# **1 Project Introduction**

Our team decided to create a remote-controlled hazard detector using a variety of sensors and technologies. We used an Arduino MKR Wi-Fi 1010 board to connect all our sensors to which streamed the data collected to a WebSocket server in Python. A Zumo 32U4 and its libraries and XBee modules to be able to control the Zumo remotely. We used a few different sensors, a light sensor to measure the brightness in a room, Temperature sensor used to measure how hot the room is and a Geiger counter which measures the amount of radiation. We used a smart phone camera which we streamed over a locally hosted server using IP Camera App. And finally we used python to create a Graphical-User Interface using Tkinter, OpenCV to display the camera feed, PySerial to allow controlling of the Zumo and threading to run it all concurrently.

We brought all this together to create a robot that could go into a room and detect different potential hazards and report back the readings. My main role was getting the Zumo controls working and getting it to be used over PySerial so we could control it from the GUI.

# **2. Project Aims**

The project aims were as followed:

* Remote control of the Zumo via XBee using Python Interface
* Remote control of the sensors (Geiger, Light and Temperature) via an MKR 1010 Arduino WIFI board.
* Remote streaming of a camera module that will give a wireless display of what the robot see, allowing us to control it from distance.
* A Python graphical interface that allows for control of the XBee, display of a video camera feed and display of streamed data from the Arduino MKR.
* A rig for all the sensors and a way to move them around using a Zumo.

# **3. Project Management**

We kept track of progress by using a Kanban agile approach. We used a Trello board which allowed us to write task cards and assign them to different project members. As a team we also used a scrum approach where we would assign a number of tasks a week to work on and would have meetings each day to discuss progress and about that day’s tasks. This meant all team members were always up to date with the progress of the project.

# **4. Project Development**

The main part of the project was the different sensors used to detect certain hazards. As mentioned above we used a Geiger counter, light sensor and heat/temperature sensor. To allow us to collect the data from these sensors, we used an Arduino MKR Wi-fi 1010 to connect them to which would allow us to send the data wirelessly to the computer.

Getting the Geiger counter to work was a fairly straight forward task, we connected it to the digital 4 pin on the Arduino MKR board and we used a program that counts the number of times radiation collides with the gas within the glass tube.

The next sensor we used was a Light dependant resistor which changes it’s voltage output based on the amount of light it receives from the area around it. The output of this was sent through the analogue pin of the Arduino board. The sensor we used was a digital temperature sensor, this was changed from our initial plan as we wanted to use an analogue sensor but the Arduino MKR 1010 created a lot of problems with interference with the readings which led to big variance in the temperature readings we received.

The data collected by the sensors were sent over the Arduino board to a WebSocket server that then took the data and sent it to the GUI which displayed the data in real time. The GUI also displayed the highest temperature reading that it took in and the highest counts per minute from the Geiger counter and showed them in different colours to indicate safe, warning and danger levels.

Another key aspect of the project was the camera used to show a real time feed of what the robot could see in front of it. We initially wanted to use a OV7670 Arduino camera module which would’ve connected to our Arduino like the sensors however, this camera did not arrive in time for us to use it so we had to go in a different direction. We ended up deciding on using an old smart phone and downloading an app called IP Camera, this allowed us to connect the camera to a computer over a server which allowed us to receive the camera feed. We were then able to use this feed to put into our GUI. The way this worked was each frame of the video feed would continuously update a canvas variable on our Tkinter GUI which made the video feed appear in real time.

A Zumo 32U4 was used as the actual robot that would navigate everything around, we created a program that would allow use to control the Zumo manually from a computer using WASD keys and we used XBee modules to allow this to be done wirelessly to allow the Zumo to move around freely. We also had to create a rig that the Zumo pushed around which would carry all the sensors and camera. We made this rig out of Lego and it housed a battery pack which powered everything, the Arduino board with a breadboard that had some of the sensors on it, the camera which was on the front and the Geiger counter which was on the underside. As this project was a prototype we used rubber bands to secure some parts in place like the camera and the Geiger counter however in a real build of this device parts would be screwed into place.

The last but probably biggest part of this project was the Python GUI. It was created using the Tkinter library along with other modules like OpenCV and PySerial to allow us to move the Zumo through the GUI and show the camera feed as well. We also used threading which allowed everything to happen concurrently.

The GUI was split into different sections. The first was a thread that allow us to control the Zumo. We used PySerial to connect to the serial port 9600 which allowed the control inputs to be received by the Zumo and move appropriately. We used WASD keys to move forwards, backwards, left and right.

The next section was a thread that allowed us to host a WebSocket localhost server. This received the data from the sensors every second which the Python GUI then updated the values for them and displayed them. Finally we had a class that displays the camera feed with the sensor data overlayed on the top.

# **5. Contributions to the project**

I was active in the initial brainstorming meetings to decide on what we should do for the project as we were struggling to make a project that would have enough aspects for us all to work on. When one of the team members suggested using a Geiger counter and the Zumo robot I suggested using other sensors like a temperature sensor and the potential use of a camera as well.

My main focus was around the Zumo and being able to control it using a computer but also have it so it can be controlled using the GUI. One of the other team members was looking into aspects of the GUI and was looking into using threading so I used that to figure out how to get the Zumo moving through the WASD key inputs in the GUI.

I was also heavily involved in creating the rig that housed everything and making the different aspects become one big device instead of the several pieces we had. The rig was somewhat difficult to design as it needed to be big enough to store everything we need but also be something the Zumo could either drag or push around. I ended up making it something that the Zumo had to push so I had to create a part of it that the Zumo could fit into. One thing I didn’t really test until towards the end was how heavy the whole rig would be with all the devices on it which meant the Zumo slightly struggled turning it unless it was on carpet.

