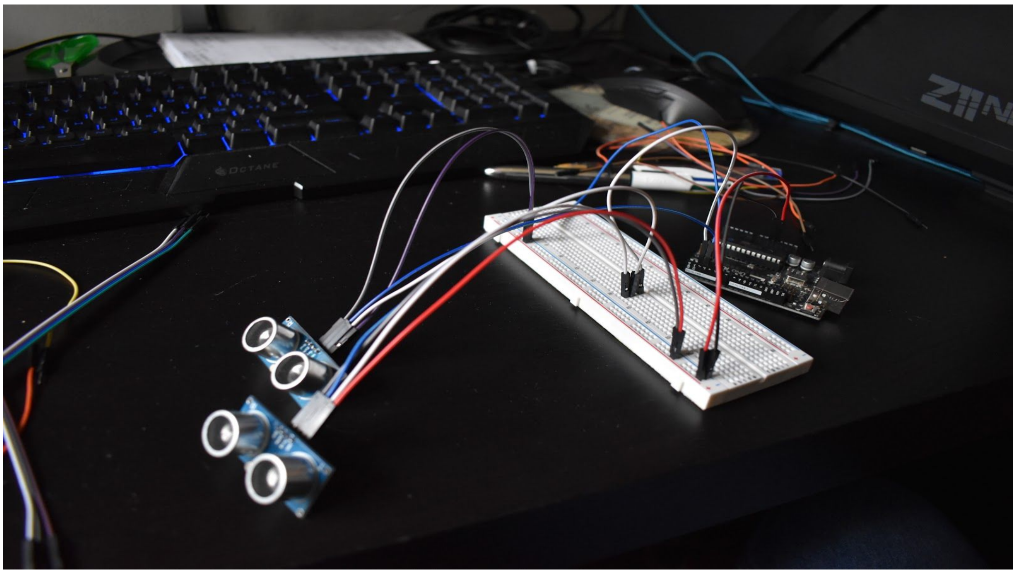
The autonomous car adapted to work with Raspberry Pi

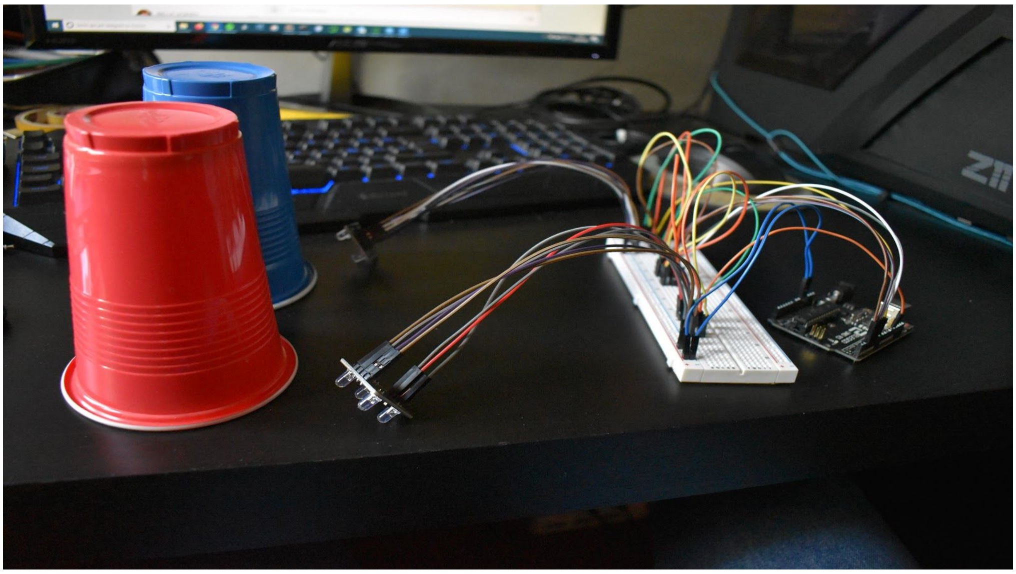
We used these two weeks to complete a wiring schema, putting all components together. While doing this, we made sure to have enough pins for everything on our board by joining a few components together (for example we used one pin as input for all the ultrasonic sensors). Here's the result of our work:



The first draft of the wiring schema

To be sure that joining the inputs wouldn't affect performances, we made some physical tests about it with color and ultrasonic sensors. Both worked perfectly fine.



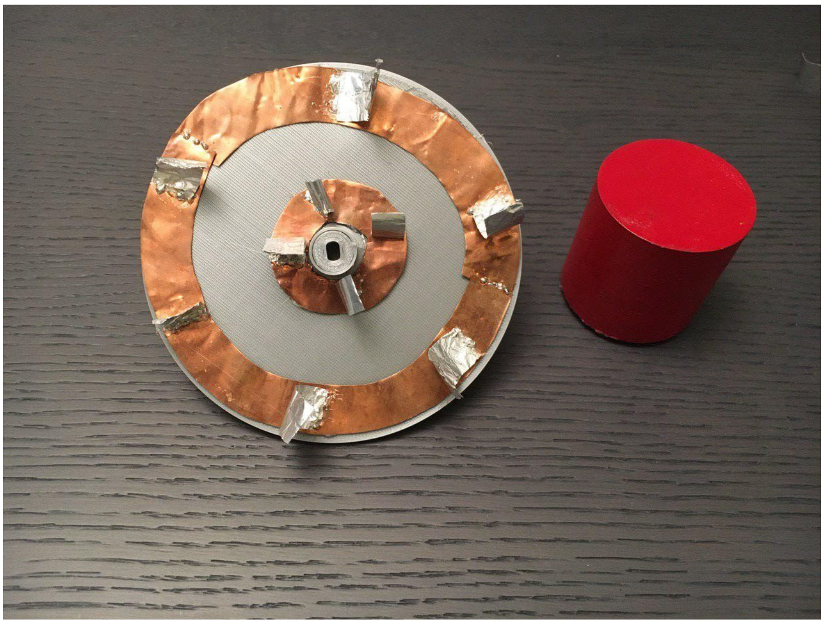


Prototypes to test the effectiveness of our wiring schema

LATER DEVELOPMENTS

When we met to build the robot, we followed the guidelines decided through the tests made in the previous weeks.
Although we encountered issues here and there, the final wiring was the same as we had decided earlier.
The main difference was in the batteries we ended up using. The website where we ordered the LiPo batteries never delivered them (edit: they delivered them at the end of June), so we had to resort to normal AA batteries to power the motors. In the end, we had to use a total of twelve of them, causing more than a few issues with weight and space.

Except for that, it is worth mentioning how we managed to get the target to work (which was basically the last element of the robot we had to develop). Our original idea was to simply use a DC motor to rotate a platform on which the target would be placed. The target would be connected to the board through two wires (one for Vcc and one for ground), which were supposed to pass through the hole inside the axis of the DC motor.
Unfortunately, once we actually got to the moment of building the target, we realized there wasn't enough space inside the motor to get the wires through.
Nevertheless, after a reasonable amount of panicking, we came up with a new idea to combine the rotating platform with the necessity of connecting wires to the target. The final design used two copper stripes on the body, together with two copper stripes on the lower side of the platform, connected between them through some aluminum leaves. The aluminum leaves would allows the current to flow through, without impeding the free rotation of the platform. The target designed by the professors completed the job closing the circuit formed by the copper stripes.



The final target with the moving platform

Another problem we encountered happened towards the end of the project when we were finalizing the last adjustments of the robot. Suddenly, the motor bridge connecting the board to the two DC motors controlling the target platform and spines stopped working. We couldn't find any reason as to what happened and the timing was too short to replace it, so we had to come up with a new solution to control the motors. Luckily we had the idea of just using two NPN transistors (which were easy to retrieve in a local electronics store) to do the job. This turned out to be not only effective but also much cleaner than the original solution. It was actually a shame not to have thought about it since the beginning, as it would have saved a lot of space in terms of wiring and as it allowed us to remove four out of the 12 AA batteries we were using.

The NPN transistor we used to replace the motor driver

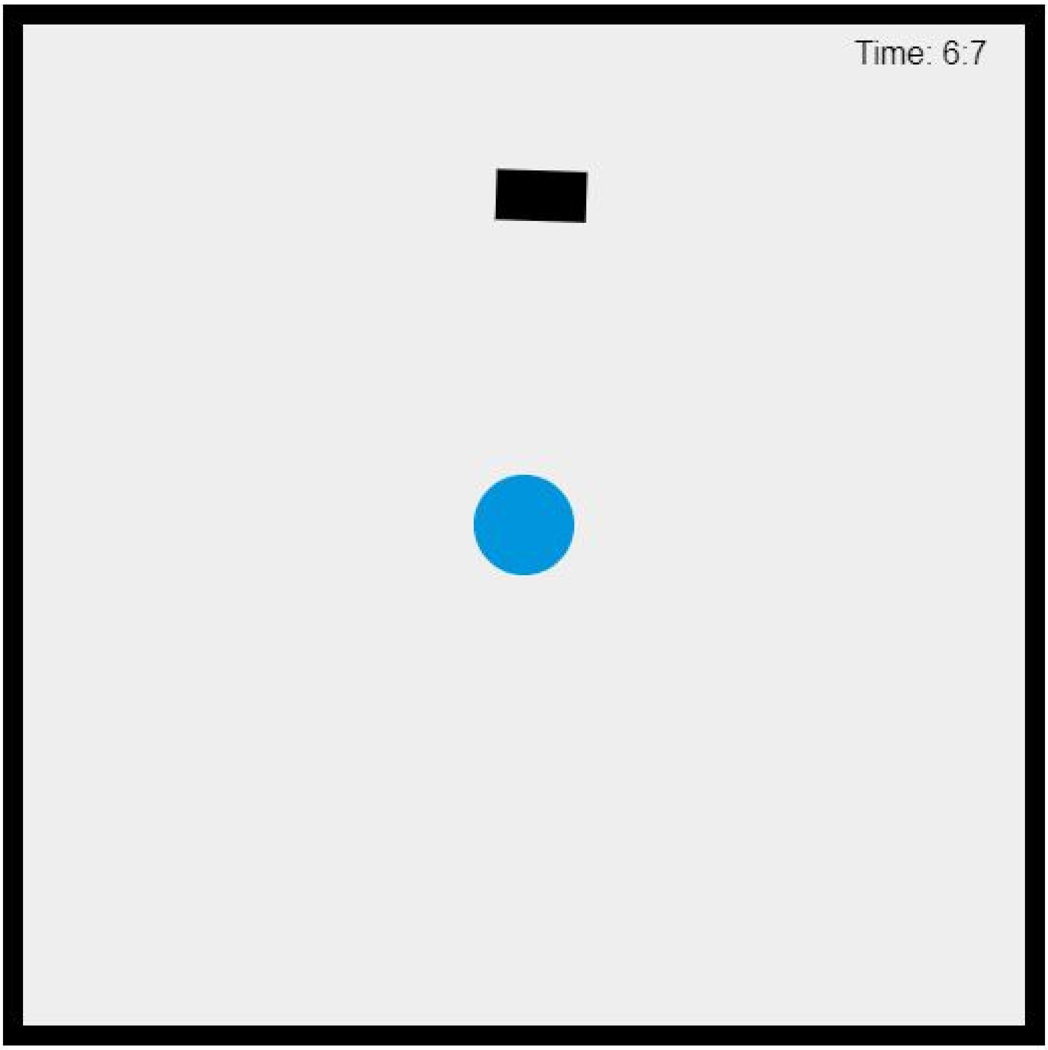
Coding

WEEK 3

Although part of our coding efforts went towards the testing of the sensors described in the previous section, the two main tasks we had to deal with were designing a flow chart and state diagram of the robot's behavior and creating a working simulation of the pond research algorithm.
We also decided to divide the designing of the software into five tasks:

