Software requirements specification

Rachel Lowe

1301408

Contents

[Contents 1](#_Toc478130035)

[1. Introduction 4](#_Toc478130036)

[1.1 Purpose 4](#_Toc478130037)

[1.2 Scope 4](#_Toc478130038)

[1.3 Terms, acronyms and definitions 5](#_Toc478130039)

[1.4 References 6](#_Toc478130040)

[1.5 Overview 8](#_Toc478130042)

[2. Overall description 8](#_Toc478130043)

[2.1 Project prospective 8](#_Toc478130044)

[2.1.1 System interfaces 9](#_Toc478130045)

[2.1.2 Users interfaces 10](#_Toc478130046)

[2.1.3Hardware interfaces 12](#_Toc478130047)

[2.1.4Software interfaces 13](#_Toc478130048)

[2.2.5 Communication interfaces 13](#_Toc478130049)

[2.2.8 Site Adaption Requirements 14](#_Toc478130050)

[2.2 Product functions 14](#_Toc478130051)

[2.3 User characteristics 28](#_Toc478130052)

[2.4 Constraints 29](#_Toc478130053)

[2.5 Assumptions and Dependences 29](#_Toc478130054)

[3.6Apportioning of requirements 30](#_Toc478130055)

[3 Specific requirements 30](#_Toc478130056)

[3.1 External interfaces 30](#_Toc478130057)

[2.2System features 46](#_Toc478130061)

[3.2.1 Feature 1 46](#_Toc478130062)

[3.2.1 Feature 2 47](#_Toc478130063)

[3.3.3 Feature 3 49](#_Toc478130064)

[3.2.5 Feature 5 51](#_Toc478130065)

[3.3.6 Feature 6 52](#_Toc478130066)

[3.3.7 Feature 7 53](#_Toc478130067)

[3.3.8 Feature 8 54](#_Toc478130068)

[3.3.9 Feature 9 55](#_Toc478130069)

[3.3.10 Feature 10 56](#_Toc478130070)

[3.2 11 Feature 11 57](#_Toc478130071)

[3.2.12 Feature 12 58](#_Toc478130072)

[3.2.13 Feature 13 59](#_Toc478130073)

[3.2.14 Feature 14 60](#_Toc478130074)

[3.3 Performance requirements 61](#_Toc478130075)

[3.4 Design constraints 61](#_Toc478130076)

[3.5 Software system attributes 61](#_Toc478130077)

[3.5.1 Reliability. 61](#_Toc478130078)

[3.5.2 Availability 62](#_Toc478130079)

[3.5.3 Security 62](#_Toc478130080)

[Maintainability, 63](#_Toc478130081)

[Portability. 63](#_Toc478130082)

[Logical database requirements 64](#_Toc478130083)

[Standards and compliance. 65](#_Toc478130084)

[Appendix A BPNM. 68](#_Toc478130085)

[Appendix B confirmation and expected mark. 1](#_Toc478130086)

[Appendix C feature as described within the SRS. 2](#_Toc478130087)

[Table 1: acronyms 3](#_Toc476899652)

[Table 2: terms 4](#_Toc476899653)

[Table 3 login use case 35](#_Toc476899654)

[Table 4 create order use case 37](#_Toc476899655)

[Table 5 append order use case 39](#_Toc476899656)

[Table 6 veiw invoice use case 39](#_Toc476899657)

[Table 7 add new customer use case 40](#_Toc476899658)

[Table 8 add new user use case 41](#_Toc476899659)

[Table 9 prepare currenr orders use 42](#_Toc476899660)

[Figure 1: Login section 8](#_Toc476761148)

[Figure 2 use case diargrem 13](#_Toc476761149)

[Figure 3 class diagram 15](#_Toc476761150)

[Figure 4 login sequence diagram 17](#_Toc476761151)

[Figure 5 create order sequence diagram 19](#_Toc476761152)

[Figure 6 append order sqecnuee diaragrem 20](#_Toc476761153)

[Figure 7 view previous orders sequence diagram 21](#_Toc476761154)

[Figure 8 prepare orders sequence diagram 22](#_Toc476761155)

[Figure 9 create new customer sequence diagram 23](#_Toc476761156)

# Introduction

This documents covers the requirements of the online purchasing process of the vapor-tek company. This will explain in detail what the requirements of the system are and how they will be implemented.

## 1.1 Purpose

The purpose of this document is to describe the requirements of the purchasing process of the vapor-tek website [www.vapor-tek.com](http://www.vapor-tek.com/). The document will outline the specific requirements of what is needed and how it will be built. The audience of this document will be different departments of the Vapor-tek Company as well as the hosting company for Vapor-tek. Additionally, it is intended for the developers of the web application. Each section of this document will be for different audiences within the audience specified above.

## 1.2 Scope

As mentioned above the project will be the purchasing process of[www.vapor.tek.com](http://www.vapor.tek.com/). It will involve the administration side of this process. This web application will not involve the payment process for orders that are made through the site.

## 1.3 Terms, acronyms and definitions

This sections describes the terms and acronym used throughout this documents

Table 1: acronyms

|  |  |  |
| --- | --- | --- |
| Acronyms | What it stands for | Definition |
| HTTP | Hypertext protocol | Protocol for transferring data |
| HTTPS | Hypertext Protocol secure | Protocol for transferring data securely |
| PHP | Hypertext pre-processor | Programming language used to develop web applications |
| SSL | Secure sockets layer | Used for secure connections |
| DRY | Do not repeat yourself | A principle used in software engineering that says do not repent yourself. |
| HTML | Hypertext Mark-up Language | Used for creating the content |
| URL | Uniform resource locator | A unique address for a website |
| CoA | Certificate of analysis | Used to define that products are safe to use. |
| RAM | Radon access memory | Memory that an application uses while it is running |
| GB | Gigabyte | Unit of storage. A gigabyte (GB) is one thousand and twenty four megabytes(MB), which one thousand and megabytes, which is one thousand and twenty four kilobytes, which is one thousand and twenty four bytes, which is eight bits which is the smallest |
| CPU | Central processing the unit | The core of a computer, |

Table 2: Technological terms used within the discussion of the Vapor-Tek web app

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Function | Used to create spectate functionality of the system |
| Variable | Used to store a partier item in the programing of the system |
| User | Someone who uses the system. |
| Primary Key | An attribute within a database entity in order to uniquely identify a record. |
| Attribute | A characteristic of entry is a database |
| Entity | A table in a database |
| User interface | Screen for the user to interact with the system. |
| Input | Information entered into a system |
| Output | Information that is produced from the system. |

## 1.4 References

# Anon (2008) *Web content accessibility guidelines* [Online] Available from*:* https://www.w3.org/TR/WCAG20/ [*https://www.w3.org/TR/WCAG20/*](https://www.w3.org/TR/WCAG20/)*[accessed 20 February]*

Anon(nd) *Identity Manager 7.0.1 - Installation Guide* [online] Available from https://support.quest.com/technical-documents/identity-manager/7.0.1/installation-guide/4#TOPIC-487760 [accessed 18 march 2017]

Anon,(nd) URL *encoding reference* Online Avail be form [accessed 6/3/17]

Clark. J, van Oorschot P C.(2013) *SoK:* SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements  *2013 IEEE Symposium on Security and Privacy* [Online] pp 511-525 Available from http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6547130 http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6547130[accessed 16 February 2017]

Gorley, D., Totty, B., Sayer, M., Aggarwal, A., Reddy, S. (2002) *HTTP the definitive guide.* Sebastopol: O’Rellly & Associates*,*

Gov (nd) data protection act. (Online): Available from <https://www.gov.uk/data-protection/the-data-protection-act> http://www.legislation.gov.uk/ukpga/1998/29/contents [Accessed 21 February 2017]

IEEE Computer society. *IEEE Recommended Practices for Software Requirements Specification* [online] Available from <http://moodle2.bolton.ac.uk/pluginfile.php/422475/mod_resource/content/0/IEEE830%20SRS.pdf> [accessed 4 February 2012]

Maciaszek, L A. (2005) *Requirements Analysis and Systems design* second edition, England, Persons education,

Microsoft(nd)  *Methods and formulas used to determine server capabilities* [online] Available from: https://technet.microsoft.com/en-us/library/cc181325.aspx [accessed 18 march 2017]

Microsoft. (n.d) *Calculate Your Server Size* [online] Available from https://technet.microsoft.com/en-us/library/bb124226(v=exchg.65).aspx [accessed 18march 2017

Miles, R, HamiltonK, (2006) *learning UML 2.0*  , Sebastopol, O’Reilly Media

Pender T. (2001) *UML Bible* Canada. Willy publishing,

Plione D , Pitman, D.(2006) *learning UML 2.0 in a nutshell*. Sebastopol, O’Reilly Media

Stevens P, Pooley R. (2000) *Using UML Software engineering with objects and components*  updated Edition, England. Pearson Education,.

