# RoboExpress Team

Members:
Shai Bing Han
Ng Yu Qin
Kaylyn Ng

## 1.0 Introduction

At UTM, students who shop online often face an unexpected inconvenience: their packages are delivered to distant central points like CPP and OPC rather than directly to their hostels. This setup contradicts the core benefit of online shopping—convenience—forcing students to travel considerable distances to collect their orders. This issue is particularly challenging for students with packed schedules or limited transportation options.

To bridge this gap, some entrepreneurial students have started offering delivery services from these central points directly to the hostels. However, the fees for these services are frequently seen as unreasonable, especially for students on tight budgets, turning a convenience into an added financial burden. The situation reveals a clear need for an improved and more affordable delivery solution that could save students time, effort, and money, enhancing their overall experience and reducing unnecessary stress associated with online shopping on campus.

Therefore, our project introduces a dual-mode autonomous robot designed to collect parcels for students, providing a reliable and time-saving solution. The robot operates in two modes: autonomous and manual. In autonomous mode, it uses an IR sensor to follow a predefined path, with an Arduino controller managing movement and navigation. A motor driver module interprets sensor readings and powers the motors for smooth movement, while a steady power supply ensures reliable operation.For more precise control, the robot can switch to manual mode with a simple button press, allowing students to navigate around obstacles or handle unexpected situations. This flexibility makes it suitable for various campus terrains, as the robot can avoid obstacles that the ultrasonic sensor might not detect accurately in autonomous mode.To fulfill its purpose, the robot is equipped with a grabbing mechanism controlled by an Arduino servo motor, allowing it to lift and carry packages of different sizes securely. This dual-mode robot offers students a more convenient, hands-free approach to retrieving their parcels, making package collection simpler and enhancing daily campus life.

## 2.0 Progress

| Date | Event |
|---|---|
| 22/10/2024 | ☑ **Grouping** |
| 23/10/2024 | ☑ **First draft proposal** |
| 24/10/2024 | ☑ **Discussion on project proposal** |
| 28/10/2024 | **Project Hardware Discussion and Lending Solutions**<br>☑ Online meeting to discuss details of project such as hardware needed.<br>☑ Discuss on hardware lending issue |
| 3/10/2024 | **Designation (Car)**<br>☑ Make the design drawing for car |
| 12/11/2024 | **Modification**<br>☑ Modify our design and do the circuit design |
| 17/11/2024 | **Designation (Arm)**<br>☑ Make the design drawing for robot arm |
| 20/11/2024 | ☑ **Hardware arrived** |
| 24/11/2024 | **Hardware Assembly**<br>☑ Check component's functionality<br>☑ Assemble a basic frame to mount all components<br>☑ Install and connect sensors |
| 26/11/2024 | **Programming**<br>☑ Do the coding for autonomous mode<br>☑ Do the coding for manual control<br>☑ Do the coding for robot arm |
| 29/11/2024 | **Test and run the coding**<br>☑ Combine the coding<br>☑ Adjust the sensitivity for IR sensor<br>☑ Run tests to ensure the robot can follow the predefined line automatically<br>☑ Make sure the manual control feature function smoothly |

| | |
|---|---|
| 1/11/2024 | **Finalize our robot**<br>☑ Assembly robot arm onto the car<br>☑ Finalize the coding<br>☑ Make sure every component interacts with each other smoothly and efficiently. |
| 3/11/2024 | **Testing at Cairo Lab**<br>☑ Testing our robot with the map that is prepared by AIROST and do adjustments to fit the provided condition. |
| 4/11/2024 | **Final checking**<br>☑ Last checking our full robot and coding |
| 5/12/2024 | ☑ **Submission Project** |
| 6/12/2024 | ☑ **Presentation** |

## 3.0 Mechanics and hardware

### 3.1 Initial Proposed Concept

Our initial concept for the dual-mode autonomous robot focuses on improving package collection for students at Universiti Teknologi Malaysia (UTM). The robot will use an infrared (IR) sensor system to detect and follow a predefined line that guides it along the path from the student hostels to the collection points. When the robot is in autonomous mode, it will monitor its position relative to the line and make real-time adjustments to maintain a steady course.

