A-lEVEL COMPUTER SCIENCE PROJECT

Inventory management system

OCR A-Level Computer Science

Townley Grammar School

Arun Patel

Table of Contents

[Analysis 2](#_Toc465842955)

[1 Problem identification 2](#_Toc465842956)

[1.1 Description of the organisation 2](#_Toc465842957)

[1.2 Description of the problem 2](#_Toc465842958)

[2 Stakeholders 3](#_Toc465842959)

[2.1 Stakeholders who will have an interest in the solution 3](#_Toc465842960)

[3 Research the problem 3](#_Toc465842961)

[3.1 Analysing the problem 3](#_Toc465842962)

[3.2 Existing solutions 5](#_Toc465842963)

[4 Data collection 6](#_Toc465842964)

[4.1 Questionnaire 7](#_Toc465842965)

[4.2 Interview 10](#_Toc465842966)

[5 Diagrams 11](#_Toc465842967)

[5.1 The current system 11](#_Toc465842968)

[6 The proposed solution 12](#_Toc465842969)

[6.1 Solution requirements 12](#_Toc465842970)

[7 Signatures 14](#_Toc465842971)

[References 15](#_Toc465842972)

REMOVE AFTER COMPLETION

|  |  |
| --- | --- |
| Coursework Log |  |
| Issues |  |
|  |  |

# Analysis

## 1 Problem identification

### 1.1 Description of the organisation

Best-One New Cross, is a convenience store selling everyday items such as food and drink to locals in the area. Having been established for more than 35 years, it has a well-established and loyal customer base all with different needs. When the shop was first opened, it featured a very small range of products, suited for a small audience with similar needs, however, as more and more customers, who had different needs, started to visit the shop, it needed to expand its product range to satisfy customers.

Best-One is a chain of convenience stores based in the UK and Jersey. The company has over 600 stores throughout the UK, with most of its stock being sourced by Best way cash and carry. Best-One New-cross was opened and ran by the current owner’s father. Once the current owner, Purnesh Patel, was of a suitable age and had enough experience, the shop was then passed on to him. As the shop grew and grew, the decision was made to buy the building next to it, allowing for more space for expansion

### 1.2 Description of the problem

When new stock is brought into the shop, a member of staff writes down the name of the item onto a piece of paper, which is stored in a filing cabinet. When stock is taken out of the storage room and put on display, the member of staff must find the item on the list and cross it out. This can be immensely difficult as there is often hundreds of items on the list all with similar names. It has also led to errors in the past when staff members cross out the wrong item. It is almost near impossible to work out who made the mistake since there is currently no way of tracking who made the changes. This can lead to blame being passed around to the wrong employees which can severely de-motivate the staff member. The prices of some items can change often, when this happens it can cause confusion as the prices of items stored in the system don’t match others.

These problems can be solved by making use of computational methods. For example, abstraction can be used during the design of the interface, by not including unnecessary information. Instead of asking the user where the item came from, it can be suitable to ask the user the name of the item and the quantity of that item.

Furthermore, decomposition can be used to solve problem of working out who made an error when managing stock. For example, by decomposing the problem, I can understand that everyone who alters the inventory has a name. They will also do this at a specific time. From this, I can come up with a solution to this problem. By implementing a security system, the solution can detect who is making changes to the inventory. In addition, the solution can also log the time of day that the staff member altered the inventory. By doing this, I can record who has made changes to the inventory and at what time.

As searching for items is a huge issue for the organisation and often leads to errors. Queries can be implemented into the application that allows the program to search the database holding all the items for the desired record. Methods of computational thinking such as thinking ahead can be used here in the form of caching. Frequently searched for items can be cached, so that the user can quickly go to the items details without having to waste time. Indexing can also be implemented into data stored in the system, this allows the system to search for data quicker, making queries on large pieces of data take a shorter amount of time. Indexing involves putting markers into sections of the list of data to indicate the data underneath are all of a same category.

Automation can be used when determining which food products have gone out of date. Rather than the user have to analyse a list of products and find which products have gone out of data , which can lead to errors such as missing an out of date item, the solution could feature an automated alert to the user , making them aware of products that have gone past their expiry date.