Zalewski, M. (2012) *The tangled web a guide to securing modem web applications* [online] San Francisco, no steach , available from: http://ebookcentral.proquest.com/lib/bolton/reader.action?docID=1137574 [accessed 6 march 2017]

## 1.5 Overview

The rest of this document is split into two main sections. The first section gives an overview of the application. This is split into 6 subsections, firstly it talks about the project prospective, then goes on to talk about the product functions. Next it discusses the user characteristics, then it discusses the assumptions and dependences followed by constraints. Finally, it discusses what can be done in future versions. The other main section discusses the specific requirements. Firstly, it gives an overview of the external interfaces. Next it details the features of the application. It then goes on to discuss the performance related requirements. Following on from this, the design constraints are discussed. After this it displays information related to the database. Next it details the software system attributes. Lastly, it discusses standards and compliances that the application has adhere to.

# Overall description

As stated this project is developing the purchasing process for the Vapour-tek website. Currently the process is not online. The project intends to make the process online though a web application. This will enable better management of orders enabling Vapour-tek to keep better track of orders.

## 2.1 Project prospective

This project is intended to functionalise the purchasing process of Vapour-tek. The current system is paper based, and this project will be a more effective process.

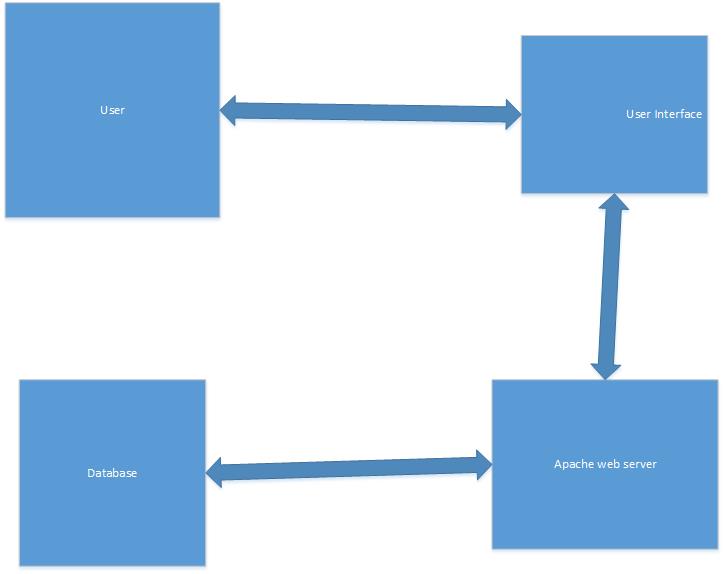


Figure : Interaction between system components.

The above diagram shows the major contents of the system and how they interact with each other. The user interacts with the user interface which interacts with the which interacts with an Apache web server which interacts with the database,

### 2.1.1 System interfaces

For the customer part of this web application there will be three main areas, first being a create order section, in this section, there will be 2 main window areas one which conveys an order form and one which allows the customer to view the invoice of the order. Additionally, there will be an area for viewing previous orders. This part of the application will have one main window area which shows the customer a list of previous orders. Furthermore, there will be a section that allows the customer to append an order, this will consist of two main window areas, one for viewing the currents orders and one for viewing an editing a specific order.

For the production manager section, there will be two main interfaces, One to view orders and another one to enter whether an order has been dispatched.

The accounts administrator section will have three main areas one for dealing with orders, within this area there will be 4 interfaces, one for creating orders, one for viewing orders, one for cancelling orders, and finally one for updating orders. Additionally, there will be a section for managing products. This will be one interface which shows information on the products. Furthermore, there will be a section for managing customers, this will consist of two interfaces one for viewing customer information and one for adding new customers.

### 2.1.2 Users interfaces

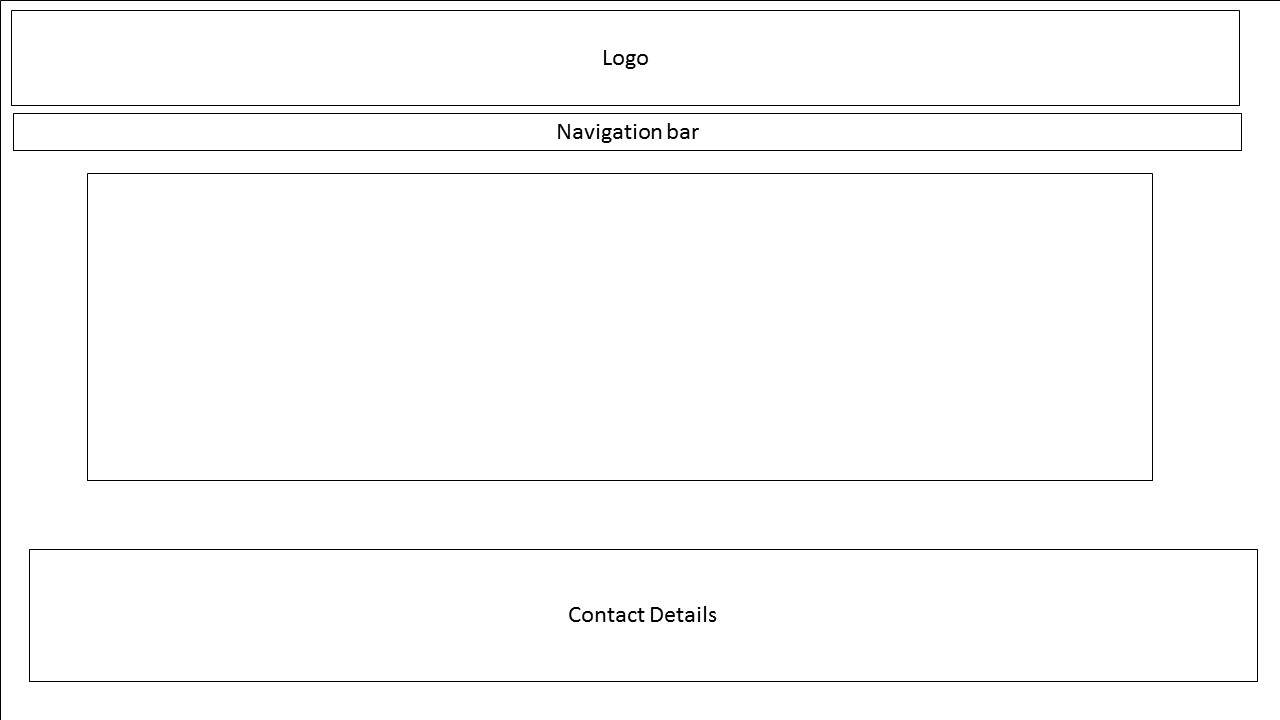


Figure 2 Whole sight template

The above diagram is a template for application as the frontend of this application has been developed and will be used in this project.