- Pond detection
- Attack mechanism
- Defense mechanism
- Emotion communication
- Score computation and communication

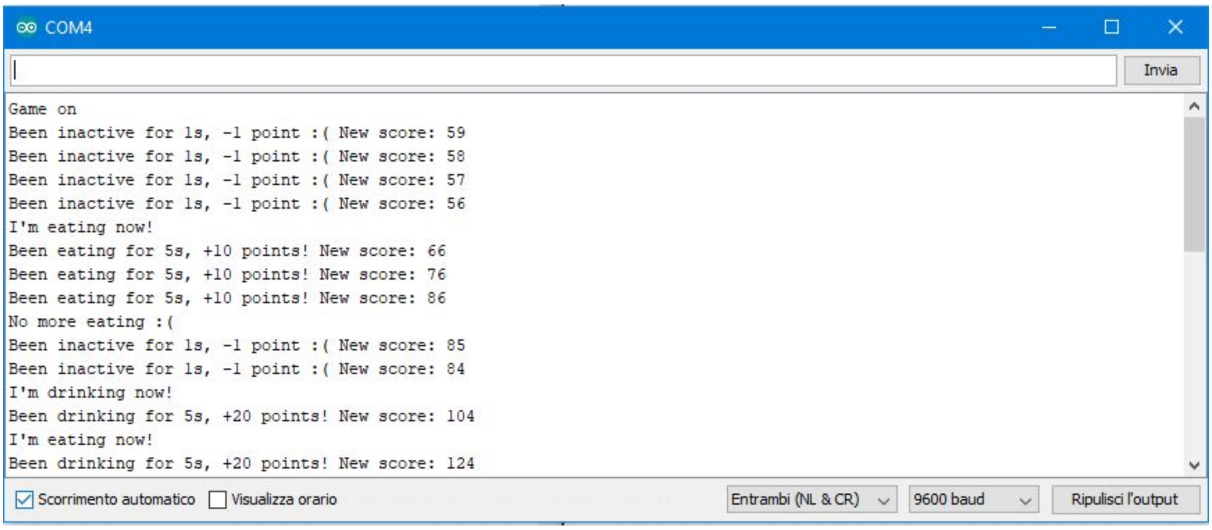
For each, we thought first of a high-level algorithm, then we worked on detailing it by dividing it into smaller subtasks until we get to commands that can be directly translated into small portions of code. Parallel to this, we tested the algorithms with virtual simulations, in order to judge their performances without having the actual robot available. Given the difficult situation due to the Coronavirus, this was of paramount importance, as we didn't know how much testing we would be able to perform before the final delivery.



Screenshot from one of the simulations for the pond research algorithm

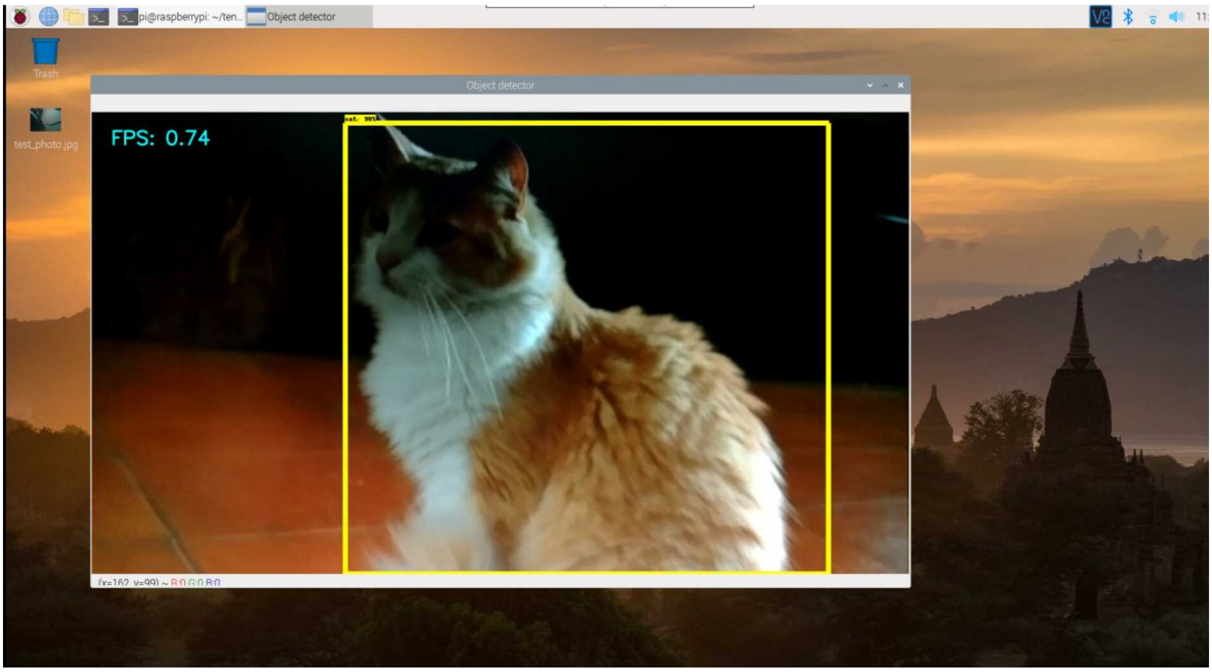
WEEK 4-5

During this period we entered deeper into the coding sections of the project.
First we started by simulating the algorithm that counts the score with an Arduino board. This was a rather simple task, but nevertheless we decided it would be useful to spend some time on it to make sure it would work smoothly. Turned out to be a wise a decision. While coding it, we realized there exists a small issue with integers in Arduino: they are encoded in 16 bits; usually this is not a problem, as rarely you store high numbers in normal applications, but in our case we had to store the value in milliseconds of the passing time. Since 16 bits can encode as maximum number roughly 32000, this meant that after 32 seconds the program just returned random numbers. The fix was extremely easy, we just used long int instead of simple integers, but it was a good reminder that even small tasks can hide traps.



Simulation for the score counting algorithm

The other main task we worked on was much bigger, and probably one of the main challenges of the entire project: developing a computer vision application inside our robot. As mentioned in the previous paragraphs, we opted to use a Raspberry PI specifically to target this task. After having it ordered and received, together with a PI camera, we began studying the problem. Luckily, the Internet is full of tutorials on how to do it, and, after a couple of days of installing and uninstalling all the required libraries, we were able to get a first working prototype capable of following a yellow ball pointed by the camera. As a side project, we thought it could be fun to extend the application to recognize other objects, so in a couple of hours we built a cat detector using some pre-built models found online.



Prototype for the computer vision part of the project

WEEK 6-7

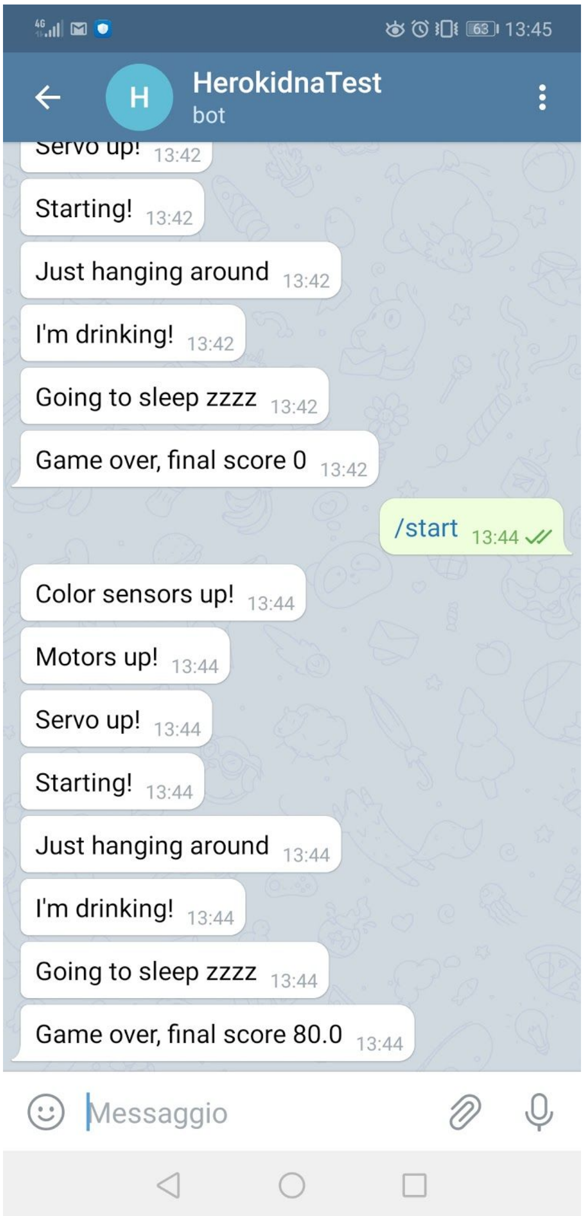
Continuing from the results of the previous weeks, we kept working on the computer vision, improving the software, and realizing a working prototype. This was still very raw - the model could just more or less follow a red glass using an extremely simple controller - but the outcome was still encouraging. However, it also confirmed our suspicion that relying only on the camera to guide the robot was way too prone to change (for example, experiments showed that having a light on the front could change dramatically the way the robot perceives a color).

A brief show of the computer vision

LATER WEEKS

Until the end of the project, we kept improving the script of the robot by correcting mistakes and adjusting the code via numerous tests. Parallel to this, we also added some more functionalities, like:

- A telegram bot used to send commands to the robot, as well as to start it during the competition.
- A connection to Spotify, to be able to use the robot as a remote speaker.
- The possibility of adding new wifi connections through the use of a USB key, without having the necessity of attaching the Raspberry Pi to an HDMI cable.
- The possibility of adding new music in the same way.