Data collection allows me to take in information from the client. This could be in the form of questionnaires and interviews. I could question my client on what the solution to their problem should include and be able to do, ensuring the final solution is correct. This can be used together with data analysis, to make sense of the data I have collected, this could be achieved by making use of graphs and charts.

## 2 Stakeholders

### 2.1 Stakeholders who will have an interest in the solution

The main stakeholder of this solution is my client, Purnesh Patel. This is the person who has asked to create a solution to a problem that they have identified. This person will make the most use solution, therefor the solution must be suitable for this person. Since this stakeholder has limited experience in computer systems and using software, it is essential that the software is as user friendly as possible, which means that I can make use of a Graphical User Interface (GUI). A GUI has features such as windows, icons, menus and pointers, which can make software much easier for the user to use, especially since they have limited experience. The solution will also require the user to input a lot of data, therefor it is essential for the system to make use of validation. Validation is an automatic computer check to ensure that the data entered is sensible and reasonable. It does not check the accuracy of data.

Another stakeholder of the solution, is the members of staff at the shop. They could potentially make use of the system as much as the main client. This stakeholder will act as the audience of the solution, the audience is the person who will look at the finished product. Since some members of staff have a lot of experience in using Computer systems and others have little experience, I will have to design the solution in such a way that can be suitable for both kinds of audiences. For example, creating an interface with a lot of help dialogs and tooltips can make it very frustrating for skilled users to use the solution, whilst removing all help features from the solution can make it very hard and even impossible for inexperienced users to use the solution.

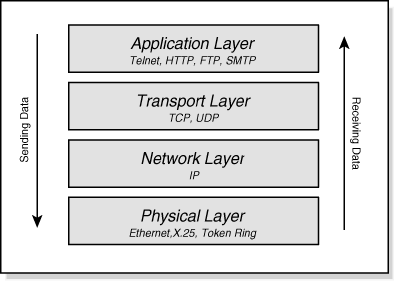
## 3 Research the problem

### 3.1 Analysing the problem

The problem which I am creating a solution to must satisfy the end user of the system. For example my client wanted the solution to be useable on different computers, where the data accessed and saved on one computer is available on another. To do this, I could make use of networking, saving the data in a small server. I could also achieve this by making use of real-time databases.

Google’s Firebase is an example of a service that offers real time databases. It is made for web based solutions but has third party wrappers for desktop programming languages such as C#. Real time databases allows each instance of the solution to be able to create, read, update and delete records in real time, so changes made by one instance of the solution will be seen by another. Initially Firebase can be used at no additional cost, however, in order to expand storage capacity and speeds you must pay a fee in order to upgrade to a higher package. This would be necessary if the solution is going to experience high traffic loads. This is unlikely since the solution is only going to be used by the client and staff members, which would not cause any delays due to traffic. If the store was to expand, offering more and more products, the Firebase package may have to be upgraded in order to expand the capacity of the database.

In order to access all the data of the system, the user must be using a computer connected to the internet, this creates a software and hardware requirement – access to the internet. This also increases the cost of the overall solution as fees would needed to be paid in order to keep the server running without stopping. To implement this feature I can make use of a computational technique, layering. “Layering is the organization of programming into separate functional components that interact in some sequential and hierarchical way, with each layer usually having an interface only to the layer above it and the layer below it”. In order to achieve layering, computational thinking methods such as decomposing the problem and abstraction must be used. Hardware such as a router are needed in order to access the internet. Furthermore, a subscription with an Internet Service Provider is needed, this can be very costly. If the internet has slow upload and download speeds the user could experience lag whilst using the solution.



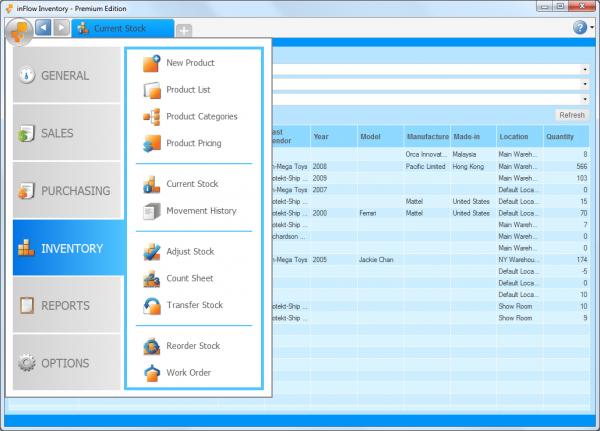
(Fletcher, 2016)

