Heriot-Watt University

School of Engineering and Physical Sciences

MSc report Cover page

MSc Program Title: Robotics

Project Title: Cloud Based Robotics Platform for Outreach Activities

Course Code: B31VZ

Author: Godfrey Inyama

Student full registered name: Godfrey Ijeomah Inyama

Student HWU Matriculation Number: H00314926

Supervisor's Name: Dr. Mauro Dragone

Signature: 

Date: 19/08/2022

Declaration of authorship – Heriot Watt University

Course code and name:	B31VZ – MSc Project
Type of assessment:	**Individual** *(delete as appropriate)*
Coursework Title:	MSc Project
Student Name:	Godfrey Ijeomah Inyama
Student ID Number:	H00314926

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment is entirely my own. I have NOT taken the ideas, writings, or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings, or inventions of others, or any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgment section.

- I confirm that I have read, understood, and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood, and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature *(type your name):* *GODFREY IJEOMAH INYAMA*

Date: *19/08/2022*

Copy this page and insert it into your coursework file in front of your title page. For group assessment, each group member must sign a separate form and all forms must be included with the group submission.

Your work may not be marked if a signed copy of this form is not included with your submission.

Abstract

Robotics and Artificial Intelligence are improving human productivity which are seen in the areas of health, education, agriculture, manufacturing, aerospace, and others. In recent times, the demand for Robotics Technology have tremendously increased with a significant pull in the Robotics Education market data which is influenced by the demands of the Public to acquire Robotics Education [97]. In a demanding environment like ours, we cannot over emphasise the benefits of utilising various Robotics Technology to help us save time, achieve maximum efficiency, reduce risks, increase performance, and work with great precision.

Robotics Education helps us to understand how to build, program, and manage Robots to maximise the full benefits gotten from using them. To some, robotics education is overly complex as it involves lots of programming, while to others, robotics education is fun due to either a passion for technology or a stern interest in the field. To make robotics education easy for all, there are a number of innovative solutions which have been implemented. These solutions range from platforms which can independently walk a first-time user from bottom to top like, "The Construct" or platforms which can help children and students learn robotics from visual programming like "MiRO Cloud." Since the late seventy's and eighty's there have been a quest to make robotics education extremely easy and accessible to all [86] [87].

An improvement into the way Robotics Education is offered, is the use of cloud-based technologies. Cloud based technologies offers an easy method for software resources to be transferred and accessed by multiple users at same time from various locations. One key role the cloud plays, is the ability to offer a parallel communication path for multiple users at once without waiting for a communication process to end. Various Educational Robotics utilising cloud technologies are creating a fast and seamless method for users not only to access resources over time but also to understand robotics better by the actions the robots perform after data transfer must have occurred in the cloud. This concept had led to the development of Internet of Robotics Things (IoRT).

Considering this, this project successfully produces a cloud-based extensible web application for school children, offering a visual programming interface and providing access to both the simulated and real robot.

Acknowledgement

Completing this Dissertation would not have been possible without the Grace and Wisdom God bestowed on me. In this journey to successfully completing my Dissertation, I specially want to thank my supervisor Dr Mauro Dragone for his guidance and mentorship all through the phases of this journey. I also want to thank all the Lecturers and Teaching Staff who have in one or two ways being a guide and impacted into me diverse levels of knowledge through my master's degree program.

In an incredibly peculiar way, I want to acknowledge my lovely wife Mmesoma Inyama for her kind support and the sleepless nights she spent with me to encourage me to keep going on even when I was almost giving up. I also want to appreciate Edinburgh Elim Church and Smplicare on how they offered their best support to give me the time to wrap up this dissertation.

Finally, I want to earnestly thank my families and friends who have been there for me offering me various levels of financial and technical support through this journey.

Contents

List of Figures

List of Tables

Abbreviations

AI	**A**rtificial **I**ntelligence
ROS	**R**obotics **O**perating **S**ystem
API	**A**pplication **P**rogramming **I**nterface
STEM	**S**cience **T**echnology **E**ngineering and **M**athematics
STEAM	**S**cience **T**echnology **E**ngineering **A**rt and **M**athematics
K-12	**K**indergarten to 12th **G**rade
SLAM	**S**imultaneous **L**ocalisation **a**nd **M**apping
RSNP	**R**obot **S**ervice **N**etwork **P**rotocol
RTT	**R**ound **T**rip **T**ime
ER4STEM	**E**ducation **R**obotics **F**or **S**TEM
NEPO	**N**ew **E**asy **P**rogramming **O**nline
DaVinci	**D**istributed **A**gents **w**ith **C**ollective **I**ntelligence
JSON	**J**ava**S**cript **O**bject **N**otation
XML	**E**xtensible **M**arkup **L**anguage
JAXB	**J**ava **A**rchitecture for **X**ML **B**inding
AST	**A**bstract **S**yntax **T**ree
SQL	**S**equential **Q**uery **L**anguage
JDK	**J**ava **D**evelopment **K**it
SSH	**S**ecure **S**hell
CPU	**C**entral **P**rocessing **U**nit
IP	**I**nternet **P**rotocol
PRISMA-P	**P**referred **R**eporting **I**tems for **S**ystematic **R**eview and **M**eta **A**nalysis **P**rotocol

1. Introduction

Robots have become increasingly common in our daily lives as computer technology and AI are rapidly advancing. The saturation of Human Robot Interactions in our day-to-day activities have significantly increased the demand of Robotics to meet certain demands and solve Human Problems ranging from Households to Industries. We have slowly come to terms with the fact that Robots will become an integral part of our lives in creating ease and helping us solve complex problems. At the moment, some of these robots which help us in our day-to-day activities are; Pepper and Toyota which are used in Homes and Health care facilities to take care of ageing groups, Paro and Kismet which are both robots designed to elicit emotional response from humans, Amazon Scout for package deliveries, and Ozobot for Educational purpose [101]. With these demands of robots in our everyday activities, the demand to upskill in this field has also increased tremendously.

Based on Global Educational Robotics Market data, the Compound Annual Growth Rate of Robotics sales is set to increase by 16.4% between 2018 and 2026 with 2018 at $784.89 million and forecast to 2026 at $2,542.43 [97]. From this Data, we can see how much investments have already gone into Robotics Education and how much it would increase in just 8 years. With a foresight from this data, it is imminent upon us that learning Robotics would indeed be an exceptionally worthwhile investment for oneself now and in the future. At the moment there are some Robotics applications and Companies which are drastically improving the ways Robotics is learnt, some of which are; the construct which is the top online robotics academy for learning ROS programming, MiRO Cloud which is a platform for delivering both virtual and facial expression training in simulated environments across the world, and Open Roberta which is a German initiative for helping School Children learn Robotics by its mass library of Robots and other electronic systems like the Arduino and the microbot.

These platforms and others are currently working with modern trends in the technology field with cloud computing, where resource sharing and data communication is done in the Cloud. This concept introduces "*The Internet of Robotics Things*." To further simplify and encourage students and the public to get Robotics education, strategies are smartly implemented every day from numerous studies and research. One of which is the introduction of Visual Programming of Robots. This is achieved by either using

blockly blocks or scratch blocks libraries. We live in a time where a vast majority of Students avoid Robotics Education because of the complexities in programming and coding. But the inclusion of a visual interface where students and other users can effectively create actions for the Robots from blocks arrangements have shown a vast improvement in the attitude of Students and the Public towards Robotics Education. With these trends serving as a background, this Project aims at building a cloud-based platform for outreach activities and improving on the data transfer methods in other platforms. The next Chapter analyses more problems in detail.

1.1. Research Problem

A suitable number of studies have been conducted in Educational Robotics and a vast number of cloud-based platforms implemented to aid in Robotics Education. Regardless, there are still several problems which have been seen from these studies and platforms. A number of these problems in which this Research aims to solve are:

- Firewalls and Network Limitations: A typical example of this is seen in the MiRO Cloud. The MiRO Cloud platform offers a bespoke cloud platform to perform certain actions on a custom-built Robot called MiRO. The data transfer from the MiRO Cloud platform to the MiRO Robot is possible via a network protocol. It has been seen that due to static firewalls in internet networks, there is a dominant restriction in accessing the MiRO cloud web application which can cause hassles when MiRO Cloud is used for outreach purposes.
- Singular and Extensible Data Transfer architecture: Many Robotics Platforms today, do not offer flexibilities of extending resources to other Robots using visual programming. Like the MiRO Cloud platform, it is impossible to use its platform to program and simulate other robots other than the MiRO robot. The Construct which may seem slightly flexible, doesn’t include visual programming in its functions. The Open-Source platform critically used throughout this study called Open Roberta seems to have solved this problem, but unfortunately it is limited in its connection model for robots. This problem is caused by the difficulties in building dedicated connectors to suit various robots’ firmware specifications to enable program transfer from the Cloud Platforms to the Robots.

- Comfort in Visual Programming based on experiments and studies: Lots of research have been conducted in evaluating the effectiveness of certain models in Robotics education, but there is still no useful resource and study which explicitly shows how students can effectively progress from Visual Programming into Non-visual programming. Visual programming which aims at creating a seamless entry for young students and others into Robotics can also be detrimental as the early age users can develop a preference for that over programming languages to program the robots at an older age.
- The final problem is seen in a wider robotics ecosystem where both the robot and the cloud platforms must be on the same network before data transfer is possible. This has shown a persistent pain point for users, researchers, and developers which always must do a reconfiguration of robot and platform IP addresses at every instance the network changes.

1.1.1. Research Questions

Based on the Research Problems stated above, these Questions have been developed to be answered in this Research:

- RQ-1: How can we develop a Cloud based Web Application for programming Robots from Open-Source?
- RQ-2: How can a Cloud based Web Application be usable to 10 years old with access to only the browser, and optionally to a real robot as part of a class activity?
- RQ-3: Besides the Robot of consideration, which is the ROSbot 2 Pro, how possible is it to develop an architecture which will be easy to add other Robots with the same data transfer model?
- RQ-4: How can we have a flexible Data Transfer from Platform to Robot without both systems being on the same network?
- RQ-5: How can we best measure and evaluate the effectiveness of this platform from Outreach activities?

1.1.2. Extended Research Questions

On the basis that an Open-Source Platform would be used, and it may have certain static features, further research Questions based on literature reviews and trends in Technology would be:

- ERQ-1: How can we manage the problem of multiple robots requesting for files to execute from the same server and prevent unauthorised users to create programs for the Robot to execute?
- ERQ-2: What are the innovative guidelines to take before choosing Cloud Services for platform deployment with respect to response time, server speed, security, and performance?

1.2. Research Methodology

The Primary aim of this Research is to build a Cloud Based Robotics Platform for outreach activities. The initial robot proposed for this platform was the Toyota, but due to further consideration of size, safety, and availability, it was changed to the ROSbot 2 Pro. The ROSbot 2 Pro, is a modification of the ROSbot 2 robot. *Figure 1.1* below shows a ROSbot 2 Pro Robot, and a back view of its components.

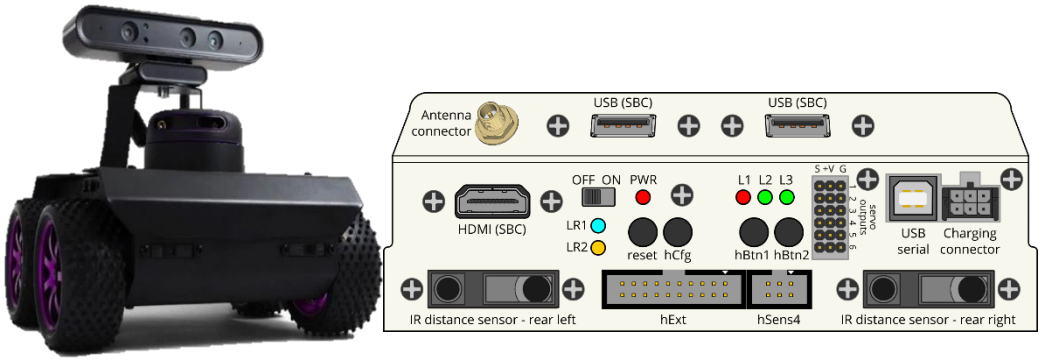


FIGURE 1.1: Front view of a ROSbot 2 Pro and Back view showing its connection ports (inputs and outputs)

Considering the Project Objectives and the Research Questions above, these methods have been followed:

1. Literature Review: The first step which was taken, was to critically analyse majority of research and similar platforms that have been developed. This review was aimed at understanding the various technologies these platforms have used, and their various limitations. This also followed a detailed outline of the similarities, differences, and uniqueness of the platforms.
2. Analysis of Robot Benchmark: The initial Open-Source platform proposed to use was Robot Benchmark. This platform was analysed to understand its components and features. After detailed analysis, I figured out that using this platform would pose lots of challenges with flexibility of adding new features and answering most of the research questions stated above, hence the switch to the Open Roberta Open-Source.
3. Understanding Open Roberta Open-Source: At this stage, I had deployed the open-source on my computer. To understand how best to implement new blocks and change the codes generated by the Plugin chosen, I had to understand how the existing blocks were created and how these blocks generated the codes.
4. Remodifying the Blocks and Code Generation: After a firm understanding of the full open-source, I was able to modify the blocks to suit the actions required and modified the codes generated from the blocks into a Python-ROS code. This was done by building out a standard Python-ROS model which was tested on the robot first to give the desired actions before it was integrated into the specific blocks.
5. Designing the architecture and implementing the new API for data transfer: At this stage, I understood how to solve the problem of placing the robot and the platform on the same network which other systems used via rosbridge. I built out the API based on the structure of the open-source to fetch and POST the generated program to another server, which would originally download into the downloads folder of end users.
6. Test and Debug: I had a ROSbot 2 Pro throughout this process of the development in which I used to test out the built-out API. A security problem was noticed during this test and the architecture was further improved to solve this problem. *Figure 1.1* below shows a simple flowchart of the methodologies.

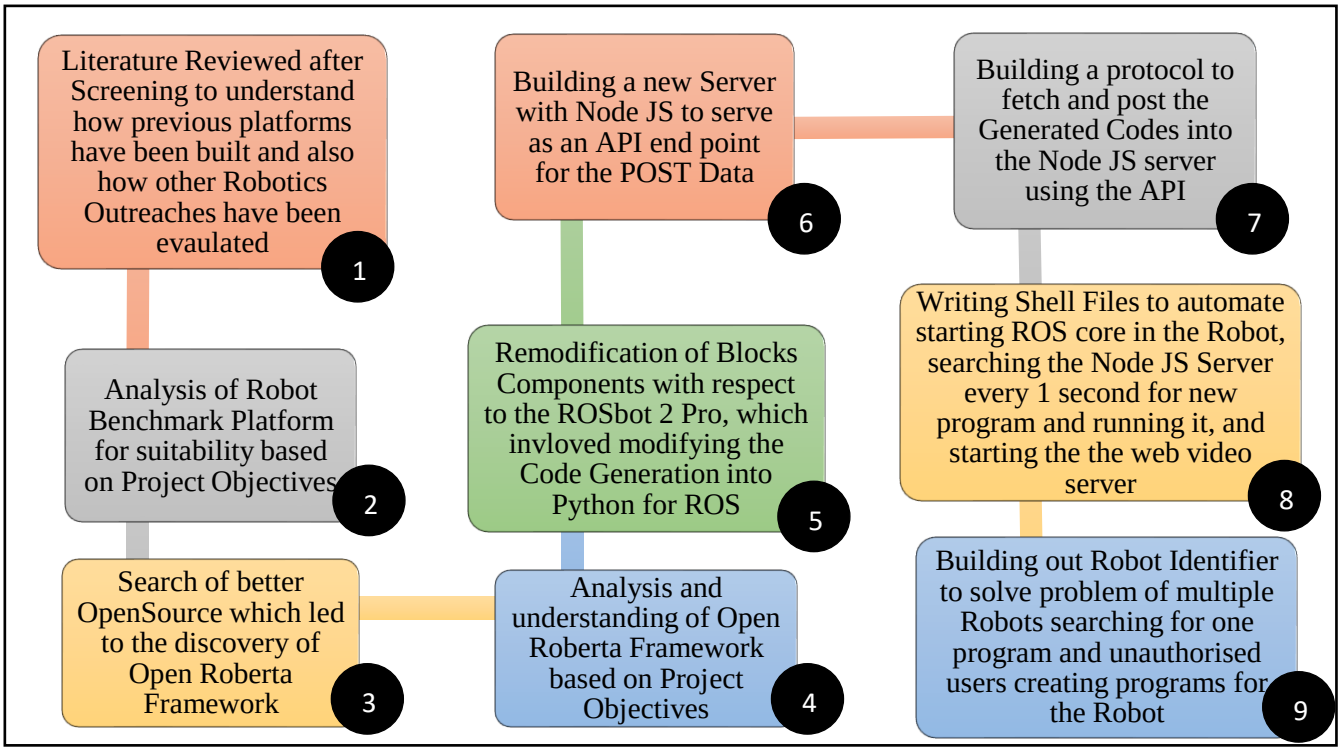


FIGURE 1.2: Flow Chart of methodologies in achieving Project Objective

1.3. Thesis Outline

This study follows this organisational pattern:

- Chapter 2 shows the literatures reviewed and the limitations of the literatures. These literatures were specifically chosen within the scope of cloud and educational robotics.
- Chapter 3 gives an extensive highlight and description of the Open Roberta Open-Source Platform.
- Chapter 4 shows in details how the new ROSbot 2 Pro was included, how the blocks were modified, and how the API was built out.
- Chapter 5 shows the designed experiments and studies to be conducted in the future with the sets of participants, and the results from the Project.
- Chapter 6 gives a general summary of the Project based on answering the research questions, its limitations, and proposed future works.

2. Literature Review

2.1. Introduction

The focus of this chapter is to extensively review previous and recent Research that have been carried out within the scope of Educational Robotics with an additional review of the platforms used for same purpose. On an extensive review aim, specific benefits achieved through the implementation of certain platforms are reviewed.

A vast majority of Research carried out on Educational Robotics aimed at:

- Improving Students interest and participation in Robotics Education.
- Understanding the outcomes on the effectiveness on certain methods used in Robotics Education.
- Reviewing Educational Robotics in both formal and informal environmental setups.
- The roles of Robotics in Educational scenarios.

2.2. Systematic Review Approach

In understanding the trends of Educational Robotics and Cloud based Robotics platforms, this review followed a systematic approach. This approach aims to create a clear understanding on the evolution of Robotics Education from early days to present days. To fulfil this review expectations, these are the outlined goals:

- To study how past and present Educational Platforms have been implemented.
- Review the current ground-breaking platforms based on the highlighted research questions.
- Extract best and useful information to guide the methodologies in implementing the current platform for this Project.

2.2.1. Review Methodology

In following a Systematic approach for this review, the first step was to analyse and synthesize the studies which had been carried out between 2010 and 2021 and select the sets of papers to be reviewed based on the expected outlined goals for this review.

The second method is analysing in depth the selected papers to understand their impacts in Educational Robotics and the platforms used in carrying out their Educational Outreaches.

The final method would be reviewing the outcomes from their research in order to understand how the platforms were developed and how their methods were evaluated with a clear expectation of extracting useful information to guide in my own implementation and evaluation.

2.2.1.1. Study Selection Criteria

With the strategy of searching, reviewing, and analysing existing literatures, the literatures were selected based on two basic criteria. The first criteria were selecting the topics which aligns to the aim of this study. Table 2.1 below shows the criteria based on topic names:

Topic Code	Topic name
TPC1	Cloud Based Robotics Platforms
TPC2	Cloud Communication
TPC3	Educational Robotics
TPC4	Open-Source Robotics Platforms
TPC5	Robotics outreach events
TPC6	Visual Programming

TABLE 2.1: Topics Search criteria

After these topics were selected, the next step was to put them together to form and develop search terms and keywords which would be used to extract the relevant literatures which met the review and study goals.

These are the list of Databases used to search for relevant papers to be reviewed:

- Google Scholar
- Core
- IEEE Xplore
- ScienceDirect
- HWU Discovery

The first two databases were accessed freely as they are opened for public use, while the last three were accessed using my Heriot-Watt University details.

After carrying out some initial search trials based on the first topics covered, these were the final phrases and keywords derived (Table 2.2):

Key words and Search Terms	Topics Covered
Blockly and Scratch Programming	TPC6
Internet of Robotics Things	TPC1, TPC2
Cloud Data communication	TPC1, TPC2
Port Forwarding and SSH	TPC2
Educational Robotics target audience	TPC3, TPC5
Educational Robotics Evaluation	TPC3, TPC5
Open-Source Technology	TPC3, TPC4
ROS Navigation	TPC1, TPC3
Internet Protocol	TPC1, TPC3, TPC4

TABLE 2.2: Combination of different search patterns to generate key words and search terms

After narrowing down to the search terms, it was seen that it was easier to focus on relevant resources in achieving Project Goals.

2.2.1.2. Exclusion Criteria

Papers were strictly excluded from this review and study if:

- They were not written in English.
- They had no clear outline of involvements in Educational Robotics.
- They were too ambiguous and did not offer any guide to developing the desired platform.
- They were implemented on Non-ROS Robots.

2.2.1.3. Study Screening

The study was screened following the PRISMA-P [60] guidelines for systematic review. The first stage was to collate all the initial search sources to be used in order get the first papers from the search key words. These were later screened based on duplicates. After which, the screening was based on relevance to study. The final stage was to get the required study materials which met all requirements before the review commenced. Figure 2.1 below shows a summary of the screening process.

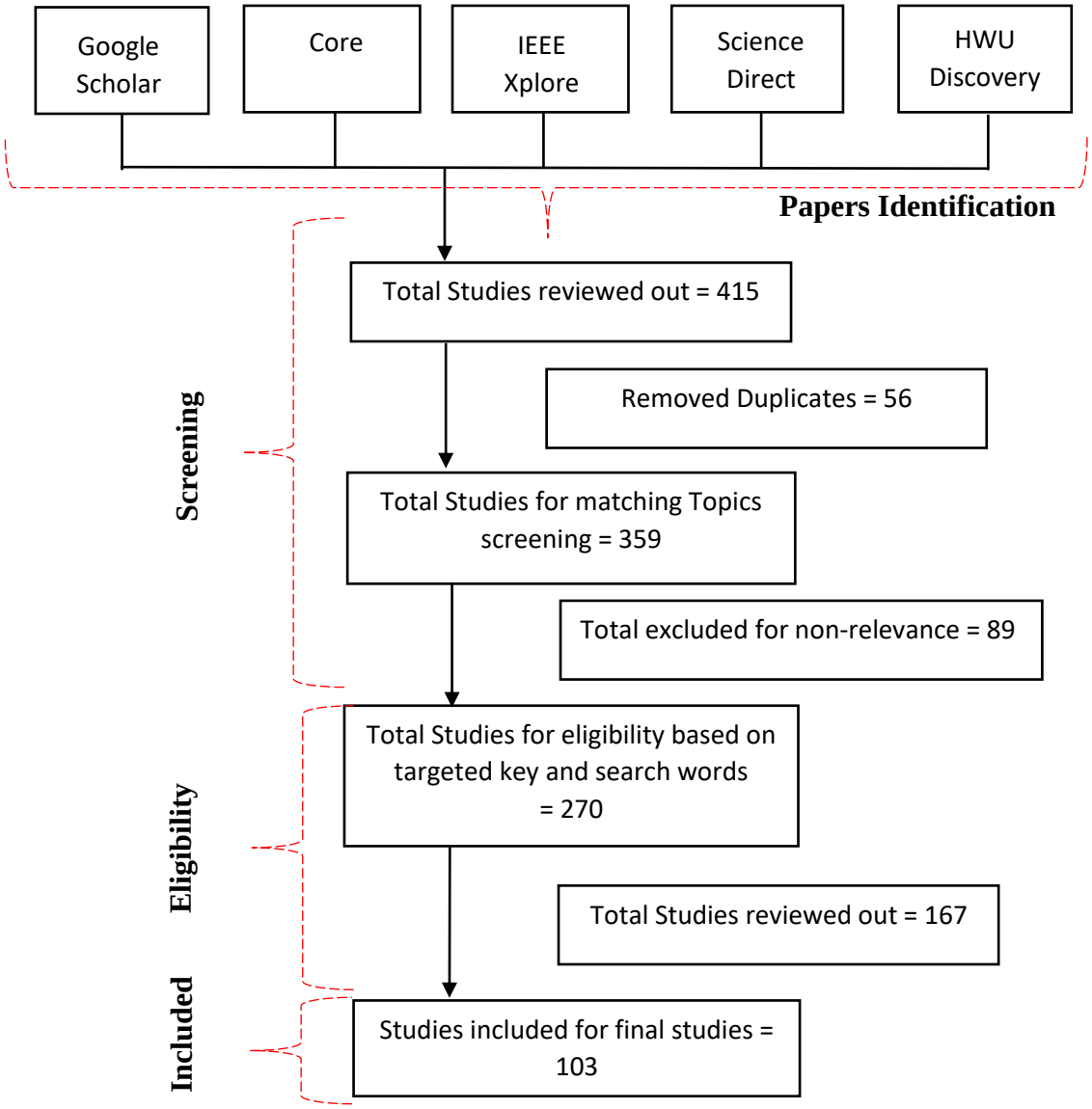


FIGURE 2.1: Screening Process Flowchart based on PRISMA-P guidelines

2.3. Background Review

2.3.1. Educational Robotics

The COVID-19 pandemic has created a disruption in educational processes and knowledge transfer methods due to the limitations of close contacts restrictions and economic breakdowns. This has impacted STEM education by creating difficult access to physical laboratories and limitations in the number of usages. A unanimous approach to solving this problem has been seen in the introduction of cloud-based platforms to assist in knowledge transfer from remote locations. To narrow the scope of STEM education is the effectiveness of these methods in Robotics Education [53]. More so, the introduction of digital resources has been invoked to meet the growing demands of distant and remote learning. Due to the dynamic nature of individual platforms, it will be difficult to conclude the most efficient robotics programming platform resources to use, hence the drive to embark on this Project to bridge all the known gaps.

In Jose Canas work in 2020, some robotics platforms for outreach and educational purposes were listed in Table 2.3 below:

Platforms	Programming Language	Supported Robots	Physical Robots	Simulator	Open Source	ROS	Evaluators
[39]	SysQuake	Three	No	No	Yes	No	No
[95]	MATLAB	Moway	Yes	RFC SIM	No	No	No
[38]	C	e-puck	No	Webots	Yes	No	No
[103]	C & C++	SIR	Remote	Stage	No	Yes	No
[85]	C & C++	KepheraIV	Remote	V-Rep	No	No	No
[27]	Python & MATLAB	Several	No	V-Rep	Yes	No	No
[94]	C++	Several	Yes	Mirage	Yes	No	No
[14]	Python	Several	No	Gazebo	No	Yes	No
[14]	Python	ThymioII, Nao	No	Webots	Yes	No	No

[16]	Python and Blockly	Several	Remote	Stage Webots	No	Yes	No
[14]	Python	Several	Yes	Gazebo	Yes	Yes	Yes

TABLE 2.3: Educational Robotics Platforms and their Resources [13]

Social Robots which are mostly used for Human Robot Interactions is a good consideration for Robotics Education [99]. In 2018, a review of the utility of social robotics in Education was conducted [9]. They cited some benefits of social robotics with regards to teaching agents, as the ability to perform one-one teaching, better interaction with the physical robots as compared to virtual agents and meeting the needs of a certain group needing support such as those with vision impairment, physical challenges, and the elderly with dementia. They conducted a meta-analysis in order analyse the results from introducing Social Robots in their environment.

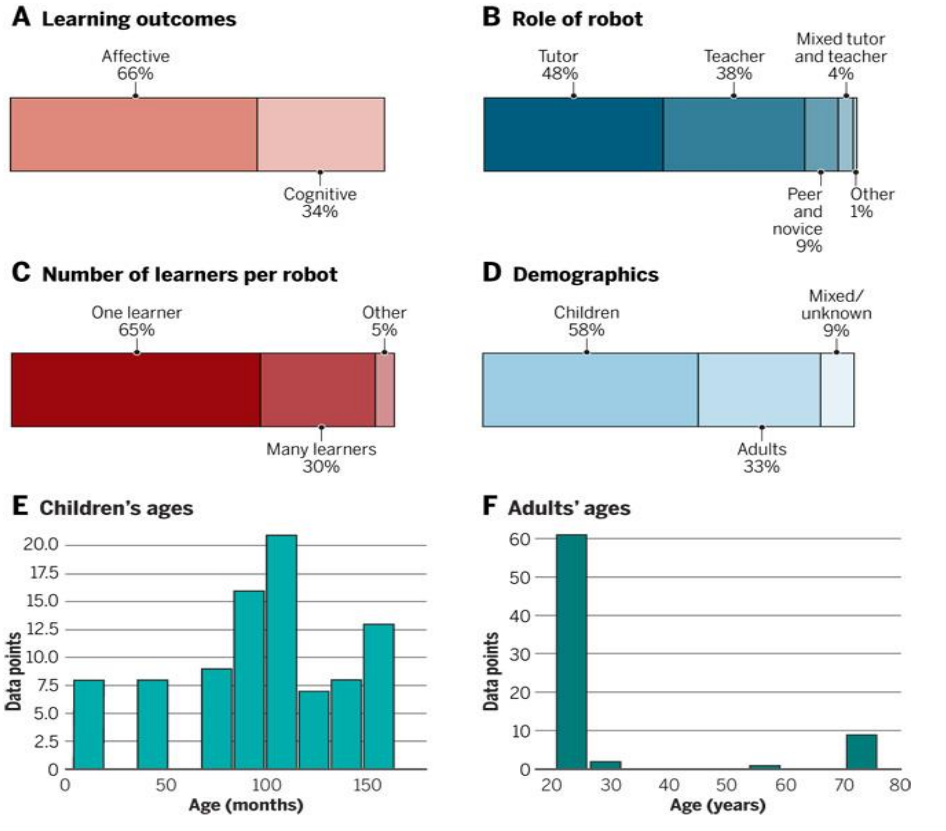


FIGURE 2.2: Data analysed from experiment [9].

CREDIT BASED ON ADAPTATION: A. KITTERMAN/SCIENCE ROBOTICS

The other educational robot is the blue bot and the Bee-Bot robot. The Blue-Bot and Bee Bot robots, which are available singly or in packs of six for classes and schools, are the ideal tool for teaching the youngest while having fun. To allow youngsters to effectively learn, they would program the robots' pathways and navigate them on the mats to learn the program. According to the mini robots and the many teaching mats accessible, the Bee-Bot and Blue-Bot educational robots are delightful bee-shaped robots that will help little ones to learn patterns, colours, numerals, letters, or even the first stages of simplified programming.

- **Miro Cloud**

A simple case study is the Miro Cloud [58]. Miro Cloud is an educational robotics coding platform. It also has a good and friendly user interface comprising of different layers.

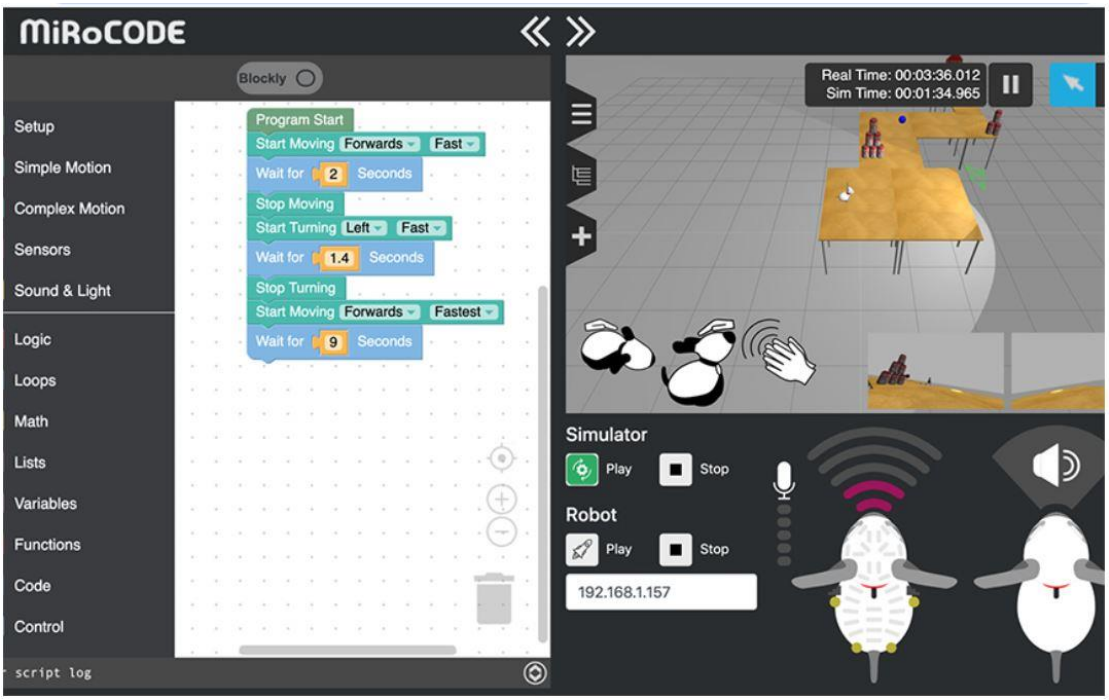


FIGURE 2.3: Miro Cloud Interface [19]

Miro Cloud is built with three basic components which are Blockly, Python, and ROS. Blockly is used to transform python codes into block layers. Python being a generic hardware and software programming language makes it easy for scaling to different system requirements. While ROS being the background framework of the Robot Software back end, is used to provide accurate MiRO-Sim [76] representation. MiRO

Cloud becomes very resourceful as it provides all components for complete education, such as the cloud, block coding, simulation, and interfacing with a physical robot. MiRO over the years have been classified as a Companion Robot due to its biomimetic layout and zoomorphic representation of a dog. The MiRO robot was first deployed for domestic and school purposes. At a larger aim, it was purposed to drive public engagement in STEM, and specifically in Robotics education.

MIRO aesthetics and morphological concept focused on behavioural perspective which guided its mammal identity replication. Animal behaviour was chosen in order to reduce the expectations of users as they would normally have a high expectation for humanoids. This personality choice was meant to reflect the famous Japanese word "kawaii" which means cute and not too toy like in nature. These expressions are seen for example in tail waggling, eye blinking, neck movement, colour lighting communication. MiRO also gives a gesture which may be seen in colour change and tail waggling to display happiness or closing eye lid and bending head to display sadness.

Basic drivers and the three Degree of freedom neck are the platform built out components. In addition to these, are the two ears, two eyes, tail, and caster which consists of different degrees of freedoms. There is also a speaker mounted on the board of MiRO which enhances its vocal effect as a mammal.

The control system of MiRO is a simple brain model consisting of lower, middle, and higher-level loops layers which are the major components of its architecture. The lower loop level is responsible for its sensory capability. The middle loop level is used to create coordination between its actions and brain. While the higher loop level is used to develop some cognitive ability based on the signals generated and memory depth.

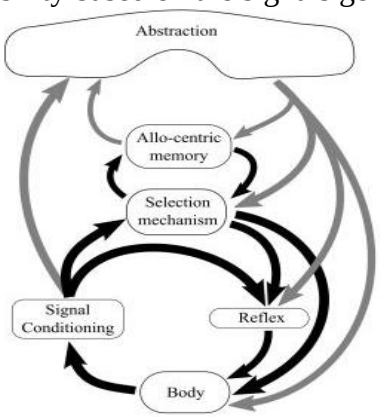


FIGURE 2.4: MIRO control system layers [18]

The MiRO code is a programming nomenclature used for the codes generated from MiRO cloud platform. The MiRO cloud is a cloud web application which is used to generate programs for the MiRO robot to act upon. This consists of sets of Blockly Blocks, block programming interface, miro robot simulator, and a connector interface for the Robot Hardware. The MiRO code can also generate the python representation of the blocks which can be ran independently of the connector.

