

(Vision for the Future)

**College of Arts and Applied Sciences**

**Department of Computer Science**

**CMPS 449 – Final Year Project I**

**TITLE OF PROPOSAL**

(**MOBILE ROBOT WITH** **PATH FINDING IN DHOFAR UNIVERSITY**)

By

|  |  |
| --- | --- |
| **ID STUDENT** | **NAME OF STUDENT** |

**202010045** **AYA YOUSRY HASSAN**

**201910963**  **AMAL ISSA ALAAMRI**

**Final Year Project Proposal Submitted to the College of Arts and Applied Sciences, Dhofar University, in Fulfilment of the Requirement for the Degree of**

**Bachelor of Science in Computer Science**

**January- 2024**

**FALL 2023-2024**

**Supervised By:**

**DR. MUKESH MADANAN**

**Appendix J: Declaration Form**

**DECLARATION**

We hereby declare that the proposal is our original work except for quotes and citations that have been appropriately acknowledged. We also declare that it has not previously been submitted and is not currently being submitted for any other degree at Dhofar University or any other institution. This proposal report may be made available within the university library and maybe photocopied and loaned to other libraries for consultation.

(Signature)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Aya Yousry Hassan**

Date:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature)

**Amal Issa Alaamri**

Date:

**TABLE OF CONTENTS**

**Page**

**CHAPTER 1 INTRODUCTION**

* 1. Background 6
  2. Problem Statement 7
  3. Objectives 7
  4. Project Questions 8
  5. Scope of the Project 9
  6. Contribution 10
  7. Limitation 11

**CHAPTER 2 LITERATURE REVIEW 11**

* 1. Introduction 11
  2. Literature Review 1- *The system of mobile robot with path finding* 11,12,13
  3. Literature Review 2 - *Models were used (Environment model) 14*
  + Grid method 14
  + Topological method 14
  + Geometric characteristic method 15
  + Mixed representation 15, 16

**CHAPTER 3 METHODOLOGY 17**

* 1. Introduction of prototype model 17
  2. Analysis Phase 18
     1. The overview of the project 18
     2. Project’s components 18,19,20
     3. Requirements analysis 20
     4. Stakeholders 20
     5. Functional Requirements 21
     6. Non-functional Requirements 21,22
  3. Design Phase 23
     1. Creating the exterior part of the robot 23,24,25
     2. Design phase (using Diagrams) 24
     + Data Flow Diagram (DFD) 25,26
     + Flowchart Diagram 26,27
     + Entity Relationship Diagram (ERD) 28,29
  4. Development Phase 30
     + Structure of the system 30
  5. Implementation Phase 31
  6. Evaluation Phase 32
  7. Tools Used 33
     + Hardware tools 33
     + Software tools 33

**CHAPTER 4 PROJECT PLAN 34**

4.1 Project Plan 34

* 1. Gantt Chart 35
  2. Network Diagram 36

**CHAPTER 5 SOFTWARE REQUIREMENTS SPECIFICATION 37**

1. Introduction 37
2. Overview of the system 37
3. Functional Requirements 38,39
4. Non-functional Requirements 39
5. Limitations 39

**CHAPTER 6 CONCLUSION 40**

**REFERENCES 41,42,43,44**

**List of figures**

**Fig1** Robot Navigation 10

**Fig2** Simple and random traversal 12

**Fig3** Environment modeling 12

**Fig4** Obstacle avoidance 13

**Fig5** Grid method 14

**Fig6** Voronoi diagram 14

**Fig7** PWM on Arduino UNO 24

**Fig8** Wireless communication 24

**Fig 9** Arduino battery level 25

**Fig10** Arduino Wi-fi ESP32 connections 25

**Fig11** Data Flow Diagram (DFD) 26

**Fig12** Flowchart Diagram 27

**Fig13** Entity relationship diagram (ERD). 29

**Fig14** Cardinality arrows 29

**List of tables**

**Table 1** Project’s components 23

**Table 2** ERD description 28

**Table 3** project plan 34

**Table 4** Gantt Chart 35

**Table 5** Network Diagram 36

**CHAPTER 1: INTRODUCTION**

* 1. Background

In various industrials, robots are widely developed in the 21st century. There are many sectors that implement robotic systems in executing technical processes **[19].** Robotic systems are getting advanced and can perform in short period of time flowless work in addition to the contribution in different factors like time, safety, quality etc. **[1].** This project explains the latest improvements in mobile robot with path finding and their applications in the field of semi-autonomous robot navigation. Path finding is an important component of many applications in the realm of robotics. GPS, crowd simulation and logistics. Also, it can be implemented in dynamic, static, and real-time environments. In addition, path finding is one of the most core domains in the research of mobile robots. With the increasing demand for intelligence and automation of mobile robots and remote control, robots are also needed to find paths of collision-free automatically.

Path finding algorithms can be used by mobile robots to identify efficient, collision-free, safe, and shortest travel paths from an origin to a destination. By choosing a suitable path finding algorithms which helps to get effective and safe point-to-point navigation and choosing the best algorithms which based on the geometry of robots as well as the computing constraints**.[1]**

Self-learning robots that can work without human guidance are demanded in automation area. Different types of robots are worked in various fields such as hospitals and houses, industries, and offices. one of the most crucial robots in the realm of robotic systems is mobile robot. Mobile robot is a machine that is controlled by software that use sensors and other technology to move and navigate in its given environments by focusing on two important tasks which are localization and navigation. Localization means the position of robot in its frame of reference. Normalization means a robot movement with avoiding obstacles in the given environment. **[2].**

The mobile robot can be classified by two categories, by the device they use to move and by environment in which they work. The type of environment mobile robot that will be covered in this project is a land robot which is used in land. The device of mobile robot will include wheels**.[2]**

* 1. Problem Statement

It is unsafe for inspectors to conduct testing and inspection work in those risky and dangerous areas. In addition, the lack of background information **[5]** about the unknown area can cause an insecure situation and difficulties for inspectors. The problems can be defined:

Develop a website to control and monitor mobile robot path finding. The website should enable users to input destination points, visualize the robot's real-time location, and receive status updates during the navigation process. The key challenge is to design an intuitive interface that allows users to interact seamlessly with the robot, ensuring efficient and user-friendly remote control for navigation in various environments.

