ENG405 (Integrated Design Project): Statement of Work

Tatyana Maltseva (**s257346**), Sakon Nadthayai (**s245002**),
and Shane Reynolds (**s262538**)

March 13, 2018

Contents

1 Introduction

As part of engineering degree at Charles Darwin University students require to undertake integrated design group project. An integrated group design project requires students to work together to complete the design task set by a client. The design project must meet clients specification and performance criteria; it must be completed within the time frame; and satisfy all of the required milestones.

For electrical engineering students the design task is to build an autonomous robotic vehicle. The realisation of the project involves making a mathematical model, system simulation, designing control loops, completing the design of the vehicle mechanically and electrically and implementing the designed system using National Instruments MyRio hardware and LabVIEW software.

2 Requirements & Scope

The main requirement of this project is to design and build an autonomous robotic vehicle that will traverse an enclosed area, which consists of obstacles. The robot will start from a randomised location, and needs to navigate to a designated finish position, in the shortest possible time. The finish location will be highlighted by a red marker, and once the robot has reached its objective, it must stop and provide a signal, physical or otherwise. These requirements can be found in more detail in the requirements matrix shown in Table 1.

Table 1: Requirements matrix for the autonomous navigation robot

Requirement Number	**Description**	**Requirement Number**	**Description**
R1	The robot must act autonomously	R6	The robot needs to stop on top of the finish zone or within 100mm from the zone
R2	The robot is designed to act in a 3m × 6m environment	R7	The robot must stop when it has reached the exit zone
R3	The robot can avoid obstacles of minimum size 55mm × 210mm × 297mm	R8	The robot must signal once it has reached the exit zone
R4	The robot can identify and navigate to an exit zone, demarcated by a red square of size 420mm × 297mm	R9	The robot chassis must use a Pololu Dagu Rover 5, two motor, tracked chassis with encoders (see Figure 1)
R5	The robot must move from start to finish within 3 minutes	R10	The robot must use an NI myRio 1900 powered by the Xilinx ZYNQ 7Z2010 (see Figure 2), or an Arduino powered by the ATmega328 for the embedded system.

The scope of the project works include the following items:

- Statement of work
- Critical analysis of options
- Demonstration & presentation which consists of:
  - Initial demonstration of obstacle avoidance, final demonstration of full system functionality, and a 10 minute presentation outlining design process, and showcasing final system.

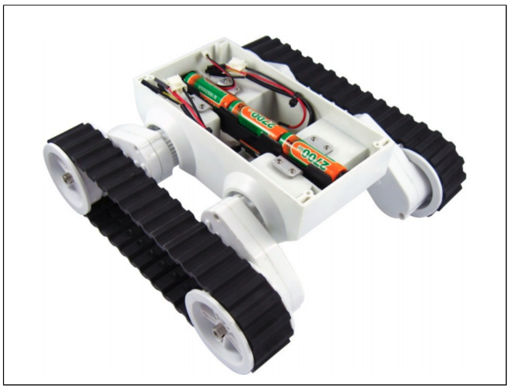

Figure 1: Pololu Dagu Rover 5, two motor, tracked chassis with encoders. Sourced: https://www.sparkfun.com/products/10336

Figure 2: NI myRio powered by the Xilinx ZYNQ 7Z2010 microprocessor. Sourced: https://www.ni.com

3 Design Options

There are a number of competing designs which would serve the project's intended objectives, and still fall within the purview of the scope. Broadly, in terms of complexity, these solutions can be placed into one of three categories: low, moderate, or high. Typically, the higher the complexity the greater the cost of the project, however, with increased complexity there is an expectation of greater performance and efficiency. This section of the report provides a high level overview for each of the three options, making elementary design recommendations for each. The section concludes by nominating a preferred solution, and proposing future lines of enquiry which need to be directed to the client.