The MiRO robot has been seen to have numerous strengths seen which are; good companion due to its dog like nature, health care due to its ability to communicate to its users in a social manner, in education by helping younger students appreciate robotics from an easier perspective of visual programming, and in research where researchers use MiRO as a benchmark for its research for Human Robot Interactions.

Although there are still some limitations and improvements which need to be done. Some of these limitations are:

Firewall issues: This issue is seen with the networks used to access the MiRO web application during research and outreach purposes. Due to the firewalls of networks, access to the application is blocked.

Server response: During my course in Human Robot Interaction, I and some other students had to wait for hours before MiRO cloud loads up, and eventually when it loads up, the simulated robot takes a lot of time to respond to the generated program.

One Robot: There are other zoomorphic robots available for public use, the MiRO cloud platform only generates actions for the MiRO robot and hence cannot be extended to other robots. Although this isn't critical as MiRO-Cloud is commercially built for the MiRO-E.

Non-ROS code generator: ROS is a very popular and most used robot operating system. The code generated from miro cloud is not a ROS code there by making it very difficult to run the code as stand-alone on a ROS robot. This is critical based on the code generation from the blocks.

The last limitation is seen in its version and updates. I experienced an interesting problem when I was working with the MiRO robot with transferring the generated program to the robot. The solution to the problem after emailing MiRO support Team was to update to the recent version of software. But this was strongly discouraged as I

was told it would update the software version of all other robots which were linked up and synchronised by a method which is still not understood.

These limitations cited above were overcome in this project work.

The next section outlines the trends of visual programming in Educational Robotics.

2.3.2. Visual Programming

Visual Programming unlike the very known conventional way of programming with codes and syntaxes, invokes a method where software programs can be created by using graphical elements. Back in the 1970s and 1980s, there was so much desire to teach children how to program. In the 1980, Seymour Papert presented Logo [86] [87] in his book "Mindstorms" as a method for educating School Children in programming [69]. Although this didn't live to see the light of the day because; it was too difficult for children who found it complex to understand the syntaxes, it was incorporated with activities which were not interesting to students, and from the context of not getting enough assistance when the kids were stuck. To disagree with this, Papert came up with an argument that programming languages should be a "low floor" (easy to get started) and a "high ceiling" (ability to improve in complexity) [32] [33]. Following this study is the attempt to introduce programming for teenagers [46]. Some of the professional used programming languages for teens were Flash, Alice, and Squeak Etoys [46] [44]. This wasn't still satisfying based on the argument that the floor could still be lowered, and the walls made even wider in-order to meet the demands of educating Children and Teenagers. Hence the introduction of Scratch, the triplet of (i) low floor (ii) high-ceiling and (iii) wide walls [32] [33].

In a very more generic case, Visual Programming is done with Visual programming languages. Some of the very popular visual programming languages are:

- **Scratch**

This is a Massachusetts Institute of Technology platform designed for School Children inspired from Lego Mindstorms [78] [23]. With regards to meeting these three desires from [32] to develop a more thinkable, more meaningful, and more social application gave birth to the concept guiding the development of Scratch. Based on the inspiration

gotten from how children responded to Lego bricks led to the design and development of Scratch to have similar feel and structure. Scratch Blocks are basically sets of virtual blocks which generate certain standard programming structures restricted from users view but provide the actions they've been configured to generate. Scratch got its name from the concept of tinkering which is a technique used by disk jockeys. This concept greatly inspired the general structure and hence creating a "*more thinkable*" programming concept.

In order to create a "*more meaningful*" programming scenario there were two significant considerations which were diversity and personalization. These considerations guided in some design concepts with most concept seen in the 2D design of the blocks rather than a 3D design. Some of the remarkable works seen from this goal is the Donkey Kong Game and the project from a 13-year-old Indian Boy who built an animated character to travel to the centre of the earth with a voice-over describing everything seen on its path [78].

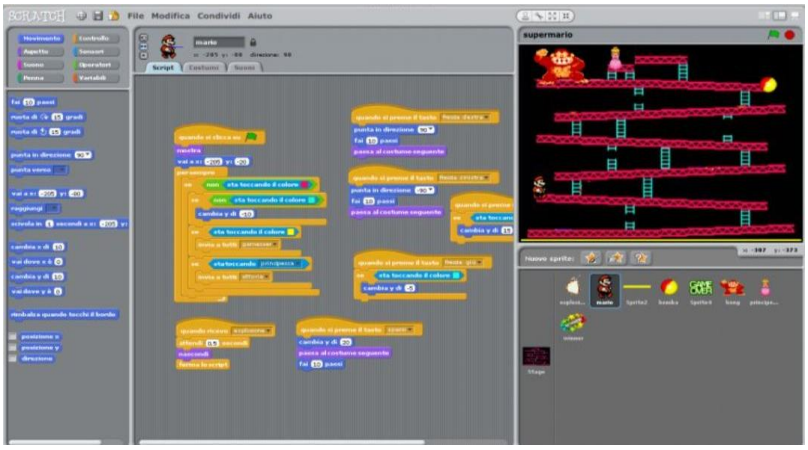


FIGURE 2.5: Donkey Kong Game built from Scratch Blocks

In an attempt to produce a "*more social*" solution, the Scratch platform offered a large community for support, collaboration, and critique. To further extend this capability is the ability to share ones project by uploading them to the Scratch website. This resulted in over 500,000 projects been shared in just 27 months after the platform was launched.

The biggest challenges considered in 2009 were more of cultural and educational which is some worth still a concern in present days as not all societies and learning institutions consider Scratch to be a limitation for K-12 Students in adopting coding [88].

FIGURE 2.6: Sets of Scratch Blocks

- **Blockly Blocks**

Blockly is a Google Open-Source framework utilized for adding blocks programming to different software applications . A remarkable feature of this framework is the ability to create python, JavaScript, and other custom programming languages. Blockly has its built stack on JavaScript. There are few uniqueness of Blockly Blocks which are:

1. Conditionals vs Loops: A persistent challenge seen in other Visual Programming are the Loops and Conditional Blocks. Blockly implemented the conditionals block by embedding them in the logic group which solved this problem [73].
2. Border Style: Initial blocks in the 2000s had an Aqua style look which was almost like an off point to modern day visual programmers. Blockly followed this trend with an improvement of its block styles by creating a highlight and shadow effect around each block.
3. Bumping Syntax Errors: Blockly blocks are in general, free from syntax errors but in some cases when wrong blocks are connected, end users get thrown a warning which help them debug the incorrectness of their program.

There are two significant problems faced with the two above visual programming languages [35]:

1. Code Ownership: In many cases, users want to remodify the outputs from the actions the blocks arrangement creates, but this isn't possible which leaves end users with no choice that accepting the best action generated. This may be

argued from the point of the ability to modify what each blocks generate, but this is not so an easy work to do and because of the rational behaviour of end users, a developer or engineer can never get all the codes included into the blocks which will satisfy all users in one go.

2. Exit Strategy: This has been repeatedly cited as a challenge for visual programming and hence this study and project also aims at solving this problem. Visual programming can be very addictive, the question remains; at what point do end users progress into real coding? as visual programming was never intended to be an end point but rather a progression to actual programming. The other question here, remains; what is the exit strategy for such users?

2.3.3. Cloud-based Robotics Platforms

- **DaVinci Framework**

In the early development of modern robotics, robotics programming platforms used the DaVinci Cloud Computing Framework to provide cloud computing activities by using one computing environment without separating any process and managing them on one ROS master. Based on this research work [7], the development, and deployment of the DaVinci cloud architecture were motivated by the drive to create an architecture that will be able to enable complex agents to share data gotten from sensors and write/upload the data for computational algorithms to be processed in the nodes. Before this project was embarked on, other frameworks were considered, but had a limitation of only computing in small environment whereas as the DaVinci framework would provide the same computation of heterogeneous robots in a large environment. During this early phase of their work, they were able to develop a backend system that make multiple robots share different resources and adopted both the Hadoop Map/Reduce framework [70] [28] and FastSLAM [62] [61] algorithm for providing computation ability in clusters, and the FastSLAM been implemented as a Hadoop Map/Reduce task. The results from the implementations were seen by a graph of execution time for different particle numbers. The execution time were set at 1, 50, and 100 for various particles. At this stage of their development, there were some challenges and limitation faced which were the non-consideration of network latencies which might result in difficulties in transporting ROS messages between the server and the robot. Future improvements

on this framework have been seen to expose algorithms for SLAM, path planning, and sensor fusing over the cloud [7].

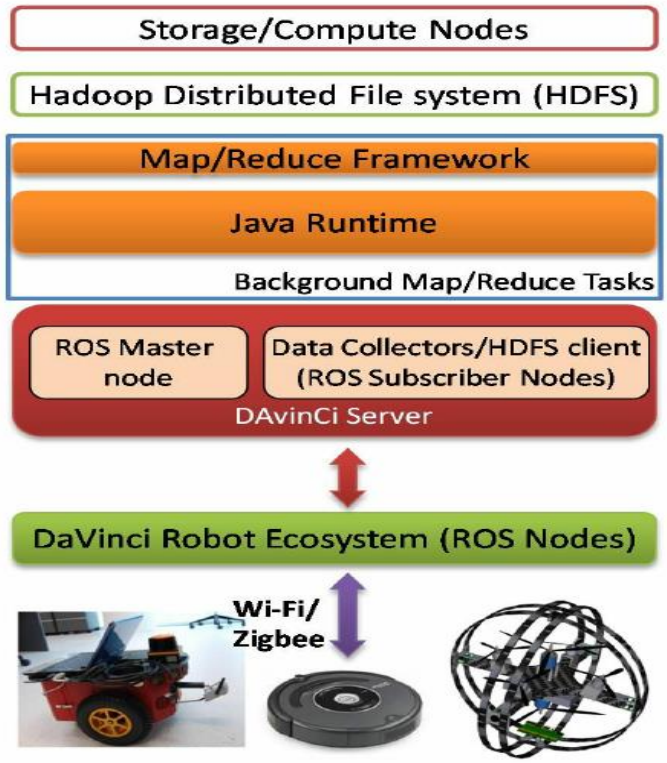


FIGURE 2.7: DaVinci Architecture Layers [82]

- **RoboEarth**

Early in 2011, the cloud-based coding platform called “RoboEarth” was embarked on by some European scientists [51]. The research had a clear objective to modify and improve the limitations with a unified technology and develop a major resources and computing power storage. Robots were able to share their knowledge and data across each operating system thereby making them possible to learn from each other quickly [43].

- **RobotBrain**

After this work, a sterling project called “RobotBrain” was developed by both Microsoft and Google [52]. The project aimed at making the RoboEarth platform as a brain of the robots which helped the robots make decisions faster and more efficiently.

Following this trend, [81] put forward a platform for robots and developed a protocol for communication known as RSNP which had the capability of attaching each task based on the users need [50][81]. Immediately after that, they put forward Jeeves which was centred around RSNP [50].

- **Rapyuta**

Another ground-breaking research is the RAPYUTA platform in 2015 which was an off shoot of the RoboBrain Project [36][41]. This research embarked on, was to demonstrate the design, benchmark results, and first use of Rapyuta [59]. Rapyuta in its unique feature helps in offloading heavy computations and grants easy access to RoboEarth knowledge repository [7].

According to Waibel [51], the deficient performance of Platform as a service platform for robotics instances are caused by web applications and robotics applications differences [98]. The Rapyuta computing environment makes it possible for robots to exchange resources and communicate be supplying efficient bandwidth privilege to RoboEarth [7]. Prior to the RoboEarth introduction, various computational processes were done locally but now, these processes can be performed in the Cloud with Rapyuta, that's why Rapyuta can also referred to RoboEarth Cloud Engine. The performance was measured under two experimental conditions. The first experiments were conducted from measuring the RTTs of various message sizes of two processes. While the virtual process was carried out on instances of the machine in Amazon's Ireland Data Centre.

- **rosbridge**

rosbridge [12] which was an open-source project like the DAvinCi project focused on external communication between one robot and one ROS environment hosted in the cloud. Comparing Rapyuta with rosbridge based on RTTs supplied the benchmarking results gotten in this study. From this work there were no limitations cited by the author neither did I notice any. Rather, the authors give an approach that can help me in building my cloud-based robotics platform, by getting servers from an Infrastructure as a service provider, by using technologies like Open vSwitch [7][15] and ROS master to connect the servers and connecting the external robots to the servers by using rosbridge [7][20].

- **Robot Benchmark**

In 2017, the Human Brain Project [54] co-founded the robot benchmark project. Robot benchmark is an open-source platform that depends on the Webots [22] open-source platform and offers various levels of robotics programming opportunities to users with an online simulated platform. Although it is still a work in progress.

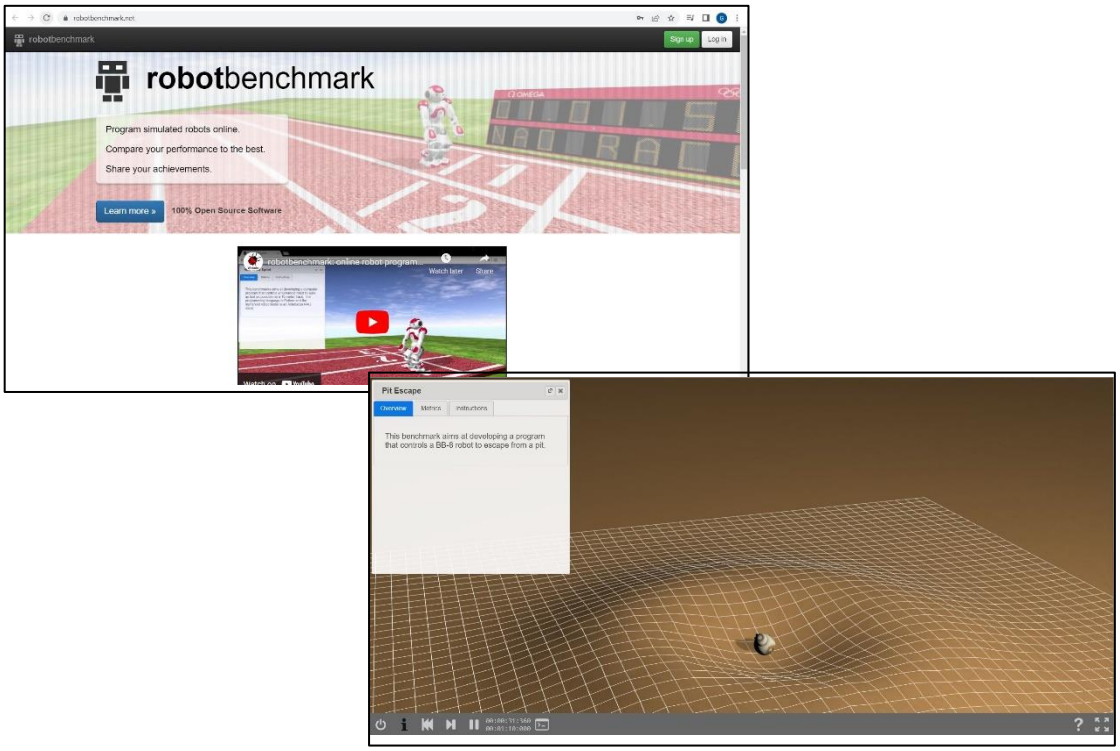


FIGURE 2.8: Robot Benchmark web landing page and simulation of the BB-8 Robot

With robotbenchmark, it is possible to share one's progress with other users, and it is 100% cloud based. With Robot Benchmark, one can interact with objects and robots, and interact with 3Dimension scenes. All resource sharing is being done in the cloud as end-users only get to interact with the Application Front-end. In general, the robotbenchmark is an improvement of the cyberbotics project – webots [67][21] in-order to make it run on the cloud.

2.4. Recent works and discussions

It is worth having some insight into the trends of Robotics Education in recent times. Based on the scope of the expected outcomes from this review, this will offer insight into understanding how Robotics Education have transitioned over time with cloud-based platforms to present days.

2.4.1. Robots Education with Social Robots

Present day organizations have a great expectancy of workers to be optimally knowledgably in Technology. One of the major issues in this regard is how to incorporate STEM education into that scenario. This is neither a topic nor an educational trend, but rather innovative methodological techniques that might interest learners. In this regard, active approaches based on mechatronics and robotics might be an attractive avenue to study. Given this backdrop, the first step in assessing the feasibility of this technique to understand the landscape of robotics and mechatronics applications in STEM Education, as well as how active techniques are used in this context. Considering three recent Educational Robotics.

- First is the EZ-Robots which is a very recent Educational Robot in the Industry. Before the development of the EZ-Robots, robots were generally divided into Robotics Platform such as NAO Robot and Stand-alone platform. EZ-Robots has its uniqueness in its ability to merge the functions and adaptive nature of a framework simultaneously with emotional perception. EZ-Robots offers students and professionals the ability to build various robotics design.
- The Second is called “First Robotics” which teaches students the fundamentals of robotics based on a curriculum. Although First Robotics have shown limitations which are seen in its expensive cost as compared to other platforms, and the result from its user’s evaluation which appealed to only 1% to 2% of Students.
- The Third platform is “Vex” which gives the opportunity to explore traditional programming such as Blockly, RobotC, and Python [90]. Sharing same draw back with the First Robotics as it still doesn’t seem appealing to a great percentage of Students.

2.5. Educational Robotics Evaluation

Getting problem solving skills and creative skills have been a persistent issue [45]. In preparation for the future of students, [102] cited creativity as an essential component. Problem solving skills have been seen from psychological research to be the highest cognitive activity and hence can motivate students to improve their thinking abilities [47][79]. In the United State, the bad results of Students in Program for International Student Assessment, made it necessary for Schools to develop programs to boost the STEM skills including problem solving skills [63]. Educational Robotics is also one of the skills all countries are steadily trying to imbibe in K-12 learning to boost students' creativity and problem-solving skills [77].

A very relevant consideration while developing a cloud-based application is the target audience, which are 10 years old students in this research work [30]. Most of the research done centred around the technological aspect of Cloud Robotics, only a few considered a defined target audience, one of which is [71]. Educational Robotics have been classified into two categories as assistive and educational robots according to [89], where Han's work in 2010 further categorised them into hands-on robots and educational service robots [40][71]. From the above two categories, this research work only comprises of hands-on robots which is same as educational robots.

To understand the adoption of robotics education there was a study conducted with less privileged children in Korea by choosing 131 vulnerable students from 8 facilities to participate in an outreach activity for one year with two different robots used for 6-months each. From this study, it was seen that 11 students dropped out with only 112 students left in the second half of the year. The results gotten and analysed were the Qualitative behavioural change which showed 16.7% of students' behaviours improvement, 66.7% improved to a degree, and 16.7% did not change [71]. In the second study conducted to deduce the Quantitative analysis for mental health with an experimental group of 50 students out of 119, and a control group of 50 students who were not green with robotics education.

From the experiment above, it was found that there was significant improvement in the mental health of the participants with an improvement in their immersion and ego-resiliency. According to Mauricio in 2020 Educational Robotics Platforms impact and benefits could be limited by the weakness which have been noticed [5] (Reported in

D1.1): Research in educational robotics lacks detailed and structured description of activities and their pedagogical design, which are of importance to their scalability and assessment. There is a need for an analytical and critical description of educational robotics activities to become more explicit and elaborate about pedagogical design, and to have activities that can be shared and interlinked. The aim of their work was to find a balance between a level of abstraction that it will make the template adaptable to different settings and a level of detail that will demonstrate the influence of a specific pedagogical approach. In their research it was possible to find three standard tests used to evaluate critical thinking. First, researchers have identified as main problem for teaching critical thinking is schools and universities are that curriculums are focus on subject, leaving small space to teach generalizable skills [4]. Pithers and Soden [4] suggest the Second ideas to teach critical thinking in a classroom is by students to think about the process of thought more explicit, making them reflect upon their thinking.

Following this experiment, in 2021 data was gathered from a classroom from the Ozobots software, which indicated that through simple programming introduction that primary school students were able to work with robots hence improving their problem-solving skills [93]. This experiment followed an improvement from a previous in 2012 which concluded that students struggled to learn robotics because the methods were amorphous [6].

Finally, Walker and Finney [3] concluded that self-awareness through reflection has helped students to improve their critical thinking and problem-solving skills. A unique strategy introduced is the “Schule 4.0. – jetzt wird’s digital” [31] involves activities for students throughout their school career as well as measurements for teachers and for schools which makes it the Ideal Strategy which this platform will be evaluated on. It can be concluded that ER4STEM needs to take students and teachers into account which revokes this method from been used [49]. Also, performing workshops for the students will be a core activity for evaluating the effectiveness of the platform.

3. Open Roberta Open Source

3.1. Overview

With the rise in everyday demand for Robotics in Educational settings, the Open Roberta Framework presents an extremely useful cloud-based approach in aiding the learning of Robotics using the New Easy Programming Online [42]. NEPO is made up of Blockly Blocks Library which makes it easy for Teachers to teach students Robotics using Visual Programming from Blockly Blocks arrangement to fit into desired Robot Output through Simulation and actions on the Physical Robot. One of the uniqueness of this framework is the ability to create an extension to accommodate other Robots hence its choice for building the Cloud-based platform for the ROSbot 2 Pro. It is worth knowing the available Integrated Systems and Robots available below.

3.1.1. Integrated Systems and Robots

We can see the numbers of available robots and integrated systems the Open Roberta opensource has. Here is a list of the available Integrated Systems and Robots with their programming languages and firmware (TABLE 3.1):

Systems	Programming Language	Firmware
LEGO Mindstorms EV3+++	JAVA, Python, C/C++	leJOS, EV3dev, c4ev3
LEGO Mindstoms NXT	Not eXactly C (NXC)	original LEGO
Calliope mini	C++	Micro:Bit runtime
micro:bit	MicroPython	Micro:Bit runtime
NAO	Python	original NAO
Bot'n Roll	C++	Arduino + Bot'n Roll libriaries
Edison	MicroPython	original Edison

WeDo2	JASON Stackmaschine	original Lego
Arduino Uno/Nano/Mega	C/C++	original Arduino

TABLE 3.1: Open Roberta Robots and Integrated Systems

3.1.2. Open Roberta Simulator

The Open Roberta Framework offers a 2-Dimensional Simulation Environment for the various available Robots Simulation. The Default 2-Dimensional Simulator is designed in JavaScript which runs bytes codes originally sourced by a generator of simulation codes. This Simulator Environment can be updated with various Client Background Choices. This in-built simulator is specifically for the mobile robots Open Roberta offers and the other systems.

In a recent release, the framework has been able to integrate the webots simulation environment which offers a 3-Dimensional simulation environment for the NAO Robot.

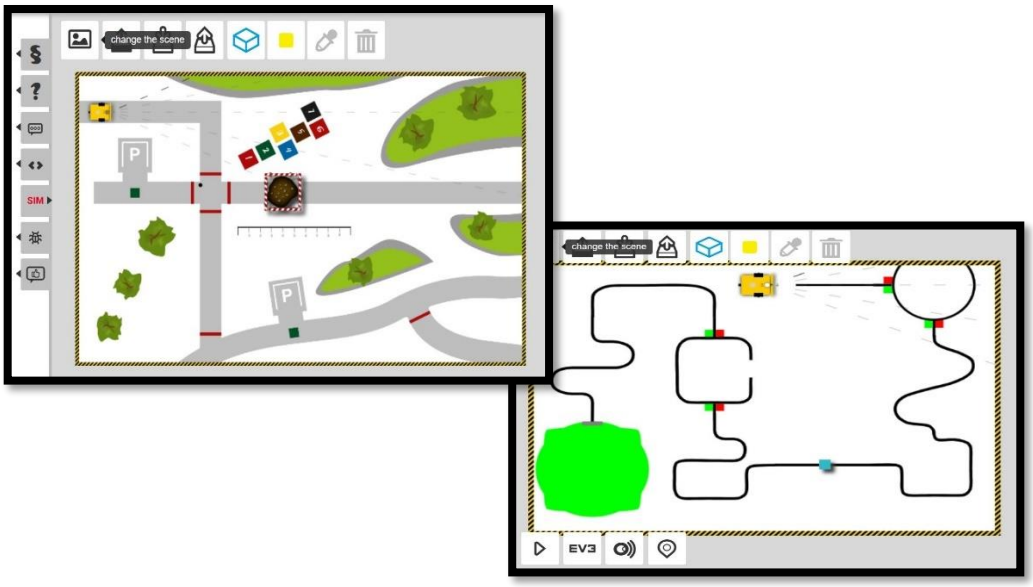


FIGURE 3.1: Open Roberta 2-Dimensional Simulator

FIGURE 3.2: Open Roberta 3-Dimensional Simulator

3.2. System Architecture

This Open-Source framework is just a simple client-facing application in a series of software and operating systems which makes the robots function in an educational environment. Its first level components are:

A) Web Browser: This serves as the end-user entry point into the Web Application.
B) Blockly Blocks: These are the Blocks which generates the Robot action.
C) Server: This provides the data transfer protocol around the System.

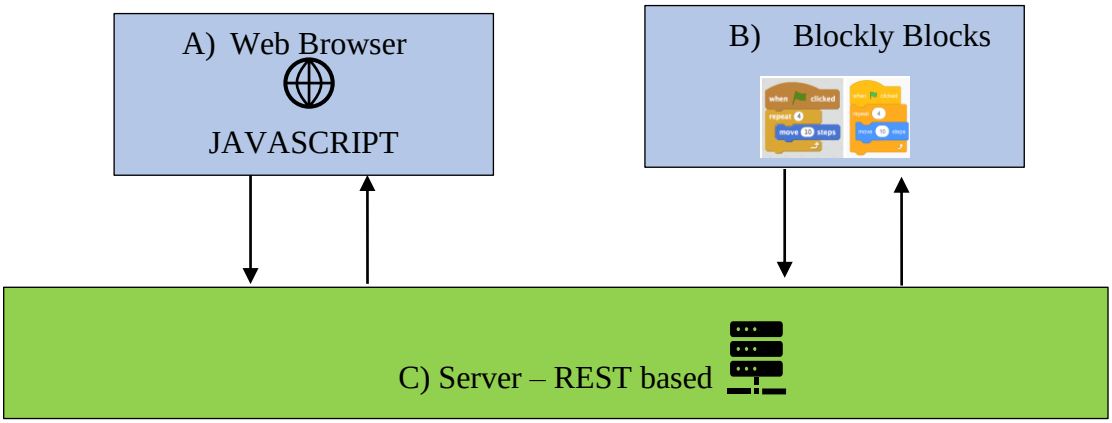


FIGURE 3.3: Open Roberta First Level Architecture

As a decoupled open source, here is an architecture of how each component and resources communicate with each other.

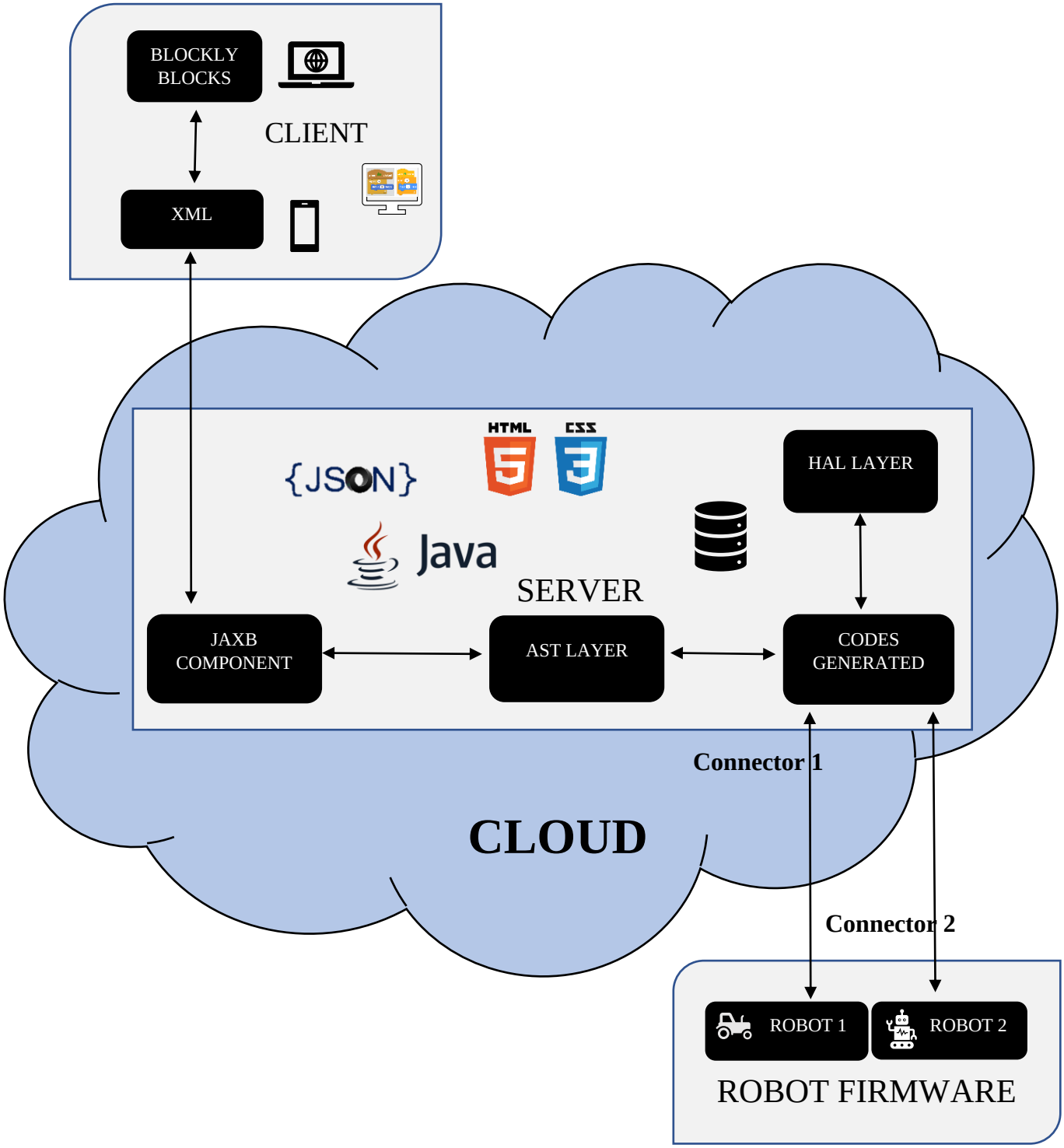


FIGURE 3.4: Open Roberta Second Level Architecture

3.2.1. Blockly Blocks in Open Roberta

Blockly Blocks are used to manage all the Client actions. These blocks are gotten by integrating Blockly APIs into the application using REST [56]. A blockly block used in this open source consists of a Toolbox, Category, Blocks, and Values. This is captured in details below.

At the first layer of the Architecture in *Figure 3.4*, the Blockly generates XML display of the blocks program generated. The XML generated is then stored in a JSON object and transfers from the web application to REST-endpoint in the server. Therefore, the Blocks Generated from the Blockly API can be represented in two formats which are the JSON and XML format.

In the existing open-source, blocks are generally generated and gotten from toolboxes, which are one of the most important resources in the components of each Plugin (Robot). Blocks and configuration components that are already being used in other robots can be reused. But since the blocks available are not configured to generate ROS based codes, a new set of code blocks are generated from the visitor resource in the JAXB component.

3.2.2. XML for Open Roberta Plugins (Robots)

The XML Resources major function is to lace or bind the data generated from the blockly codes with the client and the machine. We can basically look at the Blockly XML as the component responsible for creating a path between the Blocks specified to be used and the Application which is built on Java. This is a very important component as Developers and Researchers will need to understand how this component works to define and configure how and where each block display on the Frontend of the Web Application. A typical relevance of this component is seen when I wanted defining which blocks are to be displayed and used for the ROSbot 2 Pro. This is further seen in *section 4.4.2* below.

In a standard new plugin build, a plugin must consist of these XMLs files:

1. <robot_name>.configuration.default.xml

This is the default parameter and state configuration of the robot properties like the WHEEL_DIAMETER, MOTOR_REGULATION, and others

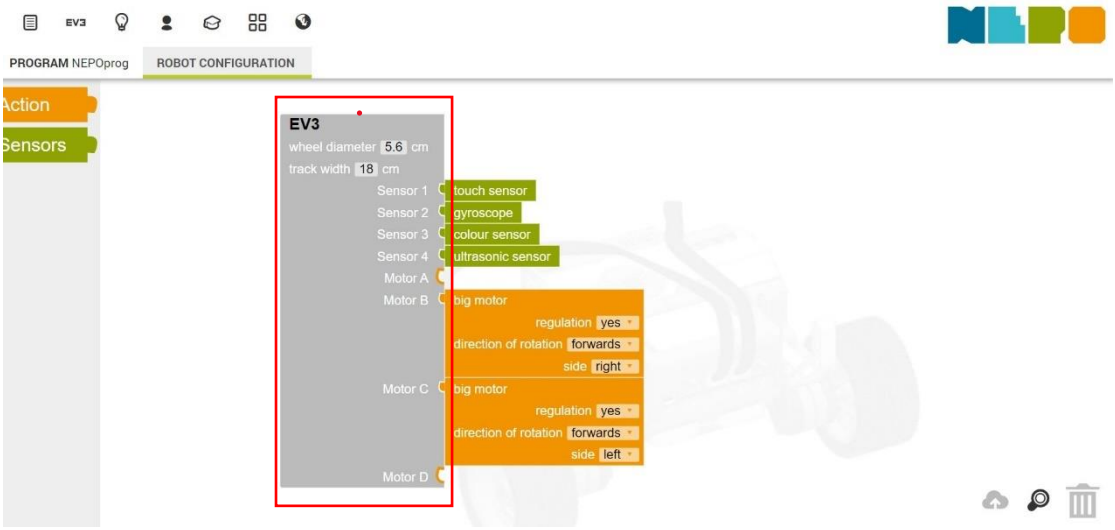


FIGURE 3.5: Open Roberta Default configuration parameters

2. <robot_name>.configuration.toolbox.xml

   This is where all the basic required blockly blocks are specified for the default configuration above.

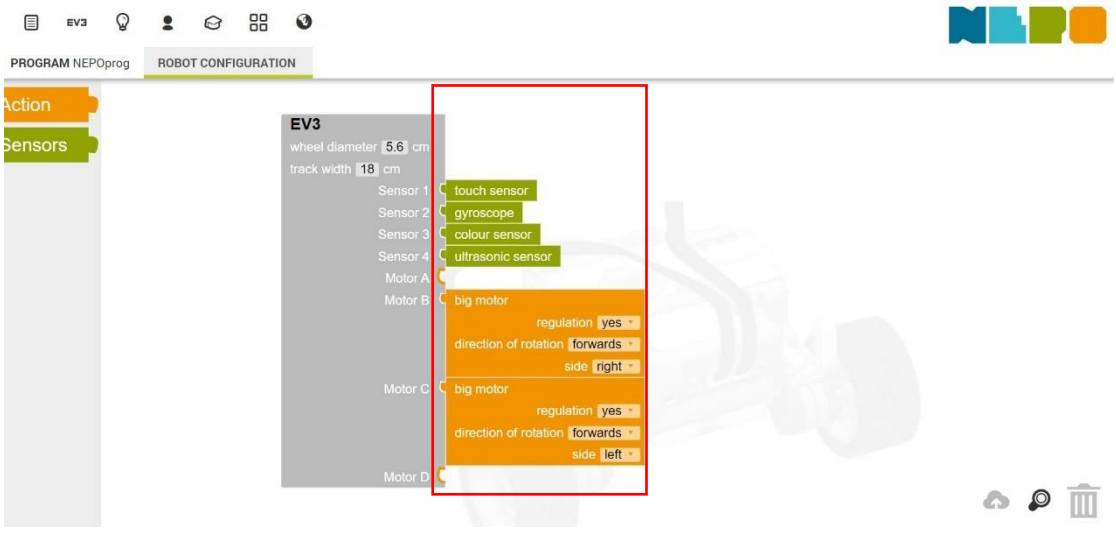


FIGURE 3.6: Open Roberta Default Toolbox configuration

3. <robot_name>.program.default.xml

   This is used to specify the default block to be used when the Robot is selected. In many cases, the “robotcontrols_start” block is used.