The Path finding Robot website project intends to provide a remedy that makes use of cutting-edge technology to handle these problems considering these difficulties. The goal of the project is to create semi-autonomous robots that can help with navigation**.**

* 1. Objectives

Below are the key objectives guiding the development of this semi-autonomous pathfinding robot:

* Develop Semi-Autonomous Navigation System enabling the robot to navigate within the university environment.
* Implement A\* Algorithm for Semi-Autonomous Path Planning considering the university environment's layout and user-defined destinations.
* Interactive Web Interface for User Guidance allowing users to input general directives.
* Utilize Arduino components to control the robot's movements.
* Semi-Autonomous Obstacle Avoidance so develop algorithms for semi-autonomous obstacle detection and avoidance, empowering the robot to make real-time decisions during navigation.
* Calibrate Sensors for Semi-Autonomous Accuracy to ensure accurate data collection and reliable obstacle detection during operation.
* Extensive Testing of robot Features: Perform thorough testing of semi-autonomous features, including decision-making processes, obstacle avoidance, and user-guided navigation.
  1. Project Questions

**Efficiency and Navigation**:

* How can semi-autonomous path finding robots improve visitors', staff members', and students' ability to navigate the campus?
* b. What are the main obstacles to better campus navigation, and how can robots overcome them?

**User Experience and Assistance**:

* How can robots help users, especially new arrivals and visitors, find their way around the college campus?

**Scalability**:

* a. How can the project be planned to be flexible enough to grow with the campus as it expands and as technology advances?
* b. What factors need to be considered for the system's expansion and adaptation in the future?
* Are there plans for integrating additional technologies or features to further improve the system?
  1. Scope of the Project

The robotics field has improved in recent years back. Robotics is referred to as some system of electronic components or semi-autonomous **[6]** system that accomplishes some tasks. The path’s plan has been taken by a robot in a closed environment which can be done in various ways. The determining methods of a path include.

**Navigation of Robotic**

Robotic navigation is the ability to generate the optimal path for finding the desired location. It includes lots of means and techniques of gathering information. Path planning /path finding algorithms are used to assist in configuration the location of the robot and predicting the next move. In the realm of robotics, navigation also involves object detection/ avoidance**.[15]**

**Object detection/ avoidance**

Object avoidance refers to when HRI being able to identify the objects that are interfering in the path. Obstacles like walls refer to static obstacles when dynamic obstacles refer to moving such as humans on a sidewalk. On the other side, object detection can be accurate via sensors which can avoid obstacles successfully. With computer vision which encompasses facial recognition, object detection, and other aspects of sensor recognition. **[8]**

**Predetermined Environment**

A path’s plan algorithm can easily take out set of coordinates for the robot to follow. Generally, it’s the means of testing many algorithms, a predetermined grid of a size is generated showing where on the map is “passable”. To assume for testing it should be safe that all borders of the grid are easily reachable through by the robot**.[17]**

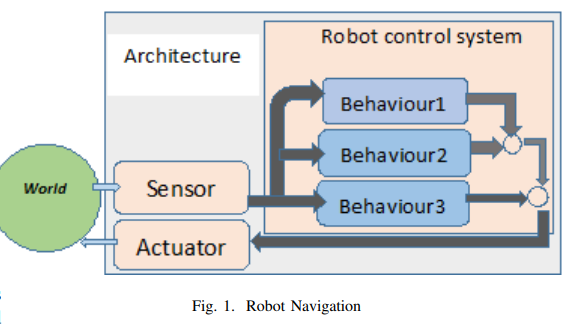
* 1. Contribution

Since decades ago, mobile robots had been applied successfully in various areas like security environment, industry, and military for executing important unmanned missions. Autonomous robots can operate and move without a human directly controlling. The purpose of path finding robot is searching a safe path for the autonomous robot additionally, it improves the efficiency of the tasks which involves navigation and movement that can done by finding an optimal paths. It also can reduce energy consumption and travel time. Path finding robot can also save time for different tasks such as taking the shortest routes, inspecting large areas quickly, and making deliveries faster. **[14]** To identify the shortest routes, can be done by using sensors or graph based on the algorithms. In the future the mobile robot can move successfully from start to goal through by avoiding the obstacles in the path without human operation. There are contributions that based on the semi-autonomous navigation problem with special emphasis on outdoor environment of the test area, these contributions are:

* **laser scanner based on sensing** involves the extraction feature that is used to separate traversable **[10]** road areas from non-traversable road aspects and other obstacles.

* **Obstacle avoidance** involves the management of obstacles, sensor fusion and determination of the short path in response to the detected obstacles**.[6]**
* **Vision based on extraction feature** shows the position determination of artificial guide marks and the determination of available road area at longer ranges.
* **Planning mission and execution** involves the scheduler of navigation which combines the mission planned with obstacle avoidance.
  1. Limitation

Robots are machines which mimic human activities through sensors for input, actuators for making decisions and effectors for output. The ability of robots to generate an optimal path is called robot navigation, the problem is to find desired location through by avoiding obstacles by considering the size and shape of the obstacles and conserve energy. To construct every part which can be tested, sensors should certainly recognize things such as sounds and pictures accurately. For effectors should be quick enough to do what they require them to do and flexible and the capacity for exploring in obscure environment in people is important. However, in a dynamic environment where there are several numbers of people and vehicles is challenging for mobile robot to navigate and that is because of the complex structure of a given environment. The following navigation environment architecture is presented. **[18]**



**CHAPTER 2: LITUERATURE REVIEW**

* 1. The system of mobile robot with path finding

Path finding robot is a technology that is related to robotics systems that robot can plan and follow a path from one place to another. The main aim of path finding robot is to move from its current place position to intended place with following a safe and efficient route and avoiding obstacles.**[9]** The techniques of the traversal path finding robot for unknown environment can be identified in several strategies:

1. **Simple and random traversal**

In this strategy like in **Fig2**, the robot runs forward continuously, crashes into objects and then turns an angle randomly then continues to run forward to achieve complete traversal.

Fig2. Simple and random traversal

1. **Modeling the environment**

After learning the edge of environmental along and performing the local pathfinding according to the existing environmental model like grid method, cell decomposition. In cell decomposition method which divides the global environment into many sub-regions, every sub-region matches with a unique base point which represents this sub-region like in **Fig3**. A complete traversal procedure is to transform into finding a path on the complex network in the map to pass through all the base points. For grid method, it divides map into a series of grids. With this case it can be avoided the complex calculations when deals with the boundaries of obstacles.