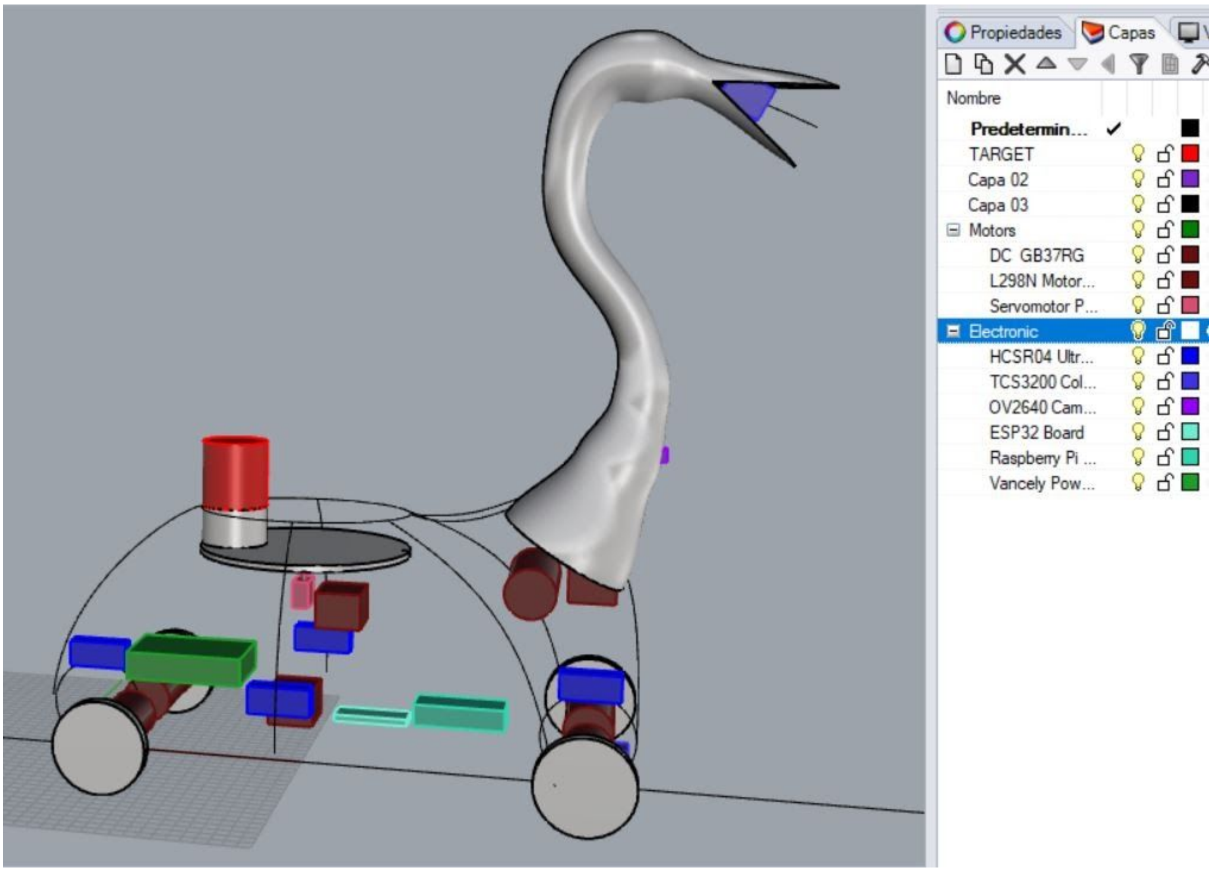


One of the tests we made with the Telegram bot

Structure

WEEK 3

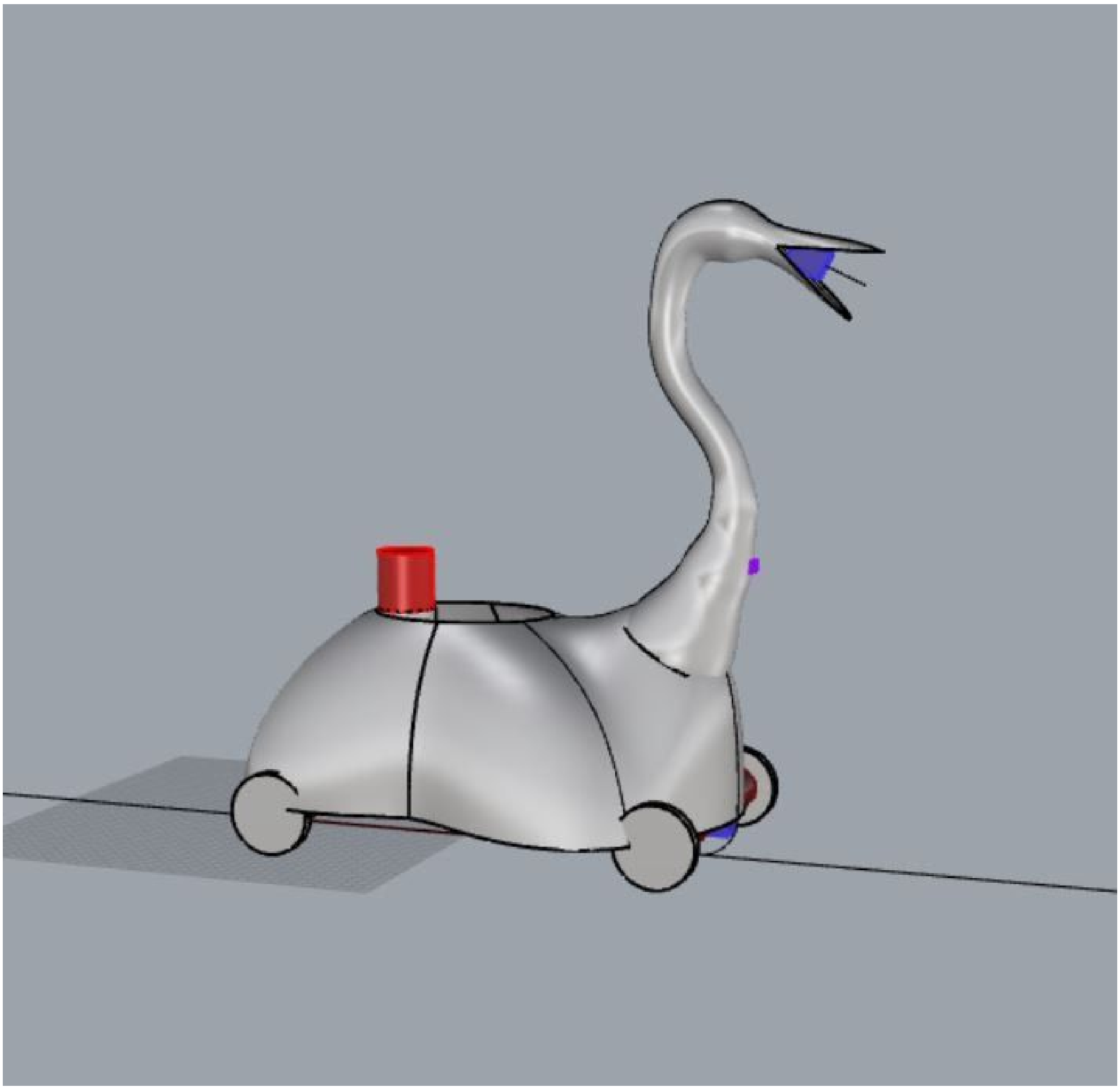
We used the third week to develop the movement mechanisms and to realize a 3d model of the robot. This gave us the opportunity to think about where we would be placing each component, from motors to batteries, as well as the mechanics inside the robot. This first approach was freer and aimed to assemble the components inside the animal shape, not been strongly attached to technical details.



A first draft of the robot's interior

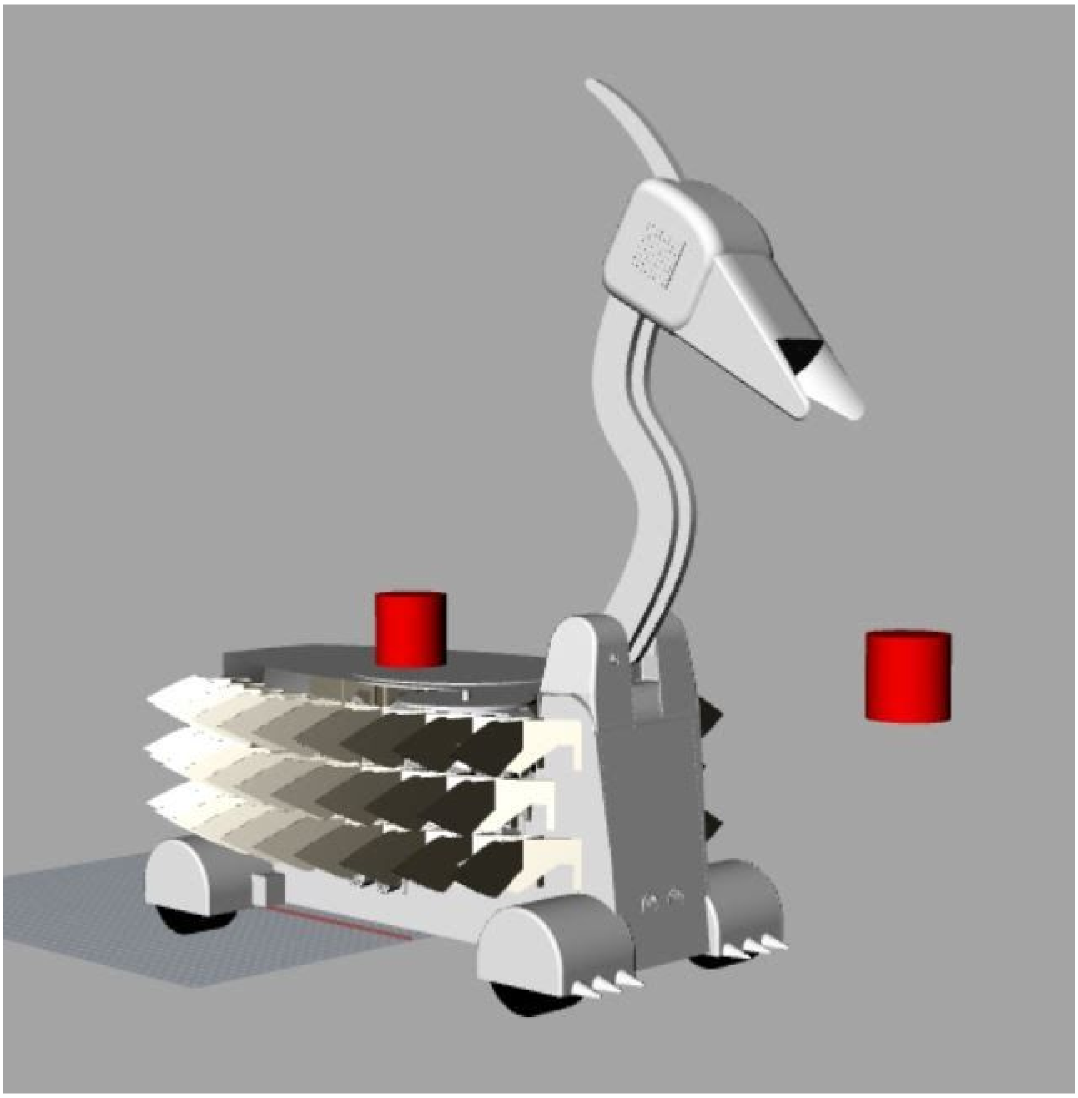
Shape

In the following three weeks, we kept working on the 3D model, obtaining, in the end, a complete simulation of what the final robot will look like. One of the challenges was to find a shape that could express the shape of the chosen animals, and translate them into a "robotic" feasible aesthetic. Below you can find the progress step by step.



One of the first designs of the external shape

In the second stage, we tried to follow a similar approach as we did for the logo. At this point, we already had in mind to produce it through 3D printing, having this in mind we developed the shape in separated modules.