FIGURE 3.7: Open Roberta Default program start block

4. <robot_name>.program.toolbox.beginner.xml

   During Program start on the Web Application, a user can select easy blocks to use. This is configured in this XML.

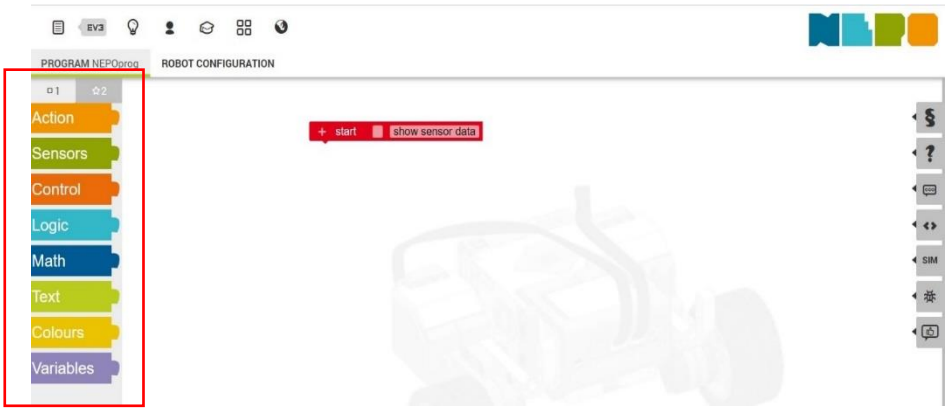


FIGURE 3.8: Open Roberta Beginner Toolbox

5. <robot_name>.program.toolbox.expert.xml

   Users can also select a category of advanced block like messages and functions. The blocks which fall into this category is configured in this XML.

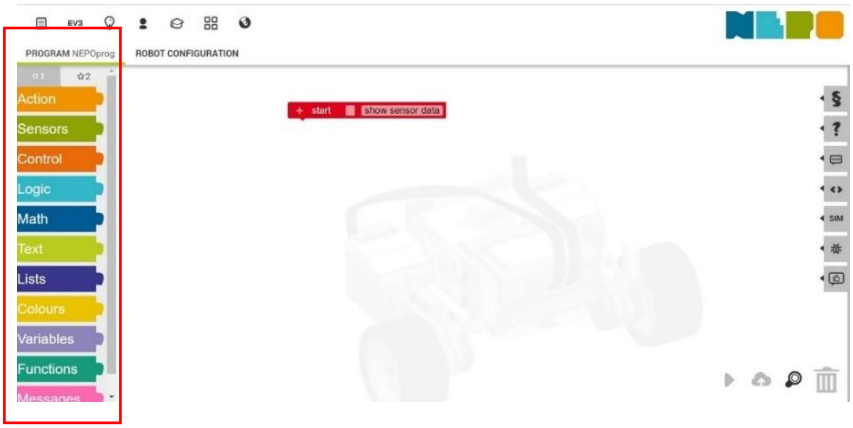


FIGURE 3.9: Open Roberta Expert Toolbox

These are the list of the available blocks in Open Roberta Open Source and the ev3 Robot:

BLOCKS CATEGORIES	OPEN ROBERTA BLOCKS	EV3 ROBOT BLOCKS
TOOLBOX_COLOURS	mbedColour_picker mbedColour_rgb naoColour_picker naoColour_rgb naoColour_rgb robColour_picker robColour_rgb robSensors_colour_getSample	robColour_picker
TOOLBOX_LISTS	robLists_create_with robLists_getIndex robLists_getSublist robLists_indexOf robLists_isEmpty robLists_length robLists_repeat robLists_setIndex	
TOOLBOX_LOGIC	logic_boolean logic_compare logic_negate logic_null logic_operation logic_ternary	logic_compare logic_operation logic_boolean
TOOLBOX_LOOPS	robControls_loopForever	
TOOLBOX_MATH	math_arithmetic math_constant math_constrain math_modulo math_number	math_number math_arithmetic

	math_number math_number_property math_on_list math_random_float math_random_int math_round math_single math_trig	
TOOLBOX_PROCEDURES		
TOOLBOX_TEXT	robText_append robText_join text text_comment	text text_comment
TOOLBOX_VARIABLES		
TOOLBOX_ACTIONS	bob3Actions_recall bob3Actions_remember bob3Actions_set_led makeblockActions_leds_off makeblockActions_leds_on	robActions_motorDiff_on_for robActions_motorDiff_on robActions_motorDiff_stop robActions_motorDiff_turn_for robActions_motorDiff_turn robActions_motorDiff_curve_for robActions_display_text
TOOLBOX_SENSORS		robSensors_touch_getSample robSensors_ultrasonic_getSample robSensors_gyro_getSample
TOOLBOX_CONTROL		robControls_if robControls_ifElse robControls_loopForever controls_repeat_ext
TOOLBOX_IMAGES		

TABLE 3.2: Open Roberta Toolboxes for the ev3 Robot

3.2.3. Java XML Binding

The JAXB creates the architecture for accessing the XML file. It also presents he XML files/documents in a Java format. From low-level perspective, XML is not a great processor, and a schema is always required for its validation. In perspective, the JAXB framework processes the XML generates Java Classes comprising of multiple objects (JavaBeans) for the XML components.

3.2.4. Abstract Syntax Tree

The AST is used to generate an additional node which provides additional information for the final code to be generated. For Code Generation, the XML is further transformed to an Abstract Syntax Tree. These transformation follows these steps:

1. XML to JAXB
2. JAXB to AST

3.2.5. Hardware Abstraction Layer

This is the layer responsible for the exchange of data between the web application and the Robots. This data exchange is made possible using connectors which is unique to various robots.

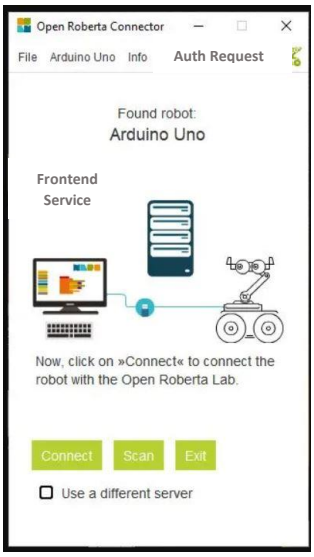


FIGURE 3.10: Connector Interface for an Arduino Uno Robot Middleware

3.3. Top-level Resources

From the High-Level System Architecture Diagram in Fig. 3.4 above, it is seen that the entire application can be sub-divided into the major components which are the CLIENT, CLOUD, and ROBOT FIRMWARE. Each of these components are further analysed below in-order to under how they function and what resources they depend on.

3.3.1. Compilers

Open Roberta Framework can be installed and ran on all Operating Systems (Windows, MAC OS, and Linux) but it is strongly recommended to run on a Linux System like Ubuntu. The framework is built on Java as its core stack although has other sub-stack used like JavaScript, JSON, HTML, CSS, Bootstrap, C++, TypeScript, and Python which supplies added resource for the effective function of the open-source framework.

To get the open source installed and running, there are a couple of tools you will need on your computer regardless of the Operating System.

These tools are:

1. Apache Maven
2. Java Development Kit
3. NPM
4. Git

3.3.2. Local Server and Database Analysis

The framework can be installed to run on the client's local machine which serves as the server. For every instance of first build using maven, a default database will be built into a directory in the "OpenRobertaServer" path called "db-embedded". At instance when this is not done, a new database can be created using:

```
"./admin.sh create-empty-db"
```

The above shell file locates the function appended to “create-empty-db” in the shell file and copies the default open Roberta database into the Database URI File ($DB_URI_FILE).

```
113   create-empty-db) java -cp $JAVA_LIB_DIR/\* de.fhg.iais.roberta.
        main.Administration create-empty-db "$DB_URI_FILE" >>
        $ADMIN_LOG_FILE 2>&1
114                     RC=$? ;;
```

The $DB_URI_FILE is for pointing to the right directory path the Database Should be installed.

```
65 DB_URI_HSQL="jdbc:hsqldb:hsql://localhost/$DB_NAME"
66 DB_URI_FILE="jdbc:hsqldb:file:$DB_PARENTDIR/$DB_NAME"
67 case "$DB_MODE" in
68   embedded) DB_URI="$DB_URI_FILE" ;;
69   server)   DB_URI="$DB_URI_HSQL" ;;
70   *)        echo 'invalid db-mode. Exit 12'
71             exit 12;;
72 esac
```

The $DB_PARENTDIR and $DB_NAME are assigned various variable names which becomes the name of the directory where the database is stored and the name of the Database.

```
19 DB_MODE='embedded'
20 DB_NAME='openroberta-db'
21 DB_PARENTDIR='db-embedded'
```

The Open Roberta Open-Source Data base is an HyperSQL database. The HyperSQL Database is a simple relational Database which is written in Java. This is setup and configured as an embedded database to get utilize in test or localhost purposes. It can also be deployed in a docker container or a Digital Ocean Droplet in server mode as described in the ROSbot 2 Pro addition below.

The database is generally accessed using REST service which receives data from the payload and performs and API call to the required parameter from the specified Database Table.

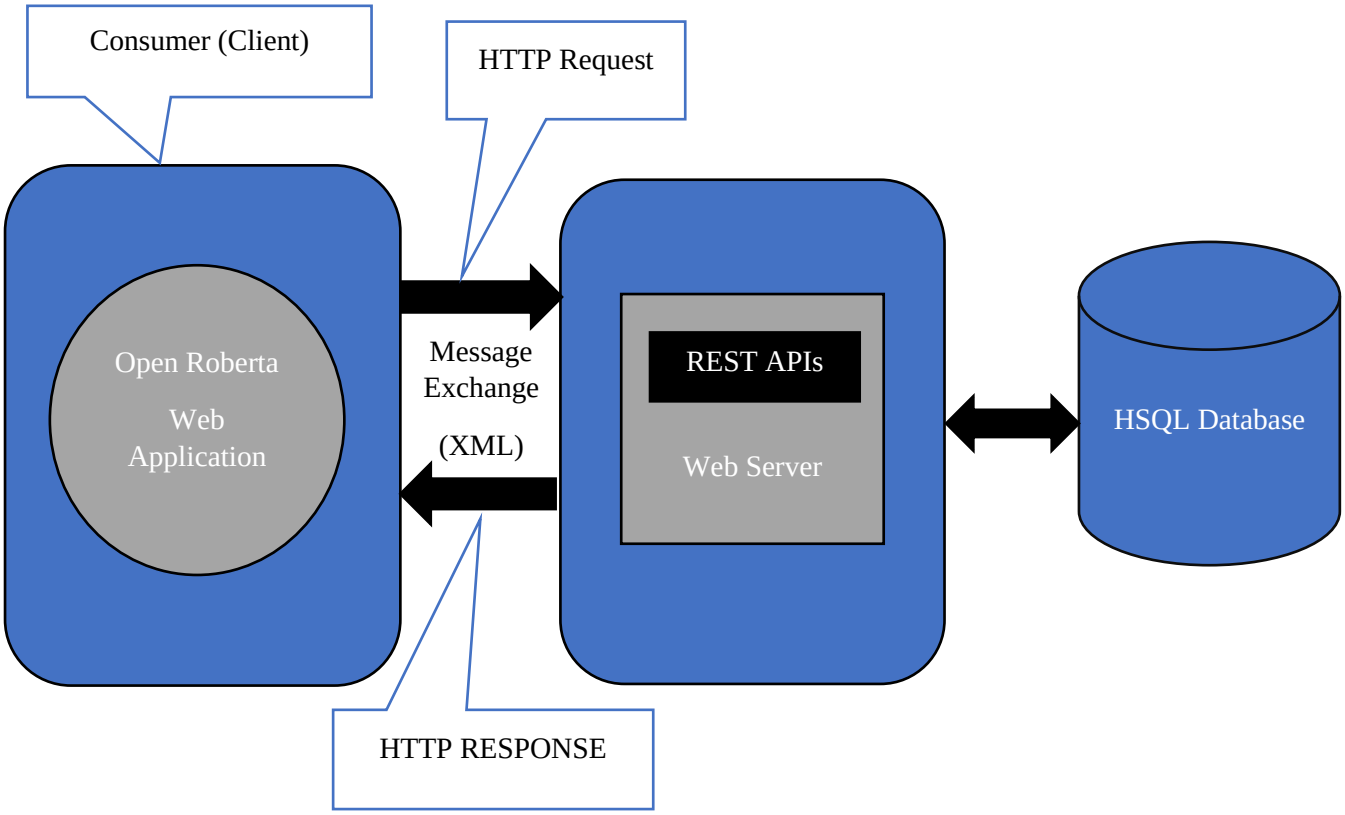


FIGURE 3.11: Server-Side Communication architecture

3.3.3. Server-side Resources

The opensource web application repositories are stored on in a Directory called the "OpenRobertaServer". All resources in this directory are responsible for the server and client services. The server services have been seen above whereas the client resources are seen below.

To understand these resources better, it is worth knowing what the static resources are in the Open Source. This repository in the server-side directory is responsible for all the services and resources which makes up the User Interface and User Interaction of the Web Application.

The branding of the application to fit desired design patterns and structures is done in this resource component of the open source. From the list below we can see what each component does.

- **Hypertext Markup Language**

The HTML resource is responsible for all the text structures and text formats of the web application.

- **Cascading Style Sheet**

The CSS resource is responsible for all the stylings, structures, and images retrieval of the Web Application.

- **TypeScript and JSON**

TypeScript is used to serve as an extension to JavaScript by adding types. In this opensource, typescript is used to create a communication protocol between the client-side and server-side during program execution.

```
/**
 * Handle server errors
 */
ALLOWED_PING_NUM = 5;
function handleServerErrors(jqXHR) {
    // TODO more?
    LOG.error('Client connection issue: ' + jqXHR.status);
    if (this.url === '/rest/ping') {
        COMM.errorNum += 1;
    }
    // show message, if REST call is no ping or EXACTLY
    ALLOWED_PING_NUM requests fail (to avoid multiple messages)
    if (this.url !== '/rest/ping' || COMM.errorNum ==
    ALLOWED_PING_NUM) {
        if (jqXHR.status && jqXHR.status < 500) {
            COMM.showServerError('FRONTEND');
        }
        else {
            COMM.showServerError('CONNECTION');
        }
    }
}
```

Whereas the JSON stack is used to store and transport data as an object.

```
{
  "name": "Open Roberta Lab",
  "short_name": "Open Roberta Lab",
  "version": "${openRobertaServer.version}",
  "description": "Roberta     Learning with Robots",
  "developer": {
    "name": "Open Roberta",
    "url": "${project.parent.url}"
  },
  "default_locale": "en",
  "start_url": "index.html",
  "launch_path": "/index.html",
  "display": "standalone",
  "orientation": "landscape",
  "icons": [
    {
      "src": "/css/img/NEPO.ico",
      "sizes": "16x16 24x24 32x32 48x48 64x64 128x128 144x144 256
    x256"
    }
  ]
}
```

3.4. Code Generation and Validations

This is done in visitor classes like the Ev3PythonVisitor.java which defines what each block represented by a visitor method is going to generate for the final program to be executed on the rosbot.

Code validation checks if there are any obvious mistakes in the program build by the user, like missing inputs and configuration components and such. This can be seen in the Ev3ValidatorAndCollectorVisitor.

3.5. Robot Benchmark and Open Roberta Open Source

Initial open source used was Robot Benchmark, but about one month into the Project, there were certain constraints which made me search and switch to the Open Roberta Framework.

Here's a summary Table with most of the constraints experienced with Robot Benchmark as compared to Open Roberta framework.

Qualities	Robot Benchmark	Open Roberta
Clear process on use	★ ☆ ☆ ☆ ☆	★ ★ ★ ★ ★
Developers Community support	★ ★ ☆ ☆ ☆	★ ★ ★ ★ ☆
Visual programming	☆ ☆ ☆ ☆ ☆	★ ★ ★ ★ ★
Server speed	★ ☆ ☆ ☆ ☆	★ ★ ★ ★ ★
Good user interface and experience	★ ★ ★ ☆ ☆	★ ★ ★ ★ ★
Easy to expand on	★ ☆ ☆ ☆ ☆	★ ★ ★ ☆ ☆
Robot Simulation	★ ★ ★ ★ ☆	★ ★ ★ ☆ ☆

TABLE 3.3: Comparison between Open Roberta and Robot Benchmark

Robot Benchmark offers quite some great features and capabilities such as the ability to simulate a good number of Robots at different levels of difficulties, and also share your developed program with other users. Regardless of the constraints seen in Table 3.3 above, there are key limitations which didn't make Robot Benchmark a suitable open source to be used based on the Project Goals. These are, limitations in understanding the opensource, frequent breaking of codes after updates have been applied, lack of visual programming interface, and inability to deploy opensource to run effectively on my cloud server. The Open Roberta overcame these limitations.

Regardless of the Open Roberta open source overcoming these limitations there were still inherent limitations in fully achieving project objectives. Some of these limitations are:

- **Non-ROS Code Generator**: None of the Robots available generated standard ROS Python codes as this was required because of the target Robot to be used which is the ROSbot 2 Pro.
- **No method to transfer Generated Codes to ROS Robot**: One of the main objectives of this Project is to be able to transfer the Generated Codes from the Blockly Blocks into the ROSbot 2 Pro via cloud to run. The open source has a connector for other Robots but nothing which could work for the ROSbot 2 Pro, hence the quest to build an API to overcome this challenge. With regards to answering the research question (*RQ-3*), I had to think of an API which can be easily extended to other Robots. The API was built to fetch and POST the generated program to a new server which I built in Node JS which serves as an end point for the API. This can be seen in *section 4.5.*
- **No interface to get the Robot Camera Feed**: Presently, the open source doesn't offer the ability to get a robot camera stream. To make the platform more fun for the target users (10 years old Primary School Students), this had to be considered and implemented for the ROSbot 2 Pro. With the way this feature was added, it is possible to stream the camera feed of other Robots.
- **Robot Identifier**: There is no generic method to create an Identifier or Token for the Robots to avoid deadlock situations and security flaws from the Robot running unauthorised users' program. I had to build a simple Client facing input field which accept the known Robot ID and appends it to the file generated before the Data is posted for the Robot to get and run.

4. Adding ROSbot 2 Pro Plugin

Plugins are referred to as the Robots and Systems added to the opensource framework. Some of the available systems are seen in Table 3.1 above. There are two ways plugins can be added to the opensource framework, which are as Standalones Plugins and Sub Plugins. For this Project, the Rosbot 2 Pro was added as a Sub Plugin. The ROSbot 2 Pro was added as a sub plugin because every resources it would need for its function was already existing in the "*ev3*" Robot Standalone Plugin.

4.1. End to End Tasks Overview

In order to achieve a part of the Project Objective, one of the key requirements is the ability to have a cloud-based platform which can generate sets of programs for the ROSbot 2 Pro. Based on the Open Source selected, the first step is to add the ROSbot 2 Pro as one of the Robot Resource available to the Open source. After this, it's ensuring that the generated Program from the platform can be transferred to the ROSbot 2 Pro, hence the building of the API with the new Node JS Server. Following this is adding a robot camera view which was done with ROS web video server. In making this application cloud based, all resources and server were deployed to a Cloud Server and all requests were HTTP based.

The flow diagram below shows a summary of the steps taken to achieve the technical aspects of the Project.

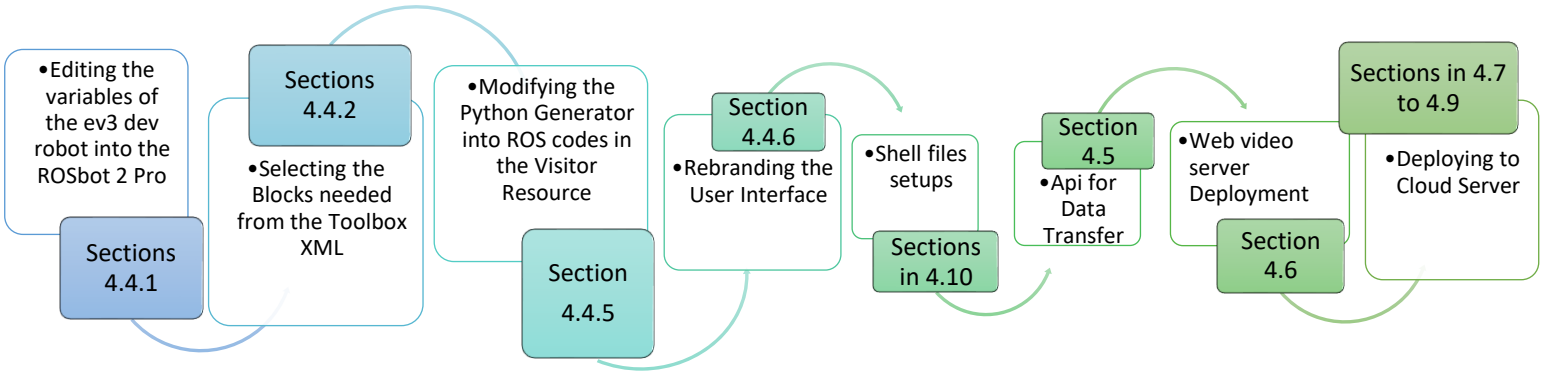


FIGURE 4.1: Summary flow process of adding the ROSbot 2 Robot

4.2. Standalone plugins

These are plugins which are completely built from ground-up where all the dependencies and resources are unique to the plugins. No many details would be given in this Paper for Standalone plugin because this method wasn't followed during the process of adding the ROSbot 2 Pro and its features.

Here's a simple and summary guide in adding a Standalone Plugin:

1. Including Plugin in build module tag: - The plugin name is first included in the module tag of the "*pom.xml*" file in the Open Source First Directory. The "<module>*robot_name*</module>" is included to specify the plugin the Java JDK and Maven Tool is to compile and build sequentially.
   These also specifies the modules which will always be built when the program is rebuilt after any update has been carried out. Should building out a new change take long of time, it can be reduced by commenting out the files which are not regularly edited.
   In this Project, only the RobotEV3 and OpenRobertaServer were made to be rebuilt as they were the two directories which were always edited to meet certain expectations.

   PATH: "~/OpenRoberta/openroberta-lab/pom.xml"
   The module structure is as follows:

```
<modules>
    <module>OpenRobertaRobot</module>
    <module>RobotEV3</module>
    <module>RobotNXT</module>
    <module>RobotArdu</module>
    <module>RobotNAO</module>
    <module>RobotMbed</module>
    <module>RobotWeDo</module>
    <module>RobotEdison</module>
    <module>OpenRobertaServer</module>
    <module>RobotCyberpi</module>
</modules>
```

2. Including the Robot "*Dependency*", "*GroupID*", and "*artifactID*":- In the Open Roberta Lab and OpenRobertaServer Directory, there's always a default "*pom.xml*" file which is responsible for defining the build and compile process of the framework using maven, and overall integration setups into the opensource framework.

   PATH: "~/OpenRoberta/openroberta-lab/pom.xml"
   PATH: "~/OpenRoberta/openroberta-lab/OpenRobertaServer/pom.xml"

   The structure is as follows:

```
<dependency>
                <groupId>de.fhg.iais.openroberta</groupId>
                <artifactId>RobotEV3</artifactId>
                <version>${project.version}</version>
            </dependency>
```

3. Artifact Item: - The artifactItem tag "<artifactItem></artifactItem>" serves as a determinant of the type and version of compiler to be used for the plugin Compiling.

   PATH: "~/OpenRoberta/openroberta-lab/OpenRobertaServer/pom.xml"

   The structure is as follows:

```
<artifactItem>
<groupId>de.fhg.iais.openroberta</groupId>
<artifactId>RobotEV3</artifactId>
<version>${project.version}</version>
<type>jar</type>
<overWrite>true</overWrite>
</artifactItem>
```

4. Robot Whitelisting: - This method is also applied to Sub Plugins. This method is useful in including the Robot as part of the available Robots in the Client

Facing Application. This name must be the same with the name assigned to the plugin in the plugin properties <plugin_name>.properties.

PATH: "~/OpenRoberta/openroberta-lab/OpenRobertaServer/src/main/resources/openRoberta.properties"

The structure is as follows:

```
57  robot.whitelist = sim, wedo, ev3lejosv1, ev3lejosv0, ev3dev, ev3c4ev3, nxt, microbit,
    botnroll, nao, bob3, rob3rta, sensebox, mbot,mbot2, edison, festobionic,
    festobionicflower, uno, unowifirev2, nano, mega, nano33ble, calliope2017NoBlue,
    calliope2017, calliope2016
58  robot.default = ev3lejosv1
```

The four files stated above are the static resources which should be done during additions of both Standalone and Sub Plugins.
The other methods of including a Standalone Plugin are stated below as they follow same process used in including the ROSbot 2 Pro as a Sub Plugin.

4.3. Sub Plugins

The ROSbot 2 Pro is added as a Sub Plugin to the "ev3" Robot by sharing all resources the ev3 Standalone plugin consist of. The ev3 is an excellent choice of Plugin for the ROSbot 2 Pro to share its resources with because from the framework, it has simulation similarity with the ROSbot 2 Pro and has a python generation Visitor which the ROSbot 2 Pro needs for its program.

4.3.1. Requirements

The functional requirements in adding this sub plugin are: -

1. A Linux Operating System Computer: An Ubuntu version with command shell and web browser installed on it was used. A Linux OS is preferred because of its flexibility, light weight, and speed.
2. Shell scripts

The non-functional requirements are: -

1. Maven Build Automation Tool
2. Java JDK, Git, Security

4.4. Adding the ROSbot 2 Pro as a Sub plugin

These tasks were performed on my local host before publishing to the Cloud Service. There are simple steps taken to include the new plugin.

- The first step was to clone the repository from GitHub into the Desktop Directory of my Computer with this command.

```
$git clone https://github.com/OpenRoberta/openroberta-lab.git
```

- The next step is to install all the application dependencies such as Apache Maven, Java JDK, npm, and Git

Once these setups are completed, the application is run by starting the server from running the shell file, "*ora.sh*".

```
$ ./ora.sh start-from-git
```

Once this shell is run, on first instance, a default empty database will be created inside the OpenRobertaServer Directory in a db-embedd directory, and the server started at a default port of "1999" and a local host IP of "0.0.0.0:1999"

After installing the open source, the next steps are seen below.

4.4.1. Robot Properties Settings

The ev3 Robot Standalone Plugin has three sub plugins which depends on its resources, which are the *ev3dev, ev3c4ev3, ev3lejosv0,* and *ev3lejosv1*. The ev3dev sub plugin was edited appropriately to fit the requirements of accommodating the ROSbot 2 Pro. As the first task, the ev3dev properties found in the RobotEv3 Directory was edited as followed:

1. Change of Robot Name
2. Change of Link to Robot Manufacturer

These changes are seen below:

```
# robot's long name to display
robot.real.name = ROSbot 2 Pro
robot.vendor = na
robot.beta = true
# robot's info for further information
robot.info.en = https://husarion.com/manuals/rosbot/
```

The next step followed was to understand the structure of the Toolboxes which consists of the categories, blocks, and values. A sample Toolbox XML is seen below.

```
<toolbox_set id="toolbox" style="display: none">
  <category name="TOOLBOX_ACTION" svg="true">
    <block type="robActions_motorDiff_on_for">
      <value name="POWER">
        <block type="math_number">
          <field name="NUM">30</field>
        </block>
      </value>
      <value name="DISTANCE">
        <block type="math_number">
          <field name="NUM">20</field>
        </block>
      </value>
    </block>
    <block type="robActions_motorDiff_on">
      <value name="POWER">
        <block type="math_number">
          <field name="NUM">30</field>
        </block>
      </value>
    </block>
```

```
    </value>
    <value name="POWER_RIGHT">
      <block type="math_number">
        <field name="NUM">30</field>
      </block>
    </value>
    <value name="DISTANCE">
      <block type="math_number">
        <field name="NUM">20</field>
      </block>
    </value>
  </block>
  <block type="robActions_motorDiff_curve">
    <value name="POWER_LEFT">
      <block type="math_number">
        <field name="NUM">10</field>
      </block>
    </value>
    <value name="POWER_RIGHT">
      <block type="math_number">
        <field name="NUM">30</field>
      </block>
    </value>
  </block>
```

4.4.2. Blocks Selection

The Robot Toolbox are the repository of blocks available for the Robot which consists of the blocks the end-user gets to see in the block list of the Web Application. At this stage, the blocks to be used for the ROSbot Pro 2 were selected.

The Toolboxes categories and Blocks used for the ROSbot 2 Pro are:

1. Action Toolbox Category
    i. Drive Block (Linear and Angular): This block is responsible for moving the robot in a forward, backward, and angular direction.
    ii. Obstacle avoidance Block: This is the block responsible for creating linear motion of the robot while using its lidar sensor to avoid obstacle.
    iii. Stop Block: This is the block responsible for generating a stop action for the robot.

These Toolbox Categories already exist, and in getting to display only the ones needed, those not needed were commented out. This procedure was same for other Toolboxes. The figure below shows the Action Toolbox XML and a front-end representation.

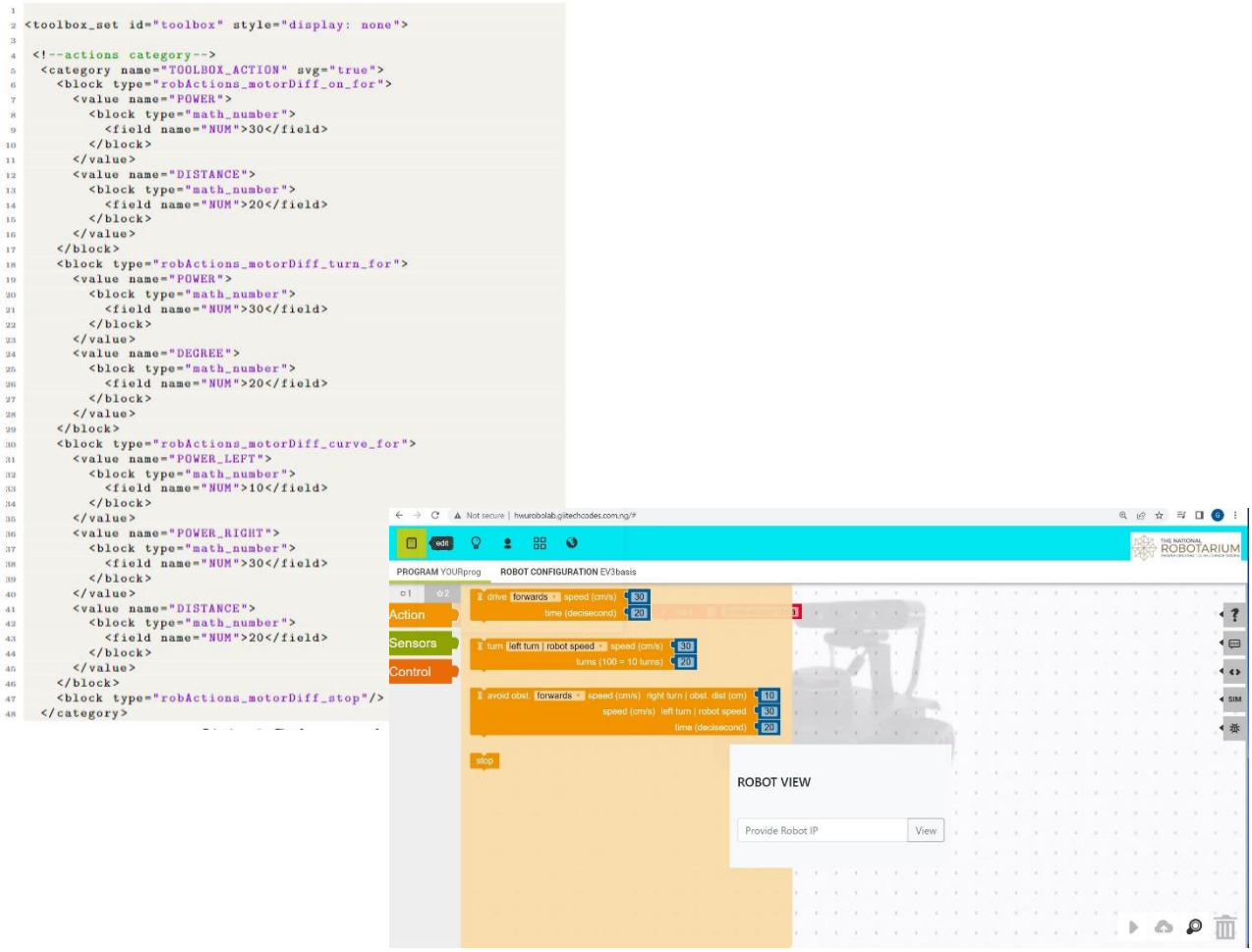


FIGURE 4.2: Action Toolbox xml and front-end representation

2. Sensor Toolbox
    i. Touch Sensor (Lidar Sensor): This sensor is responsible for responding to touches but was modified into a Lidar Sensor. This is not yet fully implemented in this study but will be in future studies.

The figure below shows the Sensor Toolbox XML and a front-end representation.

```
<category name="TOOLBOX_SENSOR" svg="true">
  <block type="robSensors_touch_getSample"/>
</category>
```

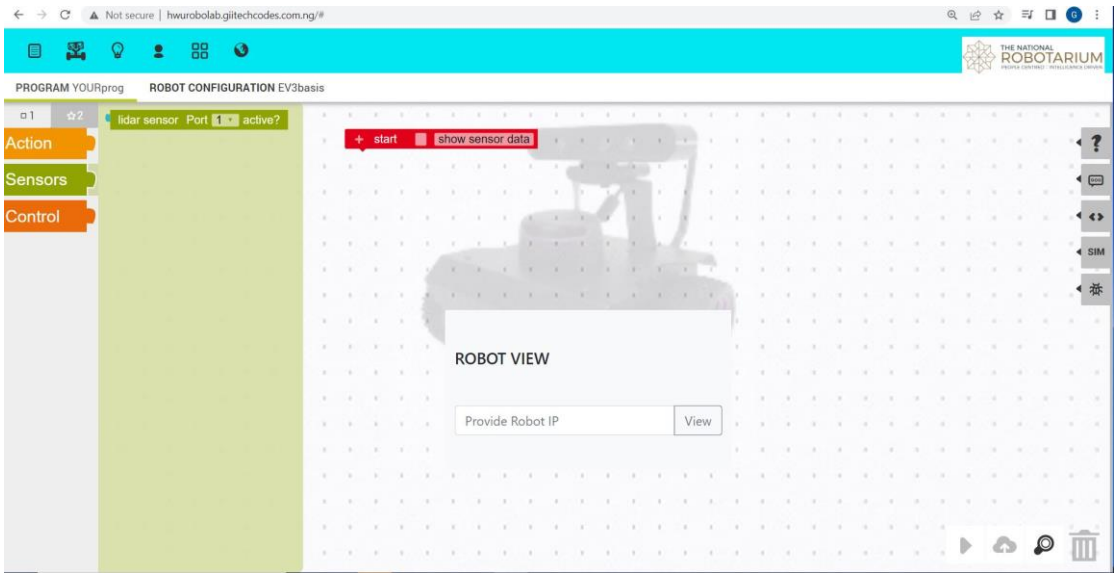


FIGURE 4.3: Sensor Toolbox xml and front-end representation

3. Control Toolbox
    i. If Block: This is the block responsible for initiating an “if” logic statement in the program.
    ii. If Else Block: This is the block responsible for initiating an “ifElse” logic statement in the program.
    iii. Loop forever: This is the block responsible for initiating a “while True” logic statement in the program.
    iv. Timed Loop: This is the block responsible for specifying the number of times a loop action will perform.