3.1 Low Design Complexity

One of the simplest and lowest cost designs, would be the implementation of a finite state machine. The robot chassis would be equipped with four sensors: one IR sensor on the front to check for an obstacle free path forward, and two on the sides of the robot to provide assurance the robot will not collide with walls to the left and right. A final sensor could be employed on rear of the robot chassis, pointing towards the floor, which would be used to detect if the robot is currently placed on the red exit panel. Kumar et alias (2016) suggest two types of IR sensor for this implementation: a SHARP GP2D120 (4cm to 30cm range) for the sides of the robot, and a SHARP GP2Y0A21YK0F (10cm to 80cm range) for the front of the robot. A very simple example of what the robots operation might look like can be seen in Figure 3 - this robot has 4 states of operation in which it can operate:

1. **Forward:** the robot operates in this mode of operation provided there are no obstacles present in the robots path (within some certain distance threshold). The robot will maintain a constant forward velocity in this mode - perhaps implemented with PID control. If the robots sensor detects an obstacle in the path, then the robot would transition to the stop state. Finally, if the robot detects that it is on the red exit panel, then the robot would move to the finish state.

2. **Stop:** in the stop state the robot would set the velocity to zero, and checking the encoder that this was true. This state would persist if the robot was not stopped. If the robot was stopped, and the path was blocked, then the robot would transition to the turn state. If the robot was stopped, and the path was not blocked, then the robot would transition back to the forward state.

3. **Turn:** in the turn state, the robot would move the left motor forward, and the right motor in reverse, to perform a 90 degree pivot turn (a turn on the spot). Once the robot has complete the 90 degree turn, the robot will automatically transition state back to the stop state.

4. **Finish:** if the robot transitions to this state, then the robot is current located over the red exit panel for the environment. The robot will signal (audio or otherwise) that it is exiting the environment. The robot cannot transition out of this state - it is persistent.

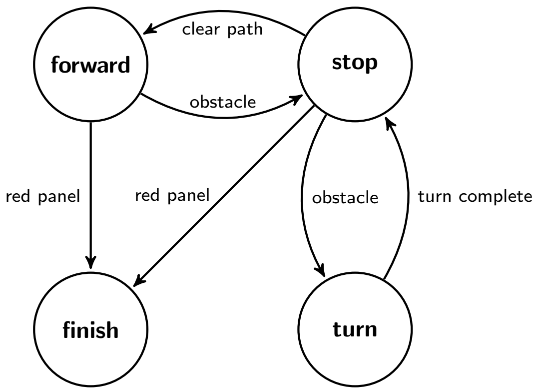

Figure 3: A state transition diagram which helps to explain how the robot moves from state to state in operation.

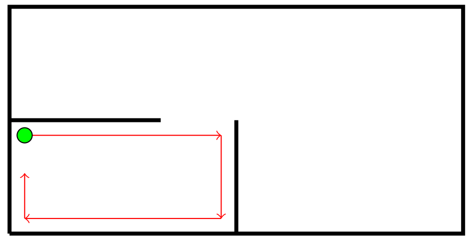

Figure 4: A simple example of environment design which would cause a basic robot, like the one detailed above, to fail. This figure shows the robot (a green circle) becoming trapped in an environment structure forever.

Whilst simple to implement, this design poses high risk of failure, and would need careful consideration of scenarios which may cause this failure. One elementary example of how this system might fail can be seen in Figure 4. Significant modifications to the scheme, and subsequent simulation or experimentation would need to be undertaken to ensure the required outcomes are met if this design is selected. A full description of the design option advantages, disadvantages, and hardware requirements can be found in Table 2.

Table 2: Advantages, disadvantages and the hardware requirements for the design option

Advantages	**Disadvantages**
• Simple to implement basic function • Low cost • Easy to source hardware • Hardware has very few points of failure	• Environment topologies may exist that causes the robot to fail, or get trapped - these cases would need to be provided for individually • The design may be too simple to navigate more complex environments completely
Hardware Requirements	
• Rover 5 chassis • NI MyRio Xilinx 7Z2010 processor • 4 Channel motor control unit	• 3 × SHARP GP2D120 • 1 × SHARP GP2Y0A21YK0F