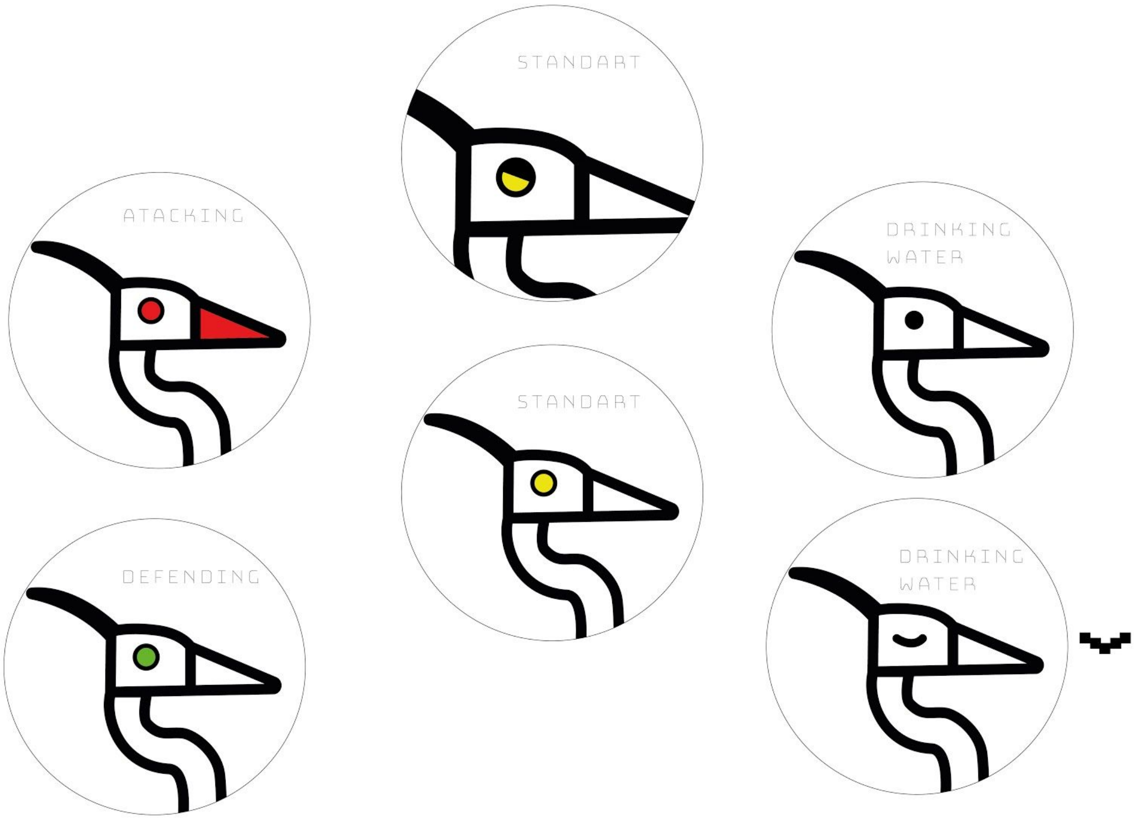
One of the designs closer to the final shape

A close representation of what the final robot would look like

The main issue was realizing a shape that could be both effective, resistant, and aesthetically pleasing. All of this had to take into consideration the limited access we had to the labs, due to the Coronavirus. To solve this problem, we contacted several shops which performed laser cutting, asking for budgets from them. To lower the final price, we resorted to using our own 3d printer to print some of the most detailed parts (like the spines or the pieces to hold the electronics).

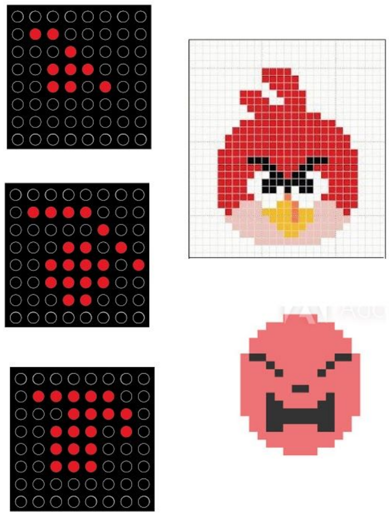
Other design aspects

Along with the body shape, we studied options to represent the emotions and give personality to the robot.
In a first try, we considered using a single lamp/led that would modify colors according to the robot's actions. Facing some color restrictions due to the target and pond color, we decided to abandon this idea to explore something more detailed. We found out the possibility to use a LED matrix, with which it would've been possible to draw more expressions and explore animation movements.
In a third try, we bought an RGB matrix, allowing us to work with any kind of color and mixed it inside the same draw. Below you can find the pictures of the tests.