The figure below shows the Control Toolbox XML and a front-end representation.

```
<category name="TOOLBOX_CONTROL" svg="true">
  <block type="robControls_if"/>
  <block type="robControls_ifElse"/>
  <block type="robControls_loopForever"/>
  <block type="controls_repeat_ext">
    <value name="TIMES">
      <block type="math_number">
        <field name="NUM">10</field>
      </block>
    </value>
  </block>
</category>
```

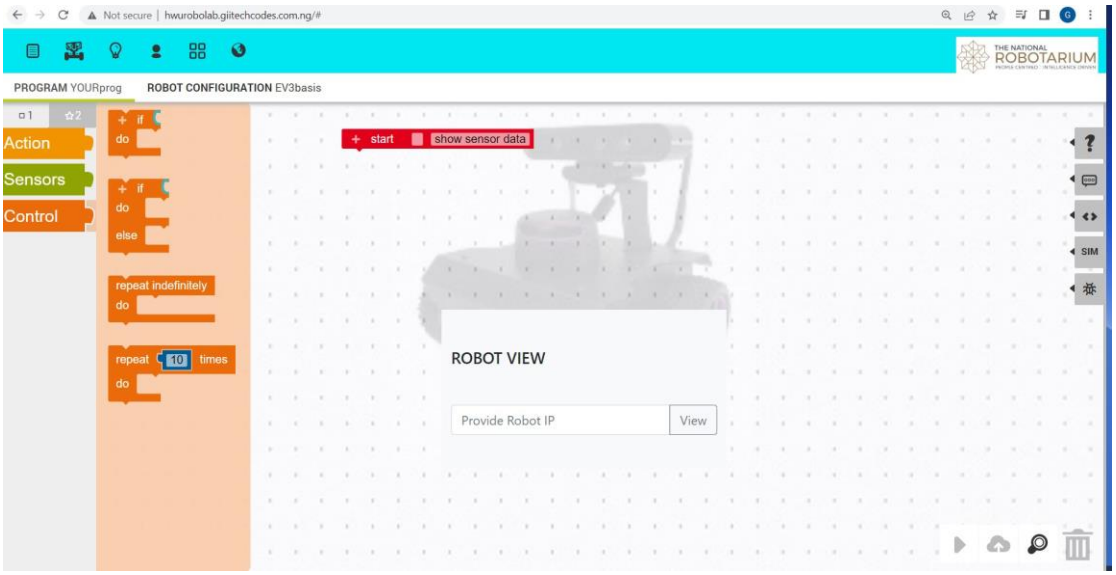


FIGURE 4.4: Control Toolbox xml and front-end representation

There are three sets of Toolboxes every plugin must have, which are the Configuration Toolbox, Beginner Toolbox, and Expert Toolbox. For the ROSbot 2 Pro, the Expert Toolbox wasn't used. To disable it, I commented out all its values. These are the path to the Toolboxes xml files:

1. Configuration Toolbox: Toolbox for Robot configuration (motors and sensors)

   /openroberta-lab/RobotEV3/src/main/resources/ev3.configuration.toolbox.xml

2. Beginner Toolbox: Toolbox for basic blocks

   /openroberta-lab/RobotEV3/src/main/resources/ev3.program.toolbox.beginner.xml

3. Expert Toolbox: Toolbox for advanced blocks. This wasn't used.

/openroberta-lab/RobotEV3/src/main/resources/ev3.program.toolbox.expert.xml

Table 4.1 below shows the Toolbox types, categories, and blocks used for the ROSbot 2 pro.

TOOLBOX TYPE	**CATEGORIES**	**BLOCKS**
Configuration	TOOLBOX_ACTION	robBrick_motor_big robBrick_motor_middle robBrick_actor
	TOOLBOX_SENSOR	robBrick_touch
Beginner	TOOLBOX_ACTION	robActions_motorDiff_on_for robActions_motorDiff_turn_for robActions_motroDiff_curve_for robActions_motroDiff_stop
	TOOLBOX_SENSORS	robSensors_touch_getSample
	TOOLBOX_CONTROL	robControls_if robControls_ifElse robControl_loopForever controls_repeat_ext

TABLE 4.1: Representation of Blocks used for the ROSbot 2 Pro

4.4.3. Robot Whitelisting

At this stage, the only robots to display on the end-user application were selected. Every other default system was commented out, only the ev3dev robot was added due to its use for the ROSbot 2 Pro.

When whitelisting is done, it chooses the Robot to be seen on the client end on the Robot Navigation menu drop down and the pop-up screen when the web application is loaded.

4.4.4. JAVA methods for Blocks

Every Plugin has a visitor Directory attributed to its resources. This is responsible for defining the functions to be used for specific blocks. The visitors give the ability to new functions and class to be added without changing the structure of the objects. The Code Generation follows this method as it retains the structure of the program the block configuration would have from an end-user activity and creates the flexibility of adding new Java Classes to the structure.

A Java default Super Class which defines all the method is created as an actor for all the Opensource Robot. This is significantly responsible for all the Object Types methods for each Class. Every Object Class has specific property which consist of class names, methods, and parameters.

The public interface is just an abstract class which contains all the methods attributed to the Class. A simple descriptor of a Public Class for the Motors "IDifferentialMotorVisitor" responsible for generating the functions for the action block can be seen below with all the methods associated with the class.

```
1
2  public interface IDifferentialMotorVisitor<V> extends IMotorVisitor
       <V> {
3
4      Void visitDriveAction(DriveAction<V> driveAction);
5
6      Void visitCurveAction(CurveAction<V> curveAction);
7
8      Void visitTurnAction(TurnAction<V> turnAction);
9
10     Void visitMotorDriveStopAction(MotorDriveStopAction<V>
       stopAction);
11 }
```

It is worth noting that “Void” is same as “V”.

The other abstract classes are: IActors4AutonomousDriveRobots, IAllActorsVisitor, IBluetoothVisitor, IDifferentialMotorVisitor, IDisplayVisitor, ILightVisitor, ISimpleSoundVisitor, ISoundVisitor, and ISpeechVisitor

The path to all the motor Abstract Classes is:

~/openroberta-lab/OpenRobertaRobot/src/main/java/de/fhg/iais/roberta/visitor/hardware

4.4.5. Python Generation for ROS

By default, a part of the ev3dev code generation is in Python but not for ROS. In-order to generate the Python Script for each block, these simple steps were followed:

1. **Designing a sudo code model for function calls and passing parameters**:
   The structure and pipeline for running the programs follows a method where a client python file is created from each block which imports, calls the function, and passes parameters to specific variables in a parent (main) python file already in the script directory of the robot. Here’s an example of the client python code:

```
1
2 #!/bin/usr/env
3 import main
4
5 main.run(10, 20)
```

   In the above example, the first action is to import the main.py file which is already which has all functions already designed for certain actions. Then passes two parameters “10 and 20” into the run function defined in the main.py file.

2. **Inserting the models into the JAVA method**:
   After designing the model above, the next step was to modify the python code generator to generate the codes I needed for various blocks. To do this, the first step was to know where the classes for each robot action was. To understand this better, we will look at how the generator for the “DriveAction” class was changed.

```
1
2       @Override
3       public Void visitDriveAction(DriveAction<Void> driveAction) {
4
5           if ( isActorOnPort(this.brickConfiguration.
        getFirstMotorPort(SC.LEFT)) && isActorOnPort(this.
        brickConfiguration.getFirstMotorPort(SC.RIGHT))){
6               decrIndentation();
7               nlIndent();
8               this.sb.append("#!/usr/bin/python");
9               nlIndent();
10              nlIndent();
11              this.sb.append("import rospy");
12              nlIndent();
13              this.sb.append("import main");
14              nlIndent();
15              this.sb.append("rospy.init_node('Godfrey_Project')");
16              nlIndent();
17              nlIndent();
18
19              boolean isDuration = driveAction.getParam().getDuration
        () != null;
20              String methodName = isDuration ? "hal.driveDistance(" :
         "hal.regulatedDrive(";
21              this.sb.append("main.run(");
22              driveAction.getParam().getSpeed().accept(this);
23              if ( isDuration ) {
24                  this.sb.append(", ");
25                  driveAction.getParam().getDuration().getValue().
        accept(this);
26              }
27              this.sb.append(")");
28
29          }
30          return null;
31      }
```

The script above is responsible for the class “visitDriveAction” and the method “driveAction”. Line 5 checks if the robot motor is active, meaning if the block has been selected. Other important syntax to note are:

i. this.sb.append(“”) : For every text appended, this Java Syntax was used. For example, I want to append “import rospy” as seen in line 11, it becomes **this.sb.append(“import rospy”).**

ii. nlIndent : This is used to start from a new line

iii. incrIndentation ; This is used to add an indentation

iv. decrIndentation : This is used to remove an indentation

Line 22 and 25 are responsible for getting the speed and duration values from the block respectively.

This method was used to add for all other blocks selected, except the Obstacle avoidance block in which the method used is seen below.

3. **Sensor Model for ROS:** The model designed is a simple ROS node which use the "/scan" topic to subscribe to the "/cmd_vel" topic using the *LaserScan* message. A simple callback function is written which consists of one threshold variable value which is passed from the block and compares these values with the ranges at three angles "0" and triggers a twist message which makes the ROSbot 2 Pro stop at an obstacle within the range and finds the best path to continue its task. Appendix C contains the used ROS models.

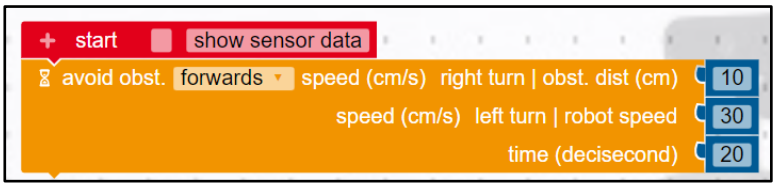


FIGURE 4.5: Obstacle avoidance Block

Here's a simple architecture showing the relation between the classes and subclasses:

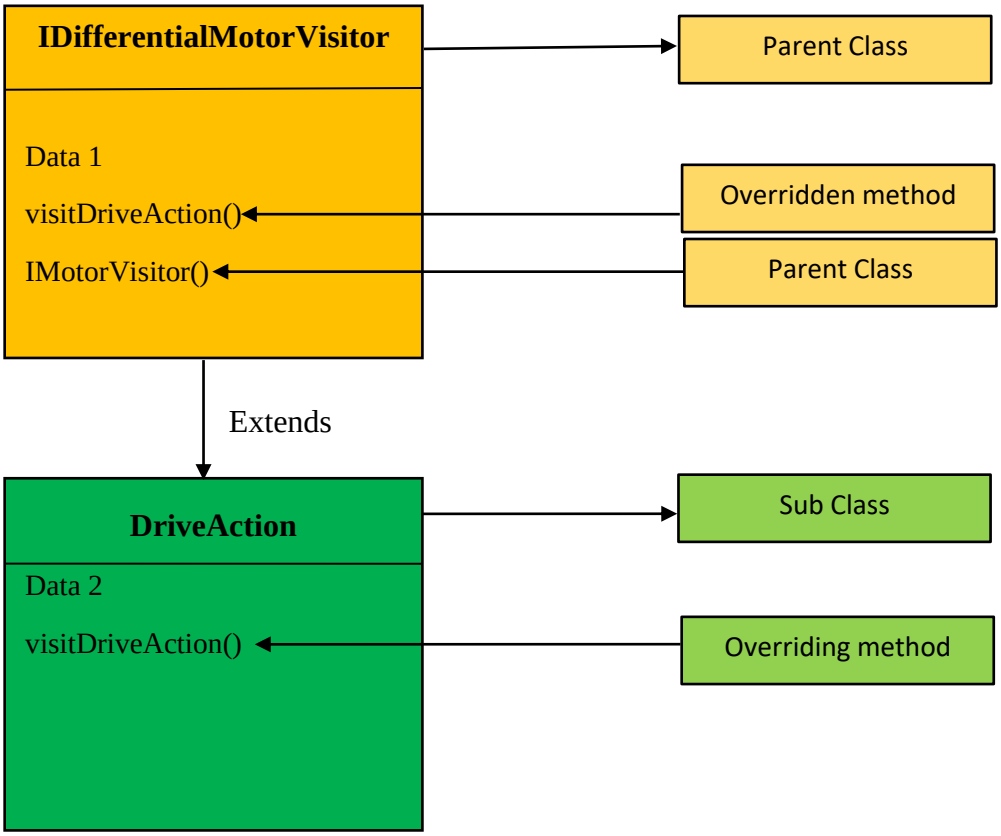


FIGURE 4.6: Architecture showing the relationship between Parent Class and a subclass

4.4.6. User Interface modification of Static Resources

Due to publishing laws and regulations by the owners of the open-source Platform, certain user interface elements needed to be updated such as the Logo, Emails, Brand name, and others which can be seen from the Open-Source Website.

- **New Robot addition to menu list**

Adding the ROSbot 2 Pro was just a simple method of updating the names and manufacturers information from the properties file as seen in (*section 4.4.1*) above.

- **Brand name and Logo update**

These were updated from the index.html file just inside the OpenRobertaServer directory.

4.5. API protocol for running generated program

The Open Roberta framework provides three ways to run the generated programs from the blocks on the Robots which are Bluetooth, USB, and Wi-Fi but not working. Now and at a larger Robotics ecosystem, there are some limitations to these methods. A very popular method of program transfer is via a network protocol, but this throws a limitation which is seen in forcing the Robot and User Application to be on the same network before data transfer is possible.

This research proposes a new and efficient way for data transfer from the application and the Robot by using a built-out API for the communication which enables Data exchange between the Robot and a non-ROS application which should not necessarily be on the same network.

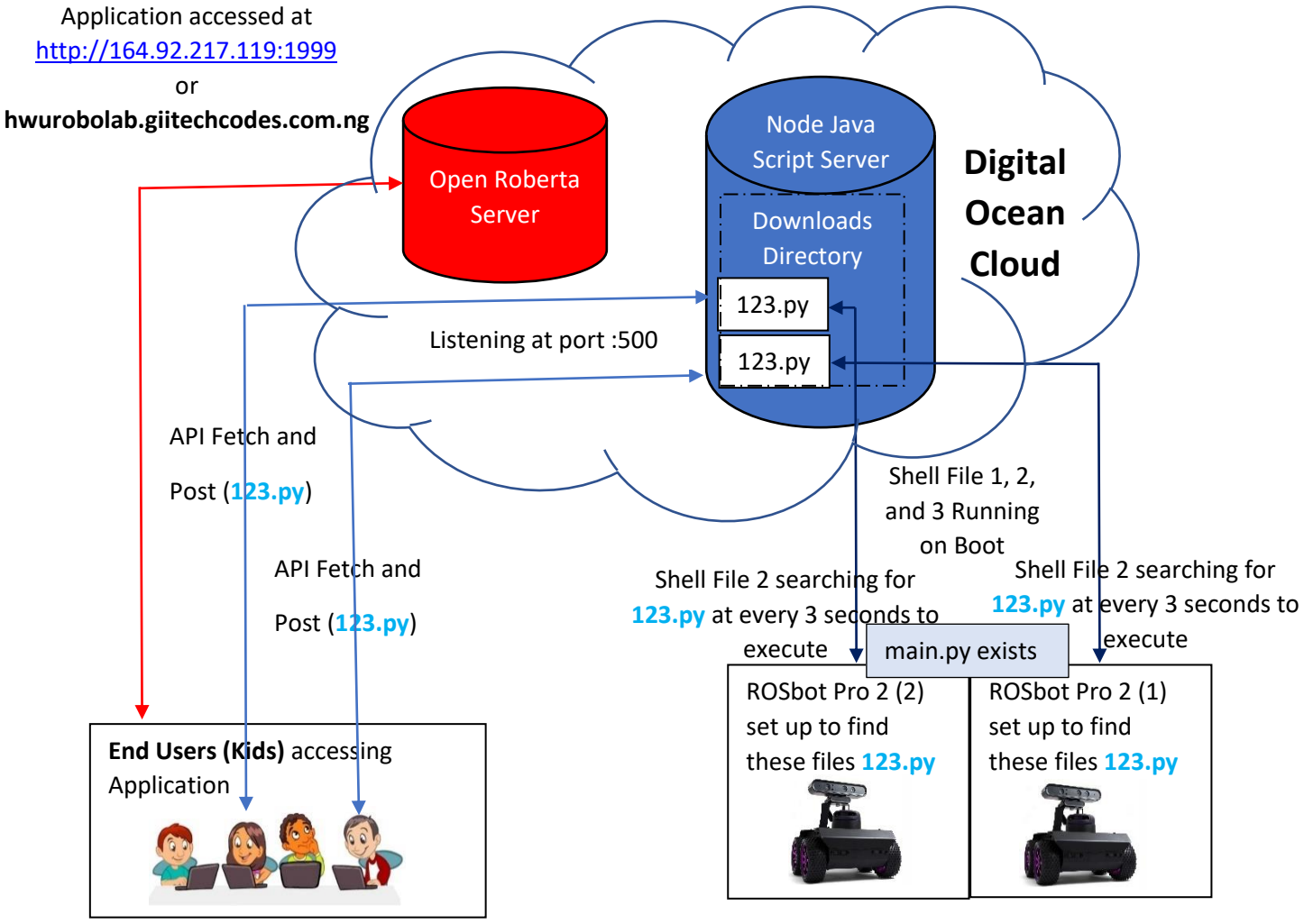


FIGURE 4.7: Full System Architecture showing the ROSbot 2 Pro communication with the Servers and Client

At the cloud application, a fetch and post request is implemented which specifies the IP of the cloud application for authorization, and on every request triggered by the User Interface button, a POST data request occurs by fetching the generated program and posting it into the specified directory on the node is server. Here is the jQuery Script responsible for performing the POST request of the Program Generated.

```
$('#codeDownload').onWrap('click', function (event) {
var filename = GUISTATE_C.getProgramName() + '.' + GUISTATE_C.
    getSourceCodeFileExtension();
            const postData = async ({ file, fileName }) => {
                //server API configuration
                fetch("http://164.92.217.119:5001/file-saver", {
                  method: "POST",
                  headers: {
                    "Content-Type": "application/json",
                  },
                  body: JSON.stringify({
                    file,
                    fileName,
                  }),
                })
                  .then((res) => console.log(res, "response"))
                  .catch((err) => console.log(err, "error"));
              };
              postData({fileName: filename, file:`${GUISTATE_C.
    getProgramSource()}`})
            console.log(GUISTATE_C.getProgramSource());

            MSG.displayMessage('MENU_MESSAGE_DOWNLOAD', 'TOAST',
    filename);
        }, 'codeDownload clicked');
```

The API has 3 routes, “index” route, “file-saver” route, and “not found or *” route. The “file-saver” route performs the file saving process. It uses a POST method, which receives the “file” and the "filename" as its request body. “file-saver” route has a controller (a functional callback) whose function is to automate the file saving process once it receives the “file” and the “filename” from the request body, which implies creating a “downloads” folder if it doesn't already exist and writing files (from the generated code) into the "downloads" folder as required.

The frontend part of the application communicates with the API (RESTful) by sending a POST request using the fetch API, with a payload of filename, and file. The Diagram below shows the User Interface for Refreshing, inserting Robot ID, and Run.

FIGURE 4.8: User interface showing the Robot ID Field and RUN Button

On the Robot, there's a Shell File always running which searches for a program with a specific ID at every 3 seconds. This is done by using the “***rsync***” command which moves the generated program from the node js server after “***ssh***” and runs it inside the Robot. Here's the shell file command structure:

```
rsync –remove-source-files root@<server_ip>:/file_path_on_server
/home/path_to_script_folder_in_rosbot
```

The limitations of initial architecture were seen in program identification for specific ROSbot 2 Pro Robot. The frontend field accepts the unique Robots Identification and appends it to the file name which will be uploaded by the Client. Before this happens, the Robot is already configured to accept only files with the unique ID from the Server when it searches. A simple architecture of this structure is seen below.

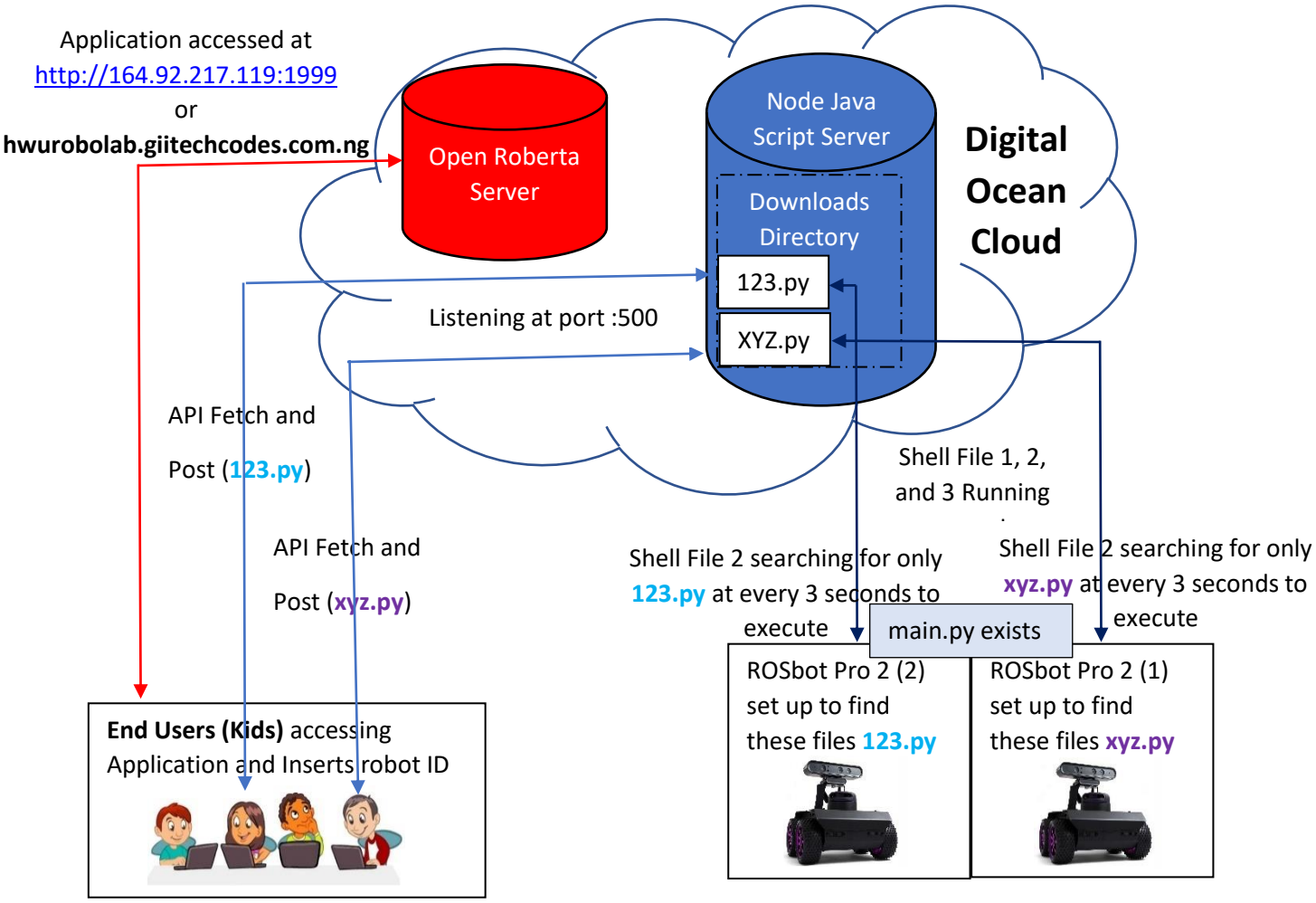


FIGURE 4.9: Improved Full System Architecture showing the ROSbot 2 Pro communication with the Servers and Client

With respect to the above modified architecture, the diagram below is a simple flow chart showing the execution flow.

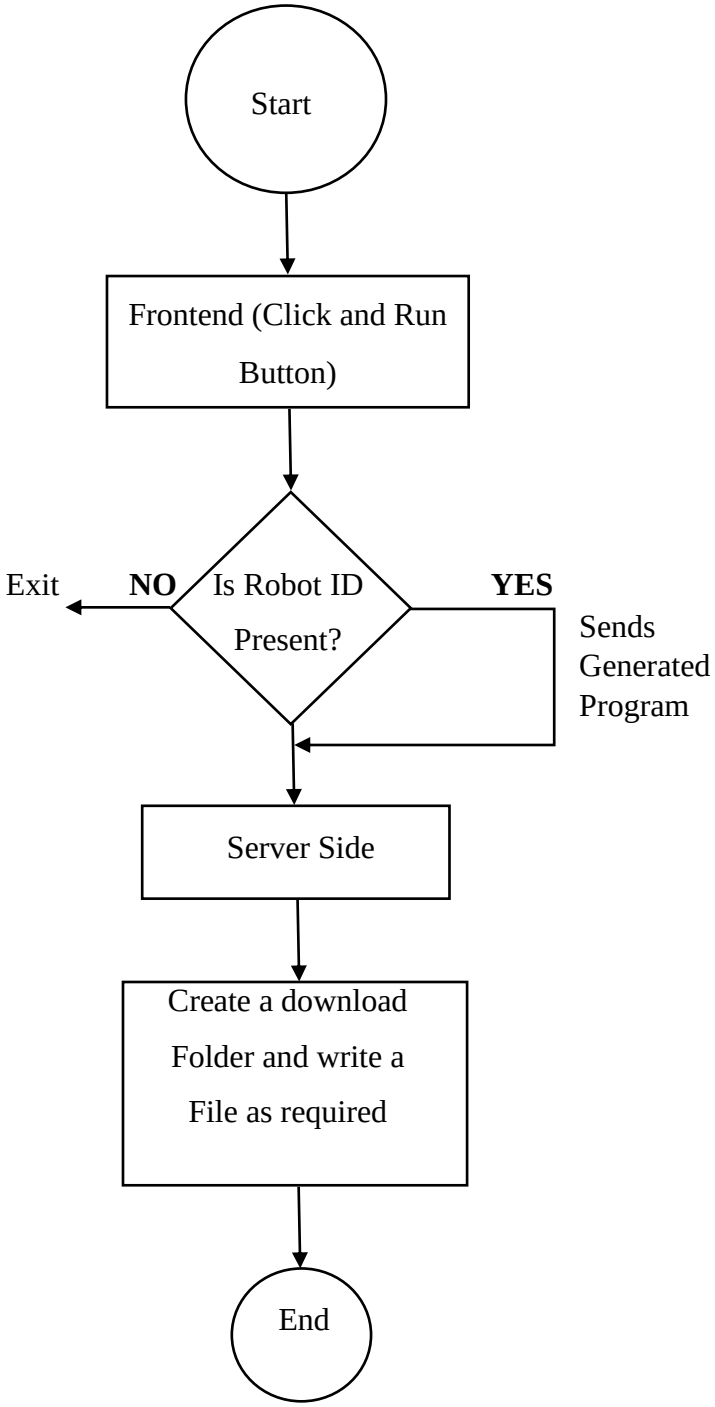


FIGURE 4.10: Flowchart Diagram based on built out API on modified architecture

4.6. ROSbot 2 Pro Camera Stream

This is also an addition to the Project. This was built by simply writing a Java Script program which has a static iframe tag with an input form which accepts the IP of the Robot and automatically passes that argument over HTTP and generates a display of the IP/URL just above the input field. It is worth nothing that for this to happen, the Robot and Computer must be on the same network as that is what servers as a protocol and pathway for the camera stream. You can see the JavaScript handler below.

```
1  (() => {
2    const isValidUrl = (urlString) => {
3      try {
4        return Boolean(new URL(urlString));
5      } catch (e) {
6        return false;
7      }
8    };
9
10   const root = document.getElementById("iFrameRoot");
11
12   const generateIFrame = (url) => {
13     const iFrame = document.createElement("iframe");
14
15     iFrame.src = `${url}`;
16     iFrame.width = "100%";
17     iFrame.height = "100%";
18     iFrame.id = "robotiFrame";
19
20     const frame = document.getElementById("robotiFrame");
21
22     if (!!frame) root.removeChild(frame);
23     root.appendChild(iFrame);
24   };
25
26   const robotIP = document.getElementById("robotIP");
27
28   const viewHandler = (e) => {
29     e?.preventDefault();
30
31     if (!robotIP.value) return;
32
33     if (!isValidUrl(robotIP.value)) return;
34     root.classList.add("robotView", "border", "border-light", "
       shadow");
35     generateIFrame(robotIP.value);
36   };
```

Line *2* to *6* above verifies the validity of the URL based on if it is served over HTTP or HTTPs. This is important when deploying this repository on your cloud server. If this is deployed to on a HTTP Server then all IP accepted on the front-end must be HTTP, but if it is deployed on a HTTPs server then all IP must be https. For a typical ROS camera topic, it is server over http, so it is advisable to deploy on a HTTP domain.

Line *10* creates an ID for the front-end input field target. Line *12* to *24* creates the design for the iframe. Line *28* to *36* is a checker which verifies the IP inputted and passes an argument into the iframe.

```
37
38    const debounce = (fn, delay) => {
39      let timer;
40
41      return function () {
42        const context = this;
43        clearTimeout(timer);
44
45        timer = setTimeout(() => fn.apply(context), delay);
46      };
47    };
48
49    const viewBtn = document.getElementById("robotIPBtn");
50    viewBtn.addEventListener("click", viewHandler);
51
52    const delay = 800;
53    robotIP.addEventListener("input", debounce(viewHandler, delay));
54  })();
```

This creates an on button click method which triggers the view of the IP. Although, this may not be necessary as the display comes up automatically once the IP is verified successfully.

In setting this up on the ROSbot 2 Pro, these are the methods followed:

- Cloning the package: The web video package needs to be cloned into the src directory of your catkin_ws. Here's the git link to clone:

  https://github.com/RobotWebTools/web_video_server.git

- Installing the web server: This was done by using this command.

  $sudo apt-get install ros-melodic-web-video-server

After installing all the required resource above, the package was built and sourced using "catkin_make" although you can use "catkin build".

The server is started by running this command:

$rosrun web_video_server web_video_server

ROS core must be running before this can be activated as it will needed to know the camera topic is available.

4.6.1. Adding camera view to Web Application

Once this is deployed on any web server, the URL can be embedded as an iframe in the application. For this project, this was deployed to:

http://robotip.giitechcodes.com.ng/robot-view/

In the Web Application, this was added as an iframe to a section of the index.html file which is responsible for the frontend of the application. It was added as:

```
<iframe styele="margin-left:-1100px;" height="315" width="560"
    title="" src="http://robotip.giitechcodes.com.ng/robot-view/"
    frameborder="0"></iframe>
```

This is what the end user gets to see after this has been implemented:

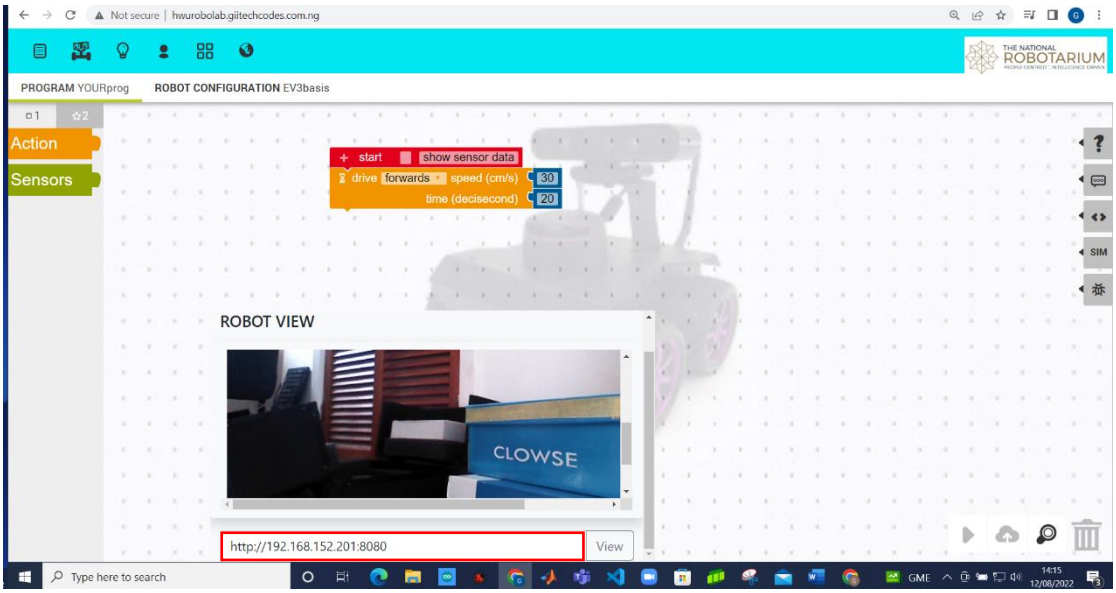


FIGURE 4.11: Camera view on Deployed Web Application

The red box mark above indicates where the Robot IP should be inputted ensuring that your Robot and Computer you are accessing the application from are on the same network.

One easy was used to get the Robot IP without connecting to an external screen is using a Command Terminal Application on your mobile phone, and inputting this command:

$ip neigh

4.7. Deployment with Digital Ocean

To make the application accessible by all users from any location, it was deployed and published on Digital Ocean Cloud Service.

4.7.1. Droplets

The Droplets of Digital Ocean are Linux Virtual Machines running on the Cloud on a Virtual Hardware. On first usage, users are to configure the Droplets.

In this project, this is the specification of the Droplet used:

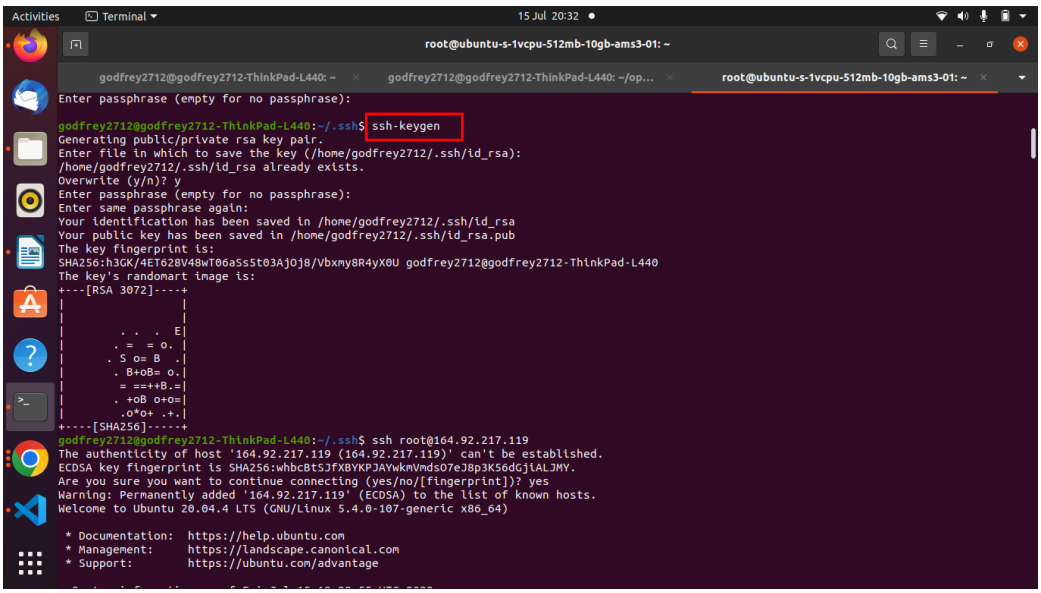
Type	CPU Type	vCPUs	Memory	SSD	Transfer	Price
Basic	Shared CPU	1 vCPU	512 MB	10 GB	0.5 TB	$4/mo $0.006/hr

FIGURE 4.12: Droplet specification for Development

Although this specification might be ideal for Development but not for Production due to the plan which is just for CPU and RAM only. The limitations are captured in detail in Chapter 6 below.