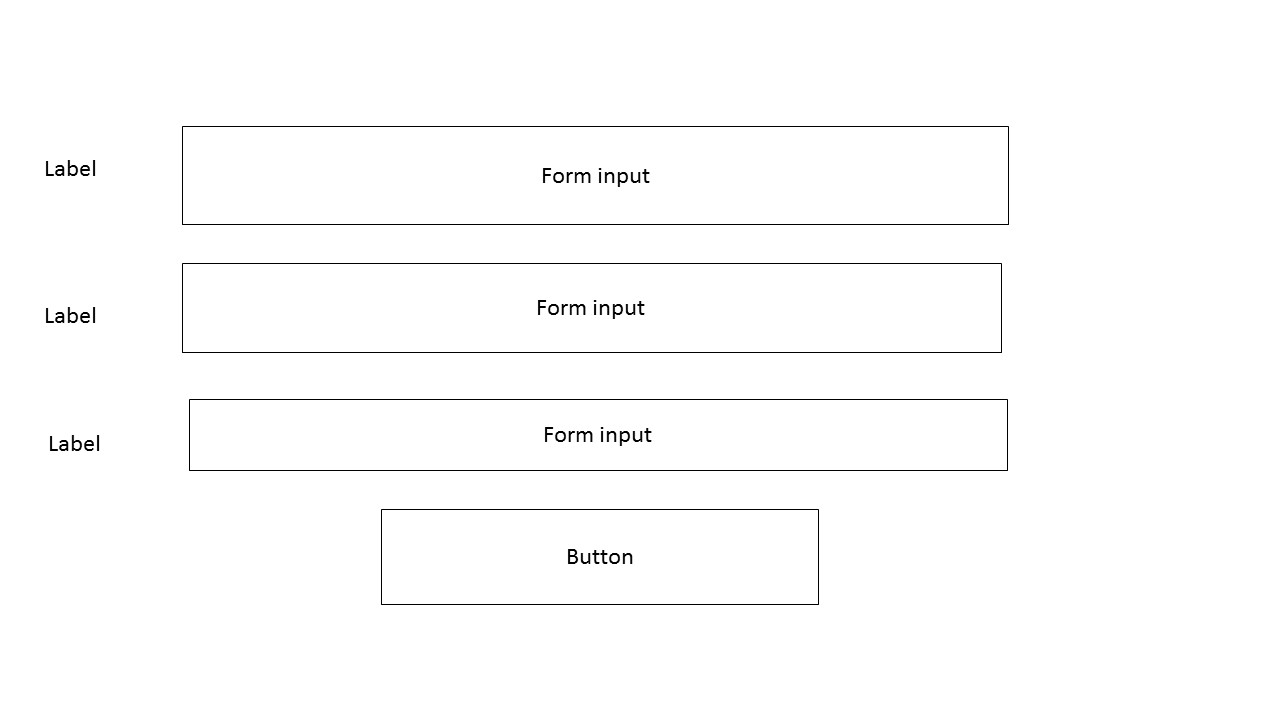


Figure Form template

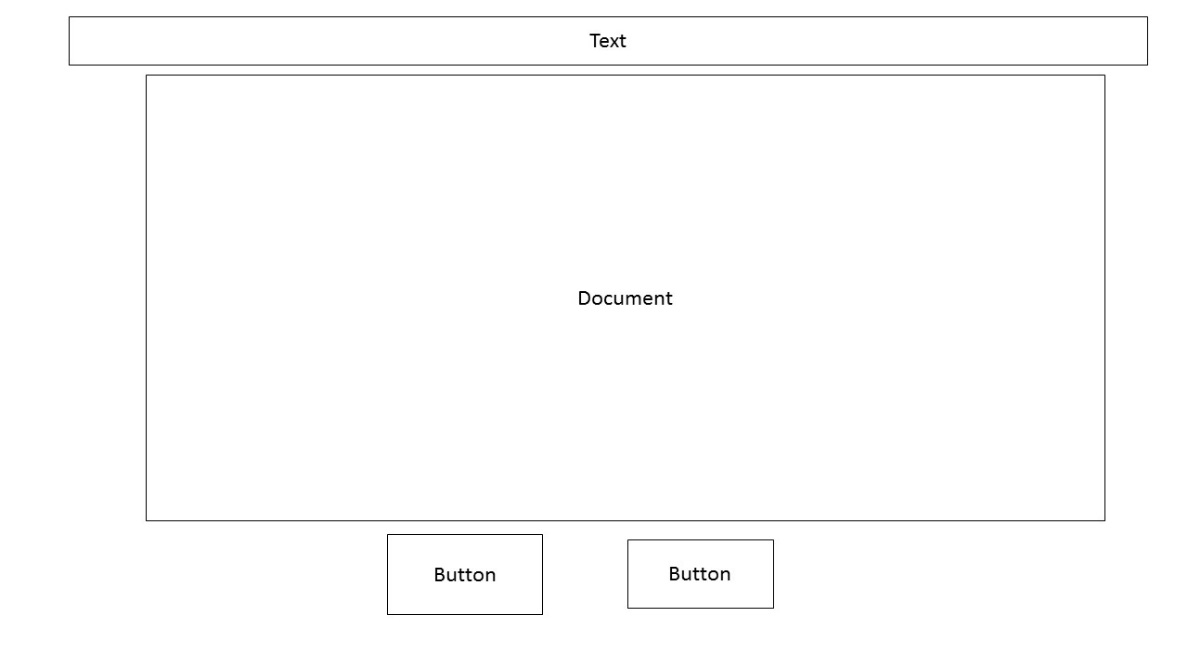
This application will consist of several forms the above diagram shows an overview of what these forms will look like. There will be various inputs for the user to enter on these pages  


Figure 4 template for viewing documents,

Figure 5

Within this application there will sections for viewing documents such as invoices. The above provides a template for these sections.

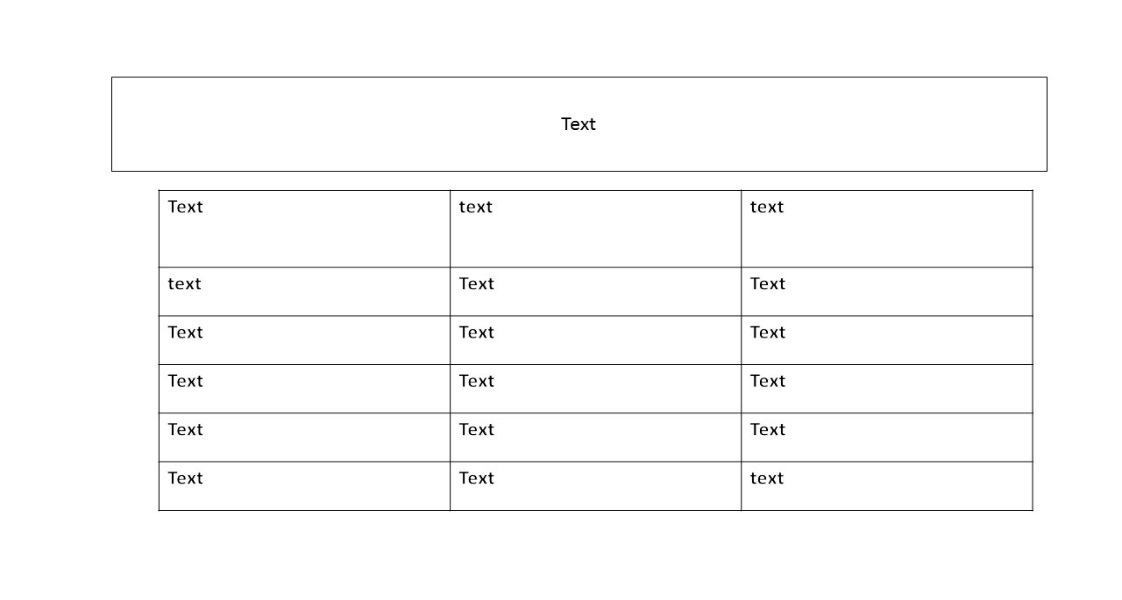


Figure 6 template for table sections

Within this application the above diagram within there will be some pages which display content in table such as previous orders of a particular user and currents orders for the production manager to view when they are preparing orders, the above diagram gives a template for this for this For father diagrams that show pages for each feature see section 3.1 external interfaces.

### 2.1.3Hardware interfaces

The database of this system will need to communicate with a server, this will be carried out though standard PHP/ database protocols such Ethernet/http. There are no definitive hardware interfaces however, the hardware requirements may vary depending on the load of the server.

In order to define the hardware requirements Microsoft have several suggestions based on different hardware usage, All of these are based on observation time(t) busy time(B) and Competitions(c) According Microsoft a way to calculate CPU utilization, which is the percentage of CPU usage, is busy time over observation time. This would relative to this application because in order the software to perform eff it need to be taken into account how much how much resource it will need to use. The expected traffic on the site would need to be confirmed in order to make the appropriate calculations.

The client must provide the hardware and must continue to be responsible for this in order to ensure smooth running o application.

### 2.1.4Software interfaces

The system will use a web server called Apache. The purpose of this is that in order for the web application to work it needs to be hosted on a web-server. The application will be connected with a MySQL database. The purpose of this is to enable the web application to retrieve and store data, for example the user’s login information. The application will be written in PHP 5 the purpose is provide the functionality for site.

### 2.2.5 Communication interfaces

The application will need access to HTTPS as this application will involve personal information such as usernames, passwords and customer information. This will require use of a security certificate, which can be done through OpenSSL which according to Gorley et al (2002) is the most commonly used open source implementation of SSL. It would be worth looking into newer releases of implementations of SSL to make sure that the security aspects of the certificate used the client are up to date and meet the latest browser requirements.

**2.1.6 Memory constraints**

In order to ensure the application runs smoothly it will need to take into the condensation memory requirements of the software, which in this application is PHP, Apache, MySQL and the Linux operating system The first memory requirement that needs to be addressed is the Linux operating system. It will depend on what version of Linux. It is assumed in order to run Linux effectively it would require between 1GB and 2GB. Additionally it is assumed that apache. Would require a minimum memory of 500mb, in order to run PHP Effectively it is assumed it was it would require a minimum of further 600mb would be required along. Therefore it is suggested that the system that this application runs on has roughly 3GB as minimum requirement.

**2.1.7 Operations**

There will be four main types of users who will use the application. These are Administrator, Accounts Administrator, Production Manager, and Customer.

The Administrator will have full access to all parts of the application, this includes access logs, product management and order management. Furthermore an Administrator will be able to access customer information.

