

This report is about the prototype of an educational and informative interactive application. I am a huge fan of space exploration, and I believe it is important to make people aware of how important exploring the universe is. This thought motivated me to create this installation that I called Space Mission. It should describe a story of humankind and their exploration of surroundings. It all started on planet Earth, our home. This part should emphasize it is crucial to take care of our planet. The next stop was the Moon, where the human made the first steps in 1969. It should familiarize the user with this event and give more information about the Moon. Human's next mission is Mars. This part would introduce the user to the current plans for landing on Mars.

This is mainly for educational and informative purpose, but it also encourages people to have fun while learning. This prototype focuses on functionality and interaction; therefore, potential education texts are missing.

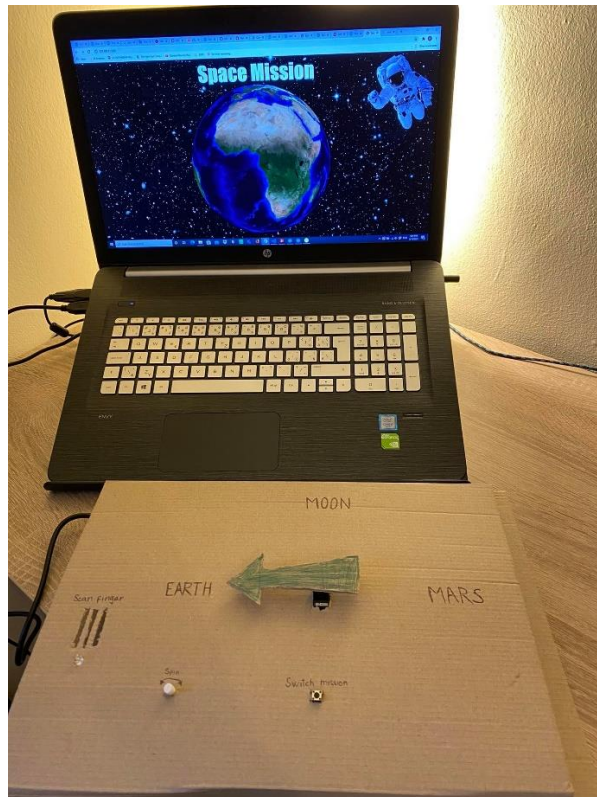


Figure 1: Prototype installation

Interaction Overview

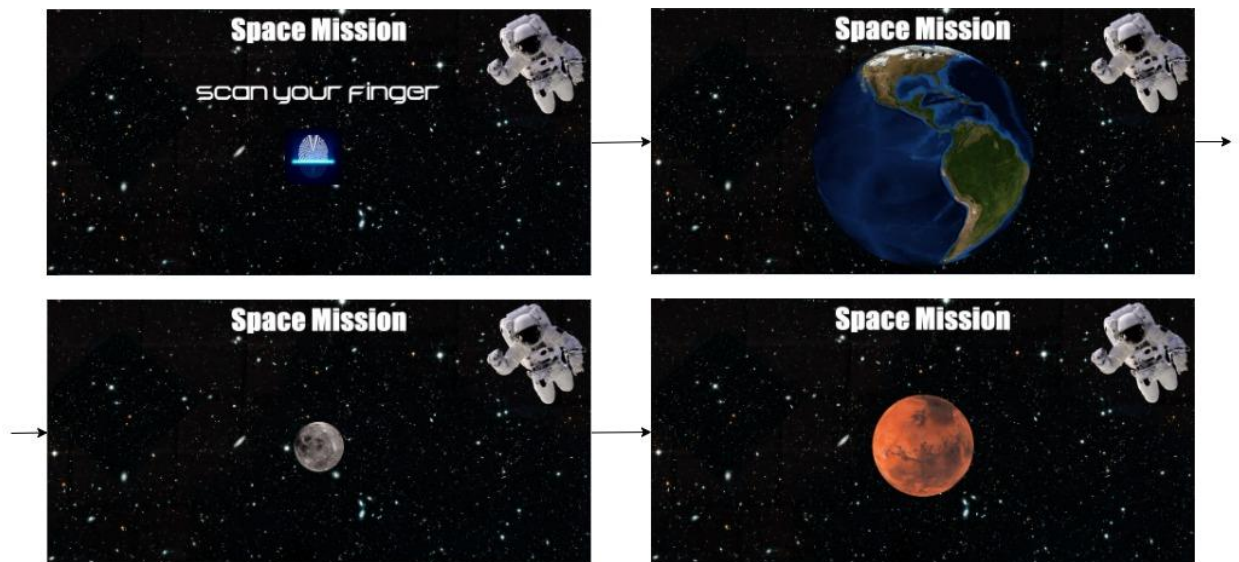


Figure 2: GUI interaction

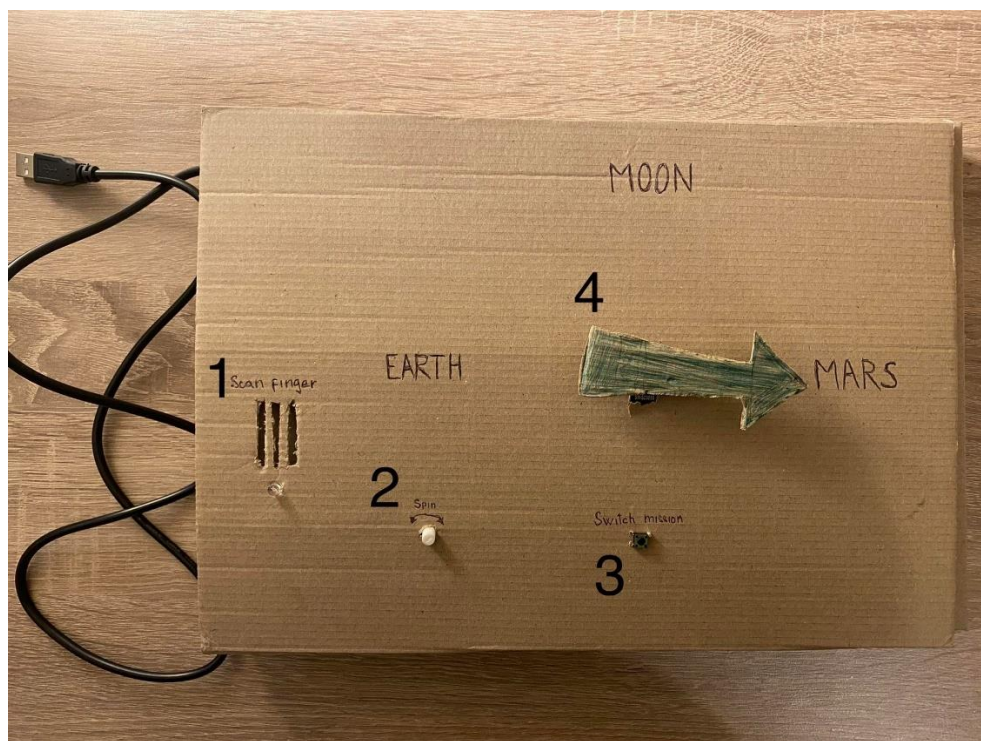


Figure 3: Prototype of the control panel

The whole interaction with a graphical user interface (GUI) is done via the control panel prototype (figure 3). The steps of the GUI are in figure 2.