4.7.2. Droplet setup

The first step in setting up your droplet is creating an SSH Key from the computer you will be using to perform read and write functions from.



Once the ssh key is generated, it is then copied into the New SSH Key menu on the Droplet. Here's an illustration:

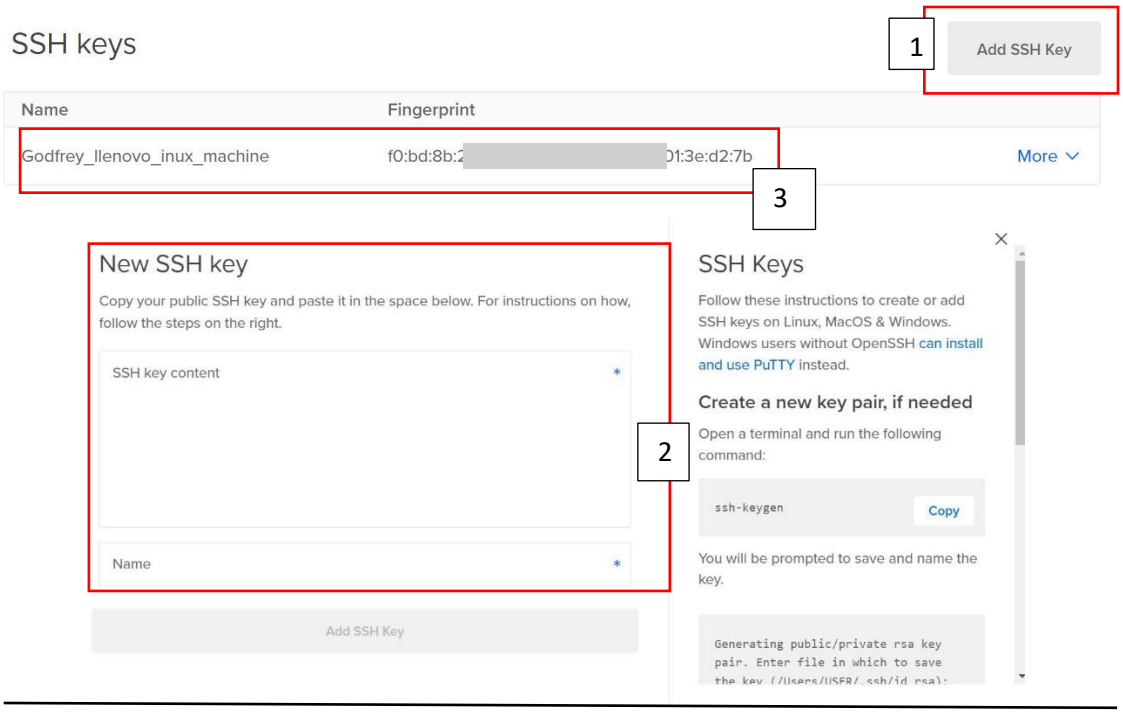


FIGURE 4.13: Adding the SSH Key to Digital Ocean

After 1 is triggered above, the new SSH Key is then inputted into 2 above with a chosen name identifier. 3 above is an indication that the SSH Key has been accepted.

At the settings page, all the network details for your droplets are generated. See below for the details for the Project:

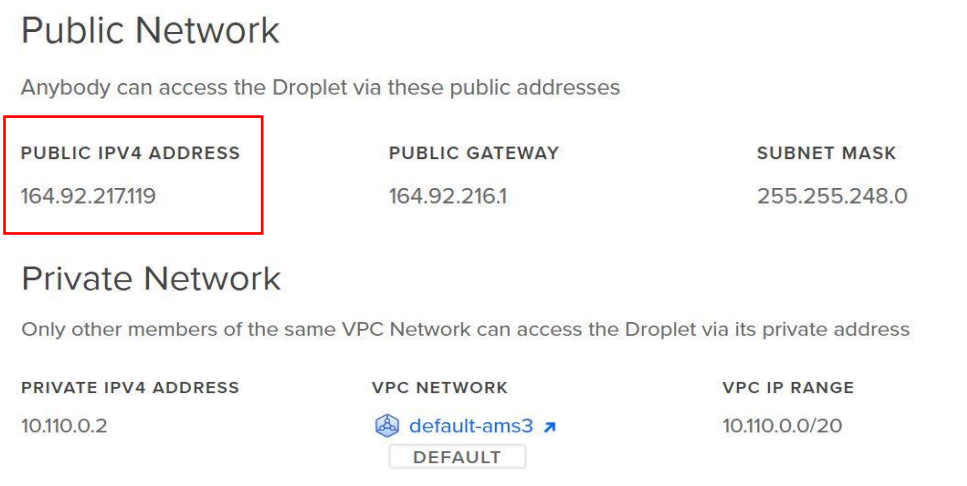


FIGURE 4.14: Public and Private IP of Cloud Server

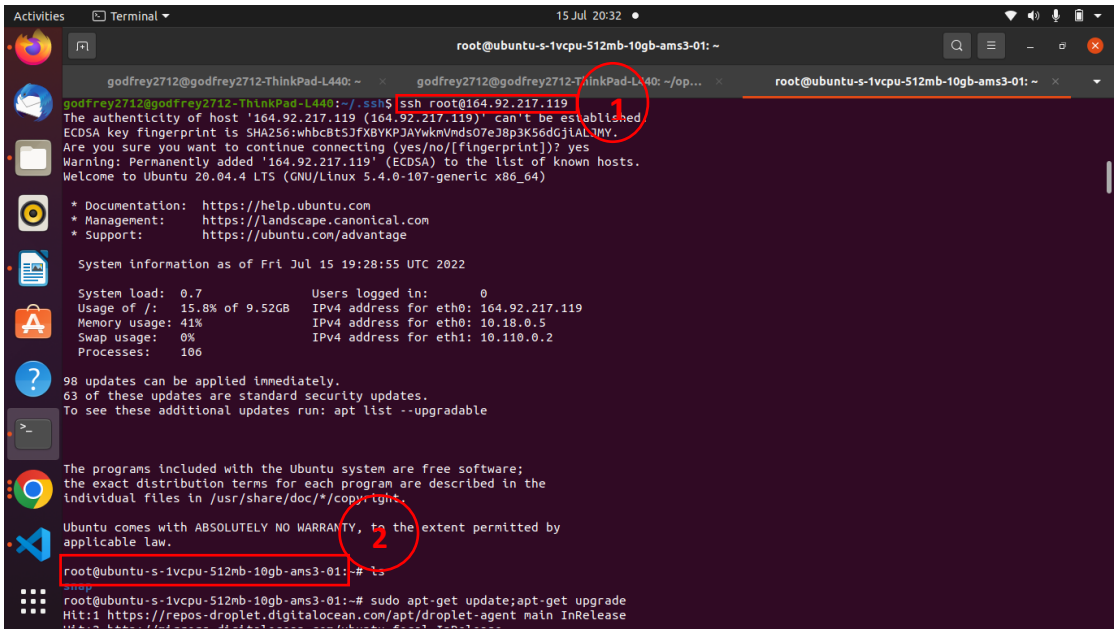
The Public IPV4 Address above is the address end users can use to access the web application. It is worth noting that the framework embedded server is started at a port of 1999, so to access the application users must enter the full address below once the server is started.

164.92.217.119:19992

4.8. Publishing from localhost

Once this is completed, you are all set to begin transferring your files from localhost to your cloud server. To copy the Directory into the Cloud Server, these steps should be followed:

1. Test your SSH Connection: During creation of the SSH Key, a user password for a default user “root” was created. Ensure you can access your server by SSHing into the server with “root” and the server IPv4 address.



1 above is the command to SSH into the server with your chosen password. 2 shows successful SSHing.

2. After testing the SSH, create a package inside the root directory with same name as the directory name you want to copy, and exit the SSH by using the command “exit”. To copy all the files and folders recursively, first get into the Directory which is openroberta-lab using “cd” from command line. Then using secure

copy command, you can copy all the files and folders into the Directory in the Cloud Server. Here's a command line example:

```
$scp -r * root@164.92.217.119:/root/openroberta-lab
```

The command above will copy all files and directory from your system path into the path specified on the Cloud Server.

4.9. Starting the Server

OpenRoberta Framework consists of an embedded server which must be started before the application can be accessed by an end user. In starting the local server, we first need to enter into the path of the openroberta-lab application, and then run this command "./ora.sh start-from-git". Same process occurs with starting the server in the cloud.

The first thing to do is to SSH into the Cloud Server from a computer which have been setup to connect with the droplet. After which, get into the openroberta-lab directory on your cloud server and type the command below:

```
./ora.sh start-godfrey-project
```

Once the server is started, end-user can access the modified web application using the Droplet Public IP Address "*164.92.217.119:19992*"

4.10. ROSbot 2 Pro Configuration

There are certain pre-configurations which must happen prior to the platform use for the ROSbot 2 Pro. These configurations are stored in three shell files which is seen in Appendix D . Below are the shell files and their functions.

4.10.1. Start Nodes Shell for ROSbot 2 Pro

To connect the platform to the ROSbot 2 Pro, certain nodes must be running at every system boot instance. These relevant nodes have been cascaded into the base.launch file in the robot bring_up directory. The ROSbot 2 Pro packages can be cloned from: https://github.com/husarion/rosbot_description

The base launch file command is embedded in a Shell File called start_robot.sh in the ROSbot 2 Pro Home Directory. This is the shell file commands:

```
#!/bin/bash
cd husarion_ws
roslaunch husarion_bring_up base.launch
```

This shell is running automatically once the ROSbot 2 Pro is turned on by including it in the *.bashrc* file and setting the command shell to launch at system boot.

4.10.2. Start program_wait Shell for ROSbot 2 Pro

This Shell file is responsible for searching the node js server every 3 seconds for the assigned program based on the Robot ID.

4.10.3. Start web_video_server Shell for ROSbot 2 Pro

This Shell file is responsible starting the web video server in the ROSbot 2 Pro.

4.10.4. Python main file for ROSbot 2 Pro

This is the python file which contains all the functions to be called from the client python files generated from the blocks.

4.11. Workflow

For first time users, here's a simple workflow on using the cloud-based application with a minimum requirement of a Computer with a Web Browser connected to the Internet.

4.11.1. Application and Node JS Server

The Application Server and Node JS Server must be running before the web application and the file saver is active. This is done by the Application Administrator.

4.11.2. ROSbot 2 Pro

Ensure that all necessary shell files and Python File are on the ROSbot 2 Pro, the necessary shell files can be seen in (*Appendix D*). Also, ensure you have performed a pre setup by including the shell files in the bashrc file and setting up the terminal to launch at system boot. This is also done by the Application Administrator.

4.11.3. End-user Computer

Ensure you have internet connection, then visit "hwurobolab.giitechcodes.com.ng".
To get the camera feed, your computer and Robot need to be on the same network.
Here's a summary Table of the tasks and the methods each user of the Web Application should perform.

Users	Tasks List	Methods
Developer	Cloning the Web Application	Appendix A
	Cloning the Node JS Server	Appendix A
	Integrating camera feed	Section 4.6.1
	Deployment to Cloud	Sections 4.7 and 4.8
	ROSbot Configuration	Appendix D
Administrator	Starting the Web Application Server	Section 4.9 and Appendix A
	Starting the Node JS Server	Appendix A
Teachers	Turning on the ROSbot	Figure 1.1 shows the ON and OFF button
	Knowing the IDs of the ROSbot Robots	Appendix D
Student	Accessing web application	Appendix F
	How to use available blocks?	Appendix F

TABLE 4.2: Summary tasks and methods for different users

5. HWU Robolab Platform Evaluation in Outreach Scenario

The User Experience of any Software Product is as relevant as having a fully functional Software System. To understand the target user experiences and behaviours, a simple user test experiment study is designed using 10 ROSbot Pro 2 which would be made available at the scene of the experiment. The primary objectives of this designed experiment are:

1. To evaluate the basic functions of the HWU Robolab Platform
2. To evaluate the user friendliness and intuitiveness of the platform in terms of ease to use with the Blockly Blocks
3. To understand how the servers responded to users' activities
4. To understand the response time of the ROSbot 2 Pro with respect to data transfer
5. To understand and learn from the experiences and feedbacks on how the Platform can best be used in outreach activities effectively, and further improvements

As an addition, this experiment and evaluation model will give an additional goal in creating a blueprint and guide on how the University and other Educational Institutions can achieve best result from using this platform for their outreach activities. A critical consideration during the design and development of this platform is the fact that it must be cloud based, hence an aspect of this experiment is also to understand how remote users respond to its use from a remote location in controlling the ROSbot 2 Pro preconfigured and stationed in the experiment environment.

5.1. Experiment setup

The robot which will be used to conduct this study is the ROSbot 2 Pro. The ROSbot 2 Pro comes with an inbuilt Linux Operating System with an Ubuntu version. It also comes in a box consisting of its Hardware components such as the Power Pack, antenna, and LAN connector. The ROSbot 2 Pro used in this study will be preconfigured using the two shell files above, and setup to run the two shell files on Robot Boot (see

Appendix D). It is also essential to connect the ROSbot 2 Pro to a stable Wi-Fi network to get an automatic connection on Robot Boot.

For a Class of above 10 Students, 5 Robots can be assigned to the Class with a Total number of 2 Classes able to handle a session at a Time. For a Class less than 10 Students, 2 Robots can be assigned to the Class with a Total number of 5 Classes able to handle a session at a Time.

In a Classroom of above 20 Students, a common Room would be used as the classroom would not be ideal to effectively carry out the experimental study.

For this study to be effective a minimum free space dimension of at least (5m*5m*5m*5m) is required in-order to carry out maximum activities with the Robots.

- **First Setup**

In the first setup step, it is ensured that the number of students and number of Robots are rightly placed in the classroom with at least a Class Teacher and a System Administrator. Here's a setup Diagram of a setting with 15 School Children.

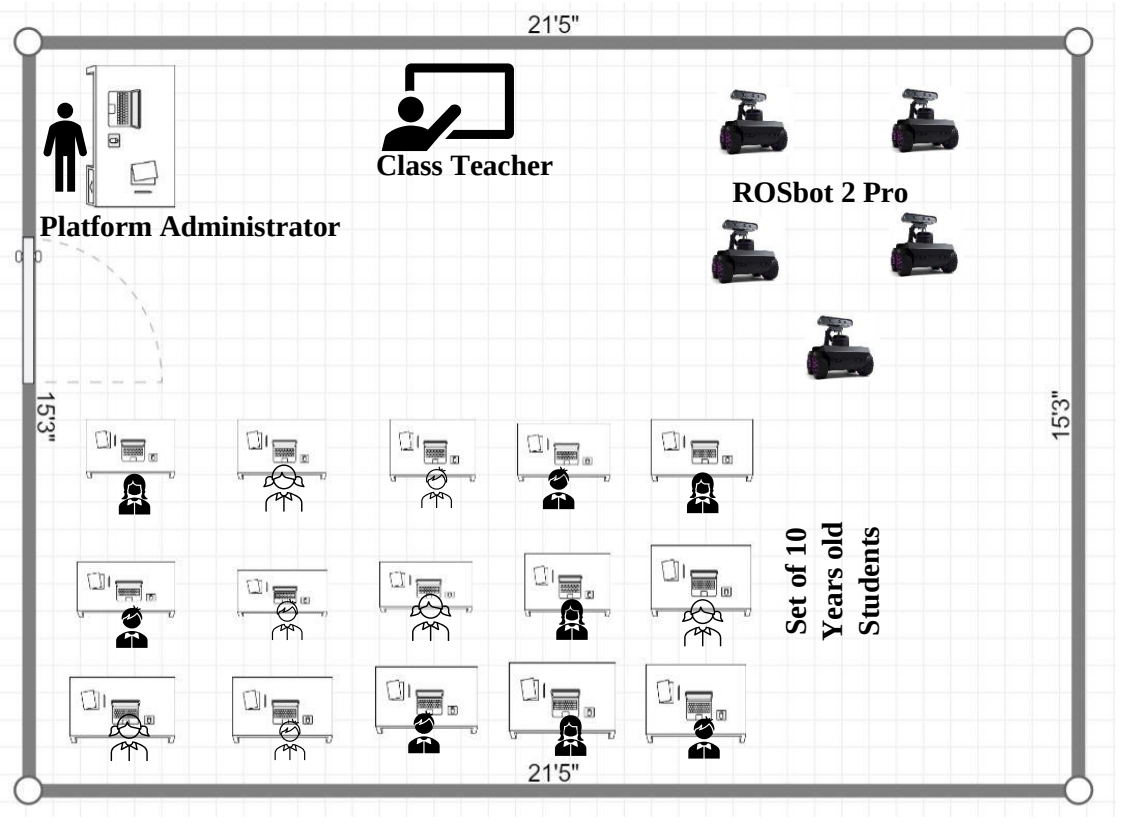


FIGURE 5.1: Setup Design in an Experimental Classroom

In the setup above we can see a small class of 10 years old students sitting in-front of their computers while facing the direction of the class Teacher who must have had a one-one session with the administrator on necessary guidelines.

The ROSbot 2 Pro is placed just in front of the class with an administrator in same class with the student. The major task of the System Administrator is to turn on the two servers in Digital Ocean (see system architecture in Appendix A) with the application started on the IP (164.92.217.119:1999) so students can access the application running on the Cloud.

- **Second Setup**

For the second setup, there were additional obstacles to the environment which can be seen in the Diagram below. Two Goal Positions is also marked out.

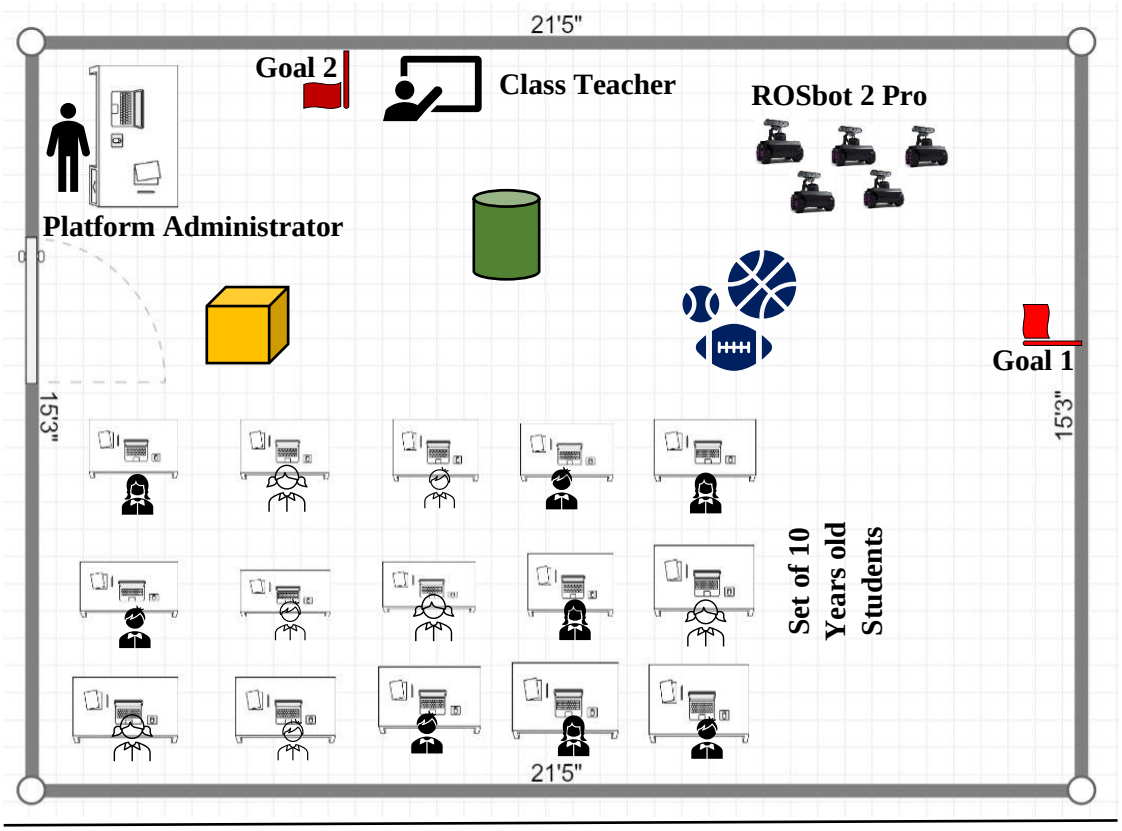


FIGURE 5.2: Setup Design in an Experimental Classroom with Obstacles

5.2. Participants

In this proposed model, the Experimental Study participants are falls within the Groups. These are the nature of the participants:

- Primary School Children of the ages10 years
- Primary School STEM Teacher
- Technical Test Participants (Alpha Testers)

The first sets of participants are Primary School Students within the ages of 10 years. There will be no split up in this pilot study as they would consist of a mix of both Technically inclined and non-technically inclined Students.

The second group would consist of the Teachers who teach STEM Subjects. For this model proposed, the Teachers would be always in the room during the experimental study to get quality data during the analysis after the experiments are over. This research participants would also be critical in understanding how Teachers can effectively utilize the platform in Remote Teaching in addressing “Schule 4.0. – jetzt wird’s digital” [83].

The last Group was specifically setup to conduct a system load analysis of the server, understand how remote users felt about controlling a mobile robot from a remote location, and see how remote teachers can be included in the model been developed based on “Schule 4.0. – jetzt wird’s digital” [83].

5.3. Experiment design

This pilot study will be conducted in one week. A total of 3 sessions is to be conducted for each group. Each session will be 20 to 30 minutes long excluding the questionaries.

The First set of participants would be in a classroom environment with initial setup as in Figure 5.1 with at least one system administrator and one classroom teacher. Two Days to the day of the experiment with the first Group, a link will be sent out to the Teachers which contains the (i) participants requirements (ii) a video clip walking through the features of the web application (iii) tasks explanations and purpose of study (iv) a how to use manual of the HWU Robolab.

The Teachers are expected to have a session with the students with the aim of conveying the experimental requirements and create an enthusiastic expectation in the 10 years old Student.

The First Session which involves the School Students will be split in two stages.

- Session 1_Stage 1

Prior to the start of the tasks the students will be asked 5 collective Questions to access their General attitude towards Robotics. The answers would come by hands raising or vocal responses.

The tasks for this Stage are as follows.

1. Five Students assigned one ROSbot 2 Pro each
2. Identify the name on the ROSbot 2 Pro chosen as this is the ID for the Robot
3. Visit the IP Address on your computer from any web browser
4. Go through the toolboxes to know which blocks are available for use
5. Understand what each block does
6. Understand how the numbers affect the actions of the block
7. Open and understand the simulator interface
8. Create a basic move forward and backward motion for the ROSbot 2 Pro and see it perform on the simulator
9. Run generated program on the ROSbot 2 Pro and see the action
10. Regulate the speed of the Robot

This task will be carried out with a Group of 5 Students each with a maximum of 20 minutes for each Group, while other Students stay around to watch.

- Session 1_Stage 2

At this stage of the experiment, the complexity of the task is increased. This would involve getting the Robot to the Goal Points at the fastest possible time. This will be conducted in the environment setup in Fig 5.2.

The tasks are as follows:

1. Perform all activities above again
2. Include the obstacle avoidance block and understand how the Robot responds to the block

3. Identification of Goal points based on assigning by the Teacher
4. Create programs with all the available blocks to get the Robots to your assigned Goal points within the shortest period

For the Second Study Group which are the teachers, Session 2 involves a simple strategy of getting them involved in all the processes from setup to actual demonstrations. To add to this basic task, the teachers will be expected to take one day to carry out a simple teaching of basic robot movement using the platform. This will be in two methods, Physically and Virtually.

The last group which are the Group for the Alpha and Technical testing of the application are HWU University Staff. This consist of a set of 3 staff carrying out the following tasks:

1. Functional tests
2. Server tests
3. Unit tests
4. User experience test

The essence of the tasks for the last group is to understand the technical loop holes the web application currently has based on their feedbacks and create a strategy on how to improve on them in the early future. This test would be carried out without the need of an ethics approval as the participants are staff of the University.

In carrying out these tasks for an assessment, the Robotics Assisted Living Testbed was used as the experimental test environment. Each of the three participants were given certain tasks to perform. Section 5.5 covers this in details and shows the result from the experiment.

5.4. Evaluation Strategy

This evaluation strategy can be used all through the stages of experimental study implementation and execution of this platform for educational outreach purposes. As the outreach programs are being developed, formative evaluations are conducted [74] [2]. The aim of the evaluation is to access the impacts of previous outreach activities which will guide in future outreaches improvements. Based on [74], summative evaluation is critical after each outreach activities in properly accessing the impacts on the target users, and understanding how their attitudes changed after participating in the program.

Through outreach activities, two strategies should be followed which are either the formative or the summative strategy. The choice of strategy is dependent on the data collection patterns in terms of the time this data is to be collected and for what group which is critical as an evaluation method. Whereas for a measurement method is particularly concerned about the means in which the data will be gotten from the end users, this can be in the form of questionnaires and interviews.

5.4.1. Formative strategy

This is seen in analysing previous outreach activities and designing a better model for future outreaches based on this model. Based on Boehm (1986) and Conan (2010), initial materials are developed and tested with a small number of participants in which the analysis of the data from the evaluation points out the changes to be made. Figure 5.3 below shows a spiral process based on [10] for program improvements.

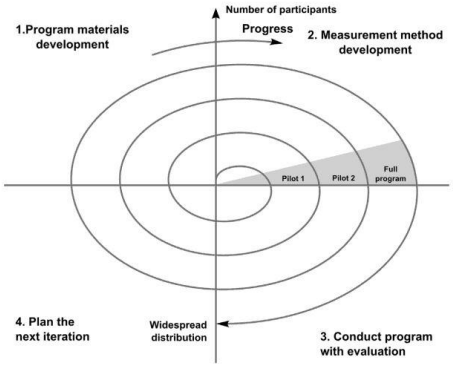


FIGURE 5.3: A spiral process for program improvement [10]

In adopting this strategy, these are the sets of Data which will be collected and analysed at the end of this pilot study to help improve in other outreaches:

1. How easy was it to buy the interest of the school administration into carrying out this Study with their 10-year-old Students?
2. Were there any Health and Safety concerns after the study was carried out?
3. How many teachers think this can be incorporated into improving the interest of Students in STEM Education with a niche in Robotics?
4. How helpful was the website and other outreach materials used in offering guidance on the requirements of the Study?
5. Time to complete each task, which will help in choosing tasks for next event.
6. General ratings on a 10-point scale from both Teachers and Students

Once this Data is collated and analysed, repeated information will be taken into immediate account as priorities in preparing for the next outreach event.

5.4.2. Summative strategy

This evaluates the effectiveness of the program after analysis based on program goals. It is very important to understand the goals of each program in order to assess its effectiveness. Following this strategy, certain data would be gotten from the participants to understand the effectiveness of this study based on the attitudinal changes of the participants after the study.

The necessary data required for this evaluation will be:

- From the 10 years old Students
    1. The ease of use of the platform
    2. The ease of the tasks given to carry out
    3. The interest level in Robotics and STEM Education
    4. Interest of visual programming as compared to normal coding
    5. General feedbacks on what they think would have been done better to improve the outreach
    6. Interest in participating in another outreach activity

- From the Teachers
    1. How they feel about incorporating this outreach into their teaching model?
    2. How they think this platform would benefit them in teaching Robotics?
    3. How easy was it to participate in the research and points of improvements for future outreaches?

Based on the two strategies above, two common measurements methods are frequently used which are Questionnaires and Interviews with Questionnaires (Surveys) been the most used. It is recommended that at the minimum that Questionnaires for programs and outreaches should contain at a minimum the profile and attitudinal information of the participants.

5.4.3. Construct interrelations

For this study, a pre-program and post-program questionnaire will be prepared to understand the attitudinal changes of the participants before and after the experiments. Based on Martin work in 2004 [55], the Questionnaire is made flexible and open-ended in order to extract useful data which can be follow both formative and summative evaluation strategies. The questionaries used for this study was divided into two sections, first for pre-program and the second for post-program evaluation. Pre-program Questions contained a set of 3 Questions whereas the post-program Questions are 5 each.

For the pre-program section:

- The first Question was to assess the participants knowledge and exposure to Robotics
- The second Question was to assess the participants interest in Robotics
- The third Question was to assess the IQ level of the participants

For the post-program section:

All three Questions above were also asked to understand the attitudinal changes which has occurred between the beginning and the end of the program. The last two Questions were to assess the points of improvements of the system and collect feedbacks on how the program can be better carried out in the future.

Here's a Table summary of methods used by some previous Authors.

Pre and Post Test	**Comparison Groups**	**Questionnaires**	**Interviews**	**Observations**	**Paper**
x	x				[8]
x					[96]
x					[29]
x					[17]
x			x		[25]
x			x	x	[26]
		x			[64]
			x		[24]
x		x			[57]
	x	x	x		[11]
		x	x	x	[34]
			x	x	[65]
x	x			x	[66]
		x			[68]
			x		[72]
x		x			[75]
x		x	x		[84]
		x	x		[37]
			x		[37]
		x	x		[37]
		x			[80]
		x			[48]
		x			[91]
		x			[92]
x			x	x	[100]
		x			[1]
11	**3**	**14**	**12**	**5**	**TOTAL**

TABLE 5.1: Evaluation methods of some studies

5.4.4. Risk Assessment and precautions

Although there were no imminent risks factored during the time of this study, there were certain precautions taken during this study. These were:

1. Ensuring that at least One Person from the administrative Team gets a Protecting vulnerable Groups certificate which is a basic requirement for working with children in Scotland.
2. Ensuring that no Child would certainly fright to Electronics and Gadgets was allowed to participate in this study.
3. Ensuring that the Children were never left on their own throughout the study.
4. Ensuring that ROSbot 2 Pro was not given a command to run it into a wall or obstacle which could cause damage of a part.
5. Using only 5 ROSbot at a time to manage technical failures

5.5. Results

A mini pilot study was carried with some Dr Mauro Dragone and two Teaching assistants in the Robotic Assisted Living Testbed Laboratory of HWU. This mini pilot study was aimed at assessing the Web Application user experience and an opportunity to take some technical measurements. The users were given a simple task to move Two ROSbot 2 Pro to targeted points using all the blocks available. Prior to the start of the experiment, the technical expert users were sent a simple Teaching manual consisting of a guide on the application usage. They were also sent the IDs of the Robots to enable them to run their generated program on the Robots, and the IPs to get the camera feeds.

5.5.1. Demonstration

As a minimum requirement, the users were expected to have a laptop with a web browser and access to an Internet Network. During the study, these were some of the steps the technical users had to take to ensure the aim of the study was actualised.

- Visit the application URL and select the ROSbot 2 Pro: At this stage, the users had to input the URL (*hwurobolab.giitechcodes.com.ng*) in their web browsers and select the EV3 Robot.

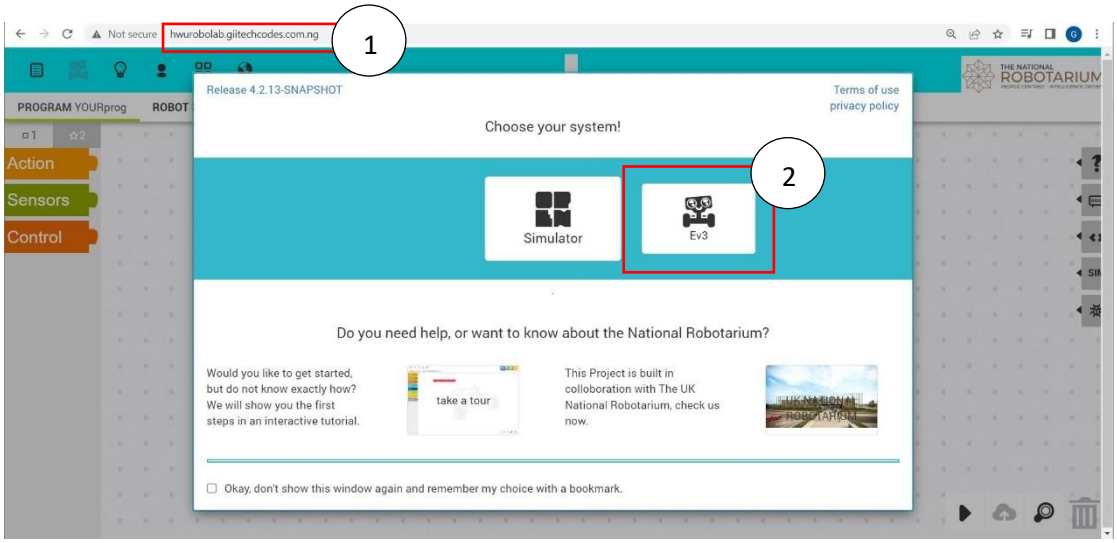


FIGURE 5.4: Visiting the URL and selecting the Ev3

After selecting the Ev3 option, the next step was to select the ROSbot 2 Pro. This is seen below.

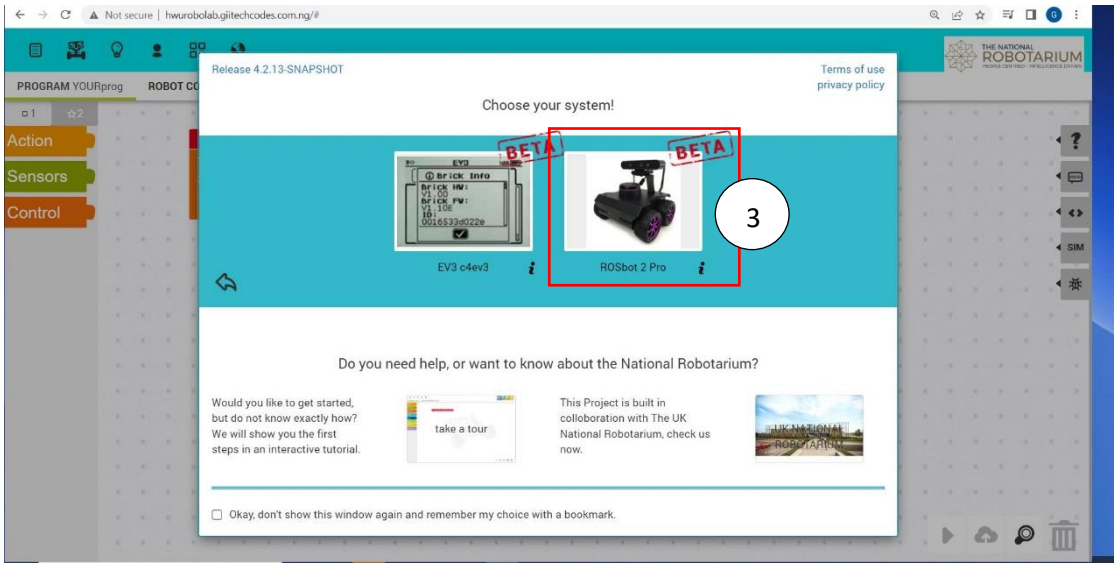


FIGURE 5.5: Selecting the ROSbot 2 Pro

- Select and Understand the Blocks to be used: At this stage, the users were expected to go through the list of the blocks available. To understand the blocks action, they were to test the block actions either from the simulator or the robot. Here's a list of the available blocks used.