The Account Administrator will have some access to the following; product management, customer management and order management. The Accounts Administrator will be able to create orders, amend orders and cancel orders of customer accounts. The Account Administrator will also be able to access product information in order to view which products customers have ordered. Additionally, the Accounts Administrator will be able to create new customers. The requirements

The Production Manager be able to view orders and enter whether or not they have been despatched.

The customer will be able, to create orders, append orders, view orders and view invoices.

### 2.2.8 Site Adaption Requirements

This site will be designed to ensure that screen readers can use it. It will be ensured that there is a way for changing the text size to enable users who are visually impaired to use the site. There will text alternatives for all non-text content. For more details on this, see section 3.7 standards and compliances.

## 2.2 Product functions

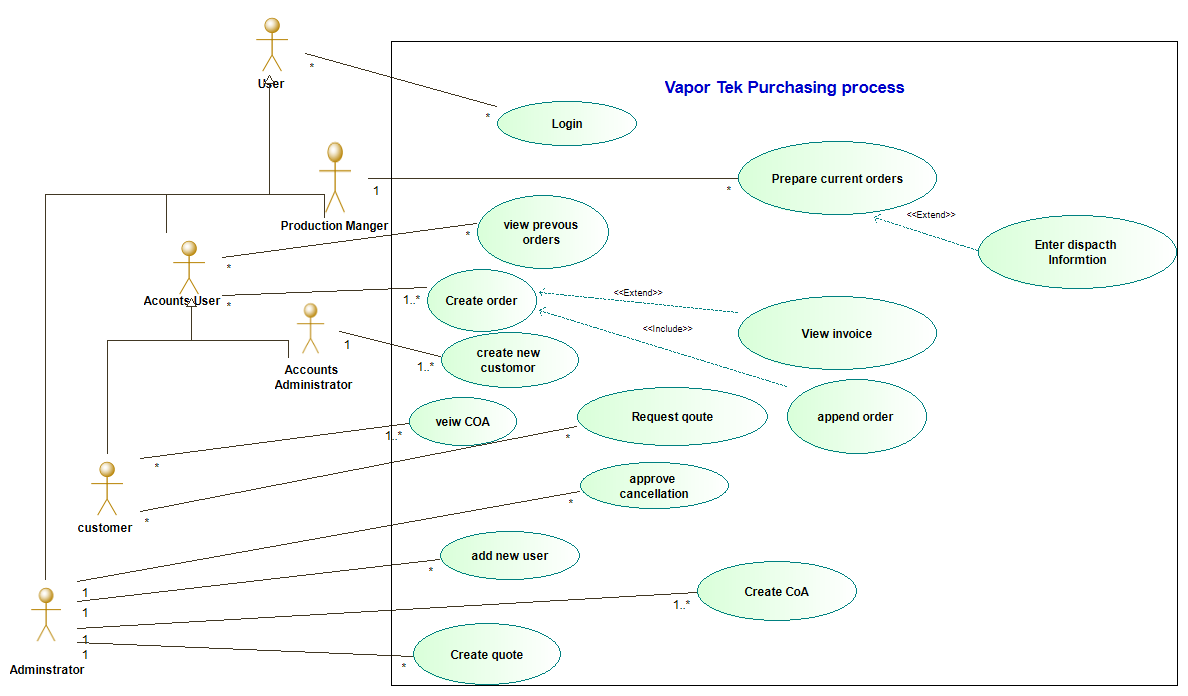


Figure 7 use case diagram

The above diagram gives an overview of the system. It shows that there is a user of the system who will use the login function of the web application. There are four types of user which use different functions of the web application. These four users are Administrator, Account Administrator, Customer and Project Manager. The four users are explained as follows

* Although the Administrator has full access to the site to oversee the process, the functions that they will be involved in that are shown in the diagram are ‘approve cancellation’ ‘add new user’, ‘create CoA. and ‘create Quote’ functions.
* The Accounts Administrator will; use the ‘create new customer’, ‘create orders’, ‘view invoice’ and ‘append order’ functions.
* The customer will have access to the ‘create order’, ‘view previous orders’, ‘append order’ ‘view invoice’ ‘request quote’ and View CoA’ functions.
* The Production Manager will use the ‘view current orders’ and ‘enter despatch information’ functions.

The Accounts Administrator and Customer are both represented in the diagram by a more general user of accounts user, this is because the accounts user and the customer use some of the same functions. However, they are users in their own right. Therefore, they inherit from the accounts user which inherits from user.

The view invoice function is an extension of the create order function, this is because in order to view an invoice the user has to create an order. The append order function will include functionality of the create order function; this means that it is a stand-alone function. The reason for this is because a user can return to the site at a later date and append their current orders. The enter despatch details function is an extension orders of the prepare orders function. This is because before the production manager can enter the despatch details of an order the orders need to be prepared first.

The relationship between the Accounts User and the create order function says that many accounts users can create many orders. The relationship between Accounts User and the view previous orders function says that many accounts Users can view many orders. The relationship between the administrator and the add new user function says that there is one administrator who can add many users. The relationship between the Production Manager and the view currents orders function says that there is one Production Manager that can view many orders. The relationship between the Accounts Administrator and the add new customer function says that there is one Accounts Administrator that can add many customers to the application. The relationship between the Administrator and the create CoA function says that is one Administrator can create one or many CoA’s. The relationships between the customer and the view CoA function says that many customers can view one or many CoA’s. The relationship between Administrator and the create quote function says that there is one Administrator that can create many quotes. The relationship between the Administrator and the create quote function says that there is one Administrator can create many quotes. The relationship between customer and the request a quote function says that many customers can request many quotes.

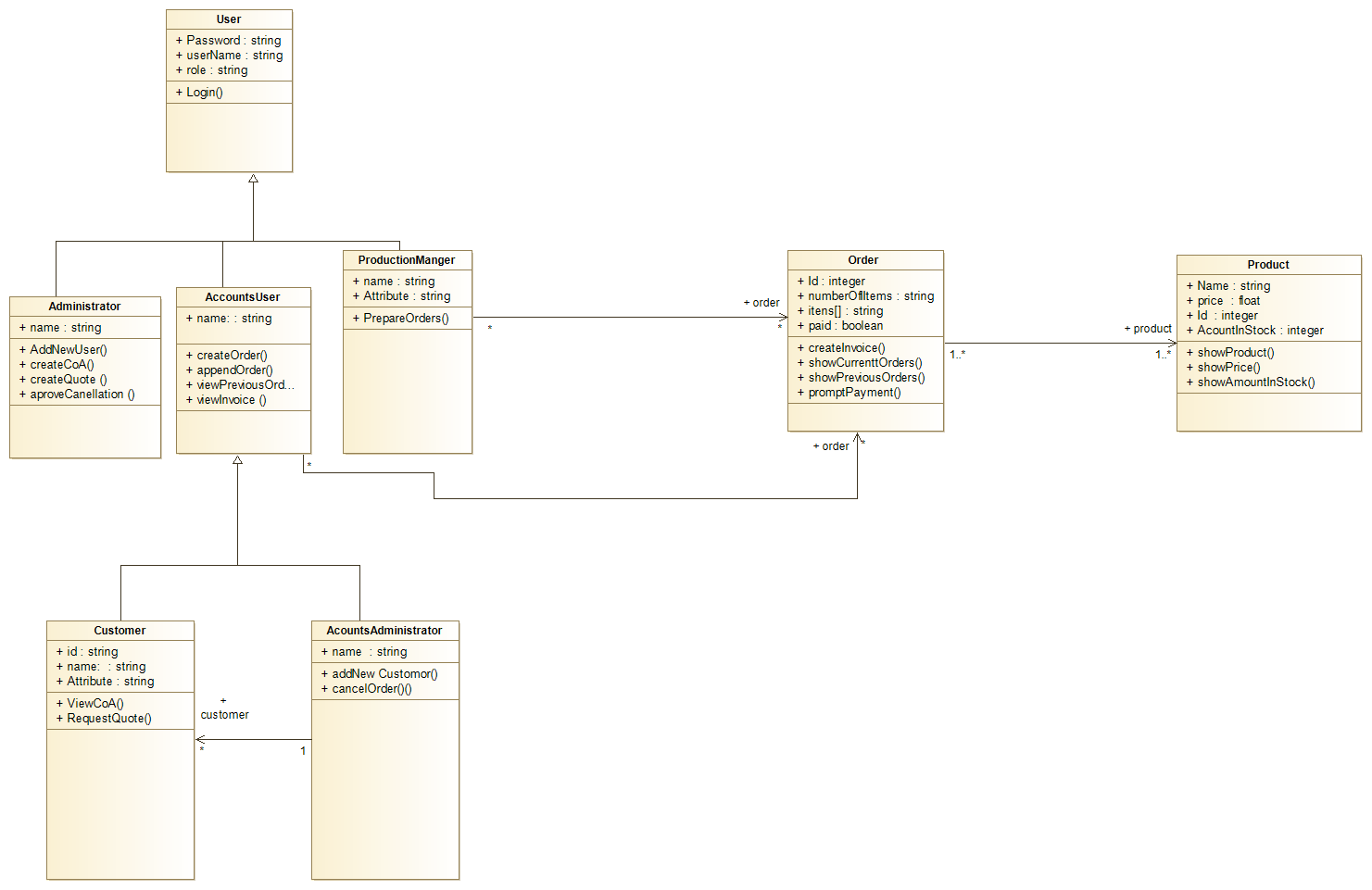


Figure 8 class diagram

The above diagram shows the classes of the system and how these classes are connected. First there is a class of user, there are classes that inherit from the user class, and these are the types of user, Administrator, Accounts User Project manager. Two additional classes inherit from the accounts user class, these classes are Accounts Administrator and Customer. The reason that these classes inherit from account user and do not inherit straight from the user class is that both the Customer and Accounts Administrator use some of the same features of the system and to have the methods written in both the Customer and Accounts Administrator classes would be breaking the DRY principle. There also two other classes which are order and product.