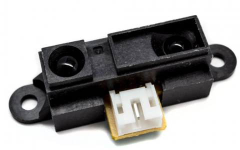

Figure 5: A picture of the Sharp GP2Y0A21YK0F IR Range Sensor

3.2 Moderate Design Complexity

The principal advancement of this design is underpinned by more sophisticated software used to traverse an environment topology. A similar IR sensor layout to the robot in Section 4.1 would be used, however, Kapoor et alias (2017) recommend an ultrasonic sensor on the front of the robot to provide longer mapping capability for more effective paths. Since environment dimensions are known, the map could be discretised into robot sized segments, as shown in Figure 6. The software would capitalise on this structure by creating a 3-dimensional `char` array to map progress. The array would be 4 times the size of the discretised map since the starting location is randomised, and depth would be 6 elements, whose contents are described in Table 3. Mapping the environment in this way would allow more intelligent options for navigating the terrain, allowing the robot to document visited areas, and regions of the environment with obstacles.

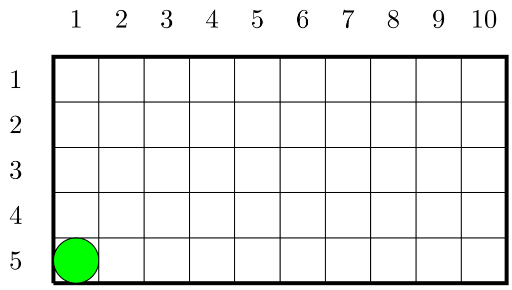

Figure 6: Since the environment dimensions are known ahead of time, the environment can be discretised into robot sized segments (note that the robot's dimensions are illustrated by the green circle). This allows the robot to exploit this structure to systematically explore the environment for the red exit panel.

Table 3: A 3D array struction is used by creating an array of length 6, which is stored in each grid location

Array Index	**Information Stored**
0	visited (v) or unknown (u)
1	direction that the robot faced the first time it visited the spot (f,b,l,r)
2,3,4	left, front, and right sides: wall (w), no wall (o), exit (x)
5	the direction from which the robot leaves the location with respect to the direction that the robot faces: forward (3), right (2), left (1)

This scheme would provide higher levels of assurance that the robot will traverse the full map and arrive at the exit panel, however, this implementation runs the risk of exceeding the 3 minute time limit - it is easy to imagine a scenario in which the exit is located in the final panel visited. See Table 4 for pros, cons, and hardware requirements.

Table 4: Advantages, disadvantages and the hardware requirements for the moderate complexity design option

Advantages	**Disadvantages**
• Robot will try to traverse the entire map, looking for the exit zone since the robot will systematically move through the environment • Relatively easy to build the software • Only requires minor additional hardware compared to low complexity design	• Runs the risk of exceeding the maximum time of 3 minutes to find the exit • Environment topologies may exist that cause the robot to believe areas of the environment are inaccessible
Hardware Requirements	
• Rover 5 chassis • NI MyRio Xilinx 7Z2010 processor • 4 Channel motor control unit	• 3 × SHARP GP2D120 • 1 × SHARP GP2Y0A21YK0F • 1 × PING))) Ultrasonic distance sensor

3.3 High Design Complexity

The previous two design options are best suited to basic environments, like the one shown in Figure 7. This environment has obstacles with surfaces that are orthogonal or parallel to other obstacle surfaces. The design options shown in Sections 4.1 and 4.2 may start to encounter difficulties when presented environments that are more complex, like that shown in Figure 8.

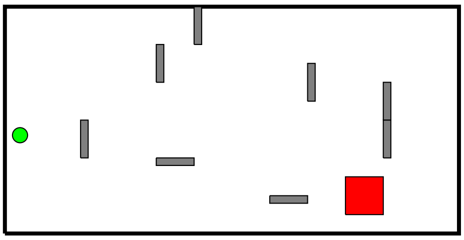

Figure 7: An example of a low complexity environment with obstacles having parallel or orthogonal surfaces

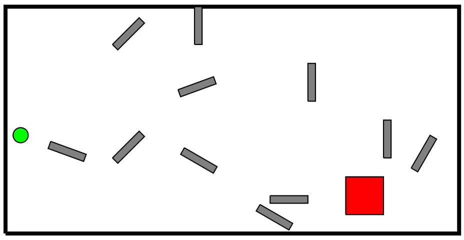

Figure 8: An example of a more complex environment which has obstacle surfaces at angles, providing more opportunities for simple system failure.