Fig3. Environment modeling

1. **Walking path finding**

A robot doesn’t need to model the environment in advance. In the movement approach, the robot gets real-time environmental information by on-board sensor like laser scanner or camera, and using an algorithm to find an optimal direction according to information and robot moves along the specific direction. A procedure to find the next local optimal direction to proceed in is called local path planning.**[18]**

1. **Mapping**

A robot has a need to have access to a map of a surrounding area. The map can be created by (SLAM) techniques which is referred to simulates localization and mapping.

1. **Obstacle avoidance**

A robot can avoid and detect obstacles by various algorithms and sensors that enable them to make informed decisions like navigation adjustments and collision detection to avoid the obstacles around the robot while walking to its determined location. **[13]**

Fig4. Obstacle avoidance

* 1. Models were used (Environment model)

The model which was used in the mobile robot with path finding was environment modeling. Environment modeling is the utilization and organization of perceived environment. An environment model can help mobile robots to search for the path effectively. There are 4 methods that can build environment models which are GM (grid method), TM (topological method), GCM (geometric characteristic method), MR (mixed representation).

1. **Grid method**: it is a method that splits a workspace of mobile robot into a series of networks cells with binary information and the size of the grid. The following grid environment covers the size of grid and the splitting workspace **[3]**.



Fig5. Grid method

1. **Topological method**: it is a basic idea to use nodes for representing a particular location and edges for representing the connection of these locations **[3].**

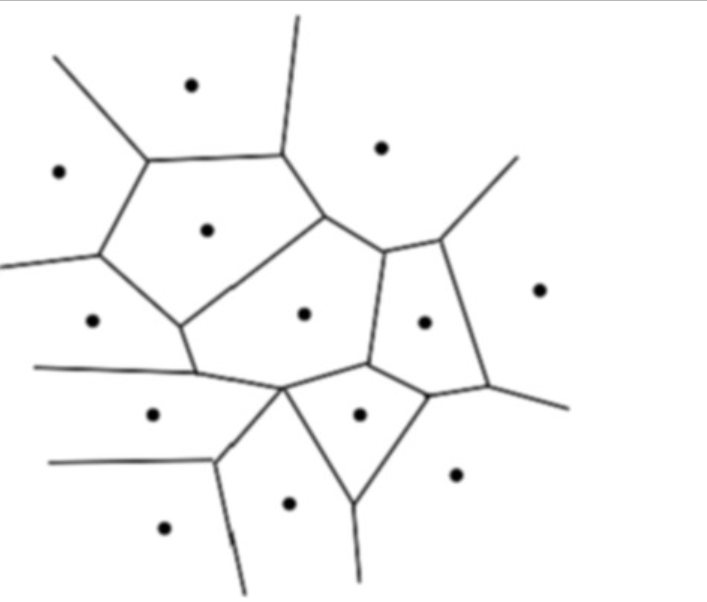


Fig6**.** Voronoi diagram

1. **Geometric characteristic method**: it is a method that how the mobile robot extracts relatively abstract geometric features from the environment like arcs, circles, and lines.
2. **Mixed representation**: to extract topological features from global metric maps is called mixed representation, by using the distance information from sensors to extract features during the navigation process **[3].**

* **Variants and Algorithm A: \***

The A\* algorithm is a popular heuristic search technique that has been successfully used in a number of pathfinding robotics projects. Scholars have investigated modifications of the A\* algorithm, like D\* (Dynamic A\*), to allow robots to adjust their paths dynamically in response to obstacles or shifting surroundings. These models have demonstrated efficacy in route optimization and traversal time reduction.**[8]**

* **Machine Intelligence for Navigation:**

Previous pathfinding robot projects have benefited greatly from the use of machine learning models. Robots that have been programmed to learn and modify their navigational strategies have been equipped with reinforcement learning algorithms such as Q-learning and Deep Q-Networks (DQNs). These models have proven to be capable of learning from mistakes, which is useful for adjusting to changing surroundings.**[9]**

* **Simultaneous Localization and Mapping (SLAM):**

A lot of pathfinding robot projects have used SLAM strategies to map their environment and figure out where they are in relation to it at the same time. SLAM algorithms, like Fast SLAM and Graph SLAM, have been used by researchers and have proven essential in developing a thorough understanding of the surroundings**.[11]**

* **Artificial Neural Networks and Swarm Robotics:**

In earlier studies, pathfinding robots were created to function in multi-agent systems, where a group of robots work together to solve challenging navigational issues. Swarm robotics models, which draw inspiration from social insects, have been used to develop cooperative behaviors in robots, enabling them to function well together in unstructured settings**.[7]**

* **User Interaction Models**:

To establish a smooth interface between people and pathfinding robots, research has concentrated on user interaction models. Human-robot communication has improved with the integration of gesture recognition models and natural language processing (NLP) for voice commands. The user experience has been enhanced by these models**.[7]**

* **Practical Applications:**

Numerous earlier initiatives have made the leap from theoretical models to practical implementations. Pathfinding robots have been used for patient assistance, warehouse automation, and autonomous farming in a number of industries, including logistics, healthcare, and agriculture. These useful applications demonstrate how flexible and feasible pathfinding robots can be in everyday situations**.[16]**

**CHAPTER 3: METHODOLOGY**

The suitable and appropriate model that will be designed in this project will be the prototyping model.

3.1 prototype model

The prototype model is a system development method where a prototype is built, tested, and re-worked until an acceptable outcome is achieved from which the product can be developed or complete system. The prototype model is the best work in scenarios where not all the requirements of the project are known in detail ahead of time. It is trial and error process that takes place between the developers and the users also it is iterative. However, to design the system with prototype model **[4]** there are main steps that need to be followed which are:

1. The requirements of the system should be defined as much details as possible. Usually, this step includes interviewing a number of users who represent all the aspects of the existing system or departments.
2. In the preliminary, a simple design should be created for the new system.
3. From the preliminary design, the first prototype is built. This first prototype is a scaled-down system which represents the approximation of the characteristics of the final product.
4. The users evaluate the strengths and weaknesses of the first prototype if anything needs to be added or removed. Then the developer collects and analyzes the changes.
5. The second prototype is constructed from the modification of the first prototype which is based on the evaluation of the users.
6. The second prototype is evaluated as the same as the first prototype.
7. Until the users are satisfied with the prototype which represents the final product desired. The final system is constructed based on the final prototype.
8. The final system is evaluated and tested. Routine maintenance is performed on a continuing basis for preventing large-scale failures and minimizing downtime.