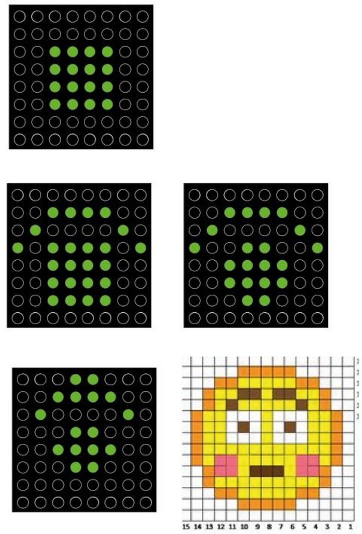


The first concept for the expression

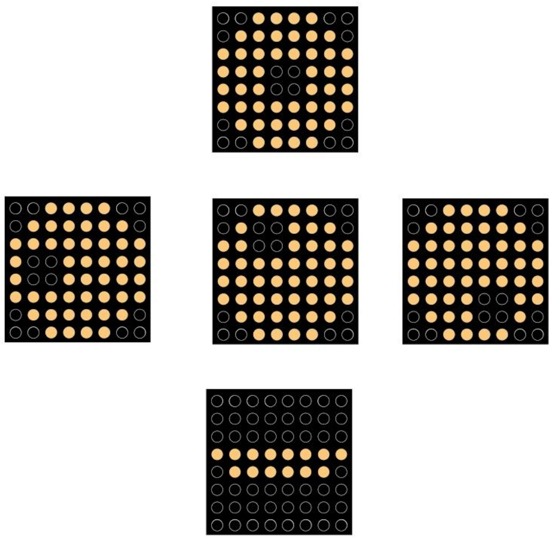
ATTACK OPTIONS



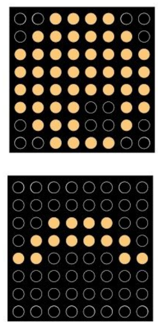
DEFENSE OPTIONS

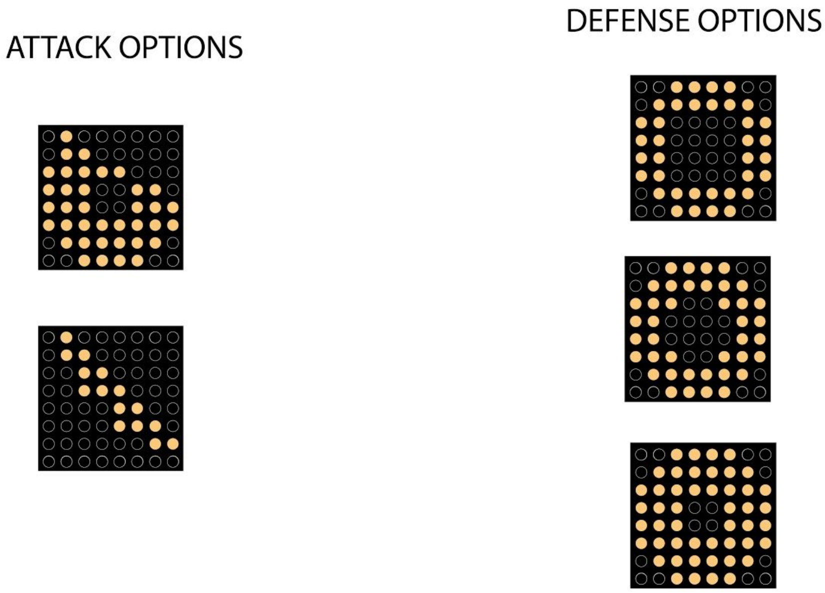


STANDART

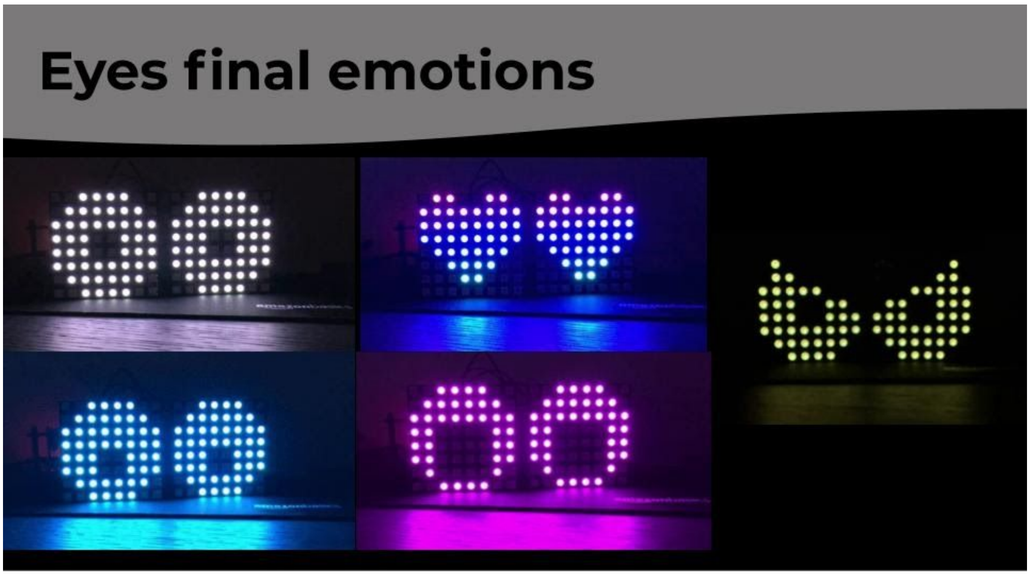


DRINK WATER





The second concept for the emotions using a unicolor LED matrix

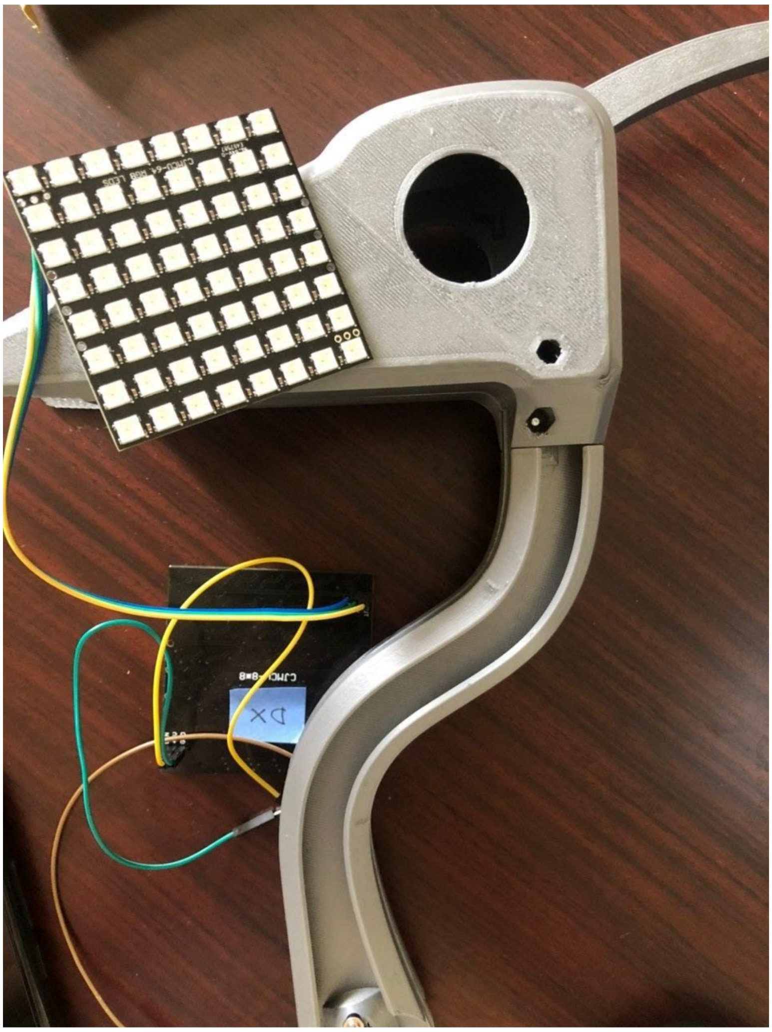


The final concept to represent emotions through the eyes

To convey more emotions, we decided to add some music related to each different emotion of the robot. Adding this was relatively simple, as we only had to buy a small speaker to be attached directly to the Raspberry PI. We opted for some videogame style music, choosing various soundtracks from the classic *Nintendo's Super Mario Bros*, each emotion represented by a different sound from the game. We instead chose another classic to close the competition, "*We are the champions*" by the Queen.

General issues and setbacks

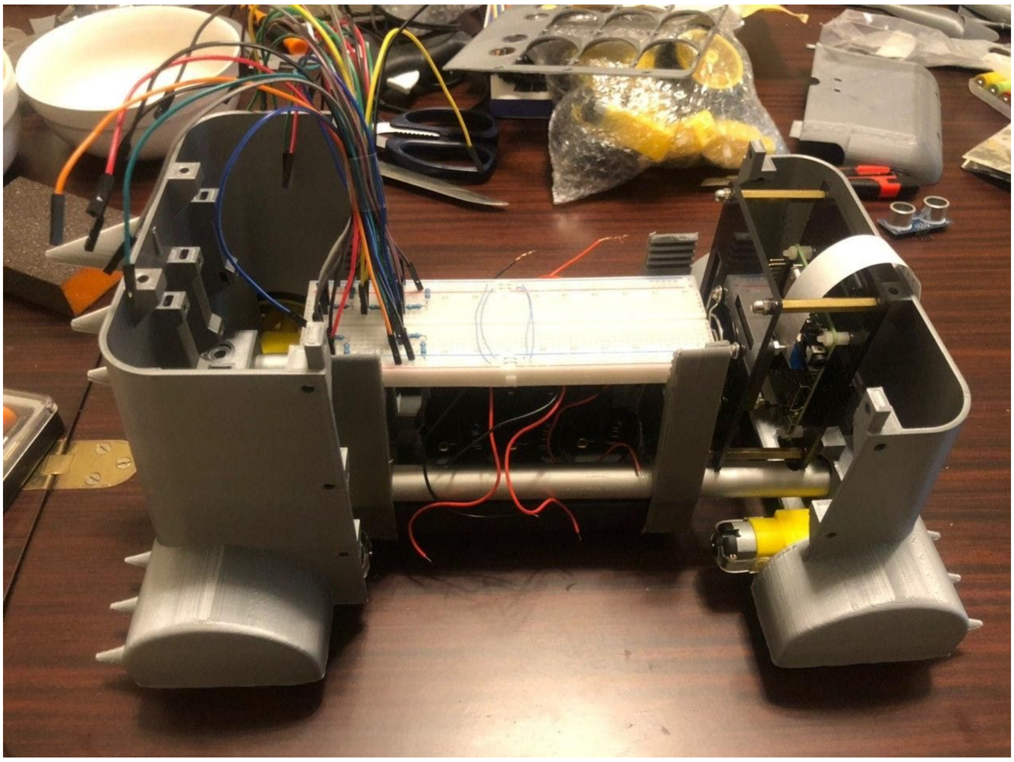
As mentioned in some of the previous paragraphs, not everything went according to plans. The coronavirus outbreak gave us a margin of error, as we had to assemble all the robots in the few physical meetings, despite those problems we were able to organize.
This caused a few issues with pieces that didn't fit, like the head which was way smaller than the RGB matrices used for the eyes, or the wheel holders, which turned out to be smaller than the actual wheels both in terms of diameter and width. This forced us to print the head again, while we had to cut the wheels and remove the tires to get them to fit. These issues were caused by the fact that our designers couldn't see the pieces with their own eyes until after the print and had to rely only on photos and measurements, which was bound to cause mistakes.



The first attempt of the head piece


The second version adjusted to the size

Another big problem was with power. We had planned to use LiPo batteries for the motors, but the ones we ordered only arrived at the end of June, making them useless for our project. We, therefore, had to use normal AA batteries, causing a surge in the robot's weight. Not only that, but the smaller space available in the interior made it difficult to get clean wiring, as well as making it more difficult to have an easy way to access them in case they needed to be changed.



First progresses to set components and the robot structure

Some photos made during the assembling process

However, although some things could've been better, we really appreciated the results: our robot turned out to be so good-looking that during our video shooting we ended up being continuously stopped by people curiously asking what we were doing and complimenting on the robot's look. We also enjoyed the way eyes and music were in sync with the robot's emotions and we were really proud of how the attacking neck and the moving target worked smoothly in the end. Finally, we were more than satisfied with how we managed to turn the spines from an aesthetic-only addition to an active and moving part of the robot.

Phase 4: Deliver

After roughly two months of development, we were able to deliver our robot in its final shape.

Final Robot description

Strategy

The first aim of the robot is to look for the pond. It does that by following a research algorithm developed by us which uses ultrasonic sensors to help guide it through the competition field. If the pond is actually found, the robot stops there and drinks.
To defend itself, we positioned the target on a rotating platform which is moving all the time. In our opinion, this alone is a sufficiently effective countermeasure to avoid other robots' attacks, but on top of that, we decided to add some checks from the ultrasonic sensors. Easily enough, if one of the sensors on the side or on the back perceives an enemy approaching, the robot starts an evasive maneuver to escape it.
As for the attacks, they are performed only when the chance of success is reasonably high, that is, the enemy target shouldn't be simply detected by the camera, but it should also be seen as big enough.

Shape

The final shape of our robot can be seen in the photos bellow:

Photos taken during the day of the video shooting, after the robot was completed

The neck and upper part of the body were built in an S shape, to recall the shape of a heron, one of the animals used as inspiration, while the lower body is more similar to the echidna, the other animal we took inspiration from.
The spines can be opened and closed, making a great mix together with the eyes for expressing our robot's emotions. We added some more spines, for aesthetic purposes, on the front and on the back, gaining an elegant looking shape that is still capable of conveying aggressiveness.

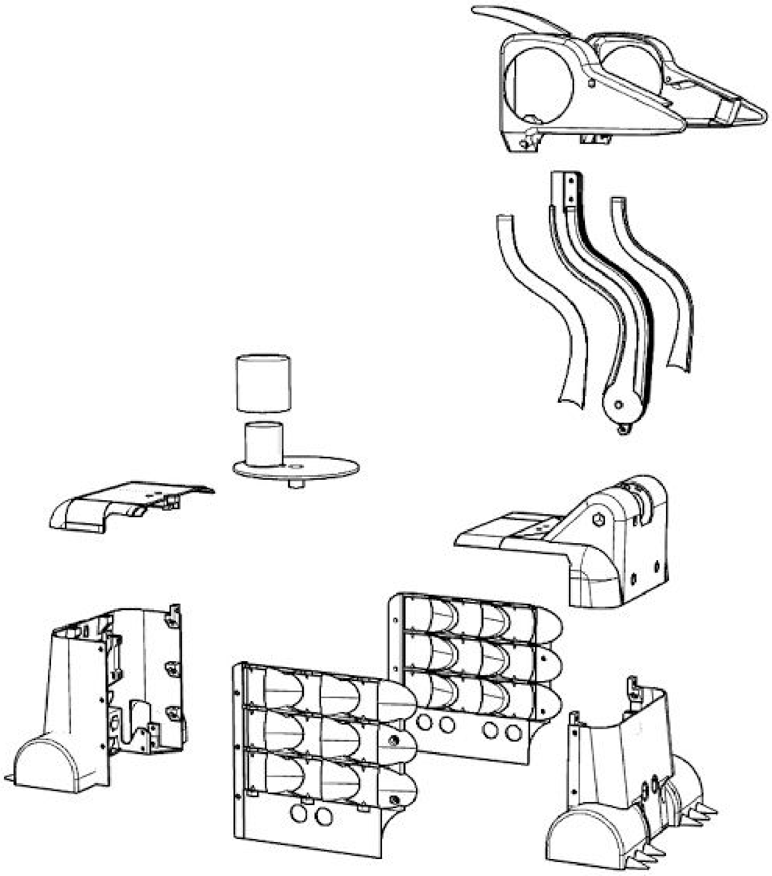
The wheels stick out from the main body but are hidden by an apposite cover which also works as a protection against hits from other robots. A few more spines complete the picture, giving a cleaner look to the final shape.
We used grey as the main color, avoiding the use of painting on top of the PLA. In our opinion, this gives the robot a nice, modern look.

Here you can find the link to the final 3D model of the robot.