### 3.2 Existing solutions Acctivate Inventory Management - Product specifications

(Anonymous, 2016)

Acctivate inventory management software is a potential solution to the proposed problem. It is an off the shelf product and does not offer bespoke features, tailored to the client. Based on the features Acctivate has to offer, I could include some of fields shown above when storing the information of a product. For example, I could also allow the user to assign a product to a category. The advantage of doing this, is faster searching time, as indexing can be used. Rather than the solution searching the entire list of products it can search for products in the chosen category, therefore reducing the searching time. Also, I can see that Acctivate gives each product an ID, this could be a feature of the final solution, since all the products will be stored in a database I can make use of IDs to give each product a unique identifier, making duplicates unique from eachother.

On the other hand, the product shown above has a cluttered user interface, which can come across as overwhelming for inexperienced users. For example it features many tabs near the bottom of the page, this can come across as overwhelming. It also allows the user to classify the product into different categories such as item type, product type and product class. This allows data to be stored in a more organised fashion allowing for quicker searching times but can be very time consuming for the user.

(Anonymous, 2016)

InFlow inventory is another potential solution to the problem. This program has a more de-cluttered interface, making it more suitable for inexperienced users. Features such as tabs, tables and menus are all features that I could include in my solution. The use of large menus on the left hand side could be beneficial to inexperienced as the icons and text are large and easy to view.

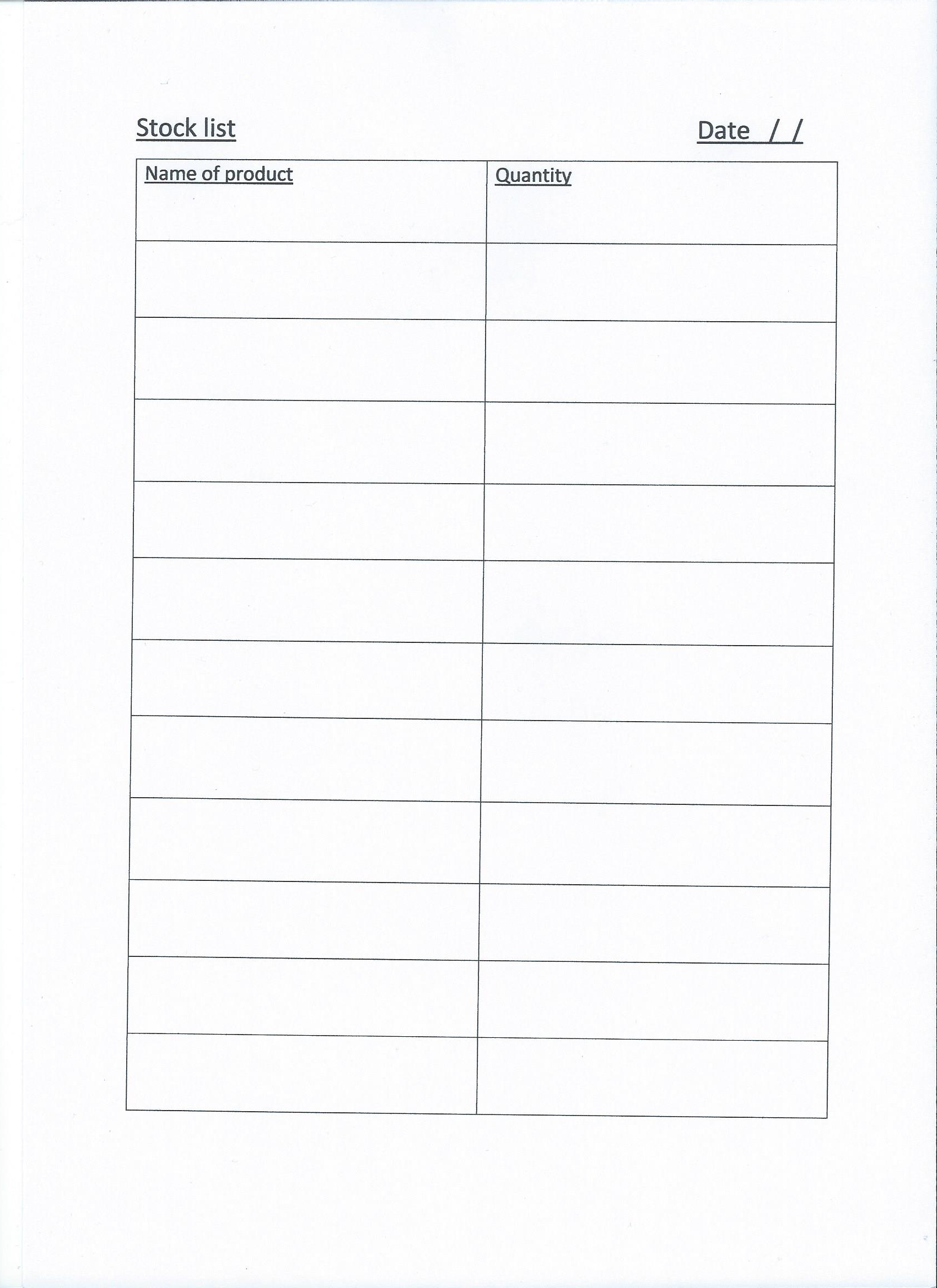
Rather than presenting all of the data entry fields on one interface like Acctivate inventory does, InFlow makes use of large icons and text, shown on the left hand side of the image. When the user clicks on one of these icons, more options appear. This is known as a menu driven interface, which is capable of displaying many options to the user whilst keeping the main user interface tidy, this can lead to the user feeling less amounts of stress due to being overwhelmed.