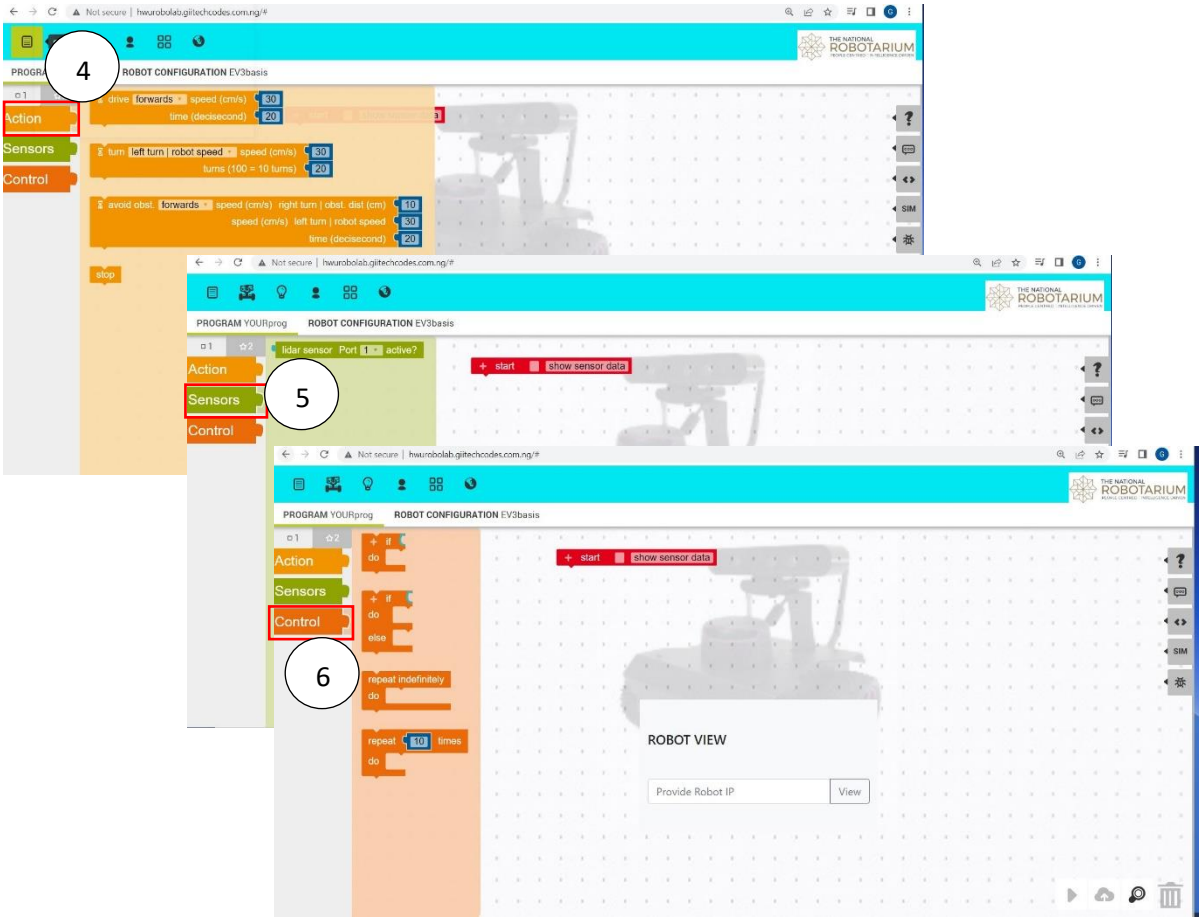


FIGURE 5.6: Blocks available for use

- View simulated Robot: After testing the blocks, the users were expected to see the outputs of their generated program on the simulator. Here's a view of a simulated program to loop forever.

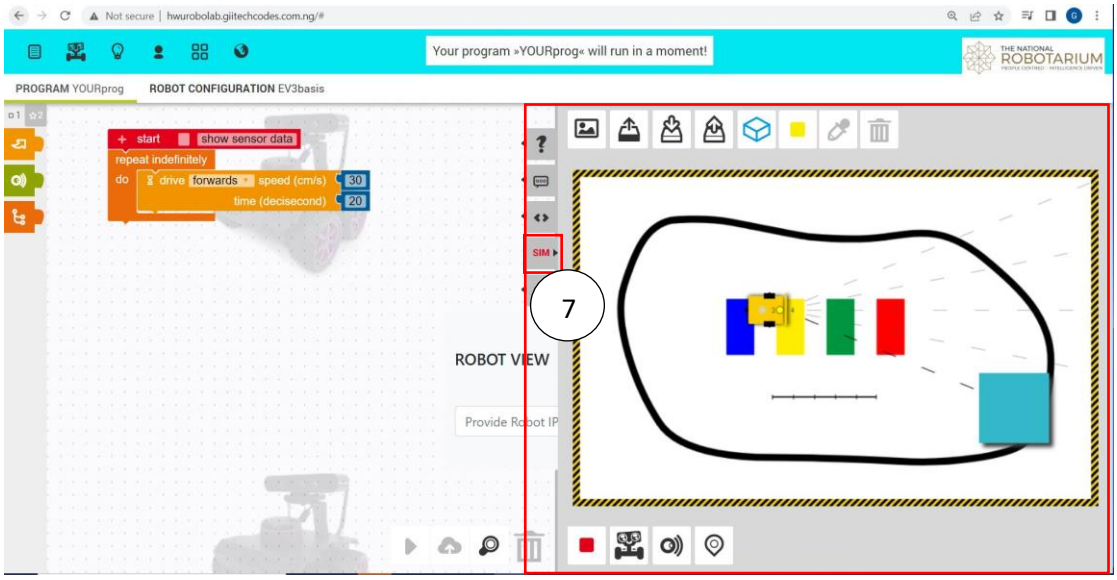


FIGURE 5.7: Accessing the Simulator

- Test Robot camera view: At this stage, the users tested the camera feed from the assigned Robot. To test this, the users had to be on the same network with the Robot and insert the Robot IP in the Robot View box in the application. The Robot View interface is seen below.

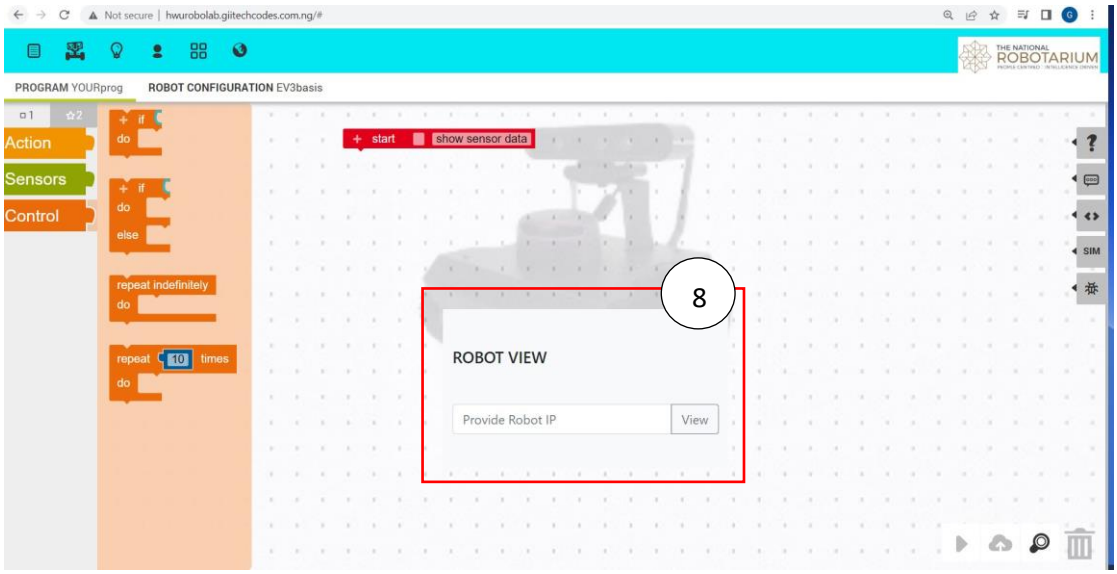


FIGURE 5.8: Remote view interface

- Run generated program on Robot: This was the last stage of the study. At this stage, the users were expected to build sets of programs which will enable the Robot to move from one end of the laboratory to the other while avoiding obstacles. Here's a view of running the program.

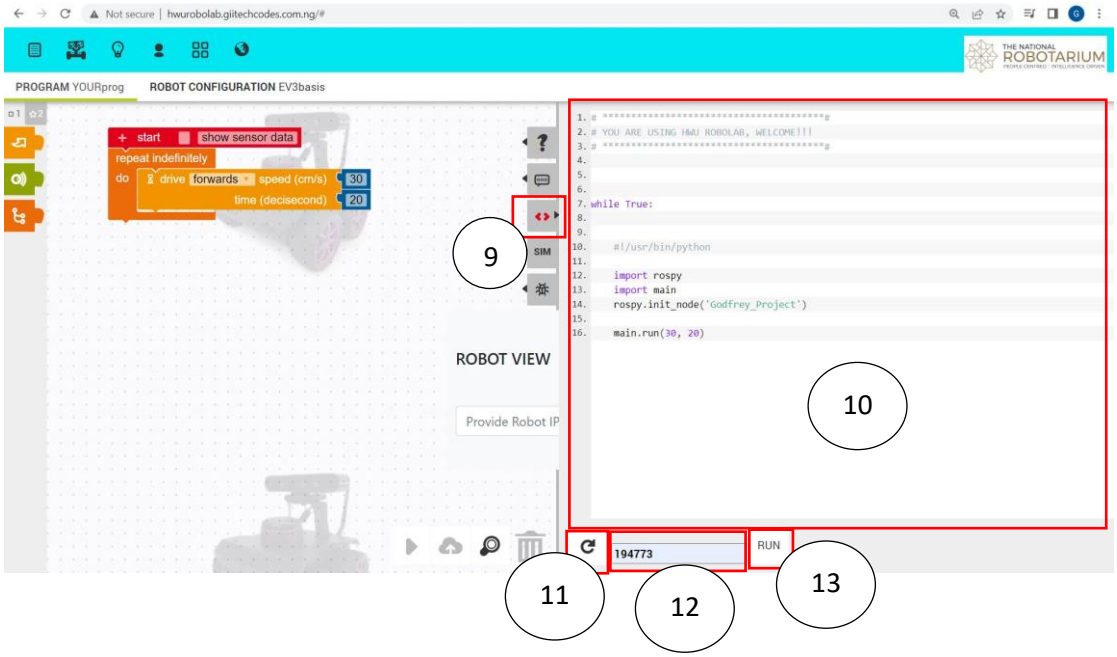


FIGURE 5.9: Running the Generated Program interface

To run a program on the robot, the users had to first click on "9" and "10" opened which shows the python code of the generated program and the interface to run the program. The next step was to input the Robot ID in "12" and click "13" to trigger the run action of the generated program on the Robot. Each time the program is changed while "9" is opened, users are expected to use "11" to refresh the program.

For a step-by-step guide on how to use the web application for further studies and activities, here's a link to a video guide:

https://youtube.com/playlist?list=PLFwip2ddaI_lfqfmYPaOGg_6kkfdhkxlU

5.5.2. Assessment and Feedback

At the end of this pilot study, the users were sent an online questionnaire which was used to assess their experiences during the experimental study.

There were a few technical feedbacks offered at the end of the experiment. These feedbacks are based on a user which was using a Chrome Browser on an Ubuntu 18.04 Linux operating system laptop. These feedbacks and technical results were:

1. Transparency in the camera view frame
2. Non-reminder of Robot and Application being on same network for video feed
3. Incomplete camera view with scroll overflow
4. Speed and Time validations
5. Inconsistency with unit of measurements with respect to Teaching manual
6. The icon of the blocks and the blocks text were not easy to understand
7. There were some delays between when the run button was pressed and when the Robot performed the actions. This was due to weak internet and loss of internet connection in the Robot during the test.

After these assessments, some fixes were made to 1, 2, 3, 5 while the others would be fixed in future works.

6. Discussion and Conclusion

During the development path of the Web Application, there were several key discoveries. The most relevant discovery is the fact that the Internet plays a very vital role in Data Transfer. In making technology usage more friendly and intuitive, the process in achieving desired results from using that technology should be very minimal. At the initial build out of this web application, the pipeline for Data Transfer from the Web Application to the Robot was done using conventional data transfer method where the user had to have some extra installations on their computer, and the administrator manually activating the required nodes in the robot by ssh. On first testing of this method, there were lots of constraints ranging from variations in operating systems, in ability to get the robot connected when the network changes, and a very long time before a user could generate a single program to run on the Robot.

To deal with the problem, I had to think of a solution where the target users which are 10 years old school students can effectively use the web application by only using a computer which had only a web browser and internet connection, and not having to install other dependencies. The solution implemented, followed a change of the method the opensource had which was to Download the generated program to the download directory, into uploading it to a server. It was because of this process I built a new server using Node JS with an API which fetched and POST the generated program to the server. This solved the problem of Data Transfer which now became Cloud based. But this opened the door to a bigger security threat where the Shell script written in the robot to search for the generated program had no authentication and made the whole architecture vulnerable to potential intruders.

In solving the problem described, an identifier system was integrated into the web application which appends a unique ID to the name of the generated python file. With this technique, every robot would be configured to search and run only specific files with their unique IDs. This project introduces a new method of Data Transfer from a Non-ROS system to a ROS system by using the Cloud.

In successfully completing the Goals of this Project, an experimental study was designed for use by the target audience, and an effective evaluation method was offered.

6.1. Project Summary

6.1.1 Primary Research Questions Answers

At the start of this Project, there were a couple of Questions which needed to be answered to achieve the Project Objectives. These answers are seen below:

- **RQ-1**: How can we develop a Cloud based Web Application for programming Robots from Open Source?

  In Chapter 3, we were able to understand how the Open Roberta open-source works and its limitations. Chapter 4 clearly presents a process on how the open source was modified and improved. A key improvement here is seen in the deployment of the open source to the Cloud. The open source has its server embedded in the application thereby creating a problem in extracting the server and deploying it as stand alone. This was overcome by deploying the whole application to a Linux virtual server on Digital Ocean cloud. The new server built out was also deployed on the same cloud platform there by making all its computations and abstractions run from the cloud. End-users didn't have to access the application anymore from the IP and the Port, rather from a sub-domain which the IP and Port was mapped to using "ngnix" and updating the A-Record of the sub-domain name.

- **RQ-2**: How can a Cloud based Web Application be usable to 10 years old with access to only the browser, and optionally to a real robot as part of a class activity?

  In this study, there was a great consideration for the end user which will be 10 years old Primary School Students. This consideration informed the design pattern of the application in such a way that it is attractive to the kids and extremely easy to use with minimal efforts and only with a web browser installed on their computers. In this study, a simple how to use manual was developed which can further guide the students on the usage of the application. This was designed with clear illustrations and a video tutorial on how to use. This was also appended to the tutorial section of the web application.

- **RQ-3:** Besides the Robot of consideration, which is the ROSbot 2 Pro, how possible is it to develop an architecture which will be easy to add other Robots with the same data transfer model?

  This study explicitly proposes a method an easy method to add more blockly blocks to the current blocks available for the ROSbot. This creates a procedural approach in adding more blocks for other Robots when they are added. Adding other Robots becomes a walk in the park as the open source creates that flexibility as seen in Chapter 3 and the API can be used for more Robots. One would wonder how this would be possible to extend to non-ROS robots. This is also possible as the code generated is not static and can always be modified to suit the Operating System of various Robots.

- **RQ-4:** How can we have a flexible Data Transfer from Platform to Robot without both Systems being on the same network?

  In this Study, a JavaScript API was built out using jQuery which was developed based on the structure of the application. At the beginning of the Project, it was seen that the generated program was downloaded to the computer of end users which did not match with the aim of the Project which should be completely Cloud based. In corporation with the API, which was designed to fetch and post the data, which was initially downloaded, a node Java script server was developed as an endpoint which runs on a static port and listens to every post request from the API. With this architecture, the Robot only searches the Path on the Node JavaScript sever for generated programs every 3 seconds and runs the program found in that path. With this model, the Robot wasn't in anyway forced to be on same network as the application at every instance before data transfer could occur. This is captured extensively in (*section 4.5*).

- **RQ-5:** How can we best measure and evaluate the effectiveness from Outreach activities?

  Effectively, this study shows us how two simple strategies such as the adaptive and summative strategy can effectively be used to analyse the outcomes from any robotics outreach activities. It is also seen that Questionnaires have been the most used and most effective tools for this analysis and as such widely recommended for Robotics outreaches as they can serve as basis of feedbacks for improvements and attitudinal analysis.

6.1.2. Extended research questions answers

- **ERQ-1**: How can we manage the problem of multiple robots requesting for files to execute from the same server and prevent unauthorised users to create programs for the Robot to execute?

  After building out the API, this problem was noticed where any user on the application can generate a program and the Robot would act on it. This presented a security problem. To solve this problem, a front end identify system was built out which enforces users to input the ID of the Robot they want to control. The ID is just any identifier randomly selected characters that each Robot has been setup with. So, if a user doesn't know the ID of the Robot, they can't generate a Program for the Robot. This creates a possibility of having multiple users controlling different Robots at same time.

- **ERQ-2**: What are the cutting-edge guidelines to take before choosing Cloud Services for platform deployment with respect to response time, server speed, security, and performance?

  Due to the computation activities which happens on the cloud, there is every tendency of having unforeseen server issues. During the test of application on the Cloud, it was seen that most build were failing as this was due to server space. This was overcome by increasing the cloud server space. In factoring the two servers the platform depended upon which must be running at every

instance, it was worth considering a cloud service which would have the capacity for parallel run action. Two options were considered based on these facts, Amazon Web Server (AWS), and Digital Ocean. AWS seemed perfect for our requirements, but it posed a challenge of ease to use and cost for every instance. Hence Digital Ocean was ideal as it is easy to use, runs on Linux Operating System, and provides a very flexible budget for scaling. For this study, it is seen that Digital Ocean can handle multiple robots SSH effectively without any down time and it had a good resilience and speed when the technical test was conducted.

6.2. Limitations

Alan Perlis said, "*There are two ways to write error-free programs, only the third one works*". In completing this Project, there were key constraints which were encountered from technical users and individual testing. Here is a list of them:

- **Internet Connection**: When I was evaluating out the platform in HWU grid lab, I noticed that due to weak internet connection from my phone hotspot in which the robot was connected to, it took a longer time for the robot to respond to the run trigger from the web application. The search and run time increased from 3 seconds to 8 seconds. This was also noticed during the technical user test.
- **Obstacle avoidance:** When testing out the obstacle avoidance block, I observed that the ROSbot Pro 2 was only avoidance obstacles at a specific height. This was due to the camera and angle height configuration of the ROSbot 2 Pro which can be modified from the Python program generated.
- **Security**: During the first alpha testing, it was noticed that anyone and everyone could control my ROSbot 2 Pro from their location. At production level this could be a serious breach of ethics and control. Although this problem was solved using Identifiers, but regardless of the unauthorised program not running on the robot, it is still stored on the Node JS server which can also cause an attack to the server. Another critical security issue is access to the files on the server. If a user gets hold of a configured Robot, they can have a gateway into the application and node js servers.

- **Experimental Study:** This limitation is seen in chapter 5. All the outlined evaluation strategies were not actually assessed due to approval of my ethics application, from the University. In as much as the model is a recommended model, it would have been good to further prove it by conducting an actual study with the target audience.
- **Robot malfunction:** In general, electronics can be very unpredictable even with error and damage sensors. While conducting various personal tests, I noticed that there were times when the ROSbot 2 Pro would not connect to the any network or fail to come on. During the technical test, one of the ROSbot 2 Pro couldn't see any active wireless network. After plugging an ethernet cable into the port behind the robot, all available networks became available.

6.3. Future Works

This study is a pilot study and hence there will be more work and improvements on the Web Application. In future studies, here are some of the improvement's points:

- **Security:** In the future this would be taken into critical consideration by creating flexible and strict restrictions in accessing the servers. I plan to create an approval request for every preconfigured robot to use the application. This will help manage access when the time of use expires. I also aim at enforcing users to sign up before they can use the application as this would help prevent users without the Robots ID from creating any program hence protecting the Node JS server from unauthorised files.

- **Integrating a Teacher Dashboard:** In the next version, I aim at creating a Teacher and Tutor Dashboard where teachers of Robotics can be assigned to Students to help manage their learnings. Students would be able to share their programs with their preferred tutors and get instant help away from their classrooms.

- **Adding more blocks:** At the end of the duration of this Study, an improvement to the way the code generation was discovered. This method made it easy to add more blocks. Currently, the method seen in Chapter 4 Code Generation section

was used to implement all the blocks available except the Obstacle avoidance block. To include this in the future, I have implemented a sensor block which should be used with the “if” logic to create a trigger for the robot sensor. This method will be followed to improve the obstacle avoidance block and followed to add more blocks.

- **Synchronising Simulator with Robot action:** Currently, not all the blocks’ actions generated are in synchronism with the actual robot and simulated robot. This is due to the complexity in matching real robot parameters with that of the simulator. This will be fixed in the next release of the application.

- **Data Transfer:** In solving the robot response time to generated program, static python files will be created automatically by the ROSbot owners on the server during registration and approval. Similar static files will also be present on the ROSbot during configuration. Program files generated by users will no longer be passed, the values will be written to the static files base on the files ID. When this is done, the ROSbot also won’t move the files to its script folder to run it after changing its mode, rather it will perform a read and write function of the content from the file data on the server assigned to it. This will reduce and improve the file transfer process from the cloud to the server hence increasing the ROSbot response time to a generated program.

- Finally, to implement the evaluation method, the proposed designed experiment will be carried out with the targeted audience with all the necessary ethics approval gotten.

APPENDIX A

HOW TO INSTALL AND RUN HWU ROBOLAB FOR DEVELOPERS AND SYSTEM ADMINISTRATORS:

- Cloning the Application

  Begin by cloning the application repository with:

  Gitclone- "$git clone https://github.com/Godfrey2712/hwurobolab.git"

- Installing the dependencies

  It is highly recommended to use a Linux machine. Here is a list of dependencies you need to have on your machine:

  1. Maven:- "sudo apt install maven"
  2. Java SDK:- "sudo apt install default-jdk"
  3. Npm:- "sudo apt install nodejs npm"

- Building the Application

  After the application is cloned, it should be built out using maven. First get into the application path : "~/openroberta-lab"

  Then build the application by using:

  "$mvn clean install –DskipTests"

  The "-DskipTests" ensure that errors are not thrown during build.

- Cloning the file-saver

  The node js server is inside the cloned directory and named "file-saver". Enter the directory path.

  Once done, build it out by using this command once you are in the server path:

  "$npm build"

- Starting the Web Application and node js Server

  The web application is started by:

  1. Get into the application path
  2. Write this command “$./ora.sh start-godfrey-project”
  3. The web application server runs at an IP of 0.0.0. and a port of 1999
  4. You can visualise the application by going to this URL on a web browser:

     “http://localhost:1999”

  The node js server is started by:

  1. Get into the file-saver path
  2. Write the command “$npm run”
  3. The server starts and is listening at a port of 500 to the API in the Application

APPENDIX B

MAPPING DOMAIN NAME TO IP ON PORT 1999

To map a domain name to the application IP running on the server, here are simple steps to follow:

- Getting a Domain name or a sub domain
- Update the A record of the Domain/Subdomain to the IP of your cloud server

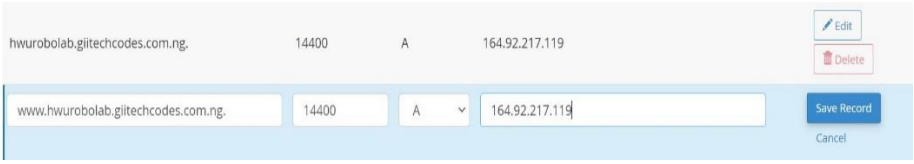


FIGURE B.1: Mapping the IP to a Sub domain

- Install nginx on your server by using, "$sudo apt-get install nginx"
- Add a new server block which does a proxy redirect in your default file in the nginx path. Here's the absolute path "$~ /etc/nginx/sites-available"

```
}


server {
    listen 80;
    server_name hwurobolab.giitechcodes.com.ng;

    location / {
        proxy_pass 164.92.217.119:1999;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

    }
}
```

FIGURE B.2: Enabling the port access over a Sub Domain

APPENDIX C

ROS MODELS FOR MOTION AND OBSTACLE AVOIDANCE:

These were the initial models of the python scripts for the blocks used before updating to current method.

- **Move Linear ROS Model**

```
#!/usr/bin/python

from geometry_msgs.msg import Twist
from sensor_msgs.msg import LaserScan
import math
import rospy
import time

rospy.init_node('Godfrey_Project')
pub = rospy.Publisher('/cmd_vel', Twist, queue_size=1)

def run():

    move = Twist()
    move.linear.x = 30 * 0.01
    for i in range(20):
        pub.publish(move)
        rospy.sleep(0.1)

run()
```

- **Move Angular ROS Model**

```
#!/usr/bin/python

from geometry_msgs.msg import Twist
from sensor_msgs.msg import LaserScan
import math
import rospy
import time

rospy.init_node('Godfrey_Project')
pub = rospy.Publisher('/cmd_vel', Twist, queue_size=1)

def turn():

    rotate = Twist()
    rotate.angular.z = 30 * 0.01
    for i in range(20):
        pub.publish(rotate)
        rospy.sleep(0.1)

turn()
```

- **Obstacle avoidance ROS Model**

```
#!/usr/bin/python

from geometry_msgs.msg import Twist
from sensor_msgs.msg import LaserScan
import math
import rospy
import time

rospy.init_node('Godfrey_Project')
pub = rospy.Publisher('/cmd_vel', Twist, queue_size=1)

move = Twist()
def callback(dt):
    if dt.ranges[0]>10 * 0.01:
        move.linear.x = 30 * 0.01
    if dt.ranges[0]<10 * 0.01:
        move.linear.x = 0.0
        move.angular.z = 30 * 0.01

sub = rospy.Subscriber('scan', LaserScan, callback)
for i in range(20):
    pub.publish(move)
    rospy.sleep(0.1)
```

For the current method of generating the codes from the blocks and running the code on the Robot, here's the link to the Python main file which should be in the Robot.

"https://github.com/Godfrey2712/hwurobolab/tree/master/hwurobolab/Python_File"

APPENDIX D

ROBOT SETUP AND CONFIGURATION WITH SHELL FILES:

- **Shell file A**

This consists of the commands to start all the ros nodes (ros core). The shell file is called "start_robot.sh"

```
#!/bin/bash
cd husarion_ws
roslaunch husarion_bring_up base.launch
```

- **Shell file B**

This consists of the commands to search the node js download path every 3 seconds, fetch the right file based on configuration, and run it till completion. The shell file is called "run_program.sh"

```
#!/bin/bash
if rsync --remove-source-files root@164.92.217.119:/root/file-saver
    /downloads/YOURprog.py /home/husarion/husarion_ws/src/
    husarion_ros/scripts; then

cd ~/husarion_ws/src/husarion_ros/scripts

chmod +x YOURprog.py

python YOURprog.py

else
  echo "no program available"
fi
```

It is worth noting that at this point the ROSbot 2 Pro ID is given by specifying the name of the python file it should search for. The names in red above can be modified to the name of the ID of the Robot. Example YOURprog_1234.py

- **Run on boot setup**

At this point, the two shell files above are included in the .bashrc file. To make this run-on boot, the start-up application has been configured to start a terminal on system boot which starts all the specified shells in the bashrc file. This is the command which starts the terminal on Robot boot:

"exo-open --launch TerminalEmulator"

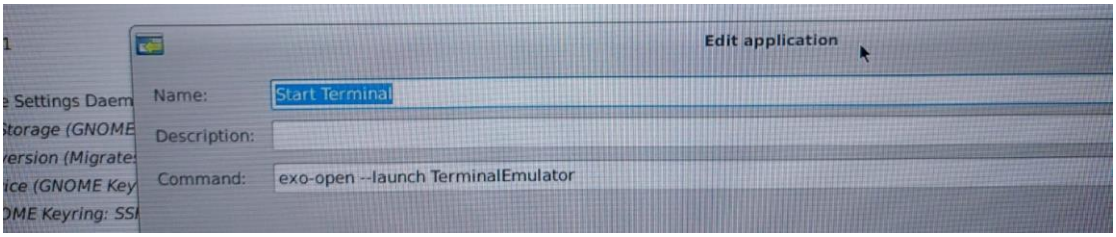


FIGURE D.1: Automatic Start-up of Terminal for bashrc

The second shell file is configured in the bashrc file to run in another type of Terminal called Terminator and this is the command used.

```
1 . ~/start_robot.sh &
2 xterm -e watch -n 3 ~/run_program.sh
```

The number highlighted above defines how long it takes the robot to search and run the generated program on the server.

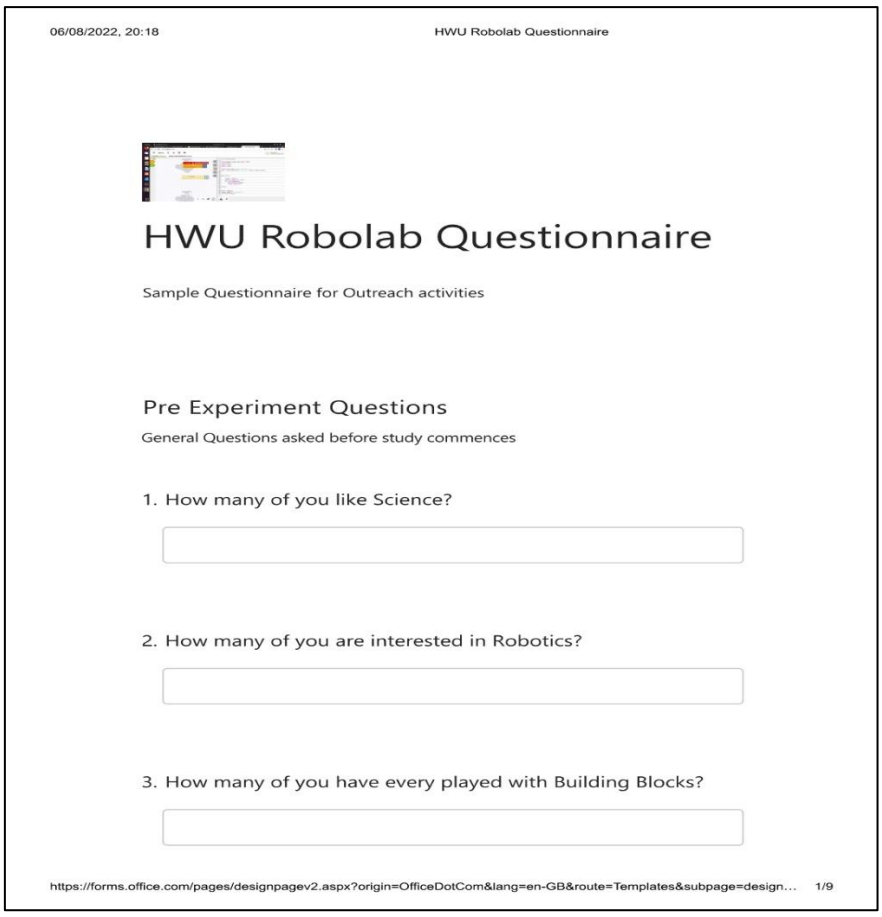
These shell files are found in:

"https://github.com/Godfrey2712/hwurobolab/tree/master/hwurobolab/Shell_Files/Shell_Files"

APPENDIX E

SAMPLE QUESTIONNAIRE:

- **Pre Session-Questions**



06/08/2022, 20:18 HWU Robolab Questionnaire

HWU Robolab Questionnaire

Sample Questionnaire for Outreach activities

Pre Experiment Questions

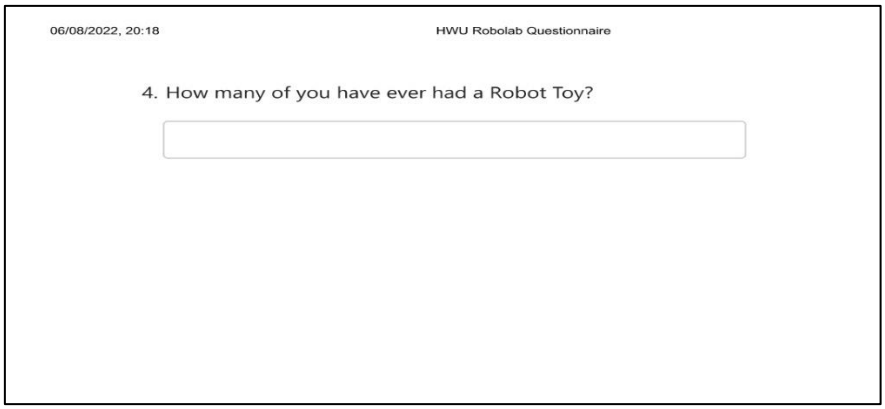
General Questions asked before study commences

1. How many of you like Science?

2. How many of you are interested in Robotics?

3. How many of you have every played with Building Blocks?

https://forms.office.com/pages/designpagev2.aspx?origin=OfficeDotCom&lang=en-GB&route=Templates&subpage=design... 1/9



06/08/2022, 20:18 HWU Robolab Questionnaire

4. How many of you have ever had a Robot Toy?

- **Session 1_Stage 1 Questions**

06/08/2022, 20:18 HWU Robolab Questionnaire

Post Experiment Questions for Session 1_Stage 1

5. Name

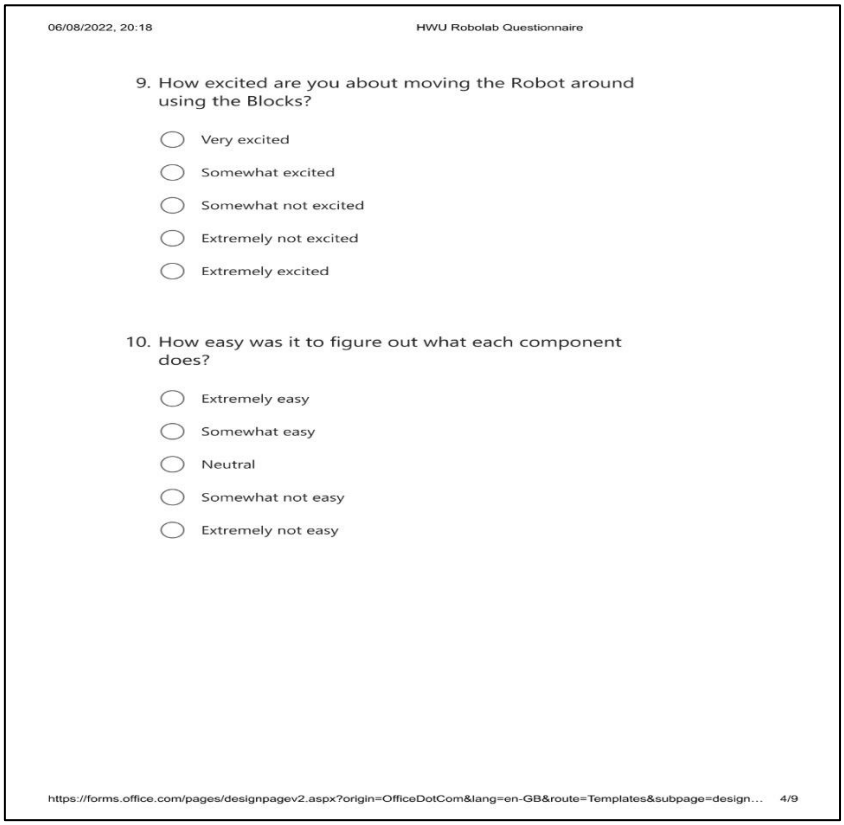
6. Age

7. Class

8. How easy was it to understand the Blocks?

- ◯ Extremely easy
- ◯ Somewhat easy
- ◯ Neutral
- ◯ Somewhat not easy
- ◯ Extremely not easy

https://forms.office.com/pages/designpagev2.aspx?origin=OfficeDotCom&lang=en-GB&route=Templates&subpage=design… 3/9



