# Task 6.2

Isaac Pittolo – 7695438
Hoang Linh Bui – 100017631
Adam Bubonya - 9986677

## Our Definition of 'Done'

### Density - KLOC - How big is it?

Aim for 5% defect density. For our experience levels this is definitely achievable while not being a figure we can underestimate – I do not see us falling behind here.

We should hit around the average of 1 mistake per 10 line of code, however we will be aiming for lower than this as we feel it is definitely achievable with some hard work.

### Testing

Each function or method in our application should be very easy to test. If we exceed 30 minutes for any single given function we will need to make the decision to go back and redevelop that section until it becomes much easier to manage.

### Functional Appropriateness and Completeness

The product at the end of the sprint must be able to facilitate all the functions discussed in this sprint-planning period.

We will be aiming for a completeness rating of around 95%

Why are we aiming for a goal of 95%?

We are not experienced programmers. We are prone to mistake as we are still learning. Therefore, a 5% margin of error has been added, and is definitely necessary to facilitate our smooth progression of the project. If we aim for 100%, I feel we will be slowed down by unnecessary complications.

### Functional Correctness

100% precision is required with this application. Due to the nature of it being a sales management software for a Pharmacy there must be no inaccuracies.

### Response Time

Our response time for this project is very hard to measure currently as we do not yet have a suitable demo available for testing. However if it takes longer than 1 second to read < 500 records we may have an issue and will need to go back and change how we store our information as well as read from it.

## Resource Utilization

### RAM

500mb RAM usage maximum for our application. The reasoning behind this is that the program may be running on a lot of low end PCs or Thin Clients (such as remote terminal sessions). Therefore the hardware impact must be as low as possible.

### CPU Utilization

Depends on the CPU as it is difficult to properly quantify due to the vast differences between CPU generations, but as an example on a more recent machine which has a slightly above average CPU, (Intel Core i7-4510U @ 2.00Ghz) we should expect this application to utilize 1-2% of the CPU when the application is merely running.

This may vary depending on the amount of data being read/written. As long as the application is not consuming a majority of system resources, such as around 80 – 100% we will be satisfied.

## Capacity

In terms of number of records, for now the inventory will contain about a hundred records, but for sales it could handle thousands a month, at an estimate 10,000 records.

## Demonstration of the Product

As the stakeholders/product owner may or may not have experience with programming, we also have to finish at least a console application that allows demonstration of the functions of the software without needing to show the actual code to the end user.

The end user does not care about what we are about to achieve, they need to see something happen in the application, something tangible. We cannot consider our project done

We decided to not aim for learnability, understand ability or attractiveness in this sprint as the purpose of this sprint was to get core functionality up and running.