In the top right hand side corner, the interface contains a question mark. If the user clicked this button when using Inflow Inventory, they would be taken to a help dialogue, where they can find information on how to solve a problem they might have. This could be a very useful feature to include in the solution, since the users of my solution are inexperienced, they are likely to need additional help to use the solution, and this could be one way in which the solution could provide this.

### 3.3 Forms of data

#### 3.3.1 Paper based

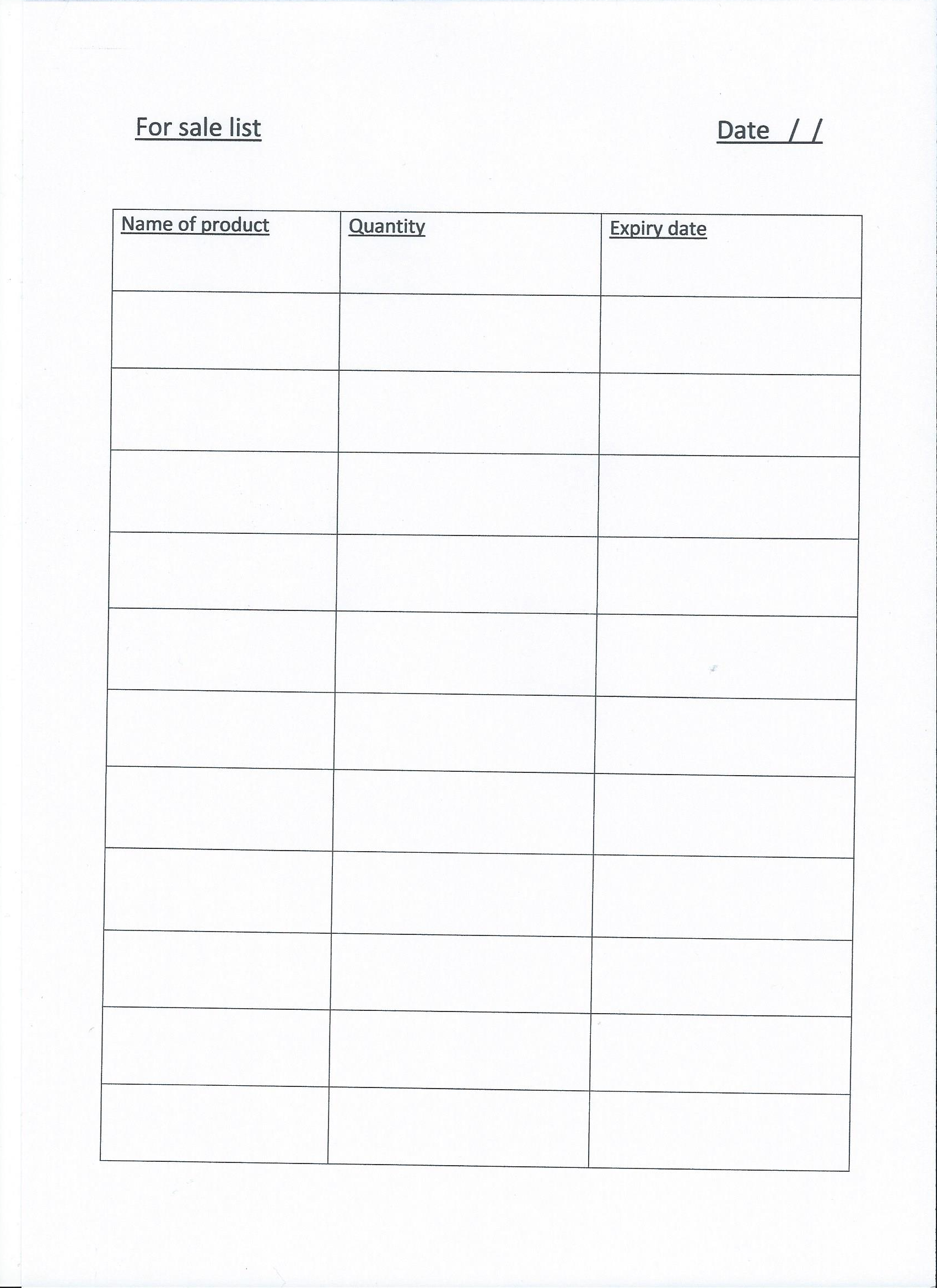
Currently, there are two paper documents in use. One of the documents, called the stock list, stores a list of the products currently stored in the shop, that isn’t available for sale. It also stores the quantity of that product in the stock room. Furthermore, the data the list was created is also stored at the top of documents.