The relationship between product and order says that many orders can have many products within them. The relationship accounts user and order says that many account Users can have many orders. The relationship between Production Manager and order says that there is one Production Manager that can view many orders. The relationship between the Accounts Administrator and Customer says that there is one Accounts Administrator who can add many customers.

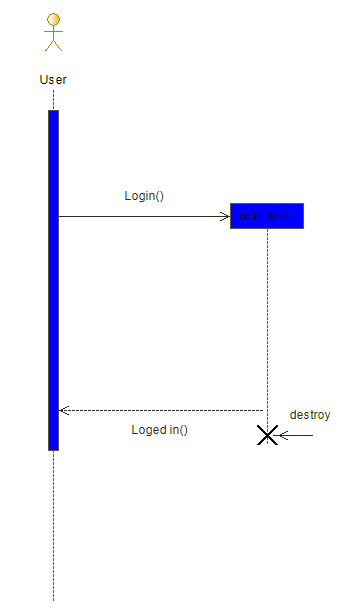


Figure 9 login sequence diagram

The above diagram shows how the login function works. It shows how the user interacts with the login feature. Until the user is at the login page the login form is not active and does interact with the user and once the user has logged in, the login form is no longer active in the process, the information.

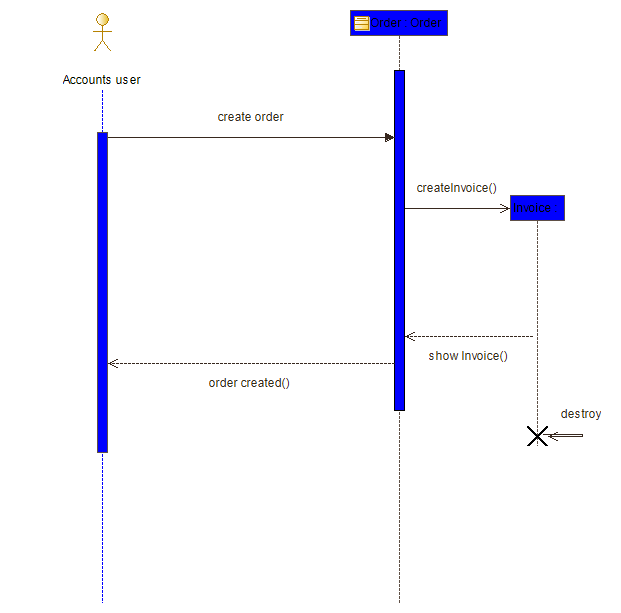


Figure 10 create order sequence diagram

The above diagram shows the behaviour of the create order function. It shows that an accounts user interacts with this function. In this instance an accounts user refers both a customer an accounts administrator. When an accounts user wants to create an order they will fill in an order form which invokes the create order method, this interacts with an order object which then creates an invoice. After this the invoice object interacts with an order object to show the invoice. Once this is done a message is sent back to the accounts user to say the order has been created.

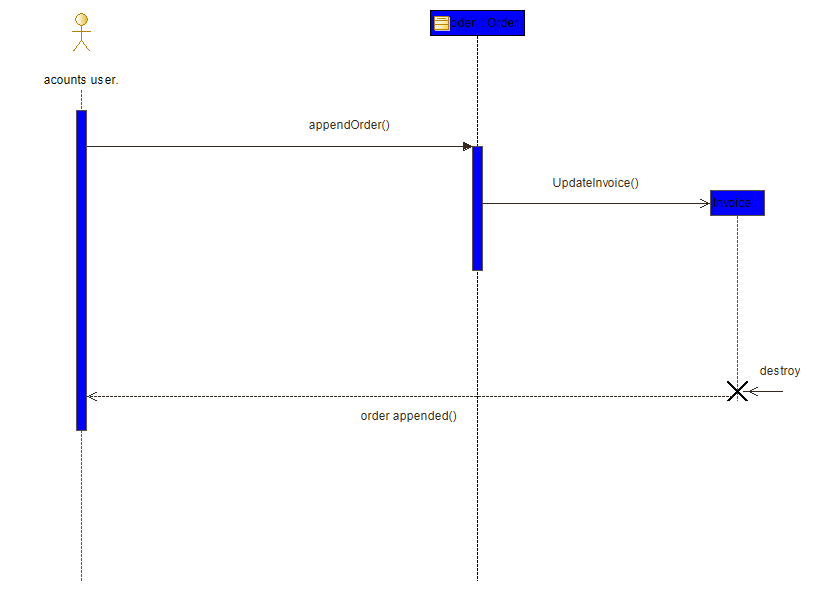


Figure 11 append order sequence diagram.

The above diagram gives an overview of the append order function. It shows that the accounts user will interact with an order by invoking the append order method. As above an accounts user refers to both a customer and accounts administrator. After this is done the invoice is updated. Once the invoice is updated a message is sent back to the accounts user to say the order has been updated.

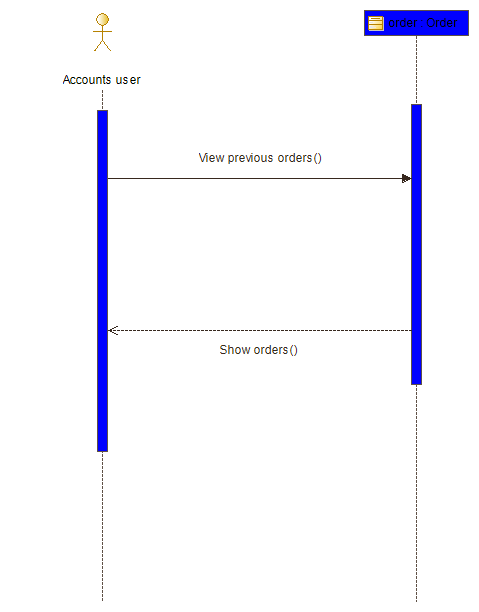


Figure 12 view previous orders sequence diagram

The above diagram shows the behaviour of the view previous orders function. This is where an account user views current orders, again in this instance an accounts user refers to both a customer an accounts administrator. An accounts user invokes the view previous orders function which interacts with the create order class. Once this is done previous orders are shown.

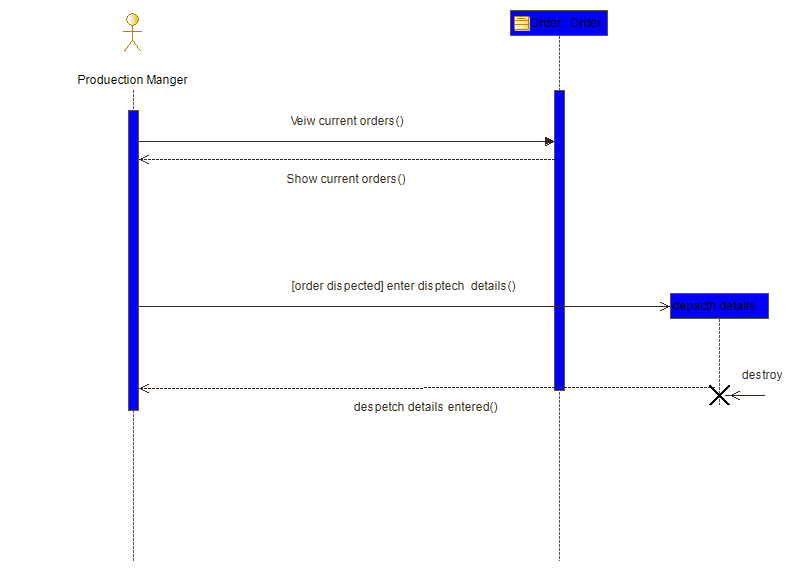


Figure 13 prepare orders sequence diagram

The above diagrams shows the behaviour of the prepare current orders function. The Production Manager will view the current orders, the order section will then display the current orders to the Production Manager in order for them to enter whether a specific order has been despatched. If an order has been dispatched the production manager will enter dispatch details. After this is done, a message will be shown to the Production Manager stating that the despatch details have been entered.

