Group 10

System Design Project

Mentor: Vesko Stefanov

Client: Iordanis Chatzinikolaidis

26th January 2018

Project Plan

Alex Shand s1530411
Grzegorz Wilk s1504889
João Catarino s1512284
Katie Worton s1530238
Rosina Paige s1407973
Campbell Scott s1530451
Stephen Waddell s1346249

1. Introduction

We at *!spam* have set out to improve office efficiency. Our robot will deliver mail to the employee's desks all by itself. In doing so, it will relieve receptionists from those duties.[1] Each letter will be scanned as it is loaded in. It will identify each recipient, find their desk and finally offload their mail.

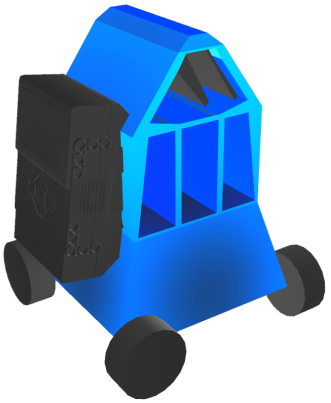

Figure 1: 3D Prototype.

This report will illustrate the market viability, outline how we will create a fully working prototype through the course (initial concept on the right) and how the group will organise itself.

1.1 Inspiration

The inspiration for the *!spam* robot comes from a few sources:

- Delivering mail in offices is a mundane and repetitive task which can lead to human errors and inefficiency. With an autonomous robot delivering the mail, there will be fewer errors allowing workers to get on with their jobs without distraction.
- As man-hours are a scarce and expensive resource, companies would rather employees stay on track or get outsourced. A mail robot would help saving money on wages.
- Freeing the receptionist from physically delivering mail will prevent more urgent tasks from holding up time-sensitive deliveries.
- Even though internet communication has become the standard, there were still 327.5 billion mail pieces delivered worldwide in 2016[2]. Within offices, the people responsible for dealing with them - receptionists - are a growing labour cost (with the US Labour Department expecting a 95,700 person increase over the next 10 years.[1]).
- Companies are looking for new technology in every aspect of the work environment. A robot will bring them up to speed in mail delivery.

Figure 2: Mailmobile.

Some offices are currently using mail delivery robots. One example is the "Mailmobile" (pictured right). Its users claim that it provides "increased mail delivery efficiency"[3]. In contrast to our proposed design, however, it doesn't identify which desk the mail is supposed to end up at.

2. Goal Description

Our mission is expressed by a few user stories and will be our focus throughout the development of the robot:

- As a **receptionist**, I want to **have the mail delivered to employees** so that **I can focus on important tasks instead**.
- As an **employee**, I want to **have the mail delivered to me** so that **I don't have to fetch it myself**.
- As the **company owner**, I want to **have automated mail delivery** so that **I reduce costs**.

2.1 Idea Description and Features

2.1.1 Base functionality

Our robot will automate mail delivery in the office. Initially, it will be loaded with letters and parcels at the reception (see state 1). It will be the receptionist's responsibility to assign desks to the mail slots using a web interface and then indicate that the robot may commence delivery.

The robot will plan the most efficient route using its internal map that was specified during initialization (see state 2). It will follow a tape along the floor and orientate itself at junctions and stopping points. This will be aided by feedback from sensors.