3.2 Analysis phase

3.2.1 The overview of the project

The analysis phase introduces the development of the robot which is equipped with the capabilities of path finding. This project aims to create a multi-use robot of navigating its environment independently and to help new students, teachers, and other workers to reach to the places as they want. The key components of this project include Arduino Uno as the central brain (microcontroller), servo motor with an ultrasonic sensor, DC motors, a toggle switch for manual power control, a Bluetooth module for remote control, an OLED screen for user interface, AF L293D Motor Shield for the movement, 3.7 V 18650 Li-ion Battery 2600MAH(2X), TP 4056 micro-USB battery charger, and 18650-battery holders for power supply, wheels and sun-board to build a physical structure of the robot.

3.2.2 project’s components

In this project, there are some components that will be considered to make a robot with path finding. These components are:

1. **Arduino UNO board**

Arduino UNO bord is the central brain of the robot and it’s responsible for executing the pathfinding algorithms, controlling the various components, and processing inputs.

1. **AF L293D Motor Shield**

It will connect to the Arduino board and DC motor, and it will provide the necessary control for the movements of the robot. It ensures precise control over motor speed and direction.

1. **Ultrasonic sensor SR04**

Ultrasonic sensor can measure the distances to create a map for the environment of the robot and it can detect obstacles. It is important for data to implement path finding algorithms and avoid collisions.

1. **Servo motor SG90**

It is used for controlling the movement of the ultrasonic sensor in different directions, by allowing the robot to scan its area and avoiding obstacles.

1. **Bluetooth modules HC05**

This Bluetooth has a feature for wireless communication between the robot and external devices like computers or smartphones. It can be used for remote control by sending commands or receiving updates.

1. **3.7 V 18650 Li-ion Battery 2600MAH(2X)**

It provides a power source for the robot. It considers the overall power requirements for extended operation. In addition, it will connect to the motor shield to power the DC motor and other electronic components.

1. **TP 4056 micro-USB battery charger**

Used for charging the Li-on batteries, it will connect to each 18650-battery holder and use a charging indicator to control the charging process.

1. **18650-battery holders**

Battery holders provide a convenient way to connect the Li-on batteries to the power system of the robot.

1. **SPST Toggle switch**

This switch monitors the power supply of the robot and it’s a manual control (turning on and turning off).

1. **Sun-board**

It’s a board for building the physical structure and mounting the components of the robot.

1. **Wheels**

The wheels help the robot to move anywhere by attaching to the DC motor.

1. **DC Motor**

The Dc Motor will be used for wheel movement.

1. **OLED Screen**

OLED Screen displays the information for user interface. It will connect to the Arduino, shows a specific message to the user in a specific situation.

1. **Arduino Wi-fi ESP32**

This Arduino Wi-fi allows the Arduino board to connect to the internet.

3.2.3 Requirements analysis

This section will cover the stakeholders of this project, functional and non-functional requirements. Stakeholders refer to the groups or individuals or other users that have an interest in using it. The functional requirements of this project refer to the capabilities and features that contribute directly to its intended functionality. On the other hand, non-functional requirements refer to the constraints and qualities that are important for the successful operation of the project but aren’t related to specific functionalities directly.

3.2.4 Stakeholders

In this project the stakeholders will include:

1. Students
2. Teachers
3. Workers
4. Administrators
5. Developers

* + 1. Functional requirements

Functional requirement describes the features of the system which should have to meet the needs of users in other word, functional requirements mean what the system should do. For a mobile robot with path finding system, the following functional requirements are:

1. **Navigation and movement:** a mobile robot should be able to navigate automatically by using a pathfinding algorithm. In addition, a mobile robot can be able to move forward, backward, turn right, and turn left.

1. **Obstacle avoidance/detection**: the system of mobile robot can adjust its path to avoid the obstacles automatically. This can be done by ultrasonic sensor which can detect the collisions in the robot’s path.
2. **Semi-autonomous exploration**: a mobile robot can explore an unknown environment automatically by using path finding algorithm to navigate and map the surrounding area systematically.
3. **Stop mechanism emergency**: a system of mobile robot includes an emergency stop feature if any errors happen or there is a problem in network connection. The robot will go back to its original place.
4. **Supporting multi-language**: a website of a robot system can support multiple languages for different users.
5. **Performance**: a path find algorithm in mobile robot system can provide accurate and quick routes in various environments. As a result, a robot can operate efficiently and respond to commands promptly.

* + 1. Non-functional requirements

Non-functional requirements describe the characteristics of a system that aren’t related to specific features or behaviors however, they are important for reliability, usability, and safety etc. The following non-functional requirements of the robot’s system are:

1. **Reliability**: in failover mechanism, a robot system can handle unexpected situations.

1. **Maintainability**: the codebase of the system will follow the best practices for maintainability, modular, and well-documented.
2. **Safety**: with an emergency stop feature which halts the movement of the robot in specific situations.
3. **Usability**: the robot system should meet the needs of users.

|  |  |  |
| --- | --- | --- |
| Arduino UNO board | AF L293D Motor Shield | Servo motor SG90 |
| Ultrasonic sensor (SR04) | Bluetooth modules HC05 | 3.7 V 18650 Li-ion Battery |
| TP 4056 micro-USB battery charger | 18650 battery holders | OLED Screen |
| SPST Toggle switch | Sun-board | Wheels |
| Arduino UNO Wi-Fi ESP32 | DC Motor |

Table1. Project’s components

* 1. Design phase

3.3.1 Creating the exterior part of the robot:

The design phase in this project will involve the steps of creating a mobile robot with path finding according to the project’s components. Firstly, Arduino UNO board will be connected to the computer for programming and testing to ensure that is working. Secondly, after testing, the AF L293D Motor Sheild will wire to Arduino UNO board using appropriate pins and ensure that the motor shield is attached to the Arduino’s headers securely. Then, the DC motor which is responsible for the movement of the robot will connect to the motor shield and ensure the polarity of proper rotation and attach the wheels to the motor shield for controlling movement.

Thirdly, the ultrasonic sensor (SR04) will connect to the Arduino board for obstacle avoidance/detection. There are two pins in ultrasonic sensor one is called Echo and the other one called Trigger, these pins provide ground connections and power. Fourthly, to control the orientation of the sensor array the Servo Motor will connect to the Arduino board by wiring the servo motor with an appropriate PWM pin on the Arduino board.

Fig7. PWM on Arduino UNO

Fifthly, wireless connection can be done by connecting the Bluetooth Module (HC05) to the Arduino board by connecting the RX pin of Arduino and TX pin of Bluetooth module conversely.