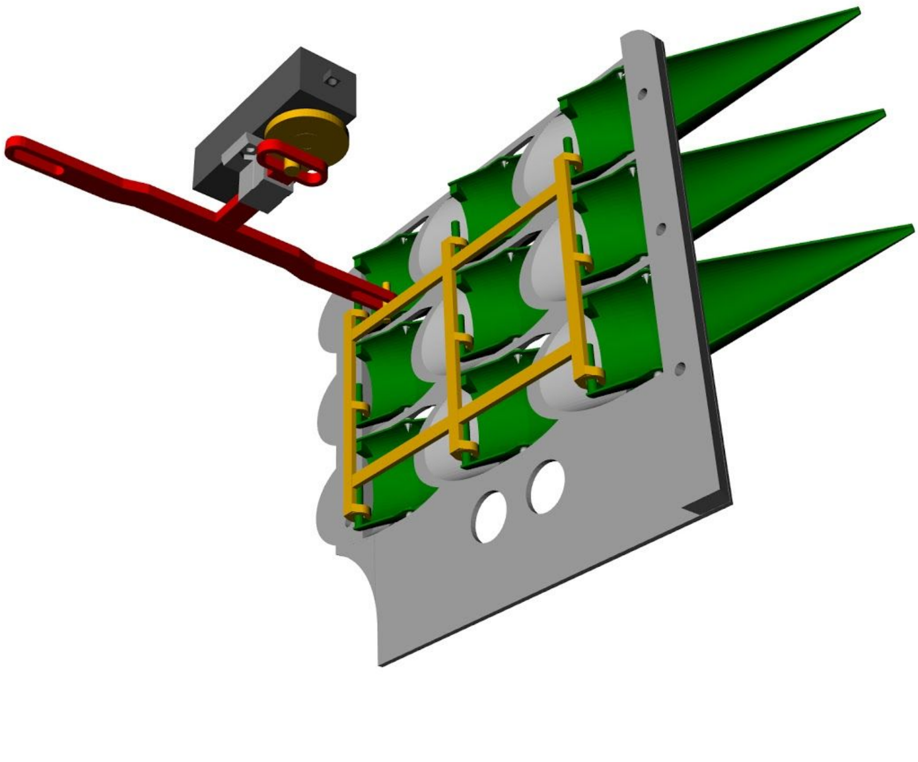
Mechanics

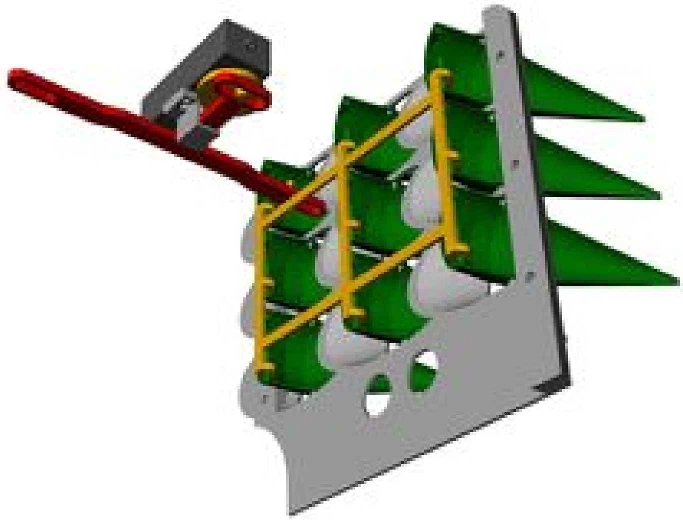
The structure of Herokidna is composed of eight 3d printed parts that make the shell and contain the internal components, plus two tubes reused from a mopping stick found at home.
The eight 3d printed parts are FRONT, BACK, TOP FRONT, TOP BACK, SIDES, NECK, HEAD.
The tubes are placed horizontally and work as joints between the BACK and FRONT parts to complete the main chassis. In addition, two 3d printed parts in the middle of the tubes help in keeping in place the inner components such as power bank, batteries, and breadboard.
In the BACK part are attached the rear wheels, the DC motors for these wheels, the rear ultrasonic sensor, and the two motor bridges.
In the FRONT part are attached the front wheels, the DC motors for these wheels, the front ultrasonic sensor, the color sensor for the pond, and the raspberry board.
The SIDES are screwed to the BACK and FRONT sections. They hold the SPINES, and the lateral ultrasonic sensors.
In the TOP BACK, part is attached the DC motor of the target. This part is screwed to the BACK.
In the TOP FRONT part we can find: the camera, the servo motor for the neck, the DC motor for the spines, and the upper part of the raspberry board. This part is screwed to the FRONT.
The NECK is connected through the servo motor to the TOP FRONT from the left, and also with a screw in the TOP FRONT, that holds it from the right. Through the NECK go all the cables for the components in the HEAD. The NECK is built by two mirrors “C” section pieces glued together, plus two caps one for each side, to cover the cables.
The HEAD is connected to the NECK by shape fitting and a screw. The HEAD is built by two mirror 3d printed shells, one left, one right, screwed together, plus the hair tail. In the HEAD we find the color sensor for the target and the eyes.

Then there is the mechanism of the SPINES that consists of 9 SPINES for each side, the SIDES parts as holders of the spines, and as rotation point. Then the INSIDE HOLDER of the spines, on each side, that keeps the 9 spines together and transmits the movement from the motor through the linking T shape piece and the rotating disc.

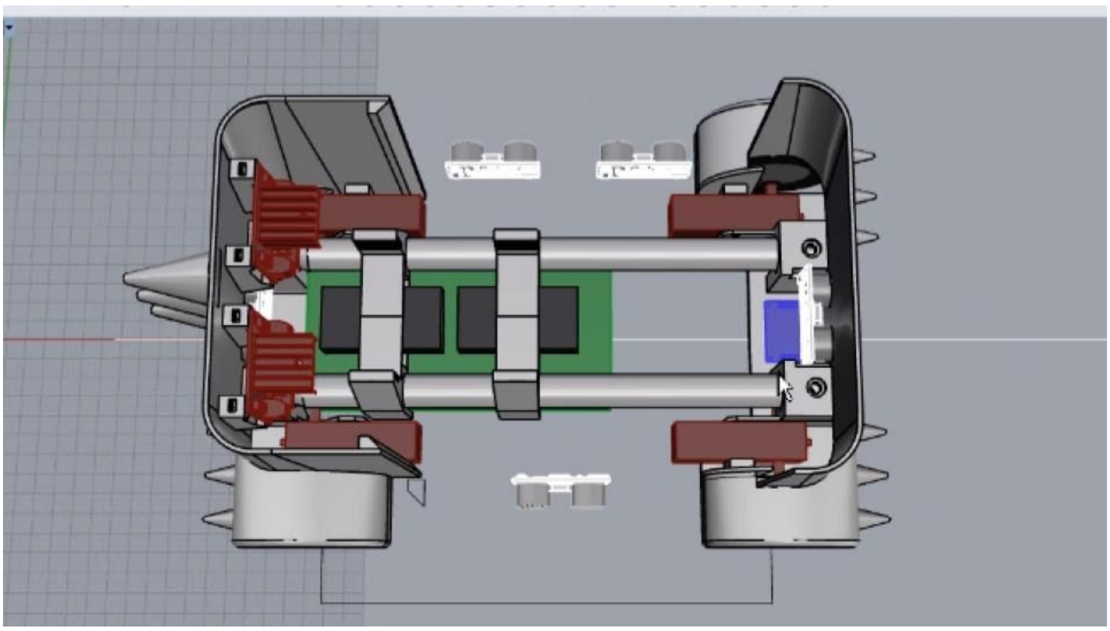


Exploded view of the robot





Spines mechanism



Internal view of the 3D model

Electronics

The electronics represented the biggest part of our budget, necessary to accomplish all the goals required to make such a complex system work. The robot required several sensors to be controlled effectively, a total of seven motors in order to move wheels, target, neck and spines (which again required drivers to communicate with the board), a board capable of analyzing images and batteries to power everything.

The sensors had to be as precise and effective as possible, so before buying them we had to try as many as possible. Unfortunately, as the pandemic was going, it has been difficult to do this, as well as to find available stores to buy them. Luckily, between sensors we already had, suggestions found on the Internet, and shopping on Amazon, we were able to retrieve everything we needed, all without buying useless components. Here's the list of all the sensors we used in the end:

- 5x Ultrasonic sensors (model HCSR04), to detect obstacles in the direction of movement, as well as approaching enemies.
- 2x Color sensors (model TC3200), to detect if the robot is placed on the pond and to detect whether it's eating.
- 1x PiCamera, used to send images to the Raspberry Pi.

As for the motors, they had to provide good torque while also requiring no more than 12V. And, since we needed at least six (they became seven once we added the moving spines), they also had to be reasonably cheap. In the end, the motors we used were:

- 6x DC motors (4x for the wheels, 1x for the moving target, 1x for opening and closing the spines), classic yellow models used often when making projects with Arduino or Raspberry Pi.
- 1x Servomotor (model SG-5010), to raise and lower the neck.
- 2x Motor bridges (model L298N), to drive the six DC motors.

As mentioned earlier, to control everything we needed a board capable of analyzing images in real-time, in order to detect enemies' targets. Initially, we had taken into consideration the ESP-32, but we quickly realized it wasn't powerful enough to do that, so we moved to the Raspberry Pi. We chose the 3A+ model, as we thought the greater computation power provided by superior models wasn't worth the price. Though initially, we had planned to integrate the board with an Arduino nano, we managed to get everything running just with the Raspberry, which saved us some more money (to do this we had to resort to some tricks, like connecting all ultrasonic sensors to the same trigger pin, but in the end, everything worked fine). To avoid overheating, we also bought a case with a fan, which would also serve to protect the Raspberry from eventual hits.

To power all of this, we had some problems. Our initial idea was to use four lithium batteries (3.7V each) to power the DC motors, while the rest would be powered by a power-bank through the Raspberry. We found cheap lithium batteries online, but they were never delivered (edit: they were delivered but at the end of June and we had some doubts on their quality). Luckily we had extra battery holders, so we just chose to use AA batteries (12 total, 8x for the wheels and 4x for target and spines). Here's the final list of what we bought:

- 1x 20000mAh Power Bank by AUKEI.
- 4x 3.4V LiPo Batteries, never delivered.
- 12x AA batteries

- 6x battery holders (2 batteries each)

Finally, we had to buy one more thing, that is, two LED matrices to be used as eyes. The first model we found turned out to have only red lights, which would go against the rule of not using the color red outside of the target. However, we managed to find a new model capable of changing color, which also resulted in a more pleasant look.

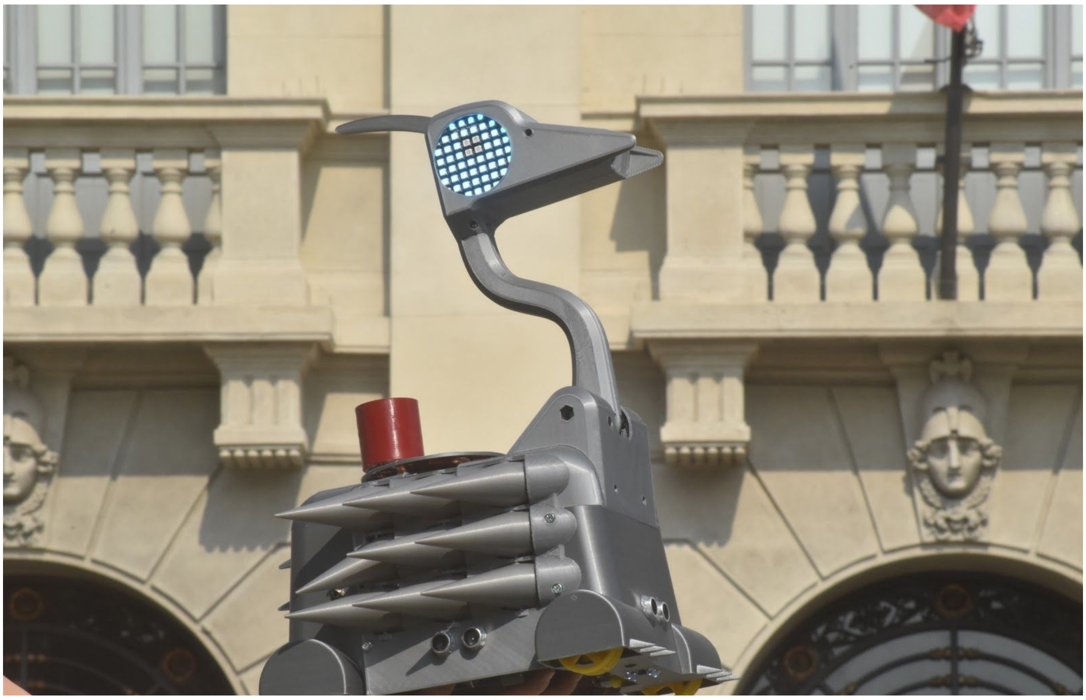
Informatics

The way we approached the problem of building software to control the robot was to tackle every single task separately, combining everything together in a later moment.
The final result was a “state” system, where the robot performs actions according to the state it’s in, which in turn changes according to measurements coming from the various sensors. In particular, the front ultrasonic sensor, the camera, the target and the color sensors are constantly checked to see if the robot is approaching an obstacle, an enemy, if it’s drinking, eating or being eaten.
All this work is performed reasonably fast by the Raspberry Pi (our experiments showed it to be in the order of a hundredth of second).
On top of that, we created a Telegram bot called *HerokidnaBot*, which we used as the robot controller. This turned out to be a really good idea, as it made testing extremely intuitive, as well as giving us an easy tool to start and calibrate the robot on the final competition.
Leaving aside the normal libraries (like gpio), we used opencv to analyse the camera images and telepot to build the Telegram bot.
The code plus some instructions to install some of the required libraries can be found in our GitHub repository (https://github.com/steflyx/Herokidna).