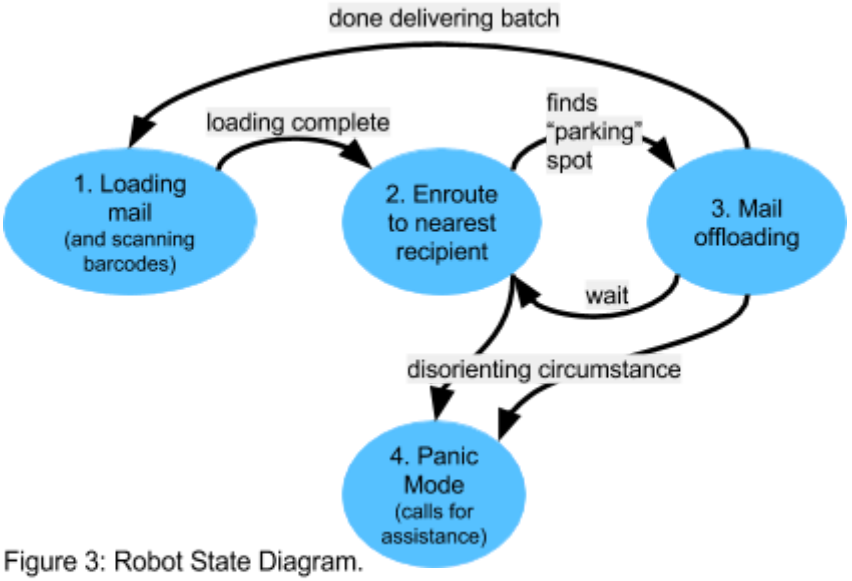

Figure 3: Robot State Diagram.

At each delivery point, the robot will offload the mail with an optional sound based alert (see state 3). It will then continue on its path seeking the next delivery point.

When the robot senses it is not at the correct location, encounters an unexpected obstacle or waits at a door it will enter panic mode (see state 4) and inform the receptionist via a notification of its current location and status.

2.1.2 Extension features

As an extension, instead of the receptionist placing letters into the correct brackets, there would be one slot where each letter would be scanned (to identify which recipients to serve) and placed in one of the brackets automatically. Parcels would still have to be loaded manually in the designated brackets. The web interface may also be further improved with recipient-desk mapping, live location tracking, summoning the robot back, and the ability to check the robot's battery level and the number of mail pieces delivered.

The robot would also be able to audibly call for help if it encounters a closed door or has to move between floors in a lift.

2.2 Limitations of the Final Prototype

Due to limitations with the structural integrity of Lego and the power of the motors, the prototype robot is going to be considerably smaller than the intended final product. For example[4], the Mailmobile mailbots are approximately 1.30m (52 inches) tall. This is the intention for our design under ideal circumstances. However, the prototype will be approximately 18 cm in height. The biggest effect this will have is the carrying capacity of the robot. There will be fewer letter slots and only one parcel slot (gray bracket in the picture) in the prototype. There also will be no outgoing mail bracket.

The mail delivery mechanism will utilise gravity and the test letters (and parcels) will be proportionally smaller and lighter than the real load. As such the prototype will deposit mail by the

desk instead of on it. Finally, the intention would be for the full sized robot to automatically read the handwritten address on each letter to classify it for delivery. Due to time constraints, this capability could be simulated via barcodes attached to each test letter.

Some assumptions made: the mail will never jam, the tape the robot follows will always be intact, there will always be a route between any two points, humans will assist the robot by opening doors, the robot will service one floor at a time and will require assistance when moving between floors.

2.3 Demo Environment

As our robot follows lines to navigate, we will require part of the floor where we can lay tape. After consulting with the technician, we were told that we will be assigned one of the 6 test areas on the 3rd floor of Appleton Tower. As our requirements are easily accommodated, we are ready to share this space with other teams or use other university rooms for testing. At every point, we will maintain the same testing conditions as will appear in the final demonstrations themselves.

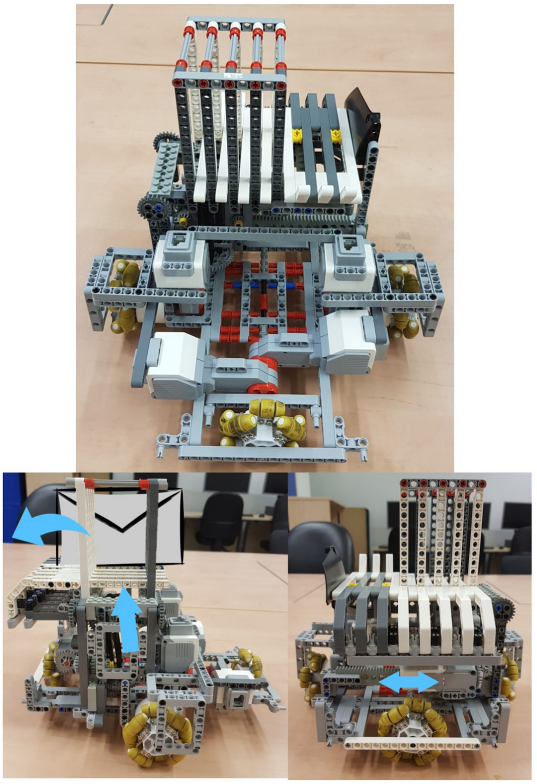

Figure 4: Lego Prototype.