Fig8. Wireless communication

Sixthly, to power the 3.7V Li-on battery on the Arduino board, the negative and positive terminals of Li-on battery will connect to the suitable pins on the Arduino board. Ind addition, the Li-on battery will secure in its holders and then it will connect with the TP4056 micro- USB battery charger for recharging.

Fig9. Arduino battery level

Seventhly, to control the power supply of the robot the SPST toggle switch which allows manual control and power management will connect to the entire robot. Eighth, the OLED Screen will connect to the Arduino board to display specific information in a specific information.

In addition, a robot can move automatically by using wi-fi. This wi-fi is Arduino wi-fi ESP32 which is associated with the Arduino board by serial communication. To process this wi-fi by uploading the library manager from Arduino IDE. Finally, all these components will be used on the Sun-board as the physical base for securing components and mounting.

Fig10. Arduino Wi-fi ESP32 connections

* + 1. Design phase (using diagrams):
* **Data Flow Diagram (DFD):**

DFD which refers to data flow diagram. DFD is visual representation using a unified set of notations and symbols that describe how information flows inside the system. It gives a high-level overview of the data flow, processes, and external entities which are involved in the functioning of the robot. It can resemble UML (Unified Modeling Language) or flow charts but can’t represent the details of the system logic.

The following DFD describes how the system of mobile robot with pathfinding is working which is done on Creately software:

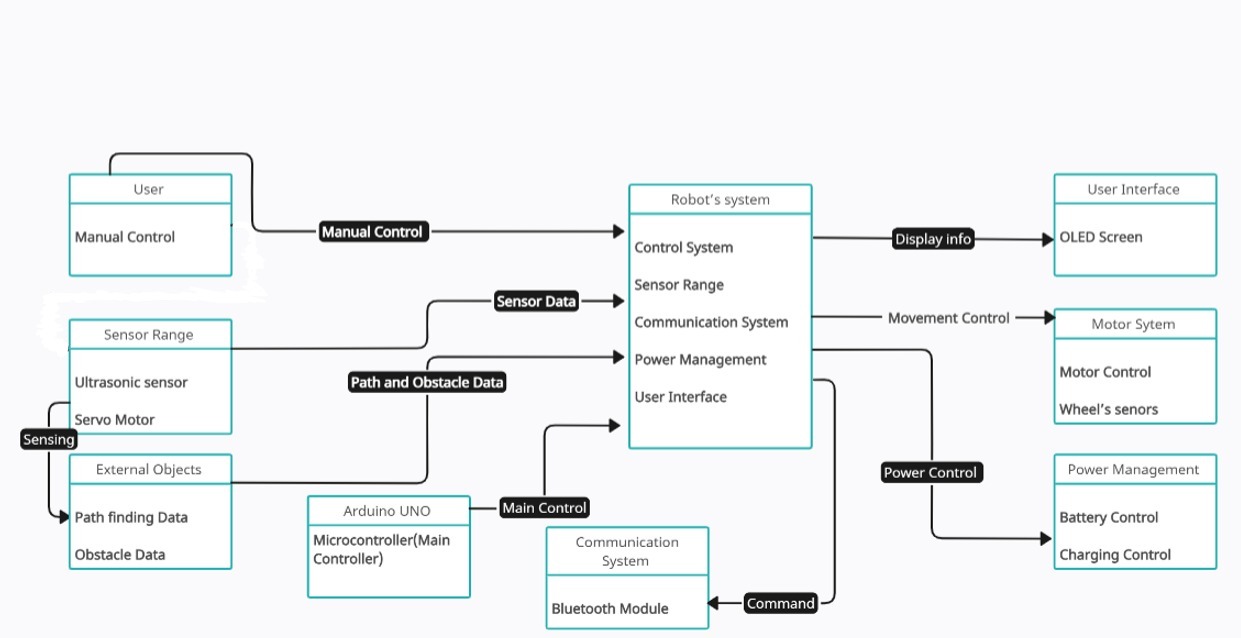


Fig11. Data Flow Diagram (DFD)

* **Flowchart Diagram:**

A flowchart is a diagram that describes a system or computer algorithm or process. It uses ovals, rectangles, diamonds, and other shapes to depict the type of step, with connecting arrows to depict flow and sequence. Flowcharts are widely used in different fields to study, plan, improve and communicate often complex process in easy to understand and clear.

Fig12. Flowchart Diagram

user scan QR code for website

Opening website of university places

User selects location

user press ‘Let’s go’ button on website

Determine robot’s current position

Determine user’s location

Robot is walking

Displays a message to the user on OLED screen to press button and floor number

A robot calculates the shortest path

Reaches to the desired user’s destination

Robot continues the destination

Start path finding algorithms.

Keep walking

If there is Obstacle

Backward

Forward

Turn right

Turn left

A robot comes back to its original place

If there’s left

Yes

No

Yes

If there is any emergency

shows a message to the user ‘Sorry, I need to come back to fix my problems’

No

No

Yes

* **Entity Relationship Diagram (ERD):**

ERD is a type of flowchart which describes how entities like objects, people, and concepts associate with each other inside the system. ERD are often used to design and debug RDBMS in different fields such as software engineering, business information system etc. ERD defined a set of symbols which are rectangles for entities, diamonds for relationships, and ovals for attributes, they are connected with each other by arrows.

In the project the ERD will be covered entities, attributes, and relationships between them which are:

|  |  |
| --- | --- |
| Entities and attributes | Relationships |
| User   * **User-ID** * **Username** * **QR-Code**     Destination   * **Destination-ID** * **Location-Name**   Robot   * **Robot-ID** * **Current-Position** * **Status** * **Battery level**   Path   * **Path-ID** * **Route** * **Obstacle**   Elevator   * **Elevator-ID** * **Floor-Number** | 1. The user chooses the destination (one to many).      1. The robot receives a request from user (one to many). 2. The robot determines its current place (one to one). 3. The shortest path can be found by a robot (one to many). 4. The robot walks on the path (one to many). 5. The obstacles can be detected by a robot (one to many). 6. The robot arrives in the elevator (one to one). 7. The user presses the button of elevator (many to many). 8. The robot continuously walks on the path. |

Table2. ERD description

Fig13. Entity relationship diagram (ERD).

