# Project Summary

## Main Goal

The efforts of this project should result in a robot that is or can perform the following:

* Can navigate a maze from a start point to an end point via the most efficient route possible without any aid after having completed an initial mapping run through the maze
* Small enough to hold in one hand
* Should have its own power supply

## Stretch Goals

The core functionality of the robot is listed above, however in the event that these goals are reached earlier than expected the robot can be extended to achieve the following targets:

* Transmission of the robot’s navigation from the robot to a phone app via Bluetooth
* Allow a photo to be taken of the maze from above and be sent to the robot via Bluetooth to allow mapping without an initial run through the maze

# The Maze

## Maze Specifications

A maze’s structure will be defined as follows:

* It should be customable
* It must be closed, i.e. have no way to get outside the maze
* All the walls must be angled on the same 180° sides
* Every wall’s length must be rationally divisible by the same number, e.g. a wall could be 10cm, 20cm, 30cm or 40cm long but not any other value that isn’t divisible by 10 exactly
* The thickness of the path through the maze should always be the minimum length of a wall (5cm for example)
* It should have a start point and end point

## Suitable and unsuitable mazes

|  |  |
| --- | --- |
|  | **Valid Maze** |
|  | **Valid Maze**  Multiple routes to the finish  Dead ends |
|  | **Valid Maze**  Loops and many routes to finish |
|  | **Invalid Maze**  Has openings |
|  | **Invalid Maze**  No possible route to the finish |
|  | **Invalid Maze**  Varying thickness of the corridors |

## Building the maze

### Possible materials

|  |  |  |
| --- | --- | --- |
| **Material** | **Pros** | **Cons** |
| LEGO® / LEGO® Duplo | Easy to setup and readjust | Expensive for the amount that would be needed |
| Cardboard | Cheap | Would require time spent to craft the maze |

## Cardboard Maze Design



The cardboard maze will be based on a large square piece of reinforced card. The size of this will vary depending on the size and movement style of the robot so this will be decided at a later stage.



This base will be raised above the surface it’s on by 1-2cm. It will then be split into a grid and slits will be cut into each section of the grid (all the black and red lines in the diagram to the right).

The walls will be made of thinner card that can be slotted into the base to form the pathways of the maze.

# The Robot

## Requirements

A suitable robot must be chosen for this project, the requirements are as follows:

* Is fully programmable
* Should be relatively small
* Capability to move forwards (backwards is ideal but optional)
* Capability to turn up to 180° (360° ideal but optional)
* Can run from a reliable, affordable source of power (alkaline battery, typical AC, USB, etc.)
* Has a suitable sensor built-in or the option of attaching one to it

## Build or Buy

One of the foreseen questions is whether the robot should be built from hand using several components such as the board, some wheels, a sensor and such, or whether a pre-built one should be bought to avoid potential complications with building one.

### Decision Table

Each point can have a maximum score of 50 and a minimum of -50. The cost of these weightings are determined based on how much impact the factor has on the overall decision.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Build** | **Score** | **Buy** | **Score** |
| **Pros** | Complete flexibility and control over the design and creation of the robot and no more | +45 | Saves time that could be spent on development | +35 |
| Only need to buy the exact components required for the project | +10 | May find a solution that is designed for the exact or similar purposes | +45 |
| Likely to be cheaper than a pre-built solution | +30 | Will be tested by previous consumers | +40 |
| A huge amount of options, ultimately deciding what the final robot will be capable of | +40 |
| **Cons** | As the separate parts may not have been tested together before, problems could occur when assembling the robot | -20 | May come with unnecessary parts | -5 |
| Lots of time and research needed to find all the best components on a limited budget and time spent assembling them | -35 | Likely to be the more expensive | -30 |
| A very limited number of choices of pre-built robots | -25 |
| **Total** |  | **70** |  | **55** |

### Building the robot

Building the robot from several components is the optimal solution for meeting the requirements and having complete control over the design. The robot would be built to meet the project’s goals making it a more favourable solution. However the opportunity cost of doing this is high as more time would need to be dedicated to this process that could otherwise be spent working on the maze and navigation algorithms.

A possibility would be to seek assistance for this process from an electronics engineer.