Video & photo shooting and development

With our robot ready, we started to design a storyboard to be followed in the video. According to the robot concept, we followed a more humoristic approach. Presenting Heronkidna in the city and in nature, we aimed to express the funny explorer character of our robot. In the video we chose a videogame-style approach, selecting soundtracks from classics like *Nintendo’s Super Mario* and *The legend of Zelda*. This could enhance the epic performances we were expecting from Heronkidna in the battle.

As a final task with the robot ready, we went to Leonardo campus to make photographs of the final outcome. We decided this place for the traditional architecture of its buildings, which would give a nice contrast to the cyberpunk style of Heronkidna.



Two photos from the day of the video shooting in Leonardo campus

We had also the opportunity of taking some shots in the nature, made by Matteo in the Aprica region. The result was quite interesting as we could create a scenario reminding the original animal heron in its natural environment.

Two more shots taken in the Aprica region by Matteo

How we handled the pandemic

We'd be lying if we said that this whole situation didn't make things more difficult, but we can say we managed to handle the situation without too many setbacks.

The main problems were in retrieving all the pieces we needed as well as finding a place to build the structure of the robot. In the end, we resorted to using mainly Amazon, which meant slightly higher costs for some products, but sure delivery times and quality of service (to be noted that one of the few products we didn't buy on Amazon, the lithium batteries, never arrived). As for the structure of the robot, we were lucky as our designer Sofia decided to buy a 3d printer for herself in May, so we ended up using that to build all of the robot.

We also had a small problem with the postal service, as some of the parts were in Stefano's house in Rome and, to get them to Milan, we used the mail, but the package was delayed by a couple of weeks. Luckily in the end it still arrived in time to mount the robot.

To meet and to coordinate the work we resorted to Microsoft Teams as it turned out to be a great software for video calls. We also used other softwares to share files and stuff, such as Google Drive and Github for the coding part. On top of that, we kept in touch through Telegram, which, being on the phone, is much more effective when fast replies are needed.

In conclusion, though the situation was complicated, we were able to overcome it, thanks to a decentralization of the work (each one of us focused on a part of the project in his own house) and to the heavy use of technological aids (like the aforementioned Microsoft Teams). When the quarantine was finally lifted, all we had to do was just putting all the pieces together, according to what we had previously planned.

Conclusion

What did we learn from this experience?

From a working dynamic perspective

When the external conditions forced us to work remotely for what was planned to be a hands-on course, we had to find digital workspaces and tools to make the communication flow work. However, what in the beginning sounded almost impossible, actually pushed us to develop new skills regarding collaborative work and an effective division of tasks.
Related to assembling logistics, we did not face as many problems as we could've since the majority of the team was in Milan and did not need to move around the robot too much.
In that, we were helped by planning in great advance which materials, components, and services we would need to buy and use, as well as planning where and how to buy them. Following the various steps of our roadmap on time was crucial to get us to complete the project.
In conclusion, the coronavirus made our lives much more difficult but didn't stop us from finishing our work.

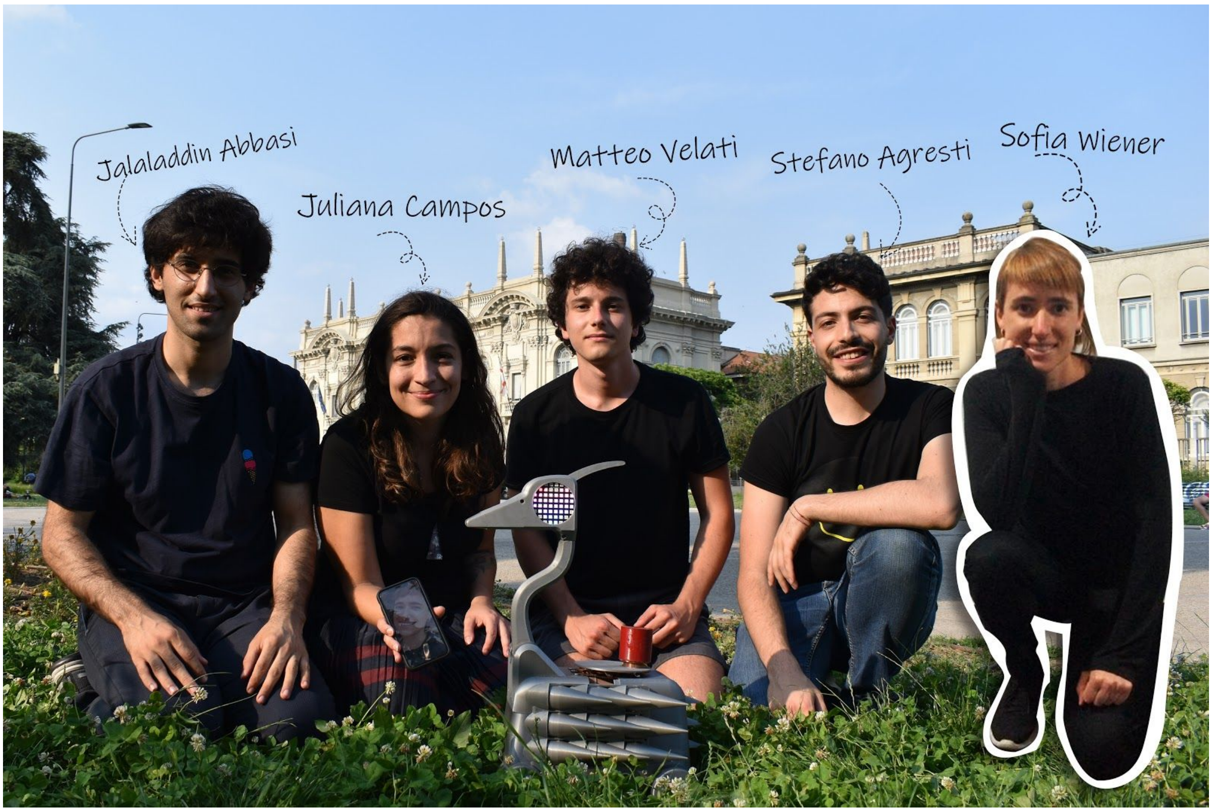
From a project perspective

The final result of our project was beyond our expectations. Although several issues arose during the development, we've been always able to overcome them one way or another.
Our main regrets are related to the power supply. We didn't have an electrical engineer in the group, so we had to choose the safest way when it came to it, which meant using two different sources for board and motors, causing a surge in the robot's weight, as well as complicating severely the wiring. As mentioned previously, we also had to improvise with the batteries, as the LiPo ones we had chosen weren't delivered on time. Going back, we could've certainly thought of better ways to tackle this part of the project, but given our lack of skills and given the small margin of error due to the coronavirus crisis, this is the best we were able to realize.
Another regret is with the dimensions of the 3D printed parts. Since we had to design the whole robot without having physical access to all the pieces until the very end, some of them turned out not to fit, which meant coming up with creative solutions to put everything together.
However, this clearly affected the quality of the final robot. Specifically, we found out that there were some issues when it tried to turn, which didn't work smoothly on all surfaces. This was caused by the combined situation of having wheels too far apart, weight too high and having the tires removed to allow the wheels to fit in the robot.
All in all, had we got the chance of reprinting the robot and rethinking the power supply, most of our troubles could've been solved, but in general, we can say we are more than happy with our project. It wasn't simple to get everything to work together, computer vision, motors, shape… , and all with the added difficulty of having just a couple of weeks to physically assemble it. Yet, we managed to realize a robot that not only does the job it was designed to do

in a decent way, but that is also more than good looking and that is actually entertaining to use, thanks to all the features we were able to add.

From a personal point of view

This class certainly required a lot more work hours than most other classes (and probably a lot more curses too), but it was a refreshing experience from the usually theory-only style that characterizes most of the Italian university.
It gave us the opportunity of challenging ourselves with (almost) real-world scenarios and it really allowed us to understand that a good dose of goodwill and hard work is all that's required to achieve good results, even when these seems unrealistic in the beginning.
It also allowed us to get a new perspective on group projects. Getting engineers and designers to work together is not something that usually happens in university, but it made possible to combine the best of both worlds when creating Herokidna. Who's writing this is an engineer who's sure that, were it up to him to design the robot's shape, Herokidna would've been little more than a box with wheels, so for sure, working with people with different backgrounds really got us to make that extra step that made our project not just ok, but rather good.
In the end, we had to work a lot, but we also had fun and enjoyed the process (and maybe - just maybe - we might build some other robot in the future!).



Unfortunately we were never able to meet all together, but we managed to find a workaround

APPENDIX

Bill of materials

This is a simplified version of the bill of materials. Here we only considered the materials that were actually used in the final robot, while we didn't insert those items we bought and later discarded during the development process. However, attached to the report is the complete version, containing detailed information on everything we used in the building of the robot.

Name	Short description	Price (total)
HC-SR04 (x5)	Ultrasonic sensor	€9,99
TCS3200 (x2)	Color sensor	€12,40
L298N (x2)	Motor bridge	€4,80
Power bank (x1)	To power the Raspberry	€19,99
PiCamera (x1)	The camera we used	€9,99
RGB Led Matrix (x2)	For the eyes	€22,98
DC motor + wheels (x6)	Motors for wheels, target, spines and wheels	€23,98
Servomotor SG5010 (x1)	For the neck	€7,40
Raspberry Pi 3A+	Board to control everything	€28,32
Case Raspberry Pi (x1)	To protect and cool down the board	€7,99
Breadboard kit + wires (x1)	To connect everything	€9,99
AA batteries (x8)	To power the motors	€3,49
Battery holders (x6)	To hold the batteries	€8,99
PLA filament	Used for the 3d printing	€45,00
Building tools	Various tools like glue, tape, screws	€18,73
Speaker	Allows the robot to speak	€14,99
Paint	Red paint for the target	€5,50
Aux cable	Cable to connect speaker and Raspberry Pi	€2,99
NPN transistor (x2)	New smaller wheels	€4,20
TOTAL		€260,72

Minutes of the Meetings

Meeting 17/03/2020

Time: 14:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We discussed the animals that could inspire the robot
- In this first phase we tried to consider as many animals and strategies as possible, eliminating those that didn't seem feasible and/or interesting. We thought on how to actually implement some ideas in a very general way.
- The first decision we took was to focus on three animals: weasel (for the distracting smoke), owl (for the rotating head) and frog (for the tongue). We distributed tasks among ourselves in order to discover a bit more about them and to find out something about the sensors and actuators we would like to use.

