Project Requirements

This supporting document will specify the range of requirements for the ILP project that were considered. This will include talking about the stakeholders, the functional requirements the measurable qualities and also user validation vs verification.

Stakeholders:

The stakeholders are the people directly involved with the process of development and the finished product of the project. This will provided context of how our pizza drone delivering app will impact them in detail and our reasoning behind it. There are currently 5 main stakeholders that were considered, but there are chances that some were missed out or are too indirect with the project itself.

The Users:

The users will be the university of Edinburgh students who will want to have a easy option to order pizzas and have it delivered to them. They are a very important aspect to consider as the product result directly affect their satisfaction as the customer of our service, whether the pizza arrives is dependent on the end result of the product.

The Pizza Restaurants:

They will be the people who are providing the pizzas and are also relying on our service to receive a profit. They are relevant stakeholders as they require our functioning, reliable service to allow more profits and pizza sales. A malfunctioning app will drive sales down.

The Edinburgh Residents:

The residents living near any possible flightpaths are also considered stakeholders. Although they may not be the target consumers for our product, the fact that drones will be flying overhead will be of concern to them. If a drone fails, the safety of the passer by may be in danger. There is also a concern of privacy, where residents may feel that their private space in being breached in their homes around the area, where they should be able to feel safe.

Drone manufacturers:

They are stakeholders due to providing our service with the products to allow to drone delivery. They will profit off the sales of the drones to allow our service to function. A good service will mean more sales for them, and we might expect to also custom order drones from them to allow a more specific and efficient autonomous drone delivery system.

Software Developers:

In this case, I am the software developer, but other informatics volunteers will maintain the project in the following years. The developers will programs, tests, and develops the system/app that allows everything to work correctly. They connect the users to the restaurant, to allow a way to communicate and provide a route for the drone to take.

Measurable Qualities:

Some qualities had to be traded off in consideration of another cost aspect of the development process, which is the total time available. Some attributes were also considered to be favoured over another. Here are measurable qualities that were taken into consideration for the project.

Fairness:

To be totally fair in the ordering process, we would have to target delivering the orders that came first. Since in our sample project, the orders don’t have time allocation for each day the order came in, a fair process may be delivering the orders after a randomisation. However fairness was traded in this case to for performance, as orders from the closest restaurants were delivered first.

Reliability:

This attribute measures how likely the system is to fail within a certain time period, which also takes into account how major the fault is.

Security/safety:

How safe is the system to attacks, if unauthorised access is given to individuals who should not have it, if information is easily leaked. And how private details (such as card expiry, number and CVV) is stored.

Functionality:

This measures how correct the project matches up to the project specifications/requirements provided.

Performance:

Performance is measured by how many pizzas the drone may deliver before battery/moves run out for the drone. This is compared on a daily basis, and deliverable orders for a day will have better performance the more pizzas that were delivered.

Efficiency:

Efficiency is measured by how long before flightpath for a given day be calculated and given out. This was measured with runtime counter, but it’s also possible to time from the time taken to run the code to when the resulting output files were printed.

Correctness:

Correctness considers how many of the orders were correctly identified as valid or invalid, based on the sample order details provided from the rest server.

Validation and Verification:

Validation considers what the users/stakeholder may really want from our product, and verification considers how well it has achieved the specifications requirements provided to us.

Validation:

Our budget is limited, both in terms of time and money, so there was no real way to present our product to real tester and users to provide feedback to us. However, we will consider user’s real needs through thinking through their eyes and see what they may want from a software perspective. This is not a perfect solution, but will still give some insight into what we may want to consider.

Users/student customers:

* Fast delivery time – may depend on the restaurant, but will also vary based on the drone flightpath
* Ease of use – how easy the interface is to use, will depend on the application graphical developers
* Privacy – if their private information is secure
* Wide range of restaurants to select from – Will depend on the availability of restaurants around the area, and how many are close enough to participate.
* Good range of pizza choice – little restrictions on the number of pizzas and type of pizzas available to purchase from.
* Cheap service – ideally no service fees on top of delivery charges.

Restaurants – pizza suppliers:

* Also fast delivery time/optimal flight paths – allows for hotter pizza to be provided, gives the restaurant a better reputation and more sales.
* Smaller service fees charged – less charges needed to be paid for the drone service.
* Fast response time from drone calculations – allows restaurant to now when and if to start preparing food.
* Good order validation – no wrong orders or invalid payments, that may cause wasted food.