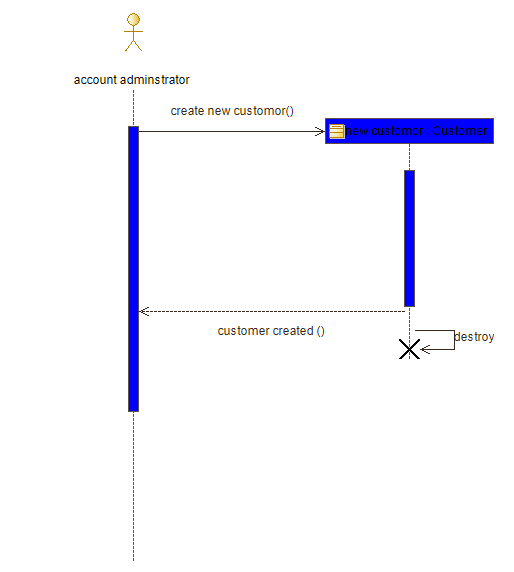


Figure 13 create new customer sequence diagram

The above gives an overview of the behaviour of the create new customer function. The Account Administrator creates a new customer, when they have done this, a new user customer object is created. After this is done, a message is sent to the Accounts Administrator to say the customer has been created.

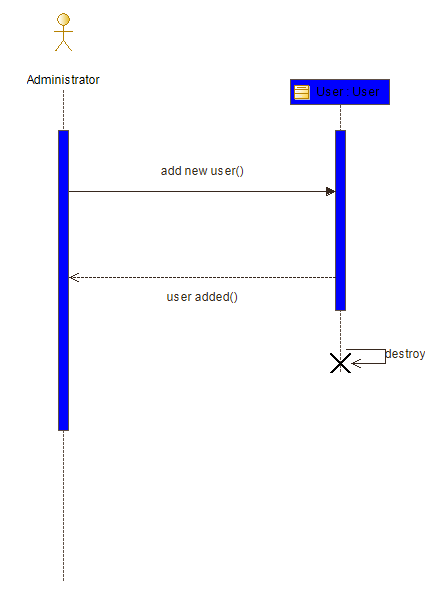


Figure 14 add new user sequence diagram

The above diagram demonstrates the behaviour of the add user function. The administrator invokes the add user method, this will then create a new user. After a new user has been added a message is sent back to the administrator to say a user has been created.

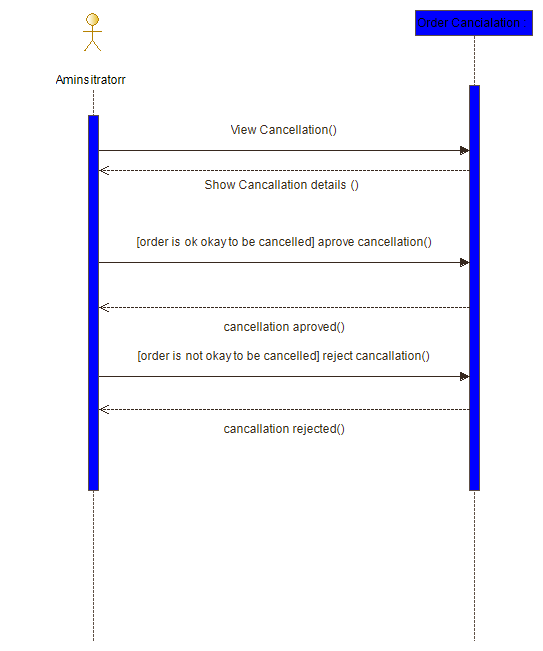


Figure 15 approve cancellation sequence diagram

The above diagram shows the process of the approve cancellation function. The Administrator invokes the function by viewing the cancellation, the order cancellation object shows cancellation details. If the order is okay to be cancelled the accounts administrator approves the cancellation. A message is then sent back to administrator saying the cancellation has been approved. If the order is not okay to be cancelled the Administrator rejects the cancellation. After this a message is sent back the Administrator to say the cancellation has been rejected.

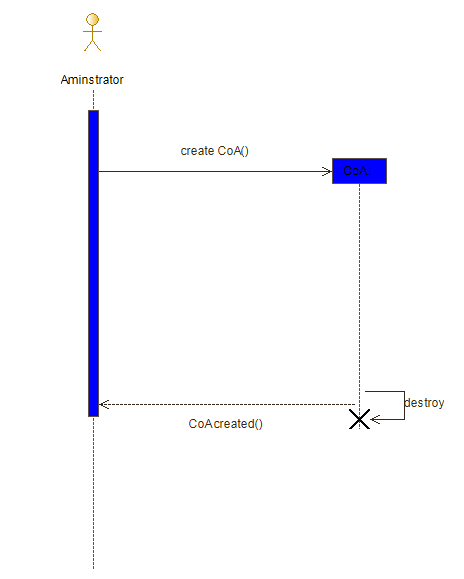


Figure 16 request CoA sequence diagram

The above diagram shows the behaviour of the request a quote function, the customer invokes the request quote method, which creates a request quote form object. After this is done, a message is sent back to the customer to say the quote has been requested.

The above diagram shows the behaviour of create CoA function, the Administrator creates an invoice, which then creates a CoA. After this is done, a message is sent back to the administrator to say the CoA has been created.

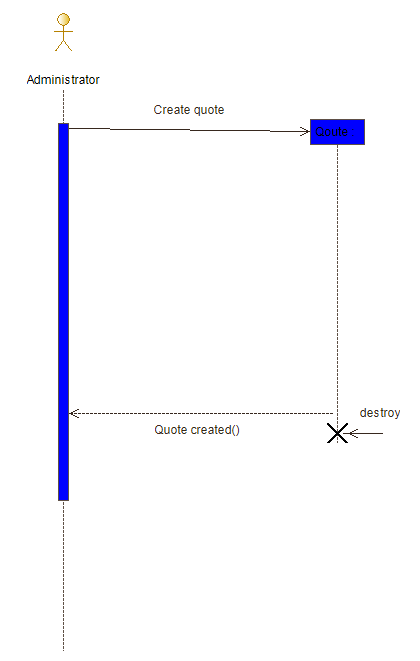


Figure 17 Create quote sequence diagram

The above diagram shows the behaviour build a quote function. The Administrator invokes the create quote method which creates a quote object, after this is a message is sent back to the Administrator to say the quote has been created.

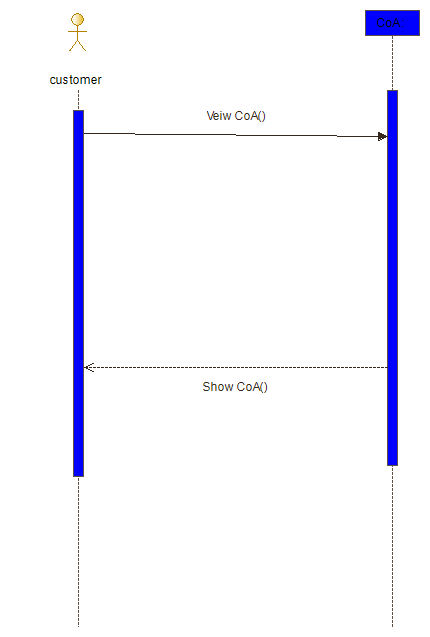


Figure 18 show CoA sequence diagram

The above diagram shows the behaviour of the view CoA feature. The customer invokes the view CoA method which interacts with the CoA object. After this a CoA is shown to the user.

## 2.3 User characteristics

As stated previously there are to be four types of users of this system, these are Administrator. Accounts Administrator, Customer and Production manager.

The Administrator is a full user of the system, they will be able to access logs and view what is happening on the site and add new users to the system. They will be able to view the logs of the site. They will also be able to create a CoA in order for the have access to this as soon as it is created

The Account Administrator has access to part of what the Administrator has access to while the Account Administrator cannot add general users to the system. However, they can add new customers to the system. They will also be able to create orders, append orders and view previous orders and view invoices.

The customer is an end user of the system, they will be able to create, append and view orders. Additionally, a customer will be able view invoices of their own orders.

The production manager will have very limited access to the system, they will be able to view current orders in order to enter a despatch status of the order. in order to view

## 2.4 Constraints

This Project will be developed using PHP. The project must implement the frontend. Within this project the Vapour-tek Logo must be maintained. The application must be designed with a mobile first approach. The application must designed to be accessible for all users. The price of the products must be displayed in sterling. The site must be deployed on a LAMP (Linux, Apache, MySQL, and PHP) stack.

The project must comply with the Data Protection Act. The site must also comply with the cookie standards.