06/08/2022, 20:18 HWU Robolab Questionnaire

9. How excited are you about moving the Robot around using the Blocks?

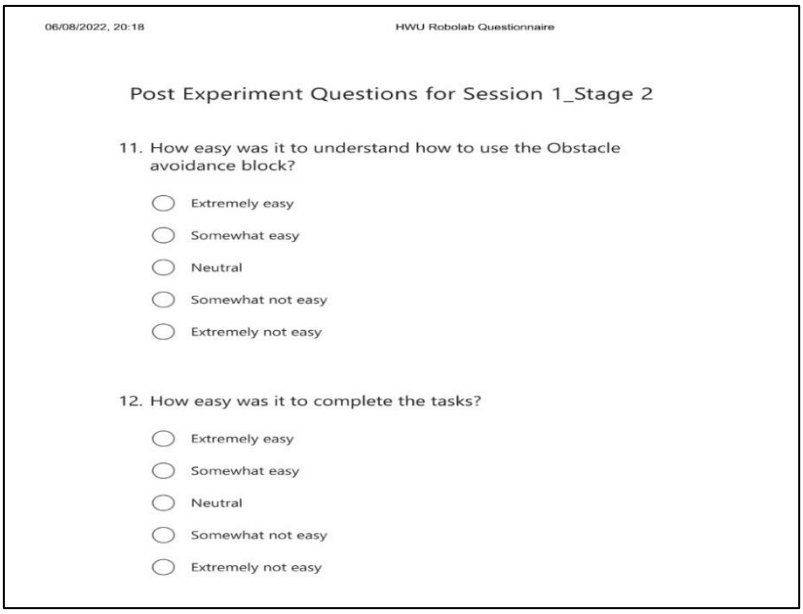
- ◯ Very excited
- ◯ Somewhat excited
- ◯ Somewhat not excited
- ◯ Extremely not excited
- ◯ Extremely excited

10. How easy was it to figure out what each component does?

- ◯ Extremely easy
- ◯ Somewhat easy
- ◯ Neutral
- ◯ Somewhat not easy
- ◯ Extremely not easy

https://forms.office.com/pages/designpagev2.aspx?origin=OfficeDotCom&lang=en-GB&route=Templates&subpage=design… 4/9

- **Session 1_Stage 2 Questions**



06/08/2022, 20:18 HWU Robolab Questionnaire

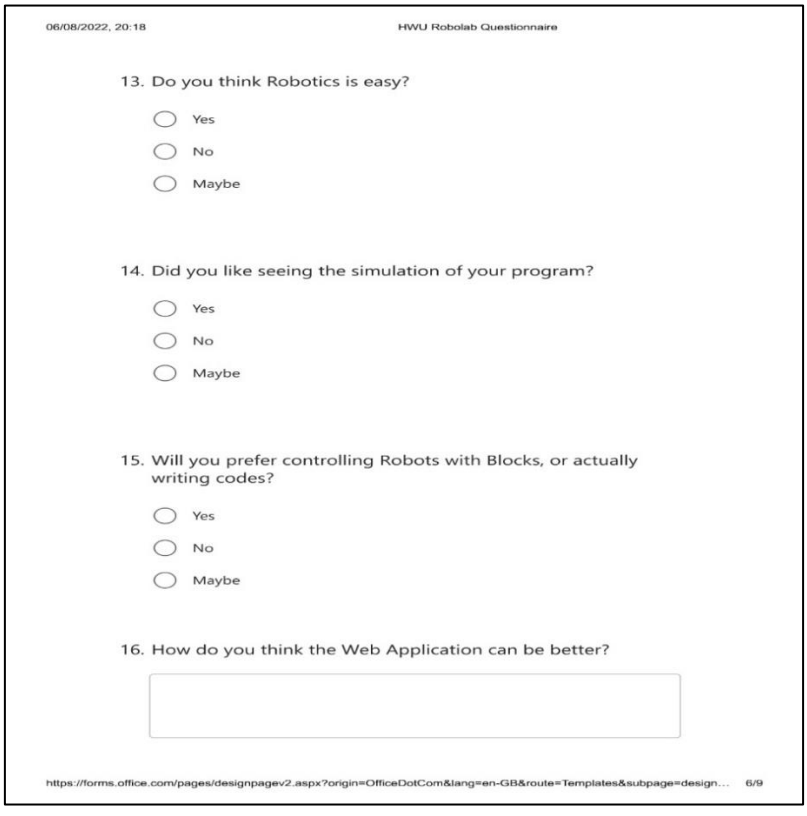
Post Experiment Questions for Session 1_Stage 2

11. How easy was it to understand how to use the Obstacle avoidance block?

- ◯ Extremely easy
- ◯ Somewhat easy
- ◯ Neutral
- ◯ Somewhat not easy
- ◯ Extremely not easy

12. How easy was it to complete the tasks?

- ◯ Extremely easy
- ◯ Somewhat easy
- ◯ Neutral
- ◯ Somewhat not easy
- ◯ Extremely not easy



06/08/2022, 20:18 HWU Robolab Questionnaire

13. Do you think Robotics is easy?

- ◯ Yes
- ◯ No
- ◯ Maybe

14. Did you like seeing the simulation of your program?

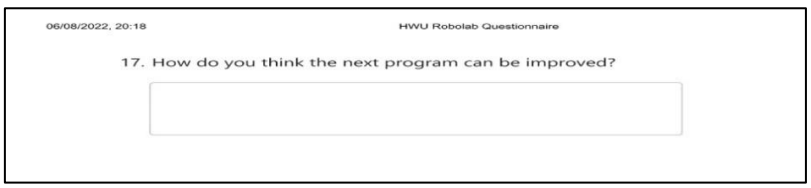
- ◯ Yes
- ◯ No
- ◯ Maybe

15. Will you prefer controlling Robots with Blocks, or actually writing codes?

- ◯ Yes
- ◯ No
- ◯ Maybe

16. How do you think the Web Application can be better?

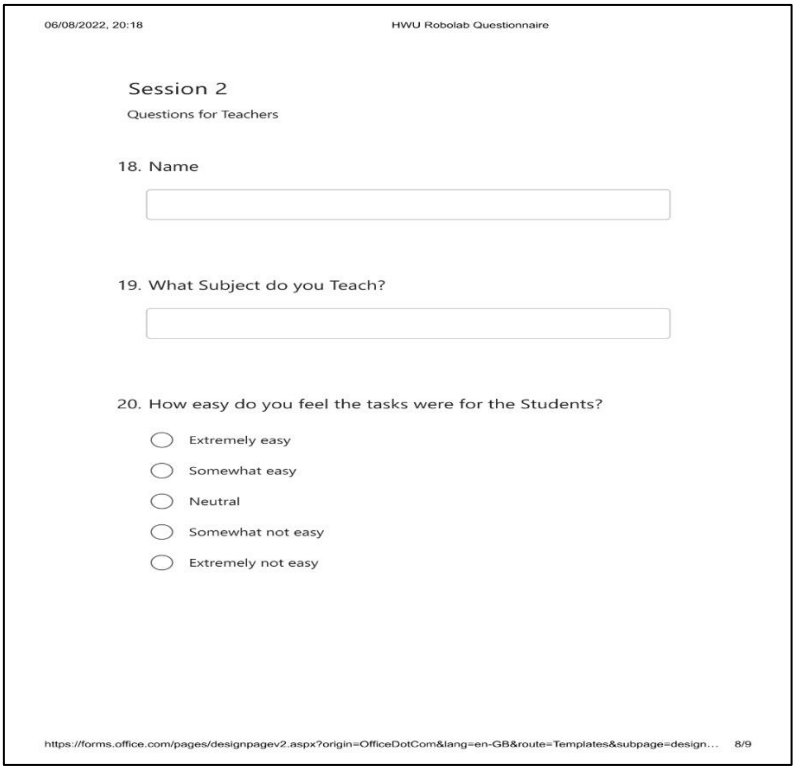
https://forms.office.com/pages/designpagev2.aspx?origin=OfficeDotCom&lang=en-GB&route=Templates&subpage=design... 6/9



06/08/2022, 20:18 HWU Robolab Questionnaire

17. How do you think the next program can be improved?

- **Session 2 Questions**



06/08/2022, 20:18 HWU Robolab Questionnaire

Session 2

Questions for Teachers

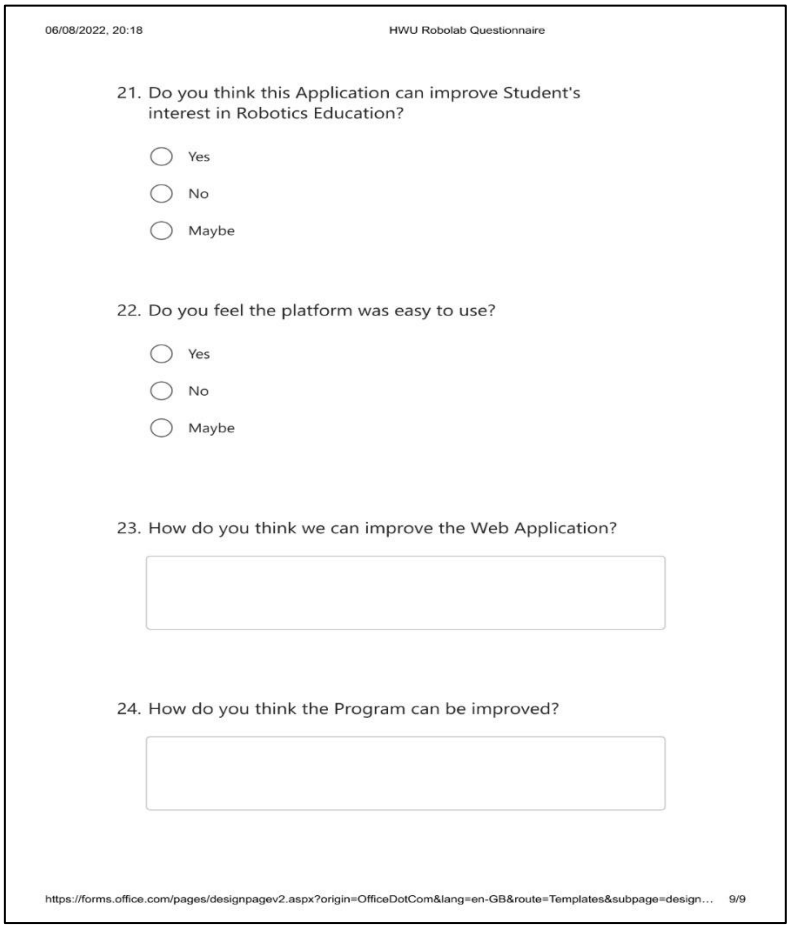
18. Name

19. What Subject do you Teach?

20. How easy do you feel the tasks were for the Students?

- Extremely easy
- Somewhat easy
- Neutral
- Somewhat not easy
- Extremely not easy

https://forms.office.com/pages/designpagev2.aspx?origin=OfficeDotCom&lang=en-GB&route=Templates&subpage=design… 8/9



06/08/2022, 20:18 HWU Robolab Questionnaire

21. Do you think this Application can improve Student's interest in Robotics Education?

- Yes
- No
- Maybe

22. Do you feel the platform was easy to use?

- Yes
- No
- Maybe

23. How do you think we can improve the Web Application?

24. How do you think the Program can be improved?

https://forms.office.com/pages/designpagev2.aspx?origin=OfficeDotCom&lang=en-GB&route=Templates&subpage=design… 9/9

APPENDIX F

TEACHING MANUAL FOR HWU ROBOLAB:

This teaching guide is prepared to guide Teachers and Instructors in effectively utilising the HWU Robolab Platform for Outreach activities.

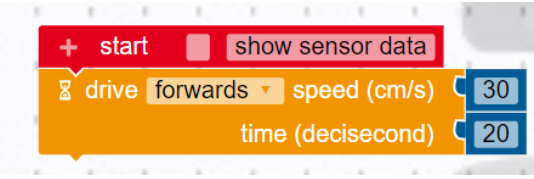
This is divided into two sections, **section 1.0** for Teachers/Instructors and **section 2.0** for Students.

SECTION 1.0: For Teachers

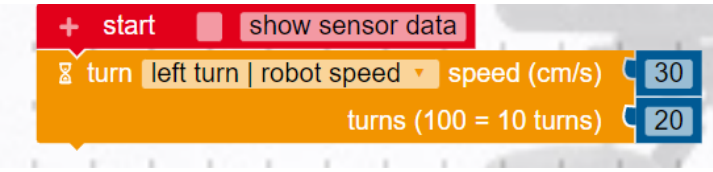
This section explains what each block does for the ROSbot 2 Pro and a preliminary guide on How Teachers and Instructors can best utilize the platform for Primary School Students (ideally ages 8 – 10 years old).

Blocks available:

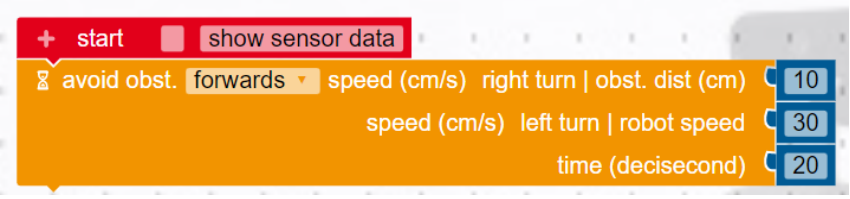
- **Drive action block:** This block is responsible for moving the ROSbot 2 Pro forward and backwards.



- **Turn action block:** This block is responsible for generating a turn action for the ROSbot 2 Pro.



- **Obstacle avoidance block:** This block is responsible for generating a forward motion for the Robot while avoiding obstacles.



Teachers Guide:

- Students are expected to use only the beginners blocks as the experts' blocks are still under development and will be available in future versions.
- To access Robot Camera view, ensure that Students are on the same network as the Robot and the IP of each robot are known. HWU Robolab support Team can further help with this. Kindly email support@hwurobolab.giitechcodes.com.ng for help.
- Ensure the IDs of the Robots are known before any class activities as this would be needed by the students to run their generated program on the Robots assigned to them.

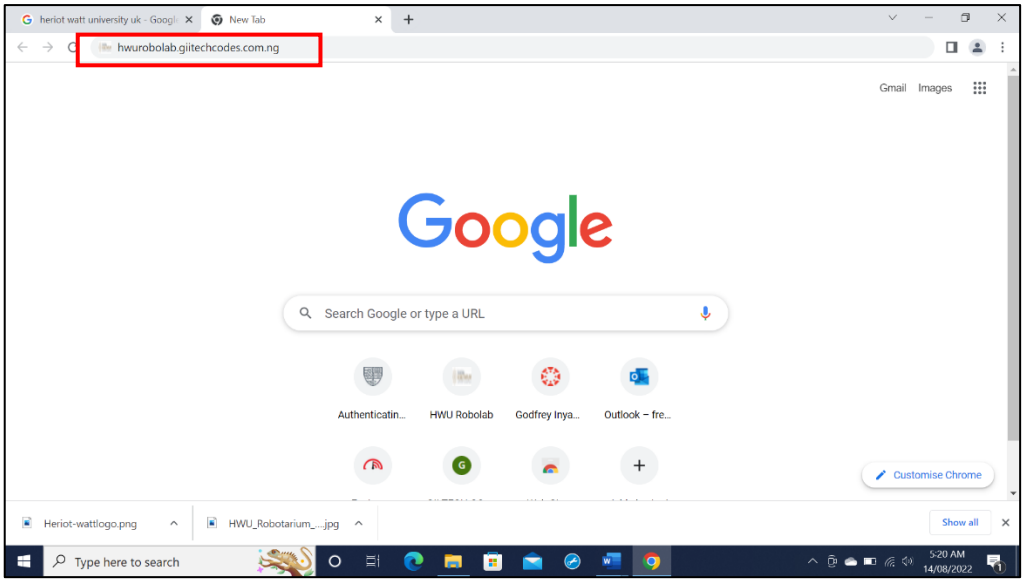
SECTION 2.0: For Students

In this section a clear guide on how to use the blocks by Primary School Students is provided, primarily for 10 years old Students.

- On first usage, ensure there's good internet connection on your laptop. The web application can also be assessed from a mobile device but for a better user experience, it is advised to use a Personal Computer.

  The application can be accessed by inputting this URL in your web browser:

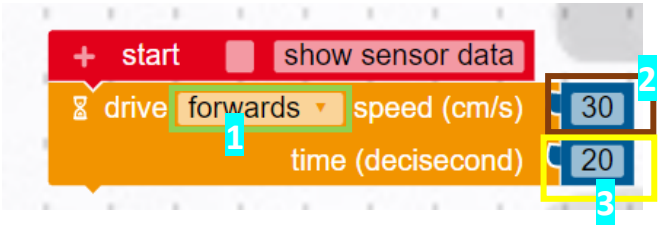
  "hwurobolab.giitechcodes.com.ng"



It is best this is accessed using **Google Chrome** Web Browser.

How to use the blocks:

- **Drive action block:** This block consists of all functions responsible to move the Robot in a forward and backward direction.



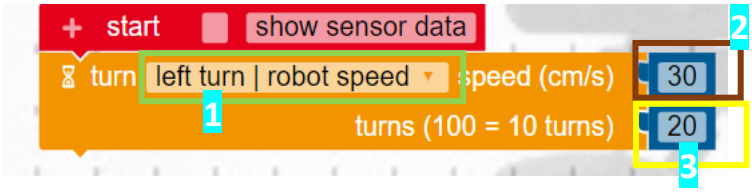
1 is used to choose the direction you want the robot to go. For a backward direction, input a (-) sign before the speed value. Example is 30 for forward and -30 for reverse motion.

2 is the value responsible for the speed which is in cm/sec. To get a moderate robot speed, ensure speed is in m/sec. Example is 0.3 and not 30.

3 is how long you want the Robot to move as its value is in decisecond. For example, if you want to move the ROSbot for 5 seconds, the value to insert is 50.

These values also cause similar action effect on the Robot in the simulator.

- **Turn action block:** This block consists of all functions responsible for turning the Robot.

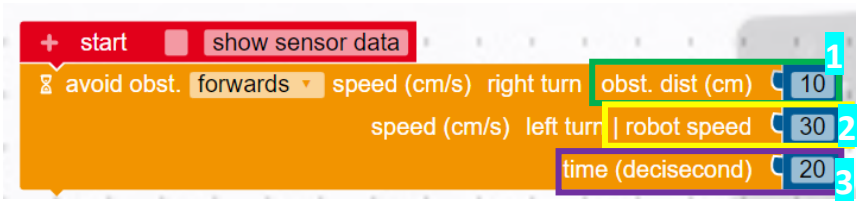


1 is used to define the turn direction of the Robot. For a left turn, the value of the speed should be positive, while for a right turn, the value of the speed should be -ve.

2 is used to define the how fast the robot should turn

3 is used to indicate the number of turns the Robot should make

- **Obstacle avoidance block:**



1 is used to indicate the distance the Robot detects an obstacle before stopping

2 is used to define the speed of the Robot

3 is used to indicate how long you want the Robot to move while avoiding obstacles.

Example Class activities:

- **Basic motion:** The ROSbot can be programmed to randomly move around a class with no specific goals point
- **Goal position:** Each Student can be assigned specific goal positions in the classroom and given the task to control the Robot to those points using only the drive and turn block
- **Move and avoid obstacles:** An environment can be set with some obstacles such as waste bins and other students standing in a fixed position, while other students are given the tasks to control the ROSbot to specific goal points within the shortest time possible.

Guide: https://youtube.com/playlist?list=PLFwip2ddaI_lfqfmYPaOGg_6kkfdhkxlU

Contact us: +44 (0)7459 457520 | godfreyinyama@giitechcodes.com.ng | EH12 9AH

APPENDIX G

ETHICS FORM:

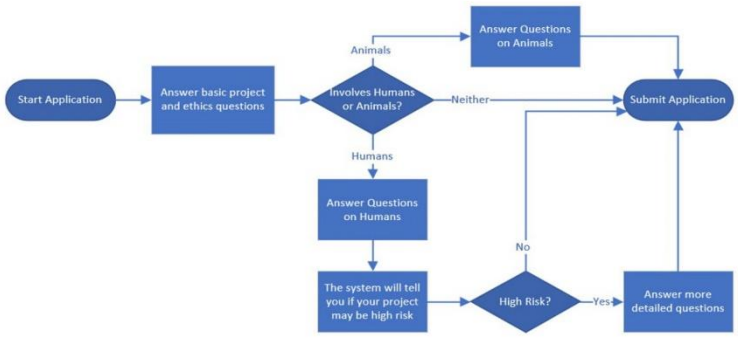
To carry out the experiment in Chapter 5 which wasn't done due to time factor, here's the ethics form filled out:

HERIOT WATT UNIVERSITY

E M S

Ethics Application Form

What is Going to Happen?



To complete an Ethics Application you will first answer some basic questions, followed by more specific questions depending on whether you are using human participants, personal data from external sources, or living animals.

"Part 1 - Ethics" asks you about ethical considerations your project may raise, and what you will do to address these ethical issues.

The section "Part 2 - Data Protection" will be asked if you are using human participants, or personal data from external sources. It will ask you about how you are using individuals data, and how you are protecting it. If your project is identified as potentially high risk you will need to answer further questions.

Eligibility to Request Ethics Approval

The Heriot-Watt Research Ethics Policy can be found here. Before proceeding with your application please check to confirm the following:

☑ I have read and understood the Heriot-Watt Research Ethics Policy

How to Navigate the System:

18 August 2022

Reference #:

Page 1 of 26

To move to the next page of the application:		To skip to different sections:		To save the application at any time:	

Project Information

Type of research project:	Undergraduate/Taught Postgraduate (MSc, MBA etc.)
Research based at:	Edinburgh
School research is for:	Engineering and Physical Sciences
Programme:	Other e.g. Combined Studies

Applicant Details

Student Details

First Name	Godfrey
Surname	Inyama
Email	gii2000@hw.ac.uk

Supervisor

Supervisor Details

First Name	Mauro
Surname	Dragone
School	School of Engineering and Physical Sciences
Campus	Heriot-Watt University
Telephone	0131 451 3325
Email	M.Dragone@hw.ac.uk

Project Summary

Project title:

Cloud based Robotics Platform for Outreach Activities

Ethical Considerations Check

Does your project involve any of the following? (Please check all that apply)

- [x] Human participants
- [ ] Personal data from external sources
- [ ] Living animals
- [ ] Medicines/Drugs/Medical Appliance
- [ ] None of the above

Type of Approval

Type of approval requested: Full Approval

External Approval

Is ethical approval required by another body linked to the research, e.g. the NHS or a collaborator?

- ( ) Yes
- (•) No

Details

State the question to be answered and the value of answering it (include the aims of the project):

This study involves testing a web-based application designed to simplify the programming of simple robot platforms, as part of STEM outreach activities targeting young school children.

The question to be answered are summarised into these two:
1. How can we improve the functionalities of the Web Application to make it more interesting and useful to 10 years old students during outreach activities?

2. How can we develop a strategy where Teachers can use the web application to teach Robotics as part of STEM Education

Outline

Give a brief outline of the proposed project, including the procedures to be used, the measurements to be made and how the data will be analysed.

Use specific names for relevant details: If you are surveying, specify how you will be surveying *(e.g. on Qualtrics, SurveyMonkey, GoogleForms)*, or if you are collecting data from Social Media specify which Social Media *(e.g. Facebook, Twitter)*

This proposed experimental study aims at evaluating the Web Application in terms of Users Experience, Technical Features, and also Server response to more users.

This will be carried out with the ROSbot 2 Pro as it is a very safe mobile robot. The ROSbot is an autonomous, open source robot platform based on ROS. It consists of four wheels and it's a four wheeled mobile robot. It also has a camera attached to it which is used to capture pictures and video around its environment when the sensor is activated. The ROSbot doesn't have any attachments which can cause any potential harm.

The study would involve Students, Teachers, and an Administrator. The Students and initial testers would be responsible for controlling the ROSbot while we access their experience at the end of the study by using Questionnaires.

The Teachers which will always be present during every study with the Students would be accessed based on the effectiveness of the Web Application in helping them teach the Students Robotics in a simpler and easier way. This Data will also be gotten from Questionnaires.

The System Administrator is basically someone with the responsible of ensuring that the servers are turned on before the experiment begin and responsible for any technical issues from the backend during the experiment. This will help me understand the common backend and server issues which may occur during future experiments and how to advert them. This will be assessed based on run time, down time, and server speed.

The method of getting these data will be from hard copies questionnaires prepared from MS Forms.

Previous Experience

Does the researcher(s) involved in the project have relevant training, or previous experience *(e.g. have previously researched or collected data)* in the field of investigation?

- (•) Yes
- ( ) No

Provide details of the previous training or previous experience. It is sufficient to provide a link to a training site, or research page etc.

This pilot study would be conducted under the supervision of Dr Muro Dragone (Project Supervisor) and with the support of the CARE group, whose members have conducted similar activities in the past.

Conflict of Interest

A "conflict of interest" is any case where the researcher(s) could profit in any way; personal or department, financial or otherwise, or if there is a pre-existing dependent relationship between a researcher and a participant (such as manager/employee, staff/student, client/consumer)

It is important to be transparent about any potential conflicts of interest to ensure any existing relationships are not conflicted by participation/non-participation.

Is there any potential "conflict of interest" relating to the proposed research project?

- ○ Yes
- ◉ No

Confidential Information

Does the project involve using confidential information that is not already in the public domain? *(This does not include the confidential information of e.g. a research participant, but considers external information for example about a company which should be used appropriately without confidential information being divulged)*

- ○ Yes
- ◉ No

Duration

State the likely duration of the project:

(Please note that the data collection can only commence following ethics approval)

One week

Location

In which country, or countries will the research take place? *If research is online select "Worldwide"*

United Kingdom

On which premise(s) or location(s) will data be collected?

If the project is to be undertaken online state this here and provide details. *E.g. if you are monitoring social media, specify which social media you are monitoring.*

In a Primary School classroom

Human Participants

State the type of participant(s) who will be involved?

Primary School Students (10 Years)

How many individuals will participate in the research? (State the maximum foreseen number).

25

Vulnerable Participants

Will any participants be from any of the following vulnerable groups?

- [x] Children
- [ ] People with learning disabilities
- [ ] Patients in hospital
- [ ] Participants with mental health issues
- [ ] Other (e.g. homeless people, refugees, people who lack capacity to consent etc.)
- [ ] N/A - Participants are not from any vulnerable groups

Disclosure

Working with vulnerable people may require a current and up-to-date Disclosure Certificate or equivalent.

Do you require a Disclosure Certificate, Protecting Vulnerable Groups (PVG) disclosure, or equivalent?

- ( ) Yes
- (•) No

Person Collecting Data

Who will collect the information from the participants?

Researcher with the help of the Teachers

Non-Standard Hardware

Will participants be using specialist hardware? *For example eye-trackers or development prototypes?*

- ○ Yes
- ◉ No

Physical Hazards

Are there any other potential physical hazards to participants including personal security?

- ○ Yes
- ◉ No

Recruitment

State how and where participants will be recruited. Be specific, if you are recruiting via Social Media please reference each Social Media Platform you will use:

Primary School invitations

How long will a participant have to decide whether to take part in the research?

One Day

Can you provide any copies of advertisements/recruiting matter you will be providing to participants?

- ○ Yes
- ◉ No

Compensation for Participation

Will compensation be provided to participants? (Financial or otherwise)

- ○ Yes
- ◉ No

Consent to Participate

Will informed consent to participate be obtained from all appropriate parties?

- ◉ Yes
- ○ No

How will you be obtaining informed consent to participate?

- ☑ Written consent
- ☐ Audio / verbal consent
- ☐ Electronic consent, e.g. via online survey
- ☐ Other

Upload the written consent form(s) here:

Documents

Type	Document Name	File Name	Version Date	Version	Size
Consent Form	Information letter and consent form - final	Information letter and consent form - final.docx			63.6 KB

As you are working with vulnerable participants, special considerations should be taken to ensure they are provided the time, opportunity and tools to make an informed decision regarding consent. Specify the special considerations to be made, *for example by providing pictures to assist understanding, or receiving consent from a parent, guardian, professional carer or relevant institution etc.*

Consent from School

Informing Participants

How will you tell individuals about the project, the use of their data, and who to contact if they want to find out more? (Select all that apply)

- ☐ Privacy Notice
- ☑ Participant Information Sheet
- ☐ Plain Language Statement
- ☐ Debrief Form
- ☐ Other
- ☐ N/A - Individuals will not be told about the project

Upload your Participant Information Sheet here:

Documents

Type	Document Name	File Name	Version Date	Version	Size
Privacy Notice	Data collection and management	Data collection and management.docx	15/08/2022	1.0	20.3 KB

De-Identification Procedures

Pseudonymisation is a de-identification procedure where personally identifiable information is replaced by an alternative identifier, or "pseudonym". When data is pseudonymised, a person can be re-identified by that data, but only with the use of additional data, or a 'key'. (Sometimes pseudonymised is known as "linked anonymised")

Anonymisation is a de-identification procedure whereby once personal data is completely anonymised it can never be used to re-identify a person (even in combination with other data).

Will you be using any pseudonymisation techniques or procedures?

- ○ Yes
- ◉ No

Will you be using any anonymisation techniques or procedures?

- ◉ Yes
- ○ No

Specify the anonymisation techniques that will be used, and at what point in the project this will be done?

Students names and profile will not be matched to any specific data and they will not be recorded

Emotional Discomfort or Distress

Will the project involve procedures that may cause emotional discomfort or distress to participants which may have long lasting or significant effects?

- ○ Yes
- ◉ No

Deception

Will the project involve deceiving a participant or providing incomplete disclosure?

- ○ Yes
- ◉ No

Will the project involve a deception or incomplete disclosure which could have any long lasting or significant effects on the participant?

- ○ Yes
- ◉ No
- ○ N/A

Mandatory or Voluntary?

Is participation in this project voluntary? *Participation may not be voluntary, for example if a participant does not know that they are being observed.*

- ◉ Yes - Participation is voluntary
- ○ No - Participation is not voluntary

Privacy Intrusive

Might the project require you to contact individuals in ways they may find intrusive to their privacy, and that may have a long lasting or significant impact on them?

- ○ Yes
- ◉ No

General Practitioner, Medical Specialist or Family Doctor Informed?

If the research may have an adverse impact on the physical or mental health of a participant, will the participants' Medical Specialist, General Practitioner or Family Doctor be informed of the recruitment of the participant before the research project begins?

(This includes any medical practitioner of whom the participant is a patient).

- ○ Yes
- ◉ No
- ○ N/A - There is no need to inform

Types of Data Collected and Analysed

Specify the categories of personal data to be collected and analysed: (Select all that apply)

- ☐ Political Opinions
- ☐ Religious or Philosophical Opinions
- ☐ Racial or Ethnic Origin
- ☐ Trade Union Membership
- ☐ Physical Health
- ☐ Mental Health
- ☐ Sex Life
- ☐ Sexual Orientation
- ☐ Alleged Offences or Proven Offences
- ☐ Gender Identity
- ☑ Other (e.g. name, age range, location, interactions, opinions etc.)

If "Other" specify the other categories of personal data:

Name, Age, Class Taught by Teacher

Potential Danger to Individuals

Will you be collecting or processing data that might endanger the individual's physical health or safety in the event of a security breach?

- ○ Yes
- ◉ No

Part 2 - Data Protection

You have now completed **Part 1 - Ethics**. The next set of questions will consider how the project plan incorporates data protection by design.

Data Protection and Privacy Notices

You were previously asked about how you would tell individuals about the project, the use of their personal data, and who to contact if they want to find out more *(for example a Privacy Notice, Participant Information Sheet, Plain Language Statement or Debrief Form)*.

Can you confirm that the given method of telling individuals about the nature and purpose of the project, either;

- Uses a template or wording approved by the Data Protection Officer; or

Includes all of the following:
- Data Protection Officer contact details: dataprotection@hw.ac.uk
- A link to the Heriot-Watt University Privacy Notice for Research Participants (or how to find it)
- Information on which party is the data controller

(•) Confirm
( ) Cannot Confirm

Privacy Invasive Technology or Algorithms

Will your research involve use of any technology or algorithm which may be perceived as being privacy intrusive, and may have a long lasting or significant effect on individuals? *For example consider any algorithms which may embed bias or discrimination?*

( ) Yes
(•) No

Systematic Monitoring

Will your research involve any systematic monitoring, which would include processes which observe, monitor or control individuals, and may have a long lasting or significant impact on individuals?

(•) Yes
( ) No

Profiling

Profiling is any form of automated processing of personal data to evaluate certain personal aspects relating to a person.
For example analysing or predicting aspects concerning that person's performance at work, economic situation, health, personal preferences, interests, reliability, behaviour, location or movements

Will your research include creating detailed profiles of individuals which may have a long lasting or significant impact on those individuals?

( ) Yes
(•) No

Decisions Against Individuals

Will the project result in you or others making decisions, or taking action against individuals in ways which can have a significant impact on them?

- ○ Yes
- ◉ No

Legality of Processing Section Statement

The following questions will be used to ensure that research projects are undertaken legally by determining who is responsible for deciding the purposes and means of data processing, and the legal basis for doing so.

Data Controller

Specify the data controller(s) for personal data processed in the course of the project - this is normally the University, unless the data is processed under contract to or in partnership with another organisation. Select at least one of the following:

- ☑ Heriot-Watt University
- ☐ Other

Legal Basis for Processing

State the legal basis for processing the personal data obtained in the course of the project. (Refer to ⓘ to determine which of the following to select).

The data subject has given consent to the processing of his or her personal data

Processing Outside of the EEA?

The following countries are currently in the European Economic Area (EEA):

Austria, Belgium, Bulgaria, Croatia, Republic of Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Liechtenstein, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden and the UK

The European Commission has designated the following countries are providing an adequate level of protection for privacy:

Andorra, Argentina, Canada (commercial organisations), Faroe Islands, Guernsey, Israel, Isle of Man, Japan, Jersey, New Zealand, Switzerland, Uruguay and the United States of America (limited to the companies operating under Privacy Shield framework)

Will the project involve transfers of data outside of the above listed countries, to organisations that are not members of Heriot-Watt

University Group?

- ○ Yes
- ◉ No

Further Questions Required?

Screening Process Indicates Further Questions May Be Required

Some of the questions you have answered in the previous sections have indicated that the research project could potentially result in high data protection risks to individuals' rights and freedoms. In this case, further questions will need to be asked which are called a Data Protection Impact Assessment (or DPIA).

It is possible a Data Protection Impact Assessment would not be required if there is another research project which:

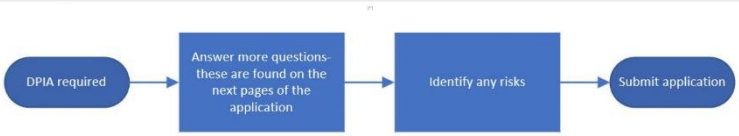
- Has the same methodology as the research project being proposed
- Has the same purpose as the research project being proposed
- Uses the same categories of data as the research project being proposed
- Uses the same risk mitigations as you will apply to the proposed project
- Has already completed a Data Protection Impact Assessment

Please advise whether a Data Protection Impact Assessment has already been completed for a research project which fulfils the above criteria, or if a new Data Protection Impact Assessment should be completed (if you are unsure please contact dataprotection@hw.ac.uk) :

- ◉ DPIA Required: The research project is unique to previous research projects, or; a Data Protection Impact Assessment has not yet been completed
- ○ DPIA Not Required: A Data Protection Impact Assessment has already been completed for a research project which fulfils the above criteria. The same risk mitigations which were applied in that project will be applied to this project.

How to Undertake a Data Protection Impact Assessment

What happens next?



As this is a potentially high risk project further questions will be asked in the next sections of this form. These questions will ask for more detail on the personal data you are processing, why you are processing it, and whether the individual understands what you are doing with their information. These questions form an assessment called a Data Protection Impact Assessment.

At the end of the form you will be presented with some potential risks for your project. You will be asked to determine whether these risks are high, medium or low risks depending on what measures you have put in place to reduce them. Information on how to decide whether a risk is high, medium or low can be found in the i-bubble on the right hand side of each question.

When you submit the form your application will be first reviewed by the School Ethics Committee, then by the Data Protection Officer who will decide whether to endorse your project, or suggest changes.