Choose

|  |  |
| --- | --- |
| **User** | |
| User-ID | Integer |
| Username | Varchar |
| QR-Code | Varchar |

|  |  |
| --- | --- |
| Destination | |
| Destination-ID | Integer |
| Location-Name | Varchar |

|  |  |
| --- | --- |
| **Robot** | |
| Robot-ID | Integer |
| Current-Position | Decimal |
| Status | ENUM |
| Battery-Level | Integer |

Determine

Receive

Press

Find

|  |  |
| --- | --- |
| **Elevator** | |
| Elevator-ID | Integer |
| Floor-Number | Integer |

|  |  |
| --- | --- |
| **Path** | |
| Path-ID | Integer |
| Route | Geometry |
| Obstacle | Varchar (Max) |

Walk

detect

**Fig14. Cardinality arrows**

* 1. Development phase

Structure of the System: the pathfinding robot is made up of these parts:

* Arduino Board: Controls the motors and sensors.
* Motor Drivers: Interface between Arduino and motors for precise control.
* Ultrasound Sensors: Detect obstacles and provide input to the pathfinding algorithm.
* Motors: The physical structure and movement mechanisms of the robot.

Software Development: Design and develop the algorithm considering the university's structure buildings, walkways, and common areas—in mind. Use Python's A\* algorithm to create the best possible routes for the robot as it navigates the campus.

Also, make the website interface easy to use so that users can communicate with the pathfinding robot. Create a user interface that is easy to navigate for the website.

Examine how well the robot can navigate the university's environment to reach various locations. Based on the testing results, debug and improve the system, fixing any hardware or software problems.

Connect and calibrate sensors like ultrasonic sensors. Implement sensor data processing in Python for obstacle detection. Integrate sensor data with the A\* algorithm to dynamically adjust the path.

* 1. Implementation phase
* Algorithm Integration:

Use real-time sensor data and human interaction to inform the implementation of A\* algorithm functions for pathfinding.

* Web Development:

Finalize web pages for destination selection, user interaction, and real-time tracking. Create a user interface that is responsive by integrating the Python backend with the website.

* Arduino Code Implementation:

Write and execute Arduino code for communication, data processing from sensors, and motor control. Create Arduino code to program the robot to move by interpreting commands from the Python backend. Implement algorithms for processing sensor data.

* Sensor Calibration:

Calibrate sensors to ensure accurate data collection and reliable obstacle detection. To ensure reliability and accuracy, calibrate each type of sensor.

* Performance Testing:

Evaluate the overall performance of the pathfinding robot system. Conduct performance tests to assess the robot's speed, responsiveness, and accuracy in navigation. Use code optimization to locate and fix any performance bottlenecks. Check that the system response time satisfies user expectations by measuring it.

* 1. Evaluation phase
* Functional Testing:

Evaluate the system's capacity to produce ideal routes by applying the A\* algorithm. Verify the responsiveness of the web interface to user commands and inputs. Verify that the robot can get to the designated locations and that its movements are accurate.

* Obstacle Avoidance Testing:

Evaluate the robot's ability to recognize and steer clear of obstacles in real time. Measure the efficacy of obstacle avoidance and detection mechanisms by running tests with different scenarios of obstacles. Ensure the robot adjusts its path dynamically in response to changes in the environment.

* User Interface and Communication Testing:

Check the usability and reliability of the website's user interface. Verify that there is smooth and responsive communication between the Arduino components, Python backend, and website. Analyze how well and clearly the status and actions of the robot were communicated.

* Performance Testing:

Assess the system's response time for path generation and user input. Evaluate the robot's speed, accuracy, and precision in navigating the university environment. Locate any performance difficulties and use optimization to fix them.

* User Acceptance Testing:

Collect feedback on the overall user experience, interface design, and system functionality. Enhance user satisfaction through iterative improvements based on feedback.

* 1. Tools used.

**Hardware tools:**

1. Robot Chassis:
   * A physical structure to house the robot's components, such as motors, wheels, sensors, and the control board.
2. Motors and Wheels:
   * DC motors and wheels for the robot's movement.
3. Arduino Board:
   * Arduino microcontroller board for interfacing with sensors, motor control, and communication with the Python backend.
4. Sensors:
   * Ultrasonic sensors for obstacle detection and avoidance.
5. Power Supply:
   * Batteries or a power supply system to provide energy for the robot's operation.

**Software tools:**

1. Web Development Tools: notepad for HTML, Visual Studio Code for JavaScript, and Visual Studio Code for CSS for creating the user interface of the website.
2. For the robot:

Arduino IDE: Arduino Integrated Development Environment for writing and uploading code to the Arduino board using C++ and PySerial python.

**CHAPTER 4: PROJECT PLAN**

* 1. Project plan

Table 3. project plan

* 1. Gantt Chart

Table 4. Gantt Chart

* 1. Network Diagram

Table 5. Network Diagram

**CHAPTER 5: Software Requirements Specification**

1. **Introduction**

* Purpose

This document serves as a guide for developing software needed to create a robot that can navigate its environment. The robot assists users in reaching specific places by navigating within an academic environment.

* Scope

The robot system consists of software components for pathfinding, obstacle avoidance, and user interaction in addition to hardware components like an Arduino UNO board, motor shield, sensors, and actuators.

1. **Overview of the System**

* System Description

The robot is a mobile device with a microcontroller, actuators, and sensors. It uses pathfinding algorithms to find its way around the campus, avoiding obstacles and arriving at predetermined locations.

* Architecture of the System

The system architecture is made up of the following essential elements:

* Arduino UNO Board
* Motor Shield (L293D)
* Ultrasonic Sensor (SR04)
* Servo Motor (SG90)
* Bluetooth Module (HC05)
* 3.7V 18650 Li-ion Battery
* TP4056 Micro-USB Battery Charger
* SPST Toggle Switch
* Sun-board
* Wheels
* OLED Screen

1. **Functional Requirements**

* User Communication
* Scan of QR Codes

The robot should be able to obtain user destination information by scanning QR codes from the university's website.

* Destination Selection

On the website, users can select their destination, and the robot will wirelessly get this data.

* Emergency Stop

The robot ought to provide an emergency stop message on the OLED screen in the event of an emergency.

* Navigation and Pathfinding
* Determining Position

The robot ought to ascertain its present location within the university environment.

* Algorithm for Pathfinding

To determine the shortest route to the user's destination, apply a pathfinding algorithm (such as the A\* algorithm).

* Avoiding Obstacles

Use obstacle avoidance techniques to get around obstructions while moving.

* User Guidance
* Message Display

Messages on the OLED screen should be displayed by the robot to direct users, particularly in elevators.

* Power Management
* Tracking of Batteries

The battery level should be tracked by the system and shown on the OLED screen.