This application will be deployed on to a Linux system, therefore, it will need to be ensured that the correct server configurations.

## 2.5 Assumptions and Dependences

This system will require a connection to the internet in order to run, it will be developed in the PHP programming language.

This site will only have one administrator who oversees the sight these is for security reasons to prevent any information been gathered by. In order for the site to run there must be an internet connection. The customer access to this site will be limited in order to for the site to r. Every user must have a user name and password in order to ensure security of the site preventing any unauthorized access of the site.

.

## 3.6Apportioning of requirements

In feature releases of this application will implement the following.

* A language feature: The site will involve a language feature for users to select their language as orders can be made for customers that are overseas who speak a different language.
* A tracking feature: this will track a product through the shipping process in order to keep track of the products’ location and how far from the customer’s location a product is. This will enable Vapor-Tek to easily locate a products location
* A history tracking feature. This will in order for Vapor-Tek to track what products are order and where they go over time.

# 3 Specific requirements

## 3.1 External interfaces

The following diagrams show pages of the site. For template of what the content showed in these diagrams will be on see section 2.1.2 user interfaces.

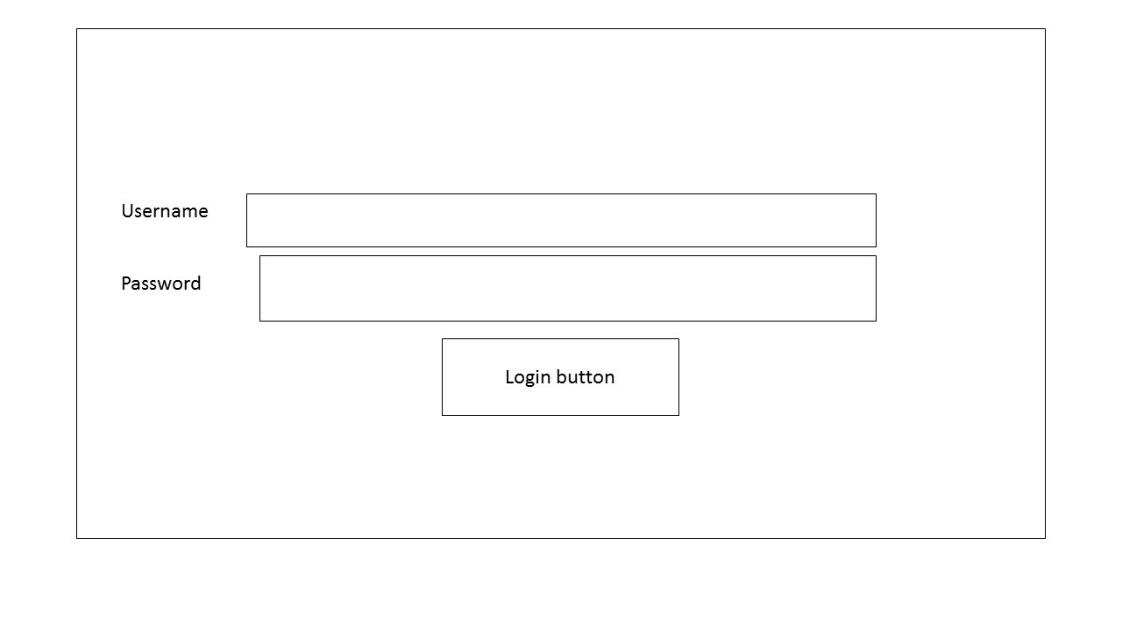


Figure 19 login page

The above diagram illustrates the login page. The page will ask users to enter their user and password in order to log in. The password field will be in a protected format and will not be shown to the user in plain text.

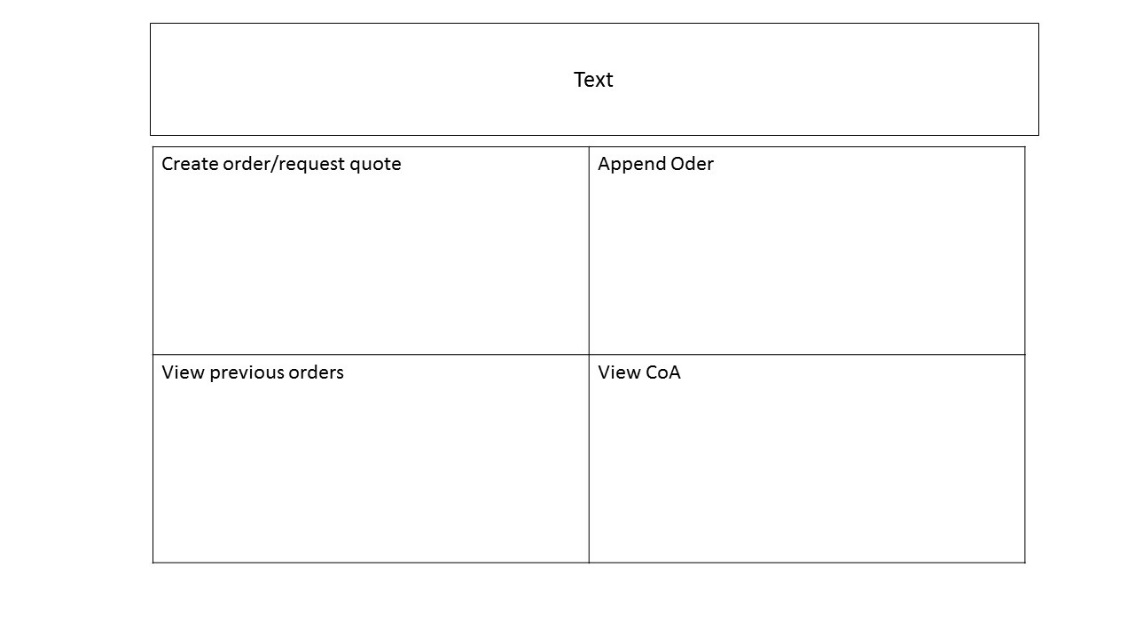


Figure 20 Customer homepage

The above diagram shows what the customer will see when they first login to the system. The page consist of table of links each one linking to the feature as displayed in the

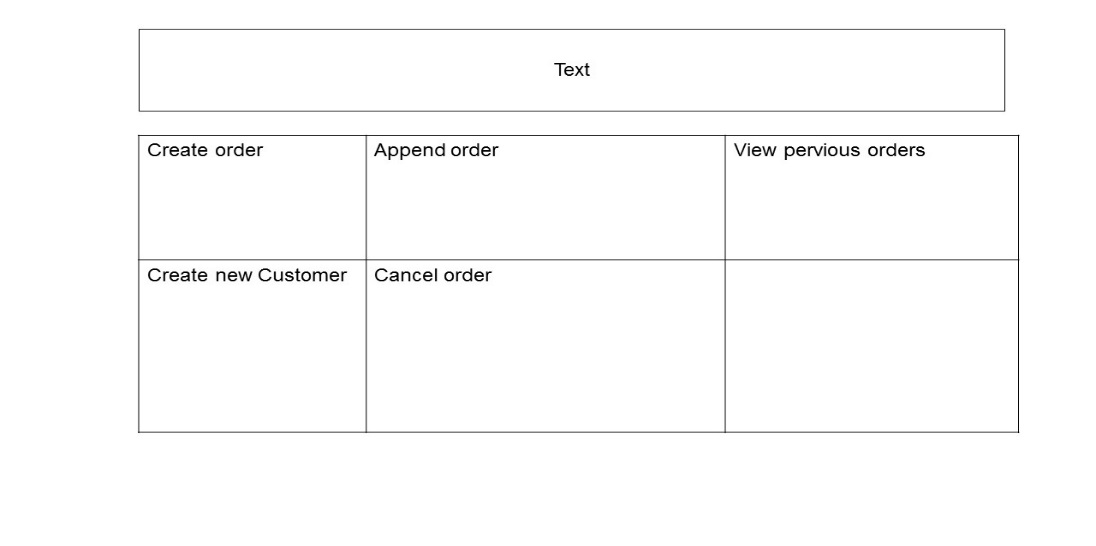


Figure 21 login Account administrator home page

The above diagram demonstrates what the Account Administrator will see when they first login to the application, this will consist of a table containing links to appropriate keys features as shown in the diagram, containing process,