Meeting 23/03/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Matteo, Juliana, Jalal

- We discussed the results of the first week of research.
- We confronted ourselves on the results obtained by each one of us.
- We decided to change the animals we were taking inspiration from, switching frog with chameleon and weasel with squid, mainly out of how interesting their natural behaviour seemed to be.

Meeting 24/03/2020

Time: 14:00-18:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We discussed changes that we should make to our project given the new information of how the target is going to be.
- We came to the conclusion that we needed a backup plan in case we wouldn't be able to build what we thought of. We then realized we needed some prototypes to see the feasibility of all our ideas.
- Therefore, we decided that in the next week we would build a prototype for the tongue and for autonomous driving. Based on the results, we will modify or not our project. In the meantime, our designers will take care of drawing more precise sketches of what the final robot will look like and look into the materials to be used, while Matteo will be defining what our strategies will be like depending on the combination chosen in the end.

Meeting 25/03/2020

Time: 16:00-18:00
Venue: Skype
Attendees: **Stefano**, Jalal

- As decided in the previous meeting, we had to check the feasibility of the mechanisms inside our project. This meeting was focused on the tongue.
- Based on the design by Jalal, we built two prototypes (each in his own home) with the materials we could find. The experience proved that the idea of the tongue works, but not as effectively as it would be needed to hit a target on an enemy's back.

Meeting 28/03/2020

Time: 11:00-12:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We discussed the results of the prototyping phase, as well as the strategies elaborated by Matteo and the sketches provided by Sofia and Juliana.
- Based on our works and results, we agreed on the final strategy we will be using, which will be a combination of three animals: heron (long neck to attack), echidna (to protect the target with rotating spines) and chameleon (for the tongue, which will be used with the different purpose of drinking water from a safe distance from the pond). We also decided what the size of the robot is gonna look like, as well as its generic shape.

Meeting 30/03/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We discussed our cardboard models, together with our findings on shape and size.
- Based on our results, we agreed on a first draft regarding dimensions of the robot, as well as the probable materials we're gonna use. We then proceeded to prepare a small presentation, summarizing what we did in the past week.

Meeting 31/03/2020

Time: 11:30-13:00/15:00-18:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- Guided by the professors, we started four new tasks: compiling a bill of materials, drawing a flow chart and state diagram, creating a 3d animation of the robot's movements and designing a 3d model of the entire system.
- We couldn't complete all the tasks in the limited time available in class, but we were able to realize a first draft of chart and diagram, as well as a first version of the bill of

materials. Jalal managed to create the first few animations, while Sofia and Juliana started working on the 3d model of the robot.

Meeting 02/04/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- This was just a quick briefing to discuss the tasks each one had to perform during the week, as well as to clarify doubts and questions. No particular issue arose.

Meeting 06/04/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- This meeting was mainly planned to realize a presentation for the 7/4 class, so we briefly discussed our results and then worked on it.

Meeting 07/04/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- Materials, orders, websites

Meeting 07/04/2020

Time: 11:30-12:30
Venue: Microsoft Teams
Attendees: **Juliana**, Sofia

- Discussion around the shape and possible solutions for the spikes

Meeting 09/04/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- Discussion on tasks for this day: websites reliability, emotions

Meeting 14/04/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- Checked how work was going: 3d model, materials and where to order them, bought some sensors and motors, organization next week's work

Meeting 16/04/2020

Time: 18:00-18:30
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- Another brief meeting just to make sure everything was going smooth. No particular problem arose, so we closed it sooner than usual.

Meeting 19/04/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Matteo

- This meeting was made just to be sure that our works on the flow charts was going well. The both of us were working on different subtasks in parallel, so we had to build a final flow chart in order to join everything together. We also used this time to buy the first battery.

Meeting 20/04/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- As on the other mondays, we used this meeting to discuss the goals each of us achieved and then we went on to create a presentation for the following class.

Meeting 21/04/2020

Time: 9:30-18:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- During this class we discussed a few minor issues together with the professors, like the use of the color red for the eyes.
- After this meeting, we decided to buy new matrices that could switch colors, instead of only their shape. We also used this time to order the LiPo batteries and to start testing some simple circuits with the Raspberry Pi.

Meeting 23/04/2020

Time: 17:00-18:00
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We made some adjustments to flowcharts and state diagram.

Meeting 24/04/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We used this meeting to ask some shops about their budgets for laser cutting and to discuss some minor details about the design (specifically the exact position of each electronics component).
- Before leaving, we defined each other's task for the two weeks ahead.

Meeting 27/04/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We had a short meeting just to see how everybody's work was going. No particular issues arose.

Meeting 01/05/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We used this meeting to choose the final emotions design (which we updated after buying the new LED matrices) and to organize the next tests for the robot (in particular there was a need to test if the motors could move neck and target as we wanted to).

Meeting 04/05/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- As every Monday, we discussed our work and prepared a presentation for the lesson of the following day.

Meeting 05/05/2020

Time: 9:30-12:00
Venue: Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We discussed with the professors our results from the previous weeks and we took the opportunity to work on some of our doubts - like the use of a voltage divider for the ultrasonic sensors or the excessive distance between the wheels.
- On top of that, we cleaned and updated the report to make it look better.

Meeting 08/05/2020

Time: 17:00-18:00
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We coordinate our efforts to integrate the software we developed, trying to merge each part coherently in order to have a general idea of the state of development.

Meeting 12/05/2020

Time: 18:00-19:00
Venue: Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**, Juliana

- We discussed technical details of the structure of the robot, taking into consideration adding an external on/off button and adding some extra battery holders.

Meeting 18/05/2020

Time: 17:00-18:30
Venue: Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**, Juliana, Jalal

- We met in order to prepare the presentation for the class on the next day.
- Sofia bought a 3D-printer and we decided to use it to produce each single part of our robot.

Meeting 23/05/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**, Juliana, Jalal

- We discussed on the target design and made some important decisions about the structure of the robot, slightly reducing it in length.

Meeting 26/05/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**, Juliana

- We met to define the last details on design aspects and came up with some ideas for the video presentation.

Meeting 27/05/2020

Time: 11:30-13:30

Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We met to run some tests with a specific version of the code that we will use to make the video presentation.

Meeting 31/05/2020

Time: 17:00-18:00
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We kept working together to the software for the presentation.

Meeting 31/05/2020

Time: 18:00-19:30
Venue: Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**

- We added some extra battery holders due to a delay on the delivery of lithium batteries, which we may replace with AA ones.
- We took some final decisions about the inner structure of the robot in order to 3D-print it and be ready to assemble everything together on next days.

Meeting 02/06/2020

Time: 16:00-22:30
Venue: Juliana's house + Microsoft Teams
Attendees: **Stefano**, Sofia, Matteo, Juliana, Jalal

- We fixed the code and prepared a simpler version for testing.
- Finally, we met to bring all pieces together and build the actual robot.

Meeting 03/06/2020

Time: 16:00-23:30
Venue: Juliana's house + Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**, Juliana, Jalal

- We met to continue building the robot.

Meeting 04/06/2020

Time: 19:00-19:30
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We carried out some tests for each component after the wiring of each piece to be sure that everything worked correctly.

Meeting 05/06/2020

Time: 19:00-20:00
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We carried out more tests to be sure that the wiring was done correctly.

Meeting 06/06/2020

Time: 16:30-18:00
Venue: Telegram
Attendees: Stefano, **Matteo**

- We met to run more tests on the robot, working on its movements.

Meeting 07/06/2020

Time: 15:00-20:00
Venue: Juliana's house + Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**, Juliana, Jalal

- We met in order to continue assembling the robot.

Meeting 08/06/2020

Time: 16:30-18:00
Venue: Telegram
Attendees: Stefano, **Matteo**

- We met to run more tests on the robot and solve mechanical issues related to the neck movements.

Meeting 09/06/2020

Time: 12:00-13:00
Venue: Telegram
Attendees: Stefano, **Matteo**

- We met to solve some target related issues and decided to add a speaker inside the shell of the robot in order to play sounds.

Meeting 10/06/2020

Time: 15:00-20:30
Venue: Juliana's house + Microsoft Teams
Attendees: Stefano, Sofia, **Matteo**, Juliana, Jalal

- We met to finish building the robot. We printed and assembled the last parts that were missing and review some software details.

Meeting 12/06/2020

Time: 17:30-19:30
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We tested the software of the Herokidna

Meeting 13/06/2020

Time: 15:30-18:30
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We tested the software of the Herokidna

Meeting 15/06/2020

Time: 19:00-21:00
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We tested the software of the Herokidna

Meeting 17/06/2020

Time: 15:00-19:00
Venue: Telegram
Attendees: Stefano, **Matteo**

- We tested the software of the Herokidna

Meeting 17/06/2020

Time: 15:00-21:00
Venue: Presencial meeting at Juliana's house
Attendees: Juliana, Sofia

- We developed the storyboard of the video.

Meeting 27/06/2020

Time: 15:00-19:00
Venue: Politecnico Leonardo
Attendees: **Stefano**, Matteo, Juliana, Jalal

- We took some shots of the robot for the video and the presentation and we took the opportunity to make a few final tests

Meeting 29/06/2020

Time: 15:00-22:00
Venue: Microsoft Teams
Attendees: Stefano, **Matteo**

- We faced hardware problems due to some broken electrical connections: we substitute some parts and test everything again.

Bibliography

Here we only placed some of the most useful tutorials we found, though we used so many of them that it was hard to keep track of all of them.

- "*Learn robotics with Raspberry PI_ build and code*" - Timmons, Brown
- https://www.progettiarduino.com/63-arduino-bluetooth-rc-car-4w.html
- https://www.raspberrypi.org/documentation/usage/gpio/
- https://tutorials-raspberrypi.com/raspberry-pi-ultrasonic-sensor-hc-sr04/
- https://howtomechatronics.com/tutorials/arduino/arduino-color-sensing-tutorial-tcs230-tcs3200-color-sensor/
- https://www.pyimagesearch.com/2018/09/26/install-opencv-4-on-your-raspberry-pi/
- https://www.youtube.com/watch?v=gGqVNuYol6o&t=310s
- https://maker.pro/raspberry-pi/projects/how-to-create-a-telegram-bot-with-a-raspberry-pi

Attachments

To Deliver
Working Robot
Printed report:

- Moodboard
- Interaction Flowchart (storyboard or state diagram if needed)
- Circuit
- Bill of materials with providers link and cost, eventually data sheets
- User instruction (1 page to explain how to:
    - power supply
    - switch ON
    - provide Music
    - Interact with.
    - Maintain (change battery, solve problems, access to boards)

Files:

- Report (on-line)
- Firmwares and libraries
- 2D or 3D models
- Short video

To fill

- Self evaluation form (same to the enrollment)
- Peer evaluation