Please confirm that you have read and understood the above:

(•) Confirm

Information Security Requirements

The following questions will be used to determine the security measures that will be applied to protect data from accidental or deliberate unauthorised disclosure, loss, or alteration.

You previously stated that the research project will involve vulnerable groups.

For vulnerable groups there is a higher risk of physical or psychological harm if there is a security breach as a result of discrimination towards vulnerable groups, please consider this within your data management plan.

Information Flows

Describe the information flows in the project:

• Data will be primarily recorded in the researcher's laptop which is not managed by the university.

• User tracking data and log data will be recorded on the same server as the application, they will be stored in a local MySQL DB database server, whose access only lies to the researcher.
• Datasets published after the research will not contain any personal information or raw data without pre-processing.
• Survey data will be held on the HWU managed Microsoft Office365 server as they will be collected using Microsoft Forms.

Would you like to attach a flow diagram to describe the information flows?

(•) Yes
( ) No

Upload the flow diagram here:

Documents

Type	Document Name	File Name	Version Date	Version	Size
Information Flow Diagram	Questionnaire Sample for Pilot Study	Questionnaire Sample for Pilot Study.pdf	11/08/2022		435.5 KB

Access Control and Training

What training, procedures, information or induction will researcher(s) receive regarding Information Security prior to the commencement of the project? (Select all that apply)

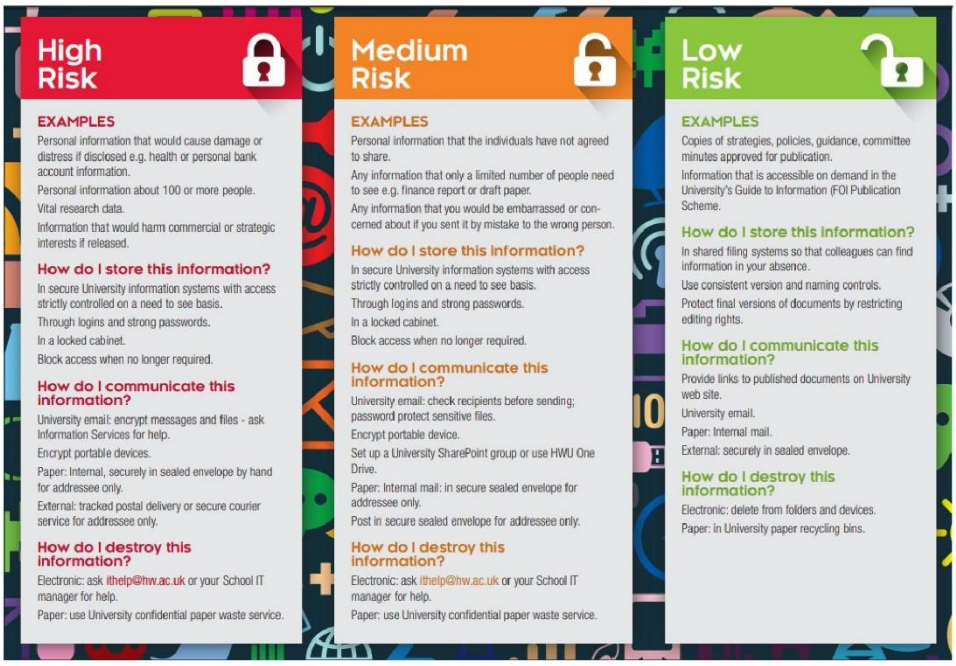
- [x] Non-Disclosure Agreements or Contractual Agreements
- [ ] Completion of Information Governance courses on Workrite
- [ ] Read the Heriot-Watt Information Governance Leaflet found https://www.hw.ac.uk/services/docs/information-governance/Infosecbasics_201605.pdf
- [ ] Other

Who will require access to the data, including Heriot-Watt staff and other partners, in order to complete work required for the research project? List any individuals and staff groups.

Supervisor and researcher

Heriot-Watt Information Security Standards

The following information is provided in Heriot-Watt Information Security guidance. It specifies the standard of controls that should be put in place to protect high, medium, and low risk information. Please read through this information carefully:



In particular:

- Data should be stored in secure Heriot-Watt University Information Systems (e.g. *Heriot-Watt University Office 365 account, Heriot-Watt Home or Shared Drive, Heriot-Watt One Drive*)
- Use strong password protection
- Portable devices should be encrypted
- Paper documents with confidential information should be disposed of using University confidential waste service
- Check email recipients before sending.

Will you adhere to all the above standards for storing, communicating and destroying information?

- (•) Yes
- ( ) No

External Third Parties

Will any data or information be sent outside of university managed IT systems (*Heriot-Watt University Office 365 account, Heriot-Watt Home or Shared Drive, Heriot-Watt One Drive etc.*)?

For example if the data is collected, stored or processed by a third party e.g. a cloud data storage company, an online survey software provider (such as Qualtrics), an application provider, market research company translator or transcriber?

- ( ) Yes – Data is stored or processed outside of Heriot-Watt IT Systems
- (•) No – Data is stored and processed within Heriot-Watt IT Systems

Remote Access

Is there a need to provide remote access to the data?

- ○ Yes
- ◉ No

Retention of Personally Identifiable Data

How long do you intend to retain personally identifiable data, *for example participant's names and contact details, or the participants' unique identifier*?

- ◉ Unless explicit consent is provided for a data subject to be named in the project outputs, personally identifiable data will only be kept for as long as it is necessary to keep the data in order to verify the integrity of the project's methods and to ensure the validity of the outputs.
- ○ Other

Consultation

Consider the practical steps you will take to ensure that you have correctly identified and addressed any data protection risks.

Identify who you have, or will, consult with regarding data protection including processing of personal data and information security:

Internal:

- ☑ Data Protection Officer
- ☑ Other Research Team Members
- ☐ Principal Investigator
- ☑ University Research Ethics Committee
- ☑ School Research Ethics Committee
- ☐ Other

External:

- ☐ Collaborative Partners
- ☐ Data Processors
- ☐ Data Providers
- ☐ Research Funders
- ☐ Contractors
- ☐ Sponsors
- ☑ Relevant Professional bodies
- ☐ Professional Accreditation for Research
- ☐ Other

Data subjects or

representatives:

- [x] Participants
- [x] Data subjects whose data has been obtained from Third Parties for use in the research
- [ ] Representatives of the data subjects
- [ ] General Public Consultation
- [ ] Targeted Consultation
- [ ] Other

Has the consultation already taken place?

- ○ Yes
- ◉ No

When will the consultation be undertaken?

In one week

Are there any individuals that it is not appropriate to consult with regarding data protection?

- ○ Yes
- ◉ No

UK Equality Act 2010

Under the UK Equality Act 2010 the University has a duty to ensure equality and diversity is embedded throughout its functions that impact on people. Where the need for a Data Protection Impact Assessment has been identified an Equality Impact Assessment may also be required. The Equality Impact Assessment form is published on the Staff Intranet here, or can be obtained by emailing equality@hw.ac.uk .

There is a risk that research projects may cause the University to be non-compliant with this act.

Are you additionally required to undertake an Equality Impact Assessment?

- ○ Yes
- ◉ No

Consent as a Legal Basis for Processing Risk

You previously stated you were relying on consent as the legal basis for processing.

There is a risk that if you are relying on consent as the legal basis for processing personal data and a participant withdrew consent, then the research project may be delayed (by having to obtain a new data set), or the project may no longer be able to progress.

If a participant subsequently withdrew their consent to process their personal data, explain the impact on the project and how you

would manage this.

No significant impact

Considering the impact to the project and how the withdrawal of consent would be managed, is the risk that the project may be delayed or may not be able to progress:

- ○ High
- ○ Medium
- ◉ Low

Previously Given Consent Risk

Are you relying on any consent that has previously been given? (The consent was given for a purpose other than this research project)

- ○ Yes
- ◉ No

Specific, Explicit and Legitimate Purposes

GDPR requires that personal data shall be collected for specific, explicit and legitimate purposes.

There is a risk that the reason for processing personal data is not explicit to participants.

Is the purpose for processing personal data in the project: "research in the public interest in order to achieve the objective of the research project"?

- ◉ Yes
- ○ No

Is the purpose of the project clearly explained to participants, *for example in the Participant Information Sheet, Plain Language Statement or Privacy Notice?*

- ◉ Yes
- ○ No

New Purposes

GDPR requires you not to process data for any reason other than the original purpose

As the research project develops there is a risk that a new purpose for processing personal data may be identified, resulting in personal data being used for that new purpose without the data subject's knowledge or consent.

Do you anticipate using data for any purpose other than for the reason it was originally intended?

○ Yes
◉ No

Excessive Data Risk

GDPR requires that the data you process is only the data that is necessary to achieve the purposes of the research project.

There is a risk that personal data is obtained that is not necessary for the purpose of undertaking the research project, resulting in higher unnecessary risk in case of data breaches, and poor data management. *For example, if you collect participant's Date of Birth when all you require is an age range.*

Are there any categories of personal data you could avoid using, without compromising the needs of the project?

○ Yes
◉ No

Insuffient Data Risk

GDPR requires that you process an adequate amount of data to complete the research project.

There is a risk that insufficient data will be collected resulting researcher being unable to complete the research project, or having inadequate data on which to base findings.

To mitigate this risk, how will you ensure you have collected sufficient data to achieve the intended purpose? *For example, explain how you have determined the number of participants required, and how you have determined the categories of data required.*

I have determined the numbers based on the available ROSbot 2 Pro available with a minimum of 10 Students as only 10 ROSbot are available

Following this, do you think that the risk of insufficient data being collected is:

○ High
○ Medium
◉ Low

Proportionality

Overall, are you collecting only data that is necessary for the research project?

◉ Yes - Only data that is necessary is being collected
○ No - More data is being collected than is necessary

Accuracy of Data Risk

GDPR requires that personal data is accurate and where necessary, is kept up-to-date.

There is a risk that inability to correct inaccurate data may result in invalid outputs of the research project.

To mitigate this risk does the software you are using allow you to amend inaccurate data when necessary?

- ○ Yes
- ◉ No

Accuracy - Participant Amendments

Is there a requirement for the research project to allow participants to update their data regarding the research (this does not include *for example, contact details*)?

- ○ Yes
- ◉ No - there is no requirement (e.g. participants updating their data may compromise the research project)

Rights of Data Subjects

A Data Subject may request that you provide them access to their data, or that inaccurate data is corrected, or that you stop processing their data or that their data is deleted.

If you receive such a request in the first instance you should email the Data Protection Team to handle the query.

If the Data Protection Team asked you to identify and retrieve all information regarding that data subject to assist them handling this query, what process would you need to do undertake to do this?

I would simply hand it over based on understanding the need for them

Vulnerable Participant Consent

Previously you stated the research project is working with vulnerable participants

As you are working with vulnerable participants, there is a risk that if those participants are not provided the time, opportunity and tools required for them to make an informed decision regarding consent, they may feel external pressures to provide consent, or that they may not have the capacity to make informed consent. Then consent may not be freely given, or it may be given uninformed.

You previously describe how vulnerable participants would be given special consideration when obtaining informed consent. Following the risk mitigation you have described, is this risk:

- ○ High
- ◉ Medium
- ○ Low

Systematic Monitoring Risk

You previously stated that the research would involve systematic monitoring, which would include processes which observe, monitor or control individuals *(including monitoring social media, CCTV etc.)*

There is a risk that systematic monitoring may infringe on the rights of the data subject; for example you may not be able to inform all data subjects of how their data is being processed and why, or individuals may not be able to avoid being subject to such processing.

Please detail any mitigations or precautions taken with regards to this risk. *For example, where possible you may seek to obtain consent to process the data post hoc.*

No risk as the monitoring will be visual

Following the risk mitigation you have described, is this risk:

- ○ High
- ○ Medium
- ◉ Low

Access Control Risk

You previously advised the data you would be processing is sensitive, or could potentially cause harm in case it is disclosed to unauthorised persons.

There is a risk that given the confidential information of the data that is being processed, that the access controls in place are not sufficient to protect the data.

Following implementation of access controls to ensure only authorised persons can access the data, is this risk:

- ○ High
- ◉ Medium
- ○ Low

Inadequate Security Controls Risk

You previously stated how you would communicate, store and delete research project data.

There is a risk that inadequate disclosure controls or security measures will increase the likelihood of data being shared inappropriately.

Following implementation of the methods you specified to secure communications, data in storage, and deletion of data, is the risk of confidentiality being compromised:

- ○ High
- ○ Medium
- ◉ Low

Risk Data is Not Truly Anonymised

You previously stated that you will be using anonymisation techniques to reduce risk to confidentiality of information being breached

There is a risk that the data will not be truly anonymised and that data subjects will still be able to be identified by other information, either from data in the data set or available elsewhere.

For example, removing a name from a data set that includes gender, ethnicity and postcode may allow that individual to be re-identified.

If it possible to remove a participant's data if they withdraw consent, then the data is not truly anonymised.

To mitigate this risk can you confirm that where necessary you have sought advice and are able verify that your data is truly anonymised?

- ◉ Confirm
- ○ Cannot Confirm

Inappropriate Retention Period Risk

You previously provided information on how long you would retain both personally identifiable and non-personally identifiable data:

If an appropriate retention period is not established this might mean information is retained or used for longer than necessary increasing risk of breaches, and non-compliance with Principle 5 of GDPR. If data is not stored for long enough there is a risk that critical data will be deleted when it is still required for the purposes of the research project.

Following implementation of your retention schedules, is this risk:

- ○ High
- ○ Medium
- ◉ Low

Additional Risks

Have you identified any other risks to individuals, the University or Legal Compliance?

- ○ Yes
- ◉ No

Other Ethics

Are there any ethical issues that you have identified which have not yet been addressed in this application?

- ○ Yes
- ◉ No

Other Documents

Are there any other documents which may support the ethics application which have not yet been uploaded?

- ◉ Yes
- ○ No

Other documents should be uploaded here:

Documents

Type	Document Name	File Name	Version Date	Version	Size
Miscellaneous Documents	Evaluation	Evaluation.docx	11/08/2022		14.6 KB

Applicant

Please confirm that the details completed in this form are accurate and a true reflection of the intended research project, and that if the details of the research project change significantly that you will seek additional ethical approval:

- ☑ Confirm

Applicant Signature:

Signed: This form was signed by Godfrey Inyama (gii2000@hw.ac.uk) on 15/08/2022 12:17

Supervisor

I (the supervisor) am satisfied that the researcher has properly considered the ethical implications of the intended research project and has taken appropriate action.

Supervisor
Signature:

Signature Request: Signature requested from Dr Mauro Dragone on 15/08/2022 12:18

How to Submit an Application

Once you have completed an application and have obtained all required signatures, click "Submit" to submit the application for review.

Submit

References

[1] Abu-Zeid, A., Glasgow, J. and Scott, S., 2007. Parameter analysis for robotic assessment of impairments in reaching due to stroke.

[2] Allen, S., Campbell, P.B., Dierking, L.D., Flagg, B.N., Friedman, A.J., Garibay, C. and Ucko, D.A., 2008, February. Framework for evaluating impacts of informal science education projects. In *Report from a National Science Foundation Workshop. The National Science Foundation, Division of Research on Learning in Formal and Informal Settings*.

[3] Angel-Fernandez, J.M. and Vincze, M., 2018. Towards a definition of educational robotics. In Austrian Robotics Workshop 2018 (Vol. 37).

[4] Angel-Fernandez, J.M., D 1.3-TOWARDS AN EXTENDED DEFINITION OF ER4STEM FRAMEWORK.

[5] Angel-Fernandez, J.M.,Lammer, L., Kynigos, C., Gueorguiev, I., Varbanov, P., Lepuschitz, W., Duca, A., Pullicino, J., Grizioti, M., Nikitopoulou, S. and Girvan, C., 2016. Best practice and requirements. TU Wien, University of Athens, ESI, Cardiff University, Across Limits and Certicos, Deliverable.

[6] Armoni, M., 2012. Teaching CS in kindergarten: How early can the pipeline begin?. Acm Inroads, 3(4), pp.18-19.

[7] Arumugam, R., Enti, V.R., Bingbing, L., Xiaojun, W., Baskaran, K., Kong, F.F., Kumar, A.S., Meng, K.D. and Kit, G.W., 2010, May. DAvinCi: A cloud computing framework for service robots. In 2010 IEEE international conference on robotics and automation (p. 3089). IEEE.

[8] Barker, B.S. and Ansorge, J., 2007. Robotics as means to increase achievement scores in an informal learning environment. *Journal of research on technology in education*, *39*(3), pp.229-243.

[9] Belpaeme, T., Kennedy, J., Ramachandran, A., Scassellati, B., & Tanaka, F. (2018). Social robots for education: A review. Science robotics, 3(21), eaat5954.

[10] Boehm, B. and Hansen, W.J., 2000. *Spiral development: Experience, principles, and refinements*. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.

[11] Burack, C., Melchior, A. and Hoover, M., 2019. Do after-school robotics programs expand the pipeline into STEM majors in college?. *Journal of Pre-College Engineering Education Research (J-PEER)*, *9*(2), p.7.

[12] C. Crick, G. Jay, S. Osentoski, and O. C. Jenkins, "ROS and rosbridge: Roboticists out of the loop," in Proc. Annu. ACM/IEEE Int. Conf. Human–Robot Interaction, 2012, pp. 493–494.

[13] Cañas, J.M., Per dices, E., García-Pérez, L. and Fernández-Conde, J., 2020. A ROS-based open tool for intelligent robotics education. Applied Sciences, 10(21), p. 4

[14] Cañas, J.M., Perdices, E., García-Pérez, L. and Fernández-Conde, J., 2020. A ROS-based open tool for intelligent robotics education. *Applied Sciences*, *10*(21), p.7419.

[15] Čejka, T. and Krejčí, R., 2016, April. Configuration of open vSwitch using OF-CONFIG. In NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium (pp. 883-888). IEEE.

[16] Cervera, E., Martinet, P., Marin, R., Moughlbay, A.A., Del Pobil, A.P., Alemany, J., Esteller, R. and Casan, G., 2016. The robot programming network. *Journal of Intelligent & Robotic Systems*, *81*(1), pp.77-95.

[17] Chu, Kar-Hai, Rachel Goldman, and Elizabeth Sklar. "Roboxap: an agent-based educational robotics simulator." In *Agent-based Systems for Human Learning Workshop at AAMAS-2005*. 2005.

[18] Collins, E.C., Prescott, T.J., Mitchinson, B. and Conran, S., 2015, November. MIRO: a versatile biomimetic edutainment robot. In Proceedings of the 12th international conference on advances in computer entertainment technology (pp. 1-4).

[19] CONSEQUENTIAL ROBOTICS. (n.d.). *MiRoCODE Characteristics* [Image]. Https://Www.Miro-e.Com/Mirocode. https://images.squarespace-

cdn.com/content/v1/5c0fcffd89c172c5222f2f07/1626251451144-PDHSZIUJCM9E9LX3EQKR/miroCloud+page-01-01.jpg?format=1000w

[20] Crick, C., Jay, G., Osentoski, S. and Jenkins, O.C., 2012, March. ROS and rosbridge: Roboticists out of the loop. In 2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI) (pp. 493-494). IEEE.

[21] Cyberbotics. (n.d.). *Webots web simulation server* [Image]. Https://Cyberbotics.Com/Doc/Guide/Web-Simulation. https://raw.githubusercontent.com/cyberbotics/webots/released/docs/guide/images/web_simulation_network_infrastructure.thumbnail.png

[22] Cyberbotics. (n.d.-a). *Synchronization of simulation and controller steps* [Image]. Https://Cyberbotics.Com/Doc/Guide/Controller-Programming. https://raw.githubusercontent.com/cyberbotics/webots/released/docs/guide/images/controller_synchronization.thumbnail.png

[23] Danahy, E., Wang, E., Brockman, J., Carberry, A., Shapiro, B. and Rogers, C.B., 2014. Lego-based robotics in higher education: 15 years of student creativity. International Journal of Advanced Robotic Systems, 11(2), p.27.

[24] DeLoach, S.A. and Matson, E., 2004, July. An organizational model for designing adaptive multiagent systems. In *The AAAI-04 Workshop on Agent Organizations: Theory and Practice (AOTP 2004)* (pp. 66-73).

[25] Demiris, Y., 2007. Prediction of intent in robotics and multi-agent systems. Cognitive processing, 8(3), pp.151-158.

[26] Demiris, Y. and Meltzoff, A., 2010. The robot in the crib: A developmental analysis of imitation skills in infants and robots. *Infant and Child Development: An International Journal of Research and Practice*, *17*(1), pp.43-53.

[27] Detry, R., Corke, P. and Freese, M., 2014. TRS: An open-source recipe for teaching/learning robotics with a simulator.

[28] Dittrich, J. and Quiané-Ruiz, J.A., 2012. Efficient big data processing in Hadoop MapReduce. Proceedings of the VLDB Endowment, 5(12), pp.2014-2015.

[29] Doerschuk, P., Bahrim, C., Daniel, J., Kruger, J., Mann, J. and Martin, C., 2016. Closing the gaps and filling the STEM pipeline: A multidisciplinary approach. *Journal of Science Education and Technology*, *25*(4), pp.682-695.

[30] Eguchi, A., 2017. Bringing robotics in classrooms. In Robotics in STEM education (pp. 3-31). Springer, Cham.

[31] Faller, T., 2018. Teaching through Videogames (Doctoral dissertation, Alpen-Adria-Universität Klagenfurt).

[32] Fincher, S. and Petre, M. eds., 2004. *Computer science education research*. CRC Press.

[33] Fincher, S. and Petre, M., 2005. Programming environments for novices. In *Computer science education research* (pp. 137-164). Taylor & Francis.

[34] Fong, T., Nourbakhsh, I. and Dautenhahn, K., 2003. A survey of socially interactive robots. *Robotics and autonomous systems*, *42*(3-4), pp.143-166.

[35] Fraser, N., 2015, October. Ten things we've learned from Blockly. In 2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond) (pp. 49-50). IEEE.

[36] Goldberg, K. and Siegwart, R. eds., 2002. Beyond Webcams: an introduction to online robots. MIT press.

[37] Goldman, R., Eguchi, A. and Sklar, E., 2004, October. Using educational robotics to engage inner-city students with technology. In *Proceedings of the 6th international conference on Learning sciences* (pp. 214-221).

[38] Guyot, L., Heiniger, N., Michel, O. and Rohrer, F., 2011, September. Teaching robotics with an open curriculum based on the e-puck robot, simulations and competitions. In *Proceedings of the 2nd International Conference on Robotics in Education. Vienna, Austria*.

[39] Guzmán, J.L., Berenguel, M., Rodríguez, F. and Dormido, S., 2008. An interactive tool for mobile robot motion planning. *Robotics and Autonomous Systems*, *56*(5), pp.396-409.

[40] Han, J., 2010, February. Robot-aided learning and r-learning services. In Human-robot interaction (p. 288). Rijeka, Croatia: InTech.

[41] Inaba, M., Kagami, S., Kanehiro, F., Hoshino, Y. and Inoue, H., 2000. A platform for robotics research based on the remote-brained robot approach. The International Journal of Robotics Research, 19(10), pp.933-954.

[42] Jost, B., Ketterl, M., Budde, R. and Leimbach, T., 2014, December. Graphical programming environments for educational robots: Open roberta-yet another one?. In *2014 IEEE International Symposium on Multimedia* (pp. 381-386). IEEE.

[43] Kato, Y., Izui, T., Tsuchiya, Y., Narita, M., Ueki, M., Murakawa, Y. and Okabayashi, K., 2011, November. Rsi-cloud for integrating robot services with internet services. In IECON 2011-37th Annual Conference of the IEEE Industrial Electronics Society (pp. 2158-2163). IEEE.

[44] Kay, A., 2005. Squeak etoys, children & learning. *online article, 2006*.

[45] Kay, K. and Greenhill, V., 2011. Twenty-first century students need 21st century skills. In Bringing schools into the 21st century (pp. 41-65). Springer, Dordrecht.

[46] Kelleher, C. and Pausch, R., 2005. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR), 37*(2), pp.83-137.

[47] Khanlari, A., 2013. Effects of robotics on 21st century skills. European Scientific Journal, 9(27).

[48] Kruger, D., Stolkin, R., Blum, A. and Briganti, J., 2007, April. Optimal AUV path planning for extended missions in complex, fast-flowing estuarine environments. In *Proceedings 2007 IEEE International Conference on Robotics and Automation* (pp. 4265-4270). Ieee.

[49] Lammer, L., Lepuschitz, W., Kynigos, C., Giuliano, A. and Girvan, C., 2017. ER4STEM educational robotics for science, technology, engineering and mathematics. In Robotics in Education (pp. 95-101). Springer, Cham.

[50] M. Narita, M. Shimamura, R. Hiura, and T. Yamaguchi. "A common platform for robot services through robot services initiative (RSi) activities". The 4th International Conference on Ubiquitous Robots and Ambient Intelligence (URAIS2007), Korea, pp. 1029-1034, 2007.

[51] M. Waibel, M. Beetz, J. Civera, R. d'Andrea, J. Elfring, D. Galvez-Lopez, B. Schiessle. "Roboearth-a world wide web for robots". IEEE Robotics and

Automation Magazine (RAM), Special Issue Towards a WWW for Robots, vol.18, no.2, pp.69-82, 2011

[52] M. Waibel, M. Beetz, J. Civera. "Roboearth". IEEE Robot, vol.18, no.2, pp.69-82,2011.

[53] Mamani, Nibeth Mena, et al. "A systematic mapping about simulators and remote laboratories using hardware in the loop and robotic: Developing STEM/STEAM skills in pre-university education." 2021 International Symposium on Computers in Education (SIIE). IEEE, 2021.

[54] Markram, H., 2012. The human brain project. Scientific American, 306(6), pp.50-55.

[55] Martin, E., 2004. Vignettes and respondent debriefing for questionnaire design and evaluation. Methods for testing and evaluating survey questionnaires, pp.149-172.

[56] Masse, M., 2011. REST API design rulebook: designing consistent RESTful web service interfaces. " O'Reilly Media, Inc.".

[57] McGrath, B., Sayres, J., Lowes, S. and Lin, P., 2008. Underwater LEGO robotics as the vehicle to engage students in STEM: The BUILD IT project's first year of classroom implementation. *American Society for Engineering Education Mid-Atlantic, Hoboken, NJ*.

[58] Mitchinson, B. and Prescott, T.J., 2016, July. MIRO: a robot "Mammal" with a biomimetic brain-based control system. In Conference on biomimetic and biohybrid systems (pp. 179-191). Springer, Cham.

[59] Mohanarajah, G., Hunziker, D., D'Andrea, R. and Waibel, M., 2014. Rapyuta: A cloud robotics platform. IEEE Transactions on Automation Science and Engineering, 12(2), pp.481-493.

[60] Moher, D., Shamseer, L., Clarke, M., Ghersi, D., Liberati, A., Petticrew, M., Shekelle, P. and Stewart, L.A., 2015. Preferred reporting items for systematic review and meta-analysis protocols (PRISMA-P) 2015 statement. Systematic reviews, 4(1), pp.1-9.

[61] Montemerlo, M. and Thrun, S., 2007. FastSLAM 1.0. FastSLAM: A scalable method for the simultaneous localization and mapping problem in robotics, pp.27-62.

[62] Montemerlo, M., Thrun, S., Koller, D. and Wegbreit, B., 2002. FastSLAM: A factored solution to the simultaneous localization and mapping problem. Aaai/iaai, 593598.

[63] Nemiro, J., Larriva, C. and Jawaharlal, M., 2017. Developing creative behavior in elementary school students with robotics. The Journal of Creative Behavior, 51(1), pp.70-90.

[64] Ngai, G., Chan, S.C., Cheung, J.C. and Lau, W.W., 2009. Deploying a wearable computing platform for computing education. *IEEE Transactions on Learning Technologies*, *3*(1), pp.45-55.

[65] Nourbakhsh, I.R., Crowley, K., Bhave, A., Hamner, E., Hsiu, T., Perez-Bergquist, A., Richards, S. and Wilkinson, K., 2005. The robotic autonomy mobile robotics course: Robot design, curriculum design and educational assessment. *Autonomous Robots*, *18*(1), pp.103-127.

[66] Nugent, G., Barker, B., Grandgenett, N. and Adamchuk, V., 2009, October. The use of digital manipulatives in k-12: robotics, GPS/GIS and programming. In *2009 39th IEEE Frontiers in Education Conference* (pp. 1-6). IEEE.

[67] Olivier, M., 2004. Cyberbotics ltd-webotstm: Professional mobile robot simulation. International Journal of Advanced Robotic Systems, 1(1), pp.40-43.

[68] Oppliger, D., 2002, November. Using first Lego league to enhance engineering education and to increase the pool of future engineering students (work in progress). In *32nd annual frontiers in education* (Vol. 3, pp. S4D-S4D). IEEE.

[69] Papert, S., 1990. Children, computers and powerful ideas. *New York: Basic Books*, *10*, p.1095592.

[70] Park, D., Wang, J. and Kee, Y.S., 2016. In-storage computing for Hadoop MapReduce framework: Challenges and possibilities. IEEE Transactions on Computers.

[71] Park, I.W. and Han, J., 2016. Teachers' views on the use of robots and cloud services in education for sustainable development. Cluster Computing, 19(2), pp.987-999.

[72] Parkes, A.J., Raffle, H.S. and Ishii, H., 2008, April. Topobo in the wild: longitudinal evaluations of educators appropriating a tangible interface. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 1129-1138).

[73] Pasternak, E., Fenichel, R. and Marshall, A.N., 2017, October. Tips for creating a block language with blockly. In 2017 IEEE blocks and beyond workshop (B&B) (pp. 21-24). IEEE.

[74] Petters, S., Thomas, D., Friedmann, M. and Stryk, O.V., 2008, November. Multilevel testing of control software for teams of autonomous mobile robots. In *International Conference on Simulation, Modeling, and Programming for Autonomous Robots* (pp. 183-194). Springer, Berlin, Heidelberg.

[75] Pivkina, I., Pontelli, E., Jensen, R. and Haebe, J., 2009, March. Young women in computing: lessons learned from an educational & outreach program. In *Proceedings of the 40th ACM technical symposium on Computer science education* (pp. 509-513).

[76] Prescott, T.J., Mitchinson, B. and Conran, S., 2017, March. Miro: An animal-like companion robot with a biomimetic brain-based control system. In Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction (pp. 50-51).

[77] Reich-Stiebert, N. and Eyssel, F., 2015. Learning with educational companion robots? Toward attitudes on education robots, predictors of attitudes, and application potentials for education robots. International Journal of Social Robotics, 7(5), pp.875-888.

[78] Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. and Kafai, Y., 2009. Scratch: programming for all. *Communications of the ACM*, *52*(11), pp.60-67.

[79] Resnick, M. and Robinson, K., 2017. Lifelong kindergarten: Cultivating creativity through projects, passion, peers, and play. MIT press.

[80] Ruiz-del-Solar, J. and Avilés, R., 2004. Robotics courses for children as a motivation tool: the Chilean experience. *IEEE Transactions on Education*, *47*(4), pp.474-480.

[81] S. Nakagawa, N. Igarashi, Y. Tsuchiya, M. Narita, Y. Kato. "An implementation of a distributed service framework for cloud-based robot services". IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society, Canada, pp.4148-4153, 2012.

[82] Saha, Dasgupta, O. P. A. (2018, August 17). *Communication Paths with Cloud Infrastructure* [Image]. Https://Www.Mdpi.Com/2218-6581/7/3/47. https://www.mdpi.com/robotics/robotics-07-00047/article_deploy/html/images/robotics-07-00047-g001-550.jpg

[83] Schön, S. and Ebner, M., 2020. Open Educational Resources in Austria. In *Current State of Open Educational Resources in the "Belt and Road" Countries* (pp. 17-33). Springer, Singapore.

[84] Scribner-MacLean, M. and Miller, H., 2009, October. Online Team Teaching: Strategies for Success for Creating an Online Learning Community. In *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education* (pp. 1957-1961). Association for the Advancement of Computing in Education (AACE).

[85] Soares, J.M., Navarro, I. and Martinoli, A., 2016. The Khepera IV mobile robot: performance evaluation, sensory data and software toolbox. In *Robot 2015: second Iberian robotics conference* (pp. 767-781). Springer, Cham.

[86] Solomon, C., Harvey, B., Kahn, K., Lieberman, H., Miller, M.L., Minsky, M., Papert, A. and Silverman, B., 2020. History of logo. Proceedings of the ACM on Programming Languages, 4(HOPL), pp.1-66.

[87] Stager, G.S., 2016. Seymour Papert (1928–2016). Nature, 537(7620), pp.308-308.

[88] Staker, H. and Horn, M.B., 2012. Classifying K–12 blended learning.

[89] Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A. and Goodrich, M., 2006, March. Common metrics for human-robot interaction. In Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction (pp. 33-40).

[90] Stewardson, G.A., Robinson, T.P., Furse, J.S. and Pate, M.L., 2019. Investigating the relationship between VEX robotics and student self-efficacy: An initial look. International Journal of Technology and Design Education, 29(4), pp.877-896.

[91] Stubbs, K.N. and Yanco, H.A., 2009. STREAM: A workshop on the use of robotics in K–12 STEM education. *IEEE Robotics & Automation Magazine*, *16*(4), pp.17-19.

[92] Taban, F., Acar, E. and Fidan, I., 2005, June. Teaching basic engineering concepts in K 12 environment using Lego bricks and robotics. In *2005 Annual Conference* (pp. 10-1195).

[93] Tengler, K., Kastner-Hauler, O. and Sabitzer, B., 2021, April. Enhancing Computational Thinking Skills using Robots and Digital Storytelling. In CSEDU (1) (pp. 157-164).

[94] Touretzky, D.S. and Tira-Thompson, E.J., 2005, July. Tekkotsu: A framework for AIBO cognitive robotics. In *AAAI* (pp. 1741-1742).

[95] Turnbull, L. and Samanta, B., 2013, April. Cloud robotics: Formation control of a multi robot system utilizing cloud infrastructure. In *2013 Proceedings of IEEE Southeastcon* (pp. 1-4). IEEE.

[96] ÜÇGÜL, M., 2013. History and educational potential of Lego Mindstorms NXT. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, *9*(2), pp.127-137.

[97] Verified Market Research. (n.d.). Educational Robots Market Size, Share, Trends, Opportunities & Forecast. [online] Available at: https://www.verifiedmarketresearch.com/product/global-educational-robot-market-size-and-forecast-to-2025/ [Accessed 15 Aug. 2022].

[98] Waibel, M., Beetz, M., Civera, J., d'Andrea, R., Elfring, J., Galvez-Lopez, D., Häussermann, K., Janssen, R., Montiel, J.M.M., Perzylo, A. and Schiessle, B., 2011. Roboearth. IEEE Robotics & Automation Magazine, 18(2), pp.69-82.

[99] Walker, E. and Burleson, W., 2012, June. User-centered design of a teachable robot. In International Conference on Intelligent Tutoring Systems (pp. 243-249). Springer, Berlin, Heidelberg.

[100] Weinberg, G. and Driscoll, S., 2007, March. The interactive robotic percussionist-new developments in form, mechanics, perception and interaction design. In *2007 2nd ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (pp. 97-104). IEEE.

[101] Wikipedia Contributors (2019). Human–robot interaction. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Human%E2%80%93robot_interaction.

[102] Yang, Y., Long, Y., Sun, D., Van Aalst, J. and Cheng, S., 2020. Fostering students' creativity via educational robotics: An investigation of teachers' pedagogical practices based on teacher interviews. British journal of educational technology, 51(5), pp.1826-1842.

[103] Zhang, Z., Huang, Y., Chen, S., Qu, J., Pan, X., Yu, T. and Li, Y., 2017. An intention-driven semi-autonomous intelligent robotic system for drinking. *Frontiers in neurorobotics*, *11*, p.48.