3. Task Planning

3.1 Milestones

We decided to align our milestones with the structure of the course, using client demos and presentations as key deadlines. Until the second client demo, we will focus on the minimum features the robot needs to perform the task. Following the second client demo, we will implement the extension features.

3.1.1 First client demo

We will have a robust base structure with integrated dispensing hardware. The robot will be able to maneuver on its holonomic wheels and accurately follow a line using two color sensors. We will demonstrate this by displaying our hardware and having the robot turn at curves.

3.1.2 Second client demo

We will have an improved robot chassis with sensors for location feedback and a sonar for collision avoidance. Route planning will be implemented. A web interface that allows the receptionist to input the recipients to be reached will be developed, as well as a notification system for when the robot gets disoriented or otherwise needs assistance. We will demonstrate the robot navigating a system of lines, delivering mail at correct designated locations, with its route determined by user input.

3.1.3 Third client demo

For the third client demo, we will work on stretch goals. These may include the initial mail classifying hardware with the barcode scanner, additional web interface functionality, and/or calling for human help. The UI will be fully finished so that the user guide can be developed. We will demonstrate this by showing the complete flow of the system, from mail input, through to route planning, navigation,

unexpected circumstances and mail offloading. Also, we will demonstrate the previously suggested extension features that we implemented.

3.1.4 Final demo

We aim to have the system as close to finished as possible by the third client demo, due to the fact that other responsibilities will require our attention at this point in the semester. However, there are sure to be things that can be improved. We intend to polish our code, robot and user interface between the third and final client demos.

3.2 Time Allocation

We've created a graph that breaks down the time resources of each member of the team throughout the semester. We will dedicate the "Delegated Work" section to building the product and each member will have different allocations, according to their function within the team.

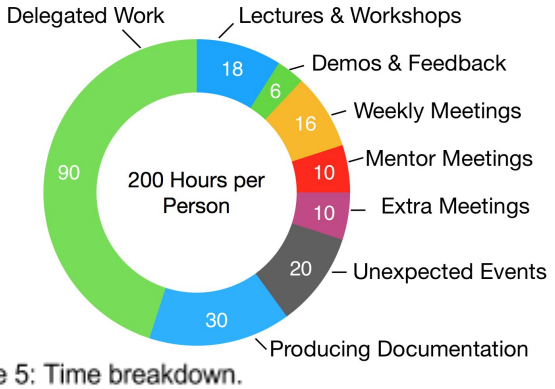

Figure 5: Time breakdown.

The Gantt Chart below displays our intended progress and includes the time redundancy outlined in the Risk Assessment. This will be used to measure deviation from the plan, visualise task dependencies and allow deadlines to adapt to challenges as the course develops.