To facilitate manual control, we decided to implement a button on the remote control that allows users to switch the robot to manual mode. When this button is pressed, the robot will stop following the line and allow the operator to navigate it around obstacles. This ensures that the user can take immediate control if the robot encounters unexpected barriers or needs to change direction quickly.

For the robot's movement and navigation, we have chosen to use a motor driver module that interprets sensor data and manages the motors' speed and direction. The robot will also be equipped with a grabbing mechanism, controlled by an Arduino servo motor, allowing it to lift and carry packages securely. The processor will handle inputs from the IR sensors and the remote control and coordinating the robot's actions.


1.0 Flowchart

## 3.2 Changes made during development

During the development process, we faced many challenges that required us to change and improve our original design. At first, we planned to use an RF controller to operate the robot, but since we couldn’t get this component, we switched to a Bluetooth module. The Bluetooth module was easier to get and worked well for our needs. Initially, our design included a gantry system for the grabbing mechanism. However, this system was expensive due to its aluminum structure and had limited movement—it could only go left-right and up-down. To reduce costs and increase flexibility, we replaced the gantry with a robotic arm. The robotic arm could rotate 360 degrees, making it much more versatile and suitable for our project.

We also planned to use a 3D printer to build the robot's body for a clean and precise look. However, due to limited time and budget, we used a wooden board instead. When testing the robot on the Cairo map, we found that the wooden base was too big and made it difficult to move around effectively. To solve this problem, we switched to an acrylic board, which was lighter and smaller, improving the robot's mobility.

Another issue we faced was with the servo motor. We started with the SG90 servo motor, but its torque of 1.8 kg/cm was too weak to handle the weight of the robotic arm. To fix this, we upgraded to the MG996R servo motor, which has a much stronger torque of 9.6 kg/cm and could handle the arm’s weight without any issues.

Finally, during testing, our motor driver module and Bluetooth module both short-circuited and stopped working. We replaced these components with new ones and made sure to improve our testing process to avoid similar problems in the future. These changes were necessary to address the issues in our original design and to make the final robot more practical and functional.

### 3.3 Hardware specifications

The hardware of the robot car consists of several key components that work together to enable both movement and robotic arm control. At the core of the system is the Arduino Uno board, which serves as the main controller. It processes inputs from the sensors and directs the actions of the other components, coordinating the robot’s functionality.

For movement, the robot utilizes a motor driver module that controls the four DC motors attached to the wheels. This module is responsible for regulating the speed and direction of the motors, allowing the robot car to move autonomously. Additionally, two IR sensors are used to enable the line-following functionality, allowing the robot to detect and follow a path.

Power is supplied by three battery holders (x3), which provide energy to the Arduino and motors. The rechargeable batteries ensure that the robot has sufficient power to operate during its tasks. A breadboard is used for wiring connections between components, providing a flexible platform to arrange and organize the circuit components without soldering.

For manual control, the robot is equipped with a Bluetooth module (HC-05), allowing for wireless communication. This module enables the operator to control both the movement of the car and the robotic arm remotely via Bluetooth.

The robot arm is driven by four servo motors, which control the precise movement of the arm to grab and manipulate objects effectively. Together, these components form a cohesive system that allows the robot car to operate efficiently in both autonomous and manual modes.

## 4.0 Software and Protocols

The software for the smart car and robotic arm is written in C++ using the Arduino IDE, with the code compiled and uploaded to the Arduino Uno. The system relies on three main components: the HC-05 Bluetooth module, the L298N motor driver, and the servo motor, each using different protocols to communicate and control the system.

The HC-05 Bluetooth module uses the Serial Port Protocol (SPP), which is part of the Bluetooth classic standard. It communicates wirelessly with the Arduino over UART (Universal Asynchronous Receiver Transmitter) at baud rates like 9600 or 115200. The HC-05 can act as either a master or slave, allowing it to send or receive data. In this setup, the module receives commands from a mobile app, sending them to the Arduino to control the car and robotic arm.