Edinburgh residents:

* Quiet drones – ideally no sound of operation and barely noticeable
* No encroaching on their privacy – avoid windows
* Safe - No flying overhead as much as possible

Verifications/Functionality:

Here are the list of functional requirements that need verification and are stated in the specifications found in the ILP coursework instructions:

1. No flying over the no fly zones in the central area campus – This is for safety reasons, as to prevent any inevitable failures of the drone to crash into unsuspecting students around the area. The no fly zones provided to us are all popular areas that students may populate and should be avoided. This is specific where the drone is not even allowed to consider having a move where the move doesn’t end up in the no fly zones but it does cut off a corner of it.
2. Only one drone is expected to carry out the delivery of orders of any given day – the flightpath should be calculated for the singular drone and that should be as optimal as possible.
3. No more than one order is being delivered at a time, where an order does not exceed 4 items – this is also for safety reason and also serves as a way to minimise mistakes on giving the wrong items for customers. Could have major consequences if someone with allergies received something slightly wrong.
4. Drone has limited flight time before needing recharge in a day – the drone only has 2000 moves in a day, where each move is either 0.00015 degrees straight line in one of the 16 major compass directions (0 degrees for east and 270 degrees for south) thought of as longitude and latitude on a flat plane, or a hover move. This will simplify the model of the drone simulation for our test product.
5. Drone is set to be at a location if its within 0.00015 degrees of it – this is a tolerance that will be made to allow the simulation of the drone reaching the desired destination.
6. Drone must hover for one move when reaching a destination and must return to Appleton tower before its battery/2000 moves runs out. – the hover allows the loading and delivering process of the service, the drone is made sure that the flightpath includes coming back safe to Appleton to avoid crashing at an unknown area for safety and reliability reasons.
7. Once the drone enters the defined central area for the university campus, it cannot leave until it has delivered the pizzas back to Appleton tower – this is to ensure efficiency and performance and restricted the drone to take long paths to go around the central area.
8. Every order has a fixed delivery charge of 1 pound – this ensures that the service we are providing is profitable, and it is easier to track the total cost of orders.
9. Check for valid pizzas and all pizzas must be from the same restaurant for a single order – need to check for if the pizza name is on any of the pizza supplier menus and if all the pizzas fall under the same menu.
10. Credit card numbers and CVV along with their expiry date is checked for validity – no attempt at transactions or deliveries will be made until these checks are done. This is for system correctness and allows the restaurant to not worry if their food orders are wrong.
11. All orders fall within the period of 2023-01-01 to 2023-05-31 – checks need to be in place to ensure that no test dates are wrong when trying to process the orders for a given date for a reliable system.
12. The total cost given by the customer is correct – need to check that the customer is not over or under paying the correct amount to the restaurant or us, and this would fail the fairness of the system and also the functionality of the service.
13. Any invalid orders detected should notify the user of the issue – this allows the user to correct the problem and try again, while not halting the program, therefore improving the reliability.
14. The valid deliveries will have to be marked as either delivered or not delivered – Some orders will not be able to be fit into the flightpath of the drone delivery despite being correct and valid for delivery. The customers may then be notified of this.
15. The runtime to produce the necessary flightpath files should have runtime of 60 seconds or less – this falls under the efficiency attribute. We would like to notify the restaurants as quickly as possible of our information so there is no delays and a smooth operation.
16. A über JAR file should be the operating point of the service and so should be generated – This is a test product and so the service is currently command line terminal operational for now, this is a functional requirement to allow the user to run the program.
17. Three output files will be generated in the resultfiles folder, under the root of the project. Two json files containing flightpath and order information, and one geojson file containing the drone path with files name created using the YYYY-MM-DD format – This is a functional requirement that gives the result of the calculations. The file date format is for easy readability and correctness.
18. Viability is judged by the average number of pizzas able to be delivered on any random day – this falls under performance and fairness, one may be considered more important than the other to satisfy the stakeholder’s needs.

Extra information:

The whole product will not just include the flightpath calculations and there is another team working with us that is focused on the web front side of things. We will not be considering the user validation for their side, such as the graphics or ease of use of the website or application.

Currently, our half of the product requires the use of a java virtual machine and is operated using the command line using a produced jar file while taking in 2 arguments (date, base url) to run the program.

Comparing the product specifications to the possible user validation requirements, we are able to determine that most of the stakeholder’s qualitative requirements. Omitting the requirements that were set for the web developers, a few things that were missing from the specification may be:

* Prevention of flying at window height.
* Limited to the number of pizzas that students can order at a time.