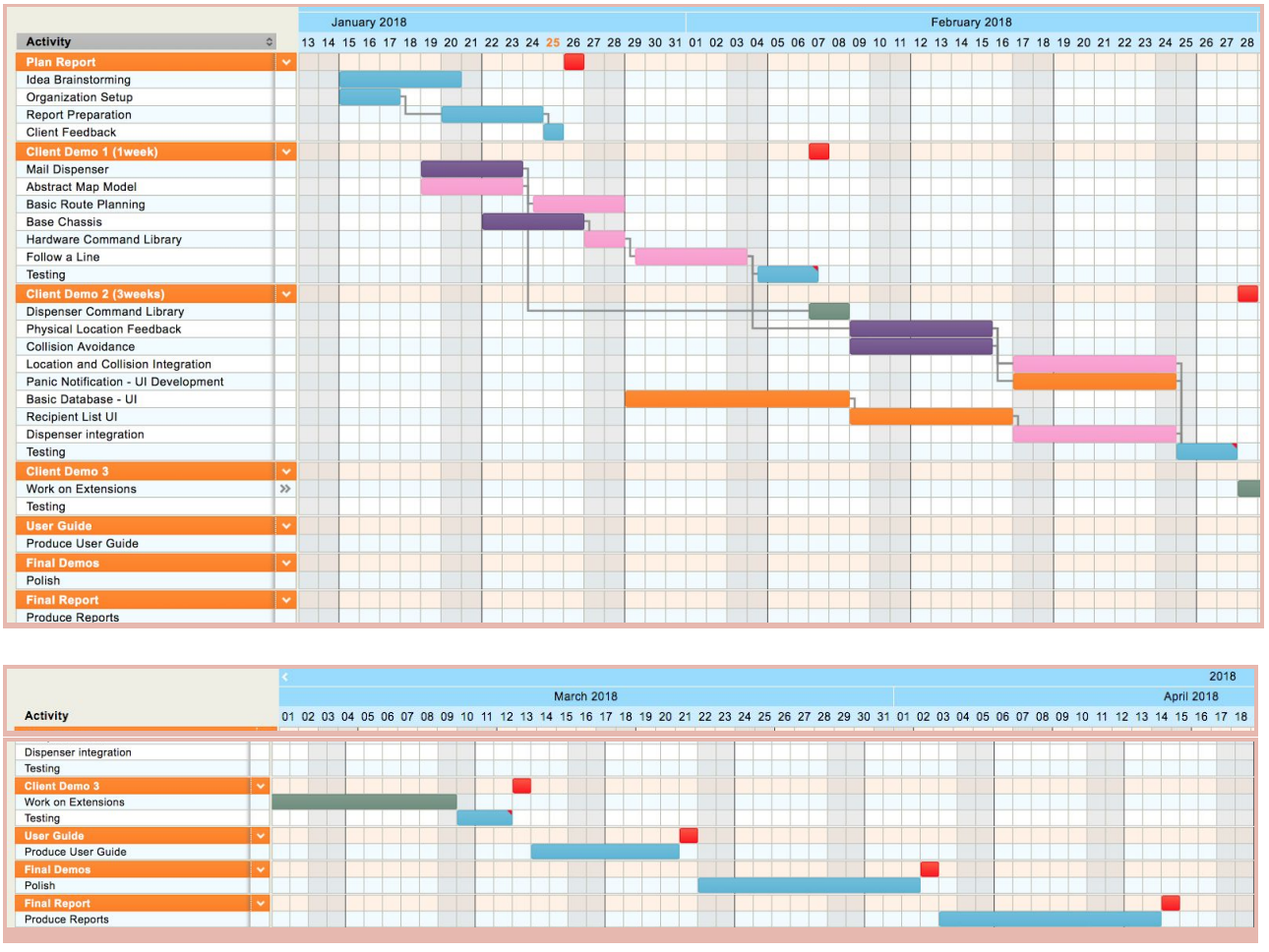

Figure 6: Gantt chart. Pink: Navigation, Purple: Hardware, Orange: UI, Blue: Organization

3.3 Team preferences

All team members expressed interest in participating in several sections of the project. The preferences demonstrated were discussed and we decided to define areas of competency, important for producing cross-functional teams when delegating responsibilities, as described below:

Alex Shand: Navigation & Software - I love all kinds of challenging programming tasks from lower level algorithms to high-level control of the robot. I have experience programming in both C++ and Python so I think I am well suited to programming the back end of the robot.
Grzegorz Wilk: Hardware - I am an aspiring software engineer, eager to gain new experiences who conveniently also has a long-lived passion for LEGO.
João Catarino: Project Manager - I hope to use my interpersonal skills to unlock the team's potential and contribute to the development of the product especially on the software side.
Katie Worton: Hardware - I hope to apply the knowledge I have acquired building custom games controllers to help with the Arduino and hardware sides of the project.
Rosina Paige: User Interface - I have experience in database management and web development.
Campbell Scott: Testing & Software - I am good at and enjoy all aspects of software engineering. I am eager to advance my skills in testing and programming.
Stephen Waddell: User Interface & Software - I live and breathe software...

3.4 Projected Budget Allocation

Table 1: Budget Allocation.