The L298N motor driver is used to control the movement of the car's wheels. Instead of using a communication protocol like Bluetooth, it is directly controlled by the Arduino via digital signals sent to its input pins (IN1, IN2, IN3, IN4). The PWM (Pulse Width Modulation) signals sent to the enable pins (ENA, ENB) control the speed of the motors. This system uses wired control from the Arduino to the L298N to determine motor direction and speed.

The servo motor operates the robotic arm. It uses Pulse Width Modulation (PWM), where the Arduino sends a signal with a specific pulse width to control the arm's position. A pulse width of 1 ms corresponds to 0°, 1.5 ms to 90°, and 2 ms to 180°. This allows precise control of the arm's movement.

Together, the HC-05, L298N, and servo motor work in harmony. The HC-05 Bluetooth module allows wireless communication with the mobile app, which sends commands to the Arduino. The Arduino processes these commands and controls the L298N motor driver and the servo motor using GPIO and PWM signals to control the car's movement and the robotic arm's position.

## 5.0 Execution and operation

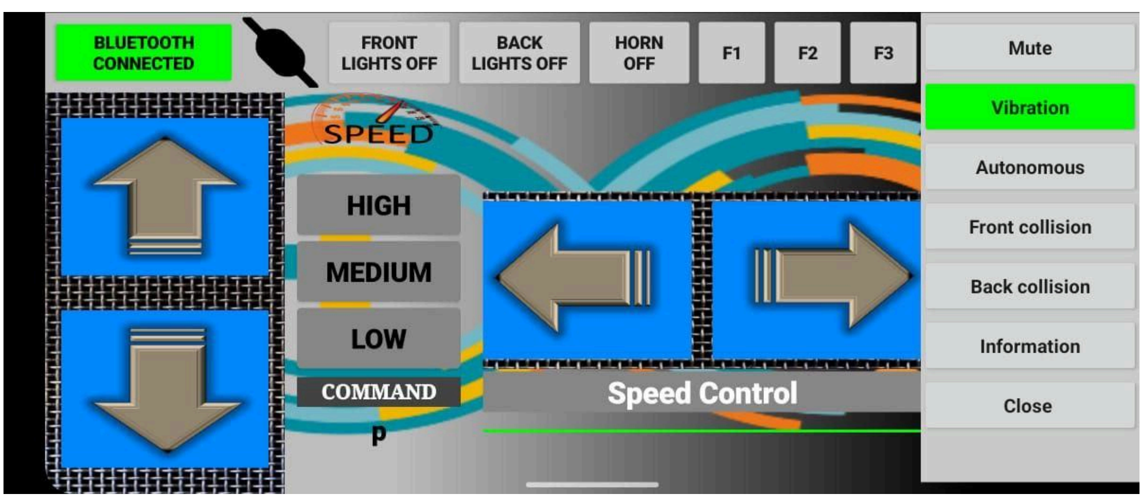
This project integrates a smart car and a robotic arm controlled by an Arduino Uno, blending automation with manual precision through Bluetooth connectivity. The system operates in four distinct modes, providing versatile functionality tailored to specific tasks.

In autonomous mode, the car autonomously navigates a path using an IR sensor. It detects light and dark surfaces, moving forward on light areas and stopping or turning when a black line is encountered. This allows the car to follow predefined tracks without manual intervention.

When switching to manual mode (Mode F1), the car's movements are controlled remotely via commands sent from a mobile app through an HC-05 Bluetooth module. Commands like 'F' (forward), 'B' (backward), 'L' (left), and 'R' (right) enable precise navigation in real time.

The robotic arm operates through two additional modes. In Mode F2 and Mode F3, the arm's base rotation and vertical movement are controlled. Commands such as 'F' and 'B' adjust the vertical position, while 'L' and 'R' rotate the arm. Switching to Mode F3 focuses on the claw, where 'F' and 'B' refine its position, and 'L' and 'R' open or close the claw for object manipulation.