### Pre-built Candidates

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Image** | **Name** | **Price** | **Notable Specs / Features** | **Other** | **Link** |
| http://www.picaxe.com/Thumbnail.ashx?image=~/Site_Resources/Media/Site_1/bot120/BOT120.jpg&h=600&w=800&mode=Absolute&k=2c044d78a8e5c71d4623dd3169dd9da000e9ff7b1efcb317174519cb7c7c104c | PICAXE-20X2 | £48 including UK shipping cost | * Bumper module * Line follower module | * Is a barebones robot but extra components are easy to attach, although this would add extra cost and complications * BASIC only, no support for other languages or frameworks | [PICAXE](http://www.picaxe.com/Hardware/Robot-Kits/PICAXE-20X2-Microbot/) |
| http://www.picaxe.com/Thumbnail.ashx?image=~/Site_Resources/Media/Site_1/kit110/kit110.jpg&h=600&w=800&mode=Absolute&k=0e00573d1a61a7119e640e40734c0d09a9f60d8adaf07e6e1eadfc81b6f263dc | PICAXE PICone | £90 including UK shipping cost | * PICAXE-28X2 microcontroller running at 64MHz | * Designed specifically for Micromouse competitions, making it a suitable choice * BASIC only, no support for other languages or frameworks | [PICAXE](http://www.picaxe.com/Hardware/Robot-Kits/PICAXE-PICone-Micromouse/) |
| Pololu 3pi Robot Kit | Pololu 3pi | £70 (free shipping) | * C-programmable Atmel ATmega328P microcontroller * 32KB flash memory, 2 KB RAM and 1 KB EEPROM * Capable of speeds exceeding 3 feet per second | * Designed for line-following, doesn’t have the necessary requirements for a maze with walls | [Pololu](http://www.pololu.com/product/975)  [Hobbytronics](http://www.hobbytronics.co.uk/3pi-robot) |
| http://ecx.images-amazon.com/images/I/81NIouhWlwL.jpg  https://www.bananarobotics.com/shop/image/cache/data/sku/BR/0/1/0/0/6/BR010062-DFRobot-2WD-MiniQ-Robot/DFRobot-2WD-MiniQ-Robot-top-600x600.jpg https://www.bananarobotics.com/shop/image/cache/data/sku/BR/0/1/0/0/6/BR010062-DFRobot-2WD-MiniQ-Robot/DFRobot-2WD-MiniQ-Robot-bottom-600x600.jpg | MiniQ 2WD | £60 including shipping cost from the US | * Fully programmable Atmel ATmega328P microcontroller robot brain. * Five bottom facing IR reflectance sensors for map navigation and edge detection. * Two forward facing IR transmitters for obstacle avoidance. * One forward facing IR receiver for obstacle avoidance and remote control. * Two forward facing CDS photocells for seeking light or dark. * Two motors with encoders to fully control the speed of movement. * Travels at speeds up to 31 inches per second. * A ball caster for stability. * Three user input buttons wired to a single analog port. * A buzzer for making beeps and playing tunes. * An ICSP interface for advanced programming. * Handy power and reset buttons | * Fulfils the requirements and would be easily extensible extra goals | [Banana Robotics](https://www.bananarobotics.com/shop/DFRobot-2WD-MiniQ-Robot)  [Amazon](http://www.amazon.co.uk/MiniQ-2WD-Complete-Kit/dp/B00E68HY88/ref=sr_1_cc_1?s=aps&ie=UTF8&qid=1417972564&sr=1-1-catcorr&keywords=4WD+MiniQ) |
| iRobot Roomba Vacuum | iRobot Create | £100 including shipping cost from the US | * Caster, left, and right wheel drop sensors * Left and right bumper * Wall sensor * Left, front left, front right and right cliff sensors * Omnidirectional IR receiver | * No longer available to purchase (was available at the start of the project) * Lacks much internal processing and computing power, would need this attached * Expensive | [iRobot (page removed)](http://store.irobot.com/education-research-robots/irobot-create-programmable-robot) |
| http://www.robotstorehk.com/micromouse/images/RJ_linetracer.gif | RS-Cruiser | £87 including shipping cost from China | * Size: 82 × 125 × 60 (H) mm * CPU Board: Atmega8 (8-Bit RISC MCU) * Sensor: Line sensor module- Infrared emitter & sensor × 6, Wall sensor module- Infrared emitter & sensor × 6 * Motor: MRM-GM03 Gearbox (DC Motor × 2) * Speed: 0 - 50 cm/s * Display: LED × 6 * Key: RESET × 1, User Push Button × 2 * Battery: AA × 4 (not included) * ATmega8 CPU (8Kb In-System Programmable flash memory) * Line detecting sensor module & wall detecting sensor module * Free C compiler(WinAVR) | * Very small website and company that is based in China – could be complications * Hardly any information about it online | [robotstorehk](http://www.robotstorehk.com/micromouse/RS-CRUISER.HTML) |

# Methodology

Rolling Wave Planning is the initial take on the project as many factors and variables have impacts on which direction the project will take to achieve the goal.

The project’s development will be carried out using an agile approach due to the lack of predictability in the analysis, design and implementation phases. Specifically, obtaining the robot may have significant time fluctuations depending on the results of the analysis of whether one can be bought or if it will need to be built.

Agile was also chosen to get a semi-functional prototype rolled out as soon as possible to get potential hurdles with the robot’s core functionality cleared up as opposed to them occurring in the later stages of the project when time constraints are limiting. Then the robot can be further developed and stretch goals can be worked towards.

<http://en.wikipedia.org/wiki/Rolling_Wave_planning>

<http://en.wikipedia.org/wiki/Agile_software_development#Adaptive_vs._predictive>