Equipment		
Part	**Function**	**Price**
All provided parts: lego, EV3, arduino	Base for the project.	£0
4x Large holonomic wheel	Movement along independent axis.	4 x £10 = £40
N150 Edimax Wi-Fi dongle	Communicating with the web user interface.	£6
Barcode scanner + Shield	Simulating the handwritten recipient analysis.	£20 + £14 = £34
2x Motor connectors	Used to connect the motors to the Arduino.	2 x £3 = £6
Timed resources		
Type	**Projected hours**	**Price**
General technical consultancy	4 hours	£40
uCreate studio time	4 hours	£40
	TOTAL	£146

3.5 Risk assessment

Table 2: Risks developing the product.

Description	Probability/ Impact	Mitigation Approaches
Team member unreliability	Medium/ High	Work in subgroups so the risk of dependence on an individual is reduced and there is collective ownership.
Budget exceeded	Low/ Medium	Keep track of the finances and look for economical alternatives to expensive equipment.
Disputes between team members	High/ Medium	Discussions and meetings to resolve conflicts. Socialising outside of project-work to promote smooth running of the team.
Tasks not being completed before deadlines	Medium/ Medium	Progress meetings and pairing people who need help with those who can assist them, not depending on one individual to complete a subtask (work in subteams).
Limited Testing/Meeting Space	High/ Low	Maintaining open communication with the technicians to establish how we can best work with the Demo space. Final decisions at the end of each meeting choosing the location and timing of the next.
Pre-Demonstration Bug Handling	High/ High	Feature Freeze. Following from Chris Paton's guest lecture, we have taken on-board his advice and will not be implementing new aspects of the design for a practical time-limit of 3 days beforehand.
Mismatched aspects of the system	Medium/ High	Also taking Chris Paton's advice, we have will ensure to integrate the subsystems we work on as early as possible.
Sonar imprecision	Low/ Low	We are considering using an IR sensor instead for obstacle identification.
Barcode scanner unimplemented	Low/ High	Create a system which works without the barcode scanner, and extend the system, time permitting.

Table 3: Risks in operation of finished product.

Description	Probability/ Impact	Mitigation Approaches
Robot runs out of battery	High/ High	Precisely monitoring the power draw during development.
Obstacle encounter/ Trip hazard for office staff	Low/ High	Sense the environment and take appropriate action if unexpected objects/humans are detected (just like the Mailmobile, it "will not start until object is removed from path"[4]).

Robot cannot detect line	Medium/ High	If the line cannot be detected, either search for a line or ask for assistance.
Robot becomes disorientated	High/ High	Panic mode - the robot would call the front desk for assistance.
Robot delivers wrong mail	Low/ High	Web interface could allow for robot error reporting.
Office workers switch desks	High/ High	The web UI could permit desk reassignment.
Barcode scan failure	Low/ High	If a barcode is not scanned correctly an error will be reported.
Various sizes of letters	High/ Low	The environment will be controlled, so that even if the letters are different sizes, the barcode is always in the same spot.
Travel between multiple floors/doors required	Medium/ High	Robot could ask for assistance when traveling between floors/doors is required.

4. Team Work Process

The team will organize according to the agile software development approach in weekly iterations, partially adopting the scrum methodology. According to our research, “Agile projects are 28% more successful than traditional”[5] and “71% of organizations report using Agile approaches sometimes, often, or always”[6]. This will allow us to adapt our goals, incorporating the client’s feedback and suggestions that we receive throughout the semester, namely in client demonstrations, into our final product.

The features of the robot are divided into small objectives and we will work on those in pairs. Each pair will be responsible for producing a requirement along with unit tests of the implemented feature (see Trello section below). In addition, we will use the pair programming techniques whenever possible. Together with the iterative process, this will show us getting closer to the final product and improve progress tracking, whilst providing enough flexibility for the changing requirements.

We will have a scheduled meeting, every week, with all team members. We will discuss the past week’s results, reflect on what to improve the following week and assign objectives that we believe are feasible to achieve that week. At this meeting, the team members will also be able to track the progress of every feature, express any concerns, discuss any problems, and discuss their task preferences.

4.1 Auxiliary tools

Continuous integration: We will use **Travis CI** as our continuous integration tool. It will mitigate the risk of having errors in stable versions of our code, increase confidence in everyone’s code within the team, save time in running tests and incentivise us to build a robust test suite.