* Charging Mechanism

Provide a system that will allow the robot to return for recharging when its battery runs low.

1. **Non-Functional Requirements**

* Performance
* Response Time

Input from the user and changes in the environment should be processed by the robot in a reasonable amount of time (less than one second, for example).

* Reliability
* Uptime of the System

In order to ensure that the robot is available for user interaction during its operational period, the system should have a high uptime.

* Safety
* Emergency Stop

In the scenario of an emergency, the robot's movement should be instantly stopped by the dependable emergency stop feature.

1. **Limitations**

* Hardware Limitations

The hardware components' limitations on sensor accuracy and motor power should be respected by the system.

* Environmental Constraints

Factors like elevators and floor transitions should be considered when designing the robot for use in indoor university environments.

**CHAPTER 6: conclusion**

All in all, this project incorporates intelligent algorithms and hardware components to provide a mobile robot capable of semi-autonomous navigating the spaces of university. The integration of these components not only helps in navigating but also lays the basis for future advancements in robotics field and semi-autonomous systems. Although the applications of this project extend beyond university campuses, at the same time, it offers a glimpse into the possibilities of smart mobile robot in various environments for the future.

The improvement of mobile robot equipped with the capabilities of path finding represents aa amazing stride in the robotics field. The integration of varied components involving an Arduino board, Servo motor SG90, Bluetooth module HC05, AF L293D Motor Shield, Ultrasonic sensor SR04, TP 4056 micro-USB battery charger, 18650 Li-on battery, SPST toggle switch, wheels, OlED Screen, and Sunboard. These components contribute to creating a semi-autonomous and intelligent robot.

The goal of this project is to empower the robot with the abilities to navigate through an environment of the university to assist users like students, workers, new teachers, visitors etc. to reach their destinations efficiently.

The algorithms of path finding which facilities by the Arduino board and ultrasonic sensor which enable the robot to choose the most optimal route and detect obstacles. In addition, a Servo motor enhances the ability of the robot for surveying its surroundings according to the given environment. However, a Bluetooth module provides the capabilities of wireless communication that allow users to interact with the robot by mobile device. The reliable and rechargeable power source is provided by the incorporation of Li-on battery with micro-USB battery ensuring easy maintenance and prolonged operation.

The OLED Screen acts as an interactive display, providing instructions and information in specific situations to the users and the DC motor which is controlled by the Motor shield to control the movements of the robot’s wheels. The structural integrity of the robot can be achieved through using a sunboard which provides a stable platform for mounting the components of the robot.

**References**

1. TechTarget. (n.d.). Mobile robot (mobile robotics). IoT Agenda. Retrieved from https://www.techtarget.com/iotagenda/definition/mobile-robot-mobile-robotics#:~:text=A%20mobile%20robot%20is%20a,as%20wheels%2C%20tracks%20and%20legs.
2. Teleweck, P. E., & Chandrasekaran, B. (2019, April). Path planning algorithms and their use in robotic navigation systems. *Journal of Physics: Conference Series, 1207*(1), 012018. https://iopscience.iop.org/article/10.1088/1742-6596/1207/1/012018.
3. Faz, M. R. M., & Hanapi, Z. M. (2013). Mobile robot path planning refers. *Expert SystemswithApplications,40*(3),828–848. https://www.sciencedirect.com/science/article/pii/S095741742300756X#:~:text=Mobile%20robot%20path%20planning%20refers,path%20planning%20techniques%20is%20presented.
4. TechTarget.(n.d.).Prototypingmodel.SearchCIO.Retrievedfrom https://www.techtarget.com/searchcio/definition/Prototyping-Model#:~:text=What%20is%20prototyping%20model%3F,or%20product%20can%20be%20developed.
5. Yong, C. S., Chong, H. Y., Siew, E. G., & Wei, W. (2013). Development of Automated Guided Vehicle (AGV) for Material Handling System. *Journal of Mechanical Engineering and Automation, 3*(2), 77–82. http://eprints.utar.edu.my/1165/1/CS-2013-0903501-1.pdf.
6. Abbass, A., & Umar, M. S. (2019). Design of Mobile Robot for Obstacle Avoidance using Microcontroller. *International Journal of Mechanical and Mechatronics Engineering,19*(5),1–9.https://www.researchgate.net/profile/Ayam-Abbass/publication/336209518\_Design\_of\_Mobile\_Robot\_for\_Obstacle\_Avoidance\_using\_Microcontroller/links/5d945296299bf10cff1f9d25/Design-of-Mobile-Robot-for-Obstacle-Avoidance-using-Microcontroller.pdf.
7. Yamamoto, Y., & Harada, K. (2012). Development of a Mobile Robot with a Novel Mechanism for Climbing Steps and Stairs. *Robotics and Autonomous Systems, 60*(8), Micr1043–1054. https://www.sciencedirect.com/science/article/abs/pii/S0045790611001091.
8. Lindroth, M., & Linder, A. (2010). Path Planning for an Autonomous Robot Using D\* LiteAlgorithm.Retrievedfromhttps://his.diva-portal.org/smash/get/diva2:1768516/FULLTEXT01.pdf.
9. Technical University of Denmark. (n.d.). Oersted: dtu2868. Retrieved from https://backend.orbit.dtu.dk/ws/portalfiles/portal/4724729/oersted-dtu2868.pdf.
10. Random Nerd Tutorials. (n.d.). ESP32 Useful Wi-Fi Functions Arduino. Retrieved from https://randomnerdtutorials.com/esp32-useful-wi-fi-functions-arduino/.
11. Thrun, S., Burgard, W., & Fox, D. (2005). Probabilistic Robotics. Retrieved from

https://link.springer.com/chapter/10.1007/978-3-540-75388-9\_3

1. Riquelme, J. C., Riquelme, F., & Jiménez, A. (2021). Evaluation of Obstacle Avoidance Algorithms for Mobile Robots Using ROS and Gazebo. Sensors, 21(23), 7898. Retrieved from https://www.mdpi.com/1424-8220/21/23/7898.
2. Wikipedia contributors. (n.d.). Obstacle avoidance. Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Obstacle\_avoidance.
3. S. Koenig and M. Likhachev, "D\* Lite," in Proceedings of the National Conference on Artificial Intelligence (AAAI), 2002. Retrieved from https://cdn.aaai.org/AAAI/2002/AAAI02-072.pdf
4. Liu, L., Wang, X., & Yang, X. (2023). Path planning techniques for mobile robots: Review and prospect. *Expert Systems with Applications*, 227, 120254 Retrieved from https://www.sciencedirect.com/science/article/pii/S095741742300756X
5. Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction the MIT Press. Retrieved from

https://books.google.com.om/books?hl=en&lr=&id=uWV0DwAAQBAJ&oi=fnd&pg=PR7&dq=%22Sutton,+R.+S.,+%26+Barto,+A.+G.+(2018).+Reinforcement+Learning:+An+Introduction.+The+MIT+Press.%22&ots=mjnHn\_W3o8&sig=MfwwOFS7YxtTMBYoBdSPJUOhtJI&redir\_esc=y#v=onepage&q=%22Sutton%2C%20R.%20S.%2C%20%26%20Barto%2C%20A.%20G.%20(2018).%20Reinforcement%20Learning%3A%20An%20Introduction.%20The%20MIT%20Press.%22&f=false