*Fig 3.3.1.A*

Shown above (fig 3.3.1.A) is a scan of an empty stock list sheet.

In addition to the stock list paper document, the shop also makes use of another paper document, called the for sale list. This document stores the list of products being taken out of storage and on to ‘the shelf’, making it available for purchase. The list stores information such as the name of the product, the quantity of that product and the expiry date of that product, should it have one. It also stores the date the list was created.



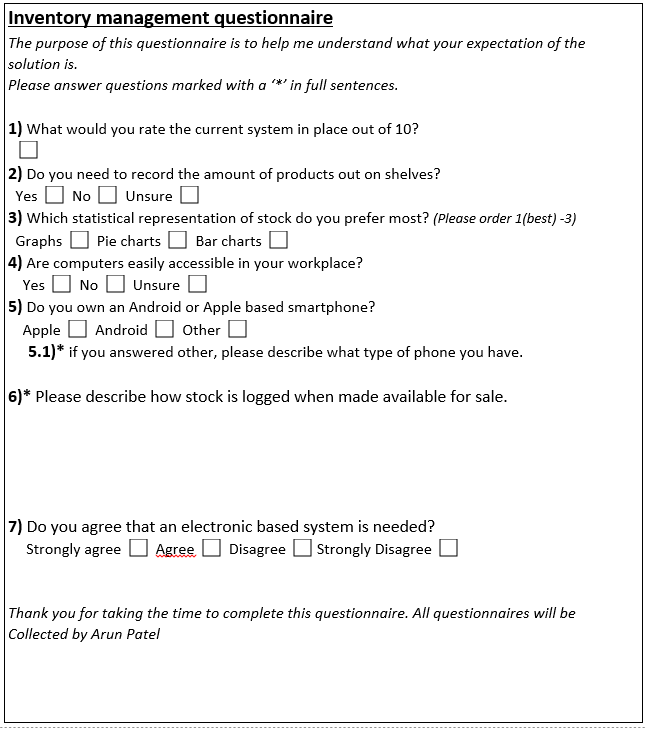
*Fig 3.3.1.B*

Shown above (fig 3.3.1.B) is a scan of an empty for sale list sheet.

## 4 Data collection

At this stage, I am going to get as much information from the client as I can regarding what the client wants from the solution. This will ensure that I produce a solution that does exactly what the client wants it to do. This will also allow for strong communication between myself and the client, meaning less time will be wasted in the future and the chances for mistakes to be made are reduced.