Modes are selected by sending specific commands ('F1', 'F2', 'F3') from the app. Each mode repurposes the same set of commands to control different components, ensuring efficient use of limited input buttons. Safety measures prevent servos from exceeding their operational range of 0° to 180°.



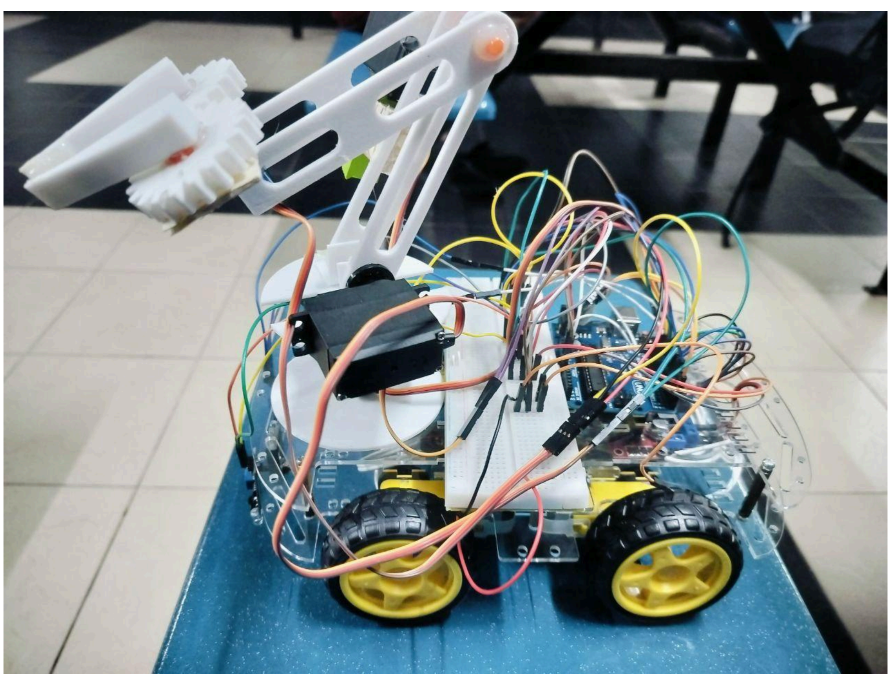

1.1User interface

## 6.0 Conclusion

The RoboExpress Project successfully addressed the logistical challenges faced by UTM students in collecting parcels, providing a practical and innovative solution through a dual-mode autonomous robot. By combining autonomous navigation and manual control capabilities, the robot enables convenient parcel collection. Equipped with an IR sensor for line following and a robotic arm for object handling, the prototype demonstrates the effective application of technology to enhance campus life. While the robot achieved its primary objectives, there remains significant potential for improvement to refine its functionality and expand its utility.

To improve precision and reliability, the accuracy robotic arm can be enhanced to handle parcels of varying sizes and shapes more effectively. Incorporating additional sensors, such as ultrasonic or proximity sensors, could further improve obstacle detection, ensuring smoother operation in autonomous mode. These adjustments would make the robot more adaptable to real-world scenarios.

In conclusion, the RoboExpress Project has laid a solid foundation for an effective parcel delivery solution on campus. With strategic refinements to it, the system has the potential to evolve into a robust and widely applicable technology that addresses real-world challenges and enhances everyday life.



1.2Picture of our robot car

## 7.0 References


1. Blum, J. (n.d.) *How to Build a Simple Arduino Robot Car*. Available at: https://www.youtube.com/watch?v=Mi31f5CzTrY.
2. Robojax (2018) *Arduino Line Following Robot Car*. Available at: https://www.youtube.com/watch?v=KP3W73wD0u0.
3. Arduino (n.d.) *Arduino Motor Driver Tutorial*. Available at: https://create.arduino.cc/projecthub.
4. Arduino (n.d.) *Servo Control Tutorial*. Available at: https://www.arduino.cc/en/Tutorial/Servo.
5. Maker Advisor (2020) *HC-05 Bluetooth Tutorial*. Available at: https://makeradvisor.com/arduino-bluetooth-hc-05 .