1. Aspholm, L., & Rolén, M. (2023).A comparison between mapless and pre-mapped path planning: Towards open-source Autonomous Mobile Robots. (Bachelor Degree Project in Information Technology). Retrieved from https://his.diva-portal.org/smash/get/diva2:1768516/FULLTEXT01.pdf
2. Kumar, F. K. (2019, December ). *Robot Navigation and Path Planning Techniques*. Retrieved from csjournals: https://www.csjournals.com/IJEE/PDF11-2/16.%20Faz.pdf

**images**

1. HowToMechatronics. (2015). Ultrasonic Sensor HC-SR04 and Arduino - Complete Guide Image. Retrieved from https://howtomechatronics.com/wp-content/uploads/2015/07/Ultrasonic-Sensor-HC-SR04-and-Arduino-Complete-Guide-2.jpg.
2. Amazon. (n.d.). Arduino Uno R3 Front Image. Retrieved from https://m.media-amazon.com/images/I/61yfIwAxe0L.jpg.
3. Tertiary Robotics. (2022). Arduino Uno R3 Front Image. Retrieved from https://www.tertiaryrobotics.com/wp-content/uploads/2022/10/arduinouno\_r3\_front.jpeg.
4. OnBuy. (n.d.). 2x 18650 2600mAh Power Battery Image. Retrieved from https://assets.onbuy.com/i14/product/87aca3cdebc04c23b68247f9be503f40-m134971718/2x-18650-2600mah-power-battery-37v-rechargeable-li-on-battery.jpg.
5. Maker's Electronics. (2018). Li-Pol Charger TP4056 Image. Retrieved from https://i0.wp.com/makerselectronics.com/wp-content/uploads/2018/02/li-pol-charger-tp4056-1s-37v-miniusb-1.png?fit=1080%2C1080&ssl=1.
6. Amazon. (n.d.). DC Motor with Wheels Image. Retrieved from https://m.media-amazon.com/images/I/619uoWwhNZL.*AC\_UF1000,1000\_QL80*.jpg.
7. Walmart. (n.d.). SPST Toggle Switch Pre-Wired On-Off Image. Retrieved from https://i5.walmartimages.com/seo/2x-SPST-Toggle-Switch-Pre-Wired-On-Off-Metal-Small-Automotive-Boat-Car-Truck\_131d277b-d3b3-42d4-9c56-86c919babdda.fa68b365c57c1a71e97ee0f0acd2be71.jpeg.
8. Amazon. (n.d.). Power Bank Image. Retrieved from https://m.media-amazon.com/images/I/81yANaUpWnL.jpg.
9. PRAlves. (2019). Facebot Image. Retrieved from https://jpralves.net/img/2019/01/PM195\_Facebot.jpg.
10. Robu.in. (2017). Large Robot Smart Car Wheel Image. Retrieved from https://robu.in/wp-content/uploads/2017/06/85MM-Large-Robot-Smart-Car-Wheel-38MM-Width-Surface-RedPink-2.jpg.
11. Circuit Basics. (2020). Arduino Pulse Width Modulation (PWM) Pins Image. Retrieved from https://www.circuitbasics.com/wp-content/uploads/2020/05/Arduino-Pulse-Width-Modulation-Pins.jpg.
12. Mechstuff. (2017). Thumbnail Video Image. Retrieved from https://i1.wp.com/mechstuff.com/wp-content/uploads/2017/01/thmbnail-video.jpg?fit=1280%2C720.
13. ArduinoCircuit. (2023). Battery Voltage Indicator using Arduino and LED Bar Graph Image. Retrieved from https://arduinocircuit.com/wp-content/uploads/2023/04/Battery-Voltage-Indicator-using-Arduino-and-LED-Bar-Graph.webp.
14. Stack Exchange. (n.d.). Image. Retrieved from https://i.stack.imgur.com/rb3Ig.jpg.
15. Europe1 Discourse. (n.d.). Arduino Image. Retrieved from https://europe1.discourse-cdn.com/arduino/original/4X/2/3/1/23100fe2317ef38100b2fd729458c9bca544ea2b.png.
16. GitHub. (n.d.). Image. Retrieved from https://repository-images.githubusercontent.com/420946327/a3ebb711-cad4-42f0-b7e6-e4506c6537a3.
17. MDPI. (n.d.). Algorithms Journal Image. Retrieved from https://www.mdpi.com/algorithms/algorithms-15-00477/article\_deploy/html/images/algorithms-15-00477-g001.png.
18. MDPI. (n.d.). Sensors Journal Image. Retrieved from https://www.mdpi.com/sensors/sensors-18-03170/article\_deploy/html/images/sensors-18-03170-g001.png.
19. Wikimedia Commons. (n.d.). Pathfinding 2D Illustration Image. Retrieved from https://upload.wikimedia.org/wikipedia/commons/4/4c/Pathfinding\_2D\_Illustration.svg.
20. ProtoSupplies. (2018). L293D Motor Driver Shield Image. Retrieved from https://protosupplies.com/wp-content/uploads/2018/04/L293D-V1-Motor-Driver-Shield-1.jpg.
21. Electronic Wings. (n.d.). HC-05 Bluetooth Module Image. Retrieved from https://www.electronicwings.com/storage/PlatformSection/TopicContent/118/icon/HC-05%20Bluetooth%20Module.jpg.
22. Automatic Addison. (2021). DC Motor Encoder Wheels Image. Retrieved from https://automaticaddison.com/wp-content/uploads/2021/04/1\_dc\_motor\_encoder\_wheels-1.jpg.

A screenshot of a cell phone

Description automatically generated

A screenshot of a web page

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