### 4.1 Questionnaire

The following questionnaire was given to my client to complete.

My client’s answers were as follows:

**1)** What would you rate the current system out of 10?

Answer: 4

**2)** Do you need to record the amount of products out on shelves?

Answer: Yes

**3)** Which statistical representation of stock do you prefer most?

Answer: Graphs (1), Bar charts (2) and Pie charts (3)

**4)** Are computers easily accessible in your workplace?

Answer: No

**5)** Do you own an Android or Apple based smartphone?

Answer: Android

**6)\*** Please describe how stock is logged when made available for sale.

Answer: When stock is put out on the shelf, it is initially taken off of the stock list. It is then

Added to a for sale list. If the product is an item that has an expiry date , that date is then recorded on to the list as well , so we know when stop selling a certain product. We also record the amount of that certain product taken out, for sales analysis purposes.

**7)** Do you agree that an electronic based system is needed?

Answer: Agree.

From the answers my client has given me for this questionnaire I can deduce that my solution should present statistics in the form of graphs and bar charts. I also understand that my client wants the new solution to be created.

I have also identified that my client does not have easy access to computers, this means I will have to find a way of allowing my client to be in use with the system whilst not a computer. One possible solution to this is to make use of a smartphone. My client has an android based smartphone, this means I could build an app to go alongside the main computer based system. This will make it easier to record the number of products out on the shelves.

I have also gained an understanding for how the stock is logged when made available for sale, this is an essential part of the system that would have to be accurately modelled in the final solution.

The questionnaire was also given to 3 other staff members at the business. The results of those questionnaires have been summarised below (excluding question 6). I processed my results using the charts feature in Microsoft Office Word (Microsoft, n.d.)

Question 1 -

Question 2 – All staff members ticked the yes box

Question 3 –

Question 4 –

Question 5 –

Question 7 –

### 4.2 Interview

#### 4.2.1 Face to face interview with client

I had arranged to have a face to face interview with my client at the workplace. I arranged for this interview by emailing my client and asking them if it was ok for me to conduct the interview and at what time and day would suit them best. The aim of the interview was to gain as much information as possible, mainly about the client themselves, such as their computer preferences, workload and computer skills. I also wanted to see, in person, exactly how some of the processes such as receiving new stock are actually conducted.

After agreeing on a time and date for the interview, I made my way to the business and conducted the interview. Initially, my client showed me the current system and how it works, he showed me the process of how new products are delivered to the store, added to the logs and stored in the stock room. After, he showed me the process of taking out stock from the stock room and making it available for sale.

The main issue with these processed seemed to be the amount of time it took to complete the tasks. The client seemed to get frustrated when trying to find items on the stock list as there was at least 100 items on the list.

After showing me a demonstration of the current processes the client undertakes, he then went on to give some ideas that could be added to the system. He suggested that the main system be able to search for products stored in the system. He also suggested that the new system maintain two separate lists, for products in the stock room and products for sale.

The client also raised an issue about the new solution, the fact that it wouldn’t be practical to use whilst performing everyday tasks. He then went on to explain that with the current system, he is able to walk around with the pen and list and make changes anywhere, whilst with a computerised solution, he would have to be at a computer which he cannot easily carry out with him.

## 5 Diagrams

### 5.1 The current system

Here is a summary of the processes undertaken in the current system in the form of an Activity diagram.

#### 5.1.1 Receiving new stock

Y:\downloads\Adding stock flow.png

#### 5.1.2 Making stock available for sale

y:\downloads\MakeForSaleDiagram.png

## 6 The proposed solution

### 6.1 Solution requirements

#### 6.1.1 Hardware requirements

* A computer with an internet connection
* This will allow the user to make use of the solution I have created.
* A smartphone
* Since there is not easy access to computers, a smartphone can be used instead when not at a computer to complete simpler tasks.
* A Keyboard
* This is a basic input device used to input characters into the computer. It is essential for both the users of the system and myself, to use the computer.
* A Mouse
* This is another basic input device used to allow the user to move a mouse pointer around the screen. This is necessary for the basic operation of the computer.