This is because obstacles may fall across multiple grid regions, or simply because the variability of obstacle angles create more permutations of environments in which a simple robot may get trapped. This would increase the difficulty involved with ensuring that each of these unique 'trapping' environments are considered. A more robust solution for dealing with complex environments, would be to use more sophisticated sensors to map the environment with higher fidelity.

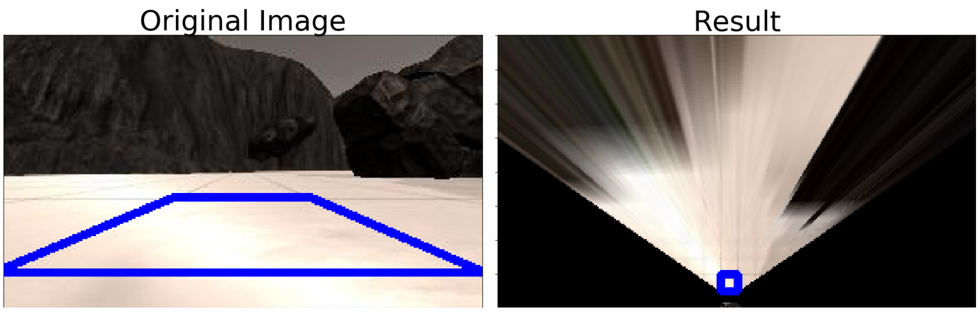

Figure 9: The picture on the left shows an image captured by an RGB camera mounted to the front of the robot, and the picture on the right shows a transformation of the original image for navigational purposes

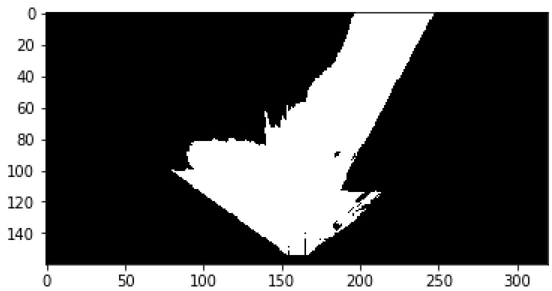

Figure 10: Using a simple RGB (or HSV) segmentation filter, the navigable terrain is highlighted to use for obstacle avoidance

There are a number of different options when looking at more sophisticated sensors. Some of these include optical (RGB) cameras, optical with IR depth (RGBD) cameras, or Lidar. The later two sensors provide advanced options like developing point clouds of an environment, but are considerably more costly. An example showing a simple RGB camera mounted to the front of the robot can be seen in Figure 9. The image from the robot camera is transformed to aid conceptualisation of the environment map - this step is purely aesthetic. Figure 10 shows the segmentation of navigable terrain and obstacles using a HSV filter. Advanced sensors provide rich environmental information sets, which allow for more sophisticated mapping and navigation algorithms. According to Cadena, et alias (2016), simultaneous localisation and mapping (SLAM) is widely employed in autonomous robotics. SLAM allows simultaneous estimation of robot state (x-coord, y-coord, z-coord, roll, pitch, yaw), and mapping of the environment. Much less sophisticated techniques, such as using odometry for mapping, tend to accumulate error over time. SLAM provides a 'reset' of these errors by revisiting previously mapped areas. This approach allows robust accurate mapping, and efficient location of the exit zone, meaning the robot can reach it's objective faster.

Table 5: Advantages, disadvantages and the hardware requirements for the design option

Advantages	**Disadvantages**
• Higher fidelity mapping • Reduced errors in navigation • Rapid navigation to exit zone	• Sensors are costly • Technically challenging to implement • Runs the risk of 'over engineering' a solution
Hardware Requirements	
• Rover 5 chassis • NI MyRio Xilinx 7Z2010 processor • 4 channel motor controller	• 3 × SHARP GP2D120 • Intel 82634DSB2P RealSense R200 Camera • (OR) Scanse Lidar Sweep Scanner

4 Project Management

4.1 Project Personnel

This project requires both technical and non-technical skills. The success of the project depends on the efficiency and productivity of the team. The delegation of tasks has been distributed amongst the group based on individual strengths and weaknesses. This will assist the team to work efficiently on each task. The expectation of the team is that each member of the group contributes to the project in accordance with the breakdown shown in Table 6.