It starts with the GUI asking a user to scan his/her fingerprint into the system (number 1). After scanning the finger, there appears a first space object which is Earth. The pointing arrow in the picture above (number 4) will be pointing on the Earth sign. The user can then interact with the space object

with the spin button (number 2) and control the object's spin. Number 3 is a push button. This button switches to another space object. The pointing arrow always points towards the current object on the screen. To end the relation, the user scans the finger again. The fingerprint scanner is only simulated. All of the space objects are displayed in their actual proportions.

Parts List

Hardware

- Arduino UNO R3
- Phototransistor
- Potentiometer
- Servo motor
- Blue LED
- Push-button
- Breadboard
- Two resistors
- HP computer screen
- USB Cable
- Jumper wires

Software

- VSCODE – live server
- Arduino editor
- P5.serialcontrol
- P5.js library and p5serialport.js library
- <https://makecode.microbit.org/#editor>

Other

- Carton box for creating the control panel prototype and the pointing arrow

Process

As mentioned, the user interacts with the GUI via the control panel. GUI works on the local server run from vscode. It uses P5.js library, HTML file and CSS styling to display graphics. The P5 sketch is written with the javascript programming language. The sketch communicates with Arduino using p5serialport that goes through the USB cable. First, I began by creating the GUI and started to understand how it all connects together. Then I started connecting electrical components to the Arduino and test their responsiveness in the GUI one by one.