Code review: Our team will focus on producing well documented, readable and maintainable code. To help on that task we will implement continuous code reviews using **Codacy**. This will help identify improvements in code style, performance, unused code, error, and security prone code.

Communication: We believe **Slack** is perfect for group work, it is multi-platform, fast and allows for a good organisation by using mentions, channels, file uploads, notifications, and reactions. This instant messaging tool allows for fast communication, an all-in-one solution and fast integration with our other tools.

Version Control: The team will use **GitHub** to control changes in code, to allow for the concurrent development of multiple features, to act as a backup of our work, to detect conflicting changes in our code and to facilitate the collective ownership principle. GitHub was chosen because of familiarity, integrability with other services and the wide range of support material available online.

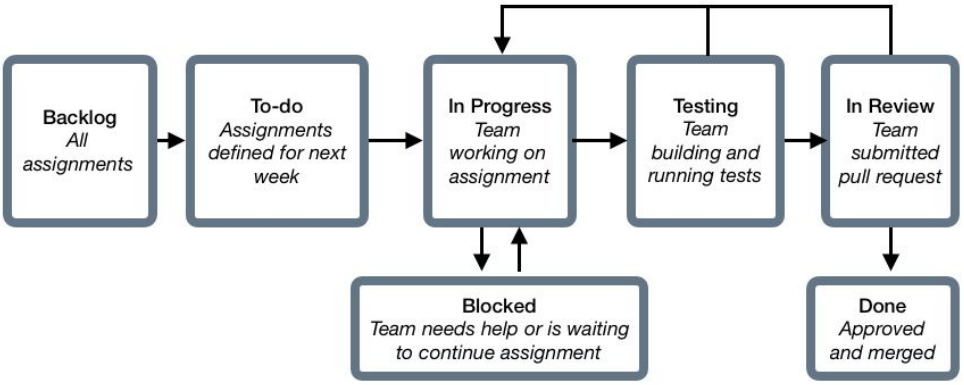

Figure 7: Trello board flow.

Task Board: The task management tool of our choice is **Trello**, a simple and easy way to organise tasks. All our tasks will be written in a backlog and every week we will decide which features to work on the following week. We will define deadlines, and delegate people and move those objectives along the lists as described.

5. Conclusion

Our team at *!spam* have extensively planned our ongoing development, building in redundancy and contingency plans where possible. We hope to reach or exceed our client's expectations, by providing the complete, functioning robot outlined in this document. We appreciate any feedback and advice so we can develop as a team throughout the process. To this effect, we have already consulted extensively with our client and mentor, and look forward to continue working with them. We all enjoy working together and are optimistic about our progress and our robot's prospects for the future.

6. Bibliography

1. Bls.gov. (2017). U.S. Bureau of Labor Statistics. [online] Available at: https://www.bls.gov/ooh/office-and-administrative-support/receptionists.htm [Accessed 22 Jan. 2018].

2. About.usps.com. United States Postal Services. [online] Available at: http://about.usps.com/who-we-are/postal-facts/size-scope.htm [Accessed 22 Jan. 2018].

3. Mailmobile Customer Cases. (1980). [ebook] USA: Bell & Howell. Available at: https://ia601508.us.archive.org/35/items/TNM_Mailmobile_office_mail_delivery_robot_1980_-_20171021_0159/TNM_Mailmobile_office_mail_delivery_robot_1980_-_20171021_0159.pdf [Accessed 24 Jan. 2018].

4. Mailmobile 4: The newest generation. (2002). [ebook] USA: Bell & Howell. Available at: http://www.egeminusa.com/wp-content/uploads/2016/10/Egemin_Mailmobile4.pdf [Accessed 23 Jan. 2018].

5. Pmi.org. (2017). Pulse of the Profession. [online] Available at: https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pulse-of-the-profession-2017.pdf [Accessed 23 Jan. 2018].

6. Pwc.com. (2017). Agile Project Delivery Confidence. [online] Available at: https://www.pwc.com/gx/en/actuarial-insurance-services/assets/agile-project-delivery-confidence.pdf [Accessed 23 Jan. 2018].