Table 6: Project Personnel

Name	**Technical Contribution**	**Non-Technical Contribution**
Shane Reynolds	Programming, software integration and hardware assembly, testing, code troubleshooting	Team Leader//Report preparation
Tatyana Maltseva	Vehicle assembly, hardware integration, testing, programming assistance	Report writing, research and information gathering, presentation preparation
Sakon Nadthayai	Vehicle assembly, hardware integration, testing, programming assistance	Report writing, research and information gathering, presentation preparation

4.2 Project Plan

The complexity of this design project requires a structural logical approach. Hence its imperative to detail a project plan to ensure that the project is completed in a timely manner and to a high standard. A sequence of tasks was created to identify all important due dates. Figure 11 shows the main milestones and the duration of each phase. Each phase consists of necessary tasks to complete the design, construction and implementation of the autonomous robotic vehicle. For more information about each phase and its tasks, please refer to Gantt Chart on page 12.

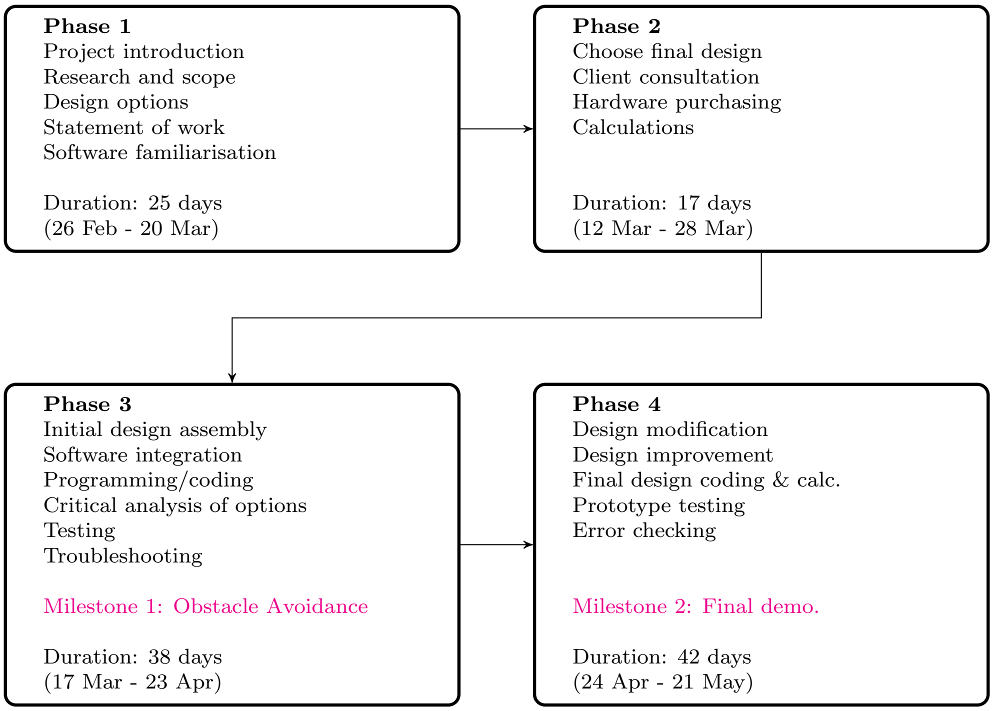

Figure 11: Phase description of the project, providing an overview of the project timeline, and highlighting key deliverables and milestones

From the Figure above it can be seen that the longest phases of the project are 2 and 3. These phases comprise the most challenging elements and two major milestones of the project i.e. completion of the robotic vehicle electrically and mechanically, development of working algorithm, design calculations, testing and troubleshooting and finally completing technical report.