I also tried my best to keep up to date with progress being made with the other parts of the project and tried helping out where I could if it was researching something or trying to help debug problems. Because of other commitments that I had I sometimes wasn’t able to meet up with the group to work which reduce my involvement in some areas however I always tried to catch up with what went on.

# **6. Conclusion**

In conclusion the project was pretty successful. However, if we had more time or did this in the future we would definitely create a more robust rig to house everything as parts of the rig looked very unstable. This would also potentially help with the struggles we had with the Zumo as it wasn’t gripping to well to smooth surfaces and worked better on carpet where there was more friction.

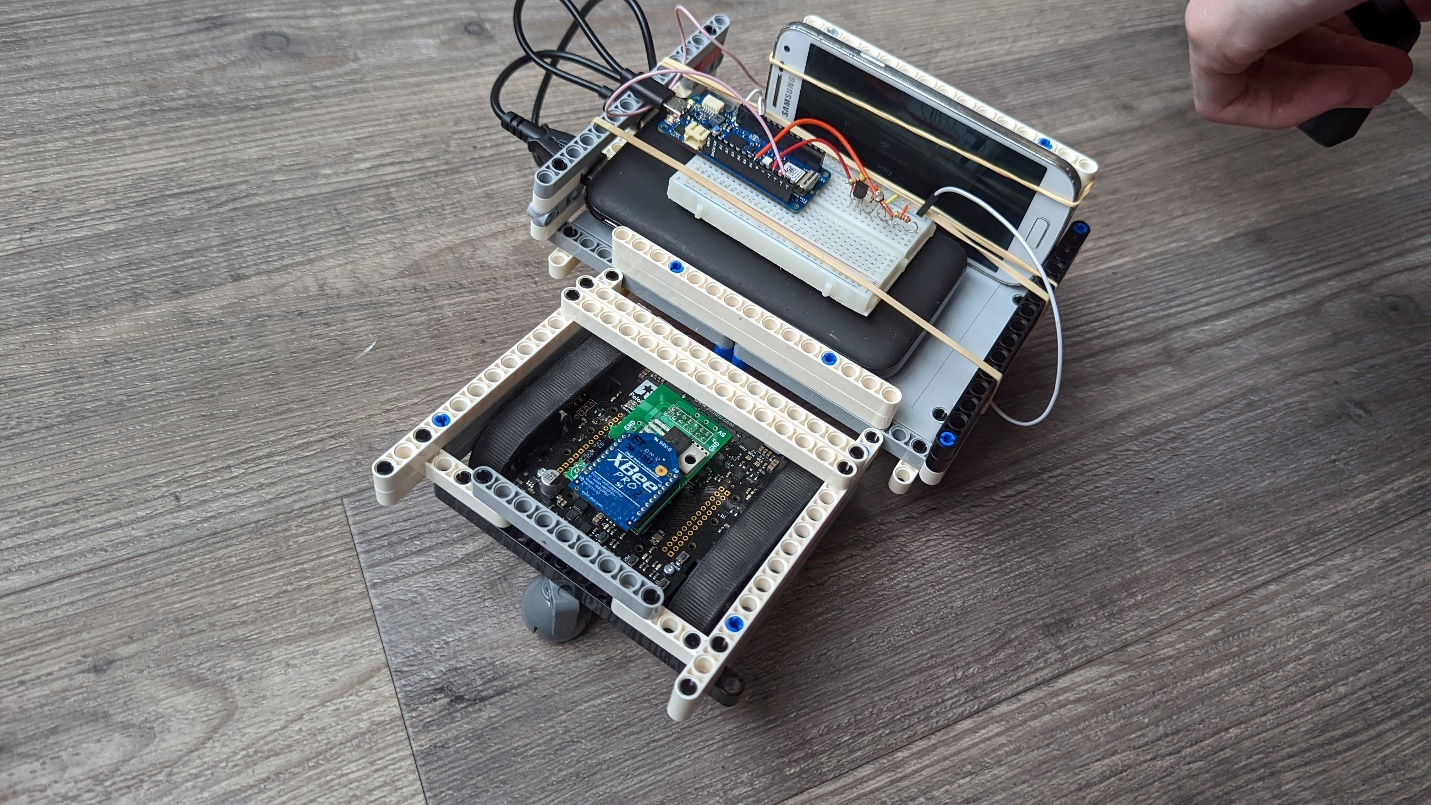
I think using the original Arduino camera module would also be an improvement we’d make as it would firstly make the whole thing lighter as we wouldn’t need the phone to act as a camera and would also mean we wouldn’t need the IP Camera server as the data from the camera would be able to transmit over the Arduino MKR board like the sensors.

As a group we worked well together for the most part. Using a Kanban style approach worked very well and would be how we’d conduct this project again in the future. There was a few times where sickness in the group caused delays and prevented meeting up which if we had considered as a possible constraint earlier we could’ve changed our approach slightly to allow more time to do the work which would’ve allowed for some areas of the project to be improved and would’ve reduce the rush we had towards the end of the project to get everything done. There was a couple instances where a lack of communication meant that some team members met up to work on the tasks without the others so some members weren’t always as involved as others.

I feel like I had a slightly smaller role than others in the project and so in the future I would put myself in a position to take on more tasks and try to be more proactive within the group.

# **7. Appendix**

### Appendix A



### Appendix B