##### Required computer specification

The minimum hardware requirements that a computer needs to run this software is the minimum hardware requirements to be able to run Microsoft’s Windows 10 64-bit (Microsoft, 2015), however, for an optical experience, the user should have a computer whose specification are in line with the recommended hardware specification.

Minimum hardware specification

Processor – any processor with a clock speed of 1GHz or more

RAM – at least 2 GB

Hard disk space – at least 21GB

Graphics – Direct X 9 or later graphics card.

Display – a minimum of 800x600

Recommended hardware specification

Processor – any processor with a clock speed of 2GHz or more

RAM – at least 2 GB

Hard disk space – 30GB or more

Graphics – Direct X 9 or later graphics card.

Display – 1920x1080 or higher.

#### 6.1.2 Software requirements

* A desktop operating system – Windows 10/8/7 OS
* The solution will be created using the C# programming language and will therefor only run on a Microsoft Windows Operating System. This means that the users of the solution must have access to either a Windows 7 , 8 or 10
* A mobile operating system – Android based
* The results of the questionnaire revealed that the client has an android based phone, therefore it a software requirement to have an android based phone.
* An Integrated Development Environment (IDE) - Visual Studio / Eclipse
* In order to create the solution efficiently and effectively, I will need a programming IDE to help me write code.
* Eclipse will be used to create java based applications, the language used to create android apps.
* Photo editing package – Adobe Photoshop
* The solution may require graphics to be created, or to edit existing graphics to suit the solution.
* Microsoft Office suite
* The office suite is used to create documents, slide shows and spreadsheets, which all necessary for the development of this project.

#### 6.1.2 Solution requirements / success criteria

##### Functional requirements

* The solution must be able to input new stock items.
* This is essential for entering new items into the system when a delivery arrives.
* The solution be able to take out stock items
* This allows the user to make stock available for purchase.
* The solution must be accessible from different computers
* This allows the user to access the database from different computers. This is needed as the client will be using different devices to make use of the solution.
* The solution should be useable by inexperienced users
* The client is not very experienced with computers therefore the solution must be user friendly enough for inexperienced users.
* The solution must be mobile
* The user isn’t going to be able to easily access a computer, therefore the solution must be useable whilst “on the move”.
* It must have a facility to change the prices of products
* The prices of products often change, therefore the system must be able to edit the price of a certain product.
* It must be able to find popular products
* In order to continue selling products that are generating the most profit, the system must be able to identify these products and tell the user.
* It must be able to find unpopular products
* Products that are not selling well should be made to the attention of the user so that they can stop buying more of it.
* Data stored in the solution must be valid
* This will ensure that the data stored in the system is free of errors , reducing the chance of an output being incorrect as a result of incorrect processing
* Data stored in the solution must be as accurate as possible
* This will prevent in-accurate data from being entered into the system, making it less likely for invalid calculations from being made.

##### Non-functional requirements

* The solution must be able to load in under 20 seconds
* This will ensure the user is not left waiting for the program to load when they initially open the program. This is dependent on the size of the database (the number of stock items stored), speed of the computer and speed of the internet connection.
* The solution must have a security login system
* This will ensure that unauthorised access does not occur. Since the solution will store data about the business, the solution must be secure enough to prevent anyone from accessing this data. Providing a login system will also allow the system to track who has been using the solution.

## 7 Signatures

Confirmation of hardware requirements Date of signature-

Confirmation of functional requirements Date of signature-

Confirmation of non-functional requirements Date of signature-

# Design

# References

Anonymous, 2016. *Acctivate Inventory Management Software.* [Online]   
Available at: http://www.softwareadvice.com/uk/inventory-management/acctivate-profile/  
[Accessed 21 September 2016].

Anonymous, 2016. *inFlow Inventory Software.* [Online]   
Available at: http://www.softwareadvice.com/uk/inventory-management/inflow-inventory-profile/  
[Accessed 21 September 2016].

Fletcher, M., 2016. *Introduction to Network Programming.* [Online]   
Available at: http://www.webbasedprogramming.com/Java-Unleashed-Second-Edition/f23-1.gif  
[Accessed 21 September 2016].

Microsoft, 2015. *Windows 10.* [Online]   
Available at: https://www.microsoft.com/en-gb/windows/windows-10-specifications  
[Accessed 2 November 2016].

Microsoft, n.d. *Office Word,* s.l.: s.n.