5 Project Stakeholders

The identification of the project stakeholders is a primary task of any project. Project stakeholders are those groups/individuals that design and execute the project; change project direction if necessary; and last but not least decide on the outcome of the project thus it's imperative to acknowledge all the stakeholders of the project.

The stakeholders of this project are as follows:

- **Client** - Erwin Chan, Rob Wolff, Charlie
- **Design team** - Shane Reynolds, Sakon Nadthayai, Tatyana Maltseva

6 Budget Estimates

Each of the three design options require the use of the Rover 5 chassis, the 4 channel motor controller, and the NI myRio Xilinx processor. The main differentiation in terms of cost comes down to the sensors that will be employed and the estimated man hours involved with the implementation. Table 7 provides the estimated budget for each option.

Table 7: Budgeted costs for the sunk costs, and each design option

Sunk Costs		**Low Complexity**	
Rover 5 chassis	$68.81	3 × SHARP GP2D120	$47.85
myRio 1900	$814.00	1 × SHARP GP2Y0A21YK0F	$15.95
4 channel motor controller	$21.95		
Prototyping equipment (breadboard, wiring, etc)	$50.00		
Total	$ 954.76	Total	$63.80

Moderate Complexity		**High Complexity**	
3 × SHARP GP2D120	$47.85	2 × SHARP GP2D120	$32.90
1 × SHARP GP2Y0A21YK0F	$15.95	Intel RealSense R200 Camera	$127.50
1 × PING))) Ultrasonic distance sensor	$29.99	(optional) Scanse Lidar Sweep Scanner	$349.00
Total	$93.79	Total	$509.40

The range of costs for the proposed options start at $1018.56 for the cheapest option, up to $1464.16 for the state of the art solution. The costs for these products were sourced from the the following online retailers:

- **Mouser:** `https://au.mouser.com`
- **National Instruments** `https://www.ni.com`
- **Pakronics:** `https://www.pakronics.com.au`
- **Parallax:** `https://www.parallax.com`
- **RS Components Australia:** `https://au.rs-online.com`
- **Scance:** `http://scanse.io/`

7 Project Constraints

This project has several constraints which restrict project options. The following are the main project constraints:

- **Cost:** as identified previously this project has three possible design options. Each design option can be categorized by its complexity level. To achieve high performance, hence the desirable outcome for the client, the design would then require technically advanced components for example, a RGBD camera and sophisticated sensors and algorithm. The tradeoff here is performance versus budget, increasing the budget would provide the best performance.

- **Time:** one of the biggest constraints of this project is time. This project must be completed within 11 weeks which limits the scope of work. To achieve high performance using advanced features would require a deeper research, more time to implement the design and its algorithm. Time constraint can sabotage the quality of the robotic vehicle and the design scope, hence careful planning and adherence to the project plan must be followed to ensure project completion on time and in accordance with the scope.

- **Scope:** if the scope creep happens during the project it will result in increased cost and time, and reduced quality. Given the nature of this project, the scope creep may potentially appear during the testing period for the second milestone.
- **Quality:** if not managed properly the identified time constraint may affect the quality of the vehicle. The vehicle may not perform in accordance with the specification or perform poorly which is always undesirable for the client.
- **Resources:** this project limits the choice of software and processor that can be used for robotic vehicle design implementation. In this project the designing team is only allowed to use NI myRio 1900 or an Arduino devices and their compatible softwares such as LabVIEW, which limits the design options and requires learning time.

8 Preferred Design & Points for Discussion

A design option which meets the project requirements and operates within the project constraints, whilst delivering an optimal outcome in terms of performance and cost is the most desirable design solution. In reality, the best solution is most likely to include elements from the low, moderate, and high complexity design options - there are useful design elements in each. The most robust solution would be the highest complexity design, however, this is likely to provide a solution which is beyond the project scope resulting in unnecessary expenditure. In contrast the low complexity design holds higher levels of risk that the solution will not be able to sufficiently meet project requirements. The preferred design option exists somewhere between the moderate and high complexity designs. Final decisions on the actual option will be dependent on additional information sourced from the client. Table 8 provides a non-exhaustive list of project aspects which need further clarification.