The control panel has four main functions that could be seen in figure 3. Number one (scanning the finger) is used as a simulation of fingerprint scanning. It uses a phototransistor which, when it is blinded, calls the function that causes blinking of the LED hidden in the control panel. It also sends the information to the p5 that the finger was scanned, which causes displaying the first space object.

Another control button is a spin button in the form of a potentiometer that sends the data about its spin to the GUI that reacts by spinning the space objects.

When the press button is pressed, it is sent to p5, which reacts by switching the space objects. The press button function was equipped with one more variable called click to avoid continuous sending of the data. The use of the click variable is in appendix 1.

The sending data from Arduino to p5 via serial is distinguished by the ending sign “!”. When the string is received in p5 it reads strings until the “!” sign.

When the space object is changed in the sketch, it sends the information back to Arduino. The Arduino then reacts by positioning the servo motor to the appropriate angle to point with the arrow on the displayed space object. This communication is done by sending a numeric value that represents a particular space object.

The wiring of all components can be seen in appendix 2.

References

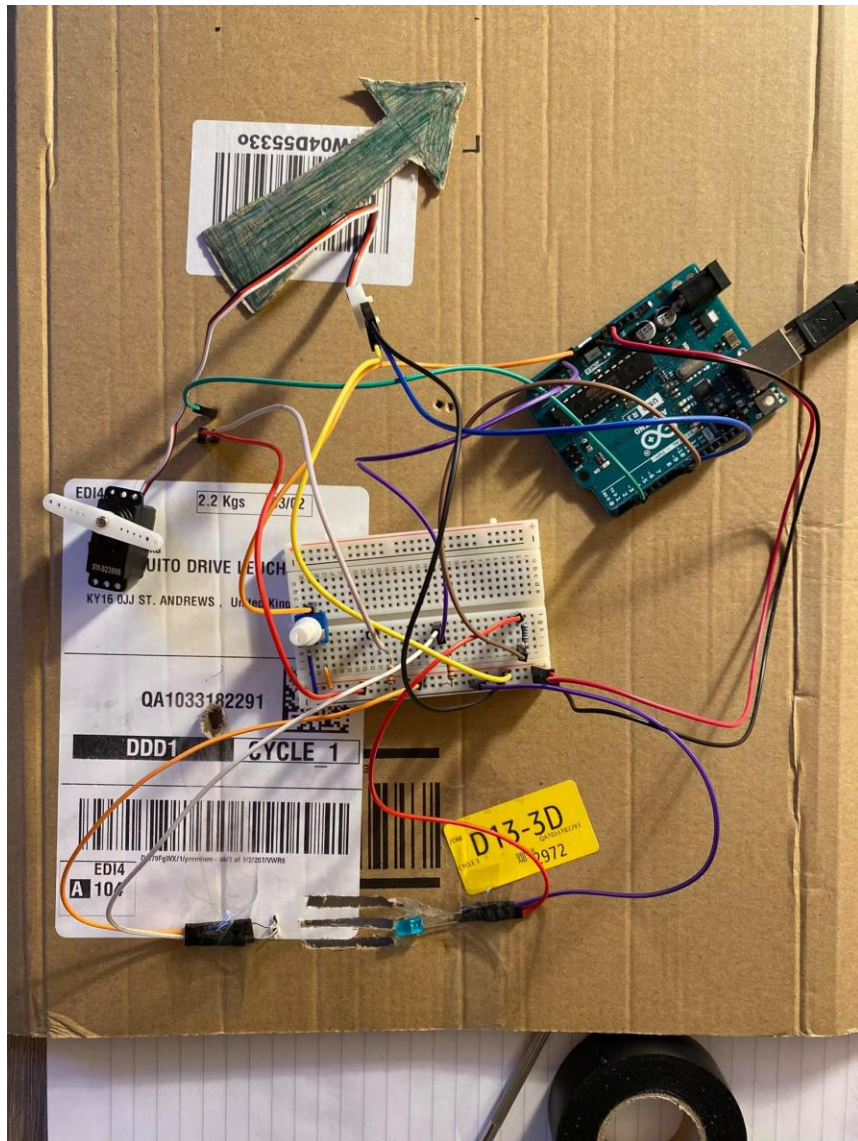
<https://www.arduino.cc>

<https://p5js.org/>

Appendix

```
// Button
buttonState = digitalRead(buttonPin);
if (buttonState == HIGH) {
  click = 1;
} else {
  if (click == 1) {
    Serial.write("#!");
    delay(10);
    click = 0;
  }
}
```

Appendix 1 – Use of click variable.



Appendix 2 – Arduino wiring.