Table 8: Additional information that needs to be sought for effective selection of robot design

Environment Questions	**Other Questions**
What materials is the environment constructed with?	Will the location of the exit zone always be in the same sport? Or will the location be random?
Does the environment change, or will it be the same topology?	Will the location of the exit have any defining characteristics (e.g. located next to a wall)
Is there access to a test environment?	Is the project restricted to using NI myRio hardware?
Does the entrance to the environment always start at the same point, or is it randomised?	
Will the walls and obstacles be assigned with parallel, and orthogonal surfaces?	
Will the environment be based on a grid? Or will the environment be random?	

9 References

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., & Leonard, J. (2016). Past, Present, and Future of Simultaneous Localisation and Mapping: Towards the Robust-Perception Age. *IEEE Transactions on Robotics*, 32 (6) pp 1309-1332

Kapoor, S., Mahesh, K., Ruzinov, D., Battipaglia, J. (2017). Autonomously Solving Mazes with Robots. Retrieved from `http://soe.rutgers.edu/sites/default/files/imce/gov2017/Autonomously%20Solving%20Mazes%20with%20Robots.pdf`

Kumar, R., Jitoko, P., Kumar, S., Pillay, K., Prakash, P., Sagar, A.,...Mehta, U. (2016). Maze solving robot with automated obstacle avoidance. Paper presented at: *2016 IEEE International Symposium on Robotics and Intelligent Sensors, Tokyo, Japan.*

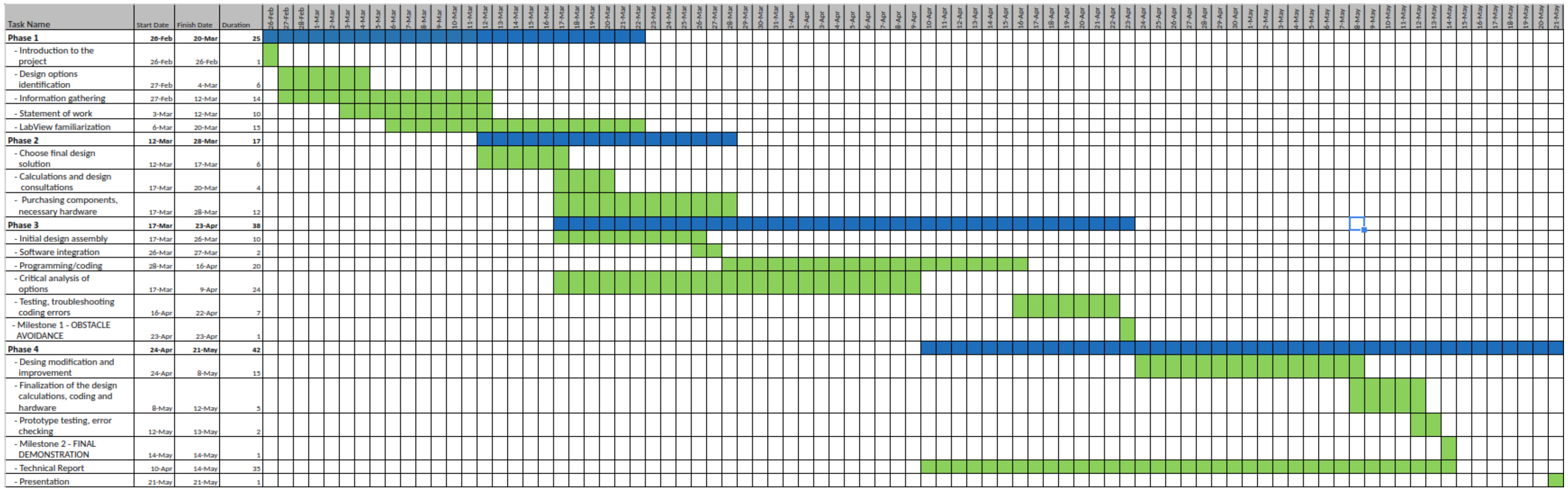

Figure 12: A gantt chart breaking down the phases of the project and detailing the labour assignment through the project life cycle.