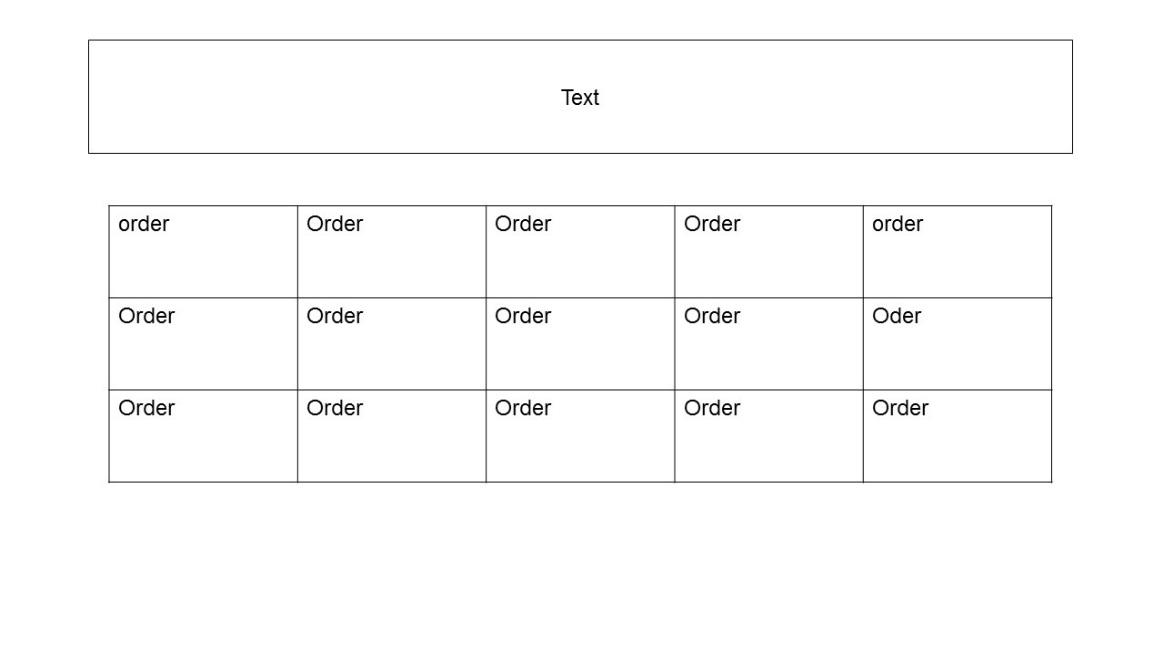


Figure 22 production manager home page

The above diagram shows what the production manager will see when they login to the system. The page will show a table of orders, which the production manager will be able to click a specific order to view information about it and enter whether that order has been despatched. This will be the part of the prepare current orders feature that the production manger

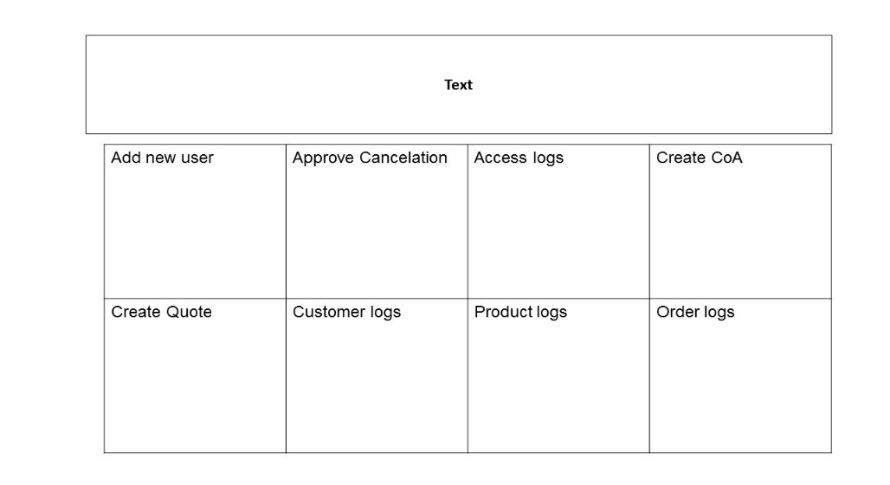


Figure 23 Administrator home page

The above diagram demonstrates what the Administrator will see when they login to the application. The first the page will consist of six buttons as displayed in the diagram, each button will link to the appropriate page.

## 

Figure 24 order from page

The above diagram shows the order form. As shown in the diagram the interface will ask for various inputs from the user. The customer Id, name and Company will need to be entered in a text format which will be checked in the application. The products will be in a drop down list, the quantity will need to be entered in a numeric format, and again this will be checked in the application. There will be two options on this page, one will be to order the products and the other will be to request a quote. The reason the create order and request quote features will not be displayed on separate pages is because that these features will take the some of the same details so it seems appropriate to have one form for this with different functionality for each button.

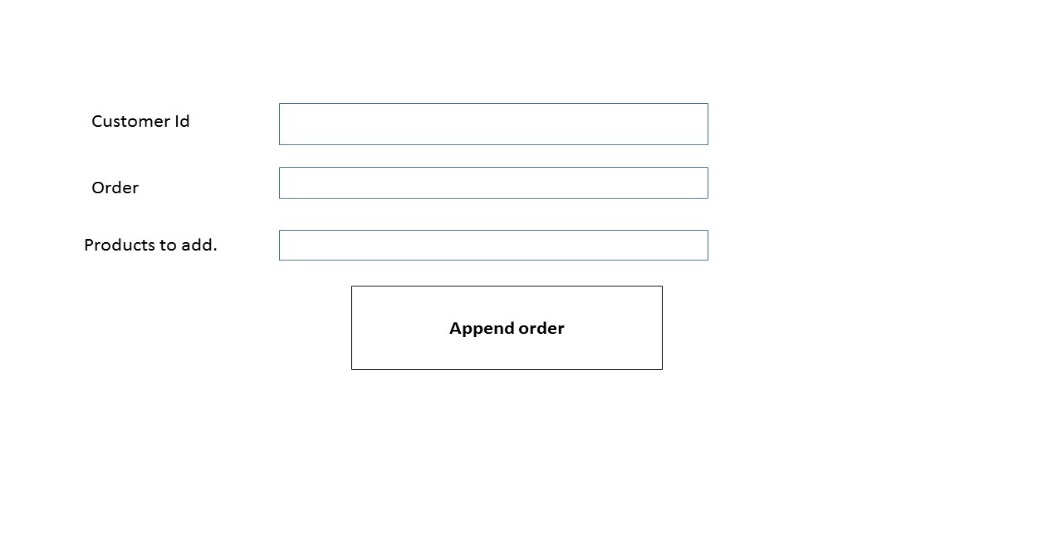


Figure 25 append order page

The above order shows the append order pages, the append order it will linked when from an accounts users home page either the accounts administrator or customer. It will ask the user to confirmed the customer id, this will be need to be entered I text format. There will need to enter the order id this will need to in numeric format. The products to add must be entered in a text format. There will be checks to ensure in the application

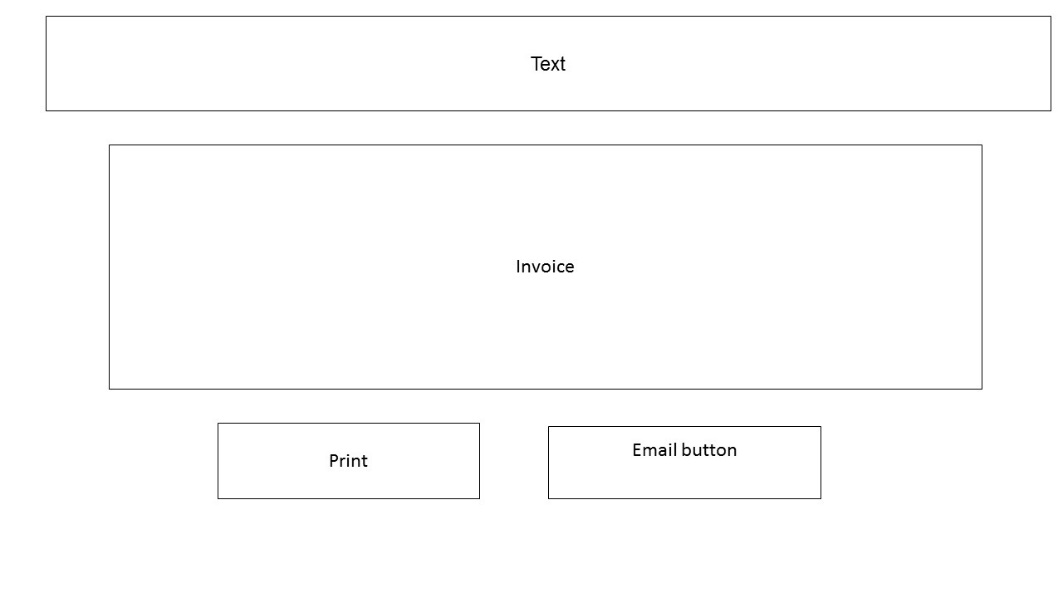


Figure 26 invoice page

The above diagram shows what the invoice page will look like, as the diagram shows it will display an invoice and give users the option to email and print the invoice. The invoice will also be able downloaded. The invoice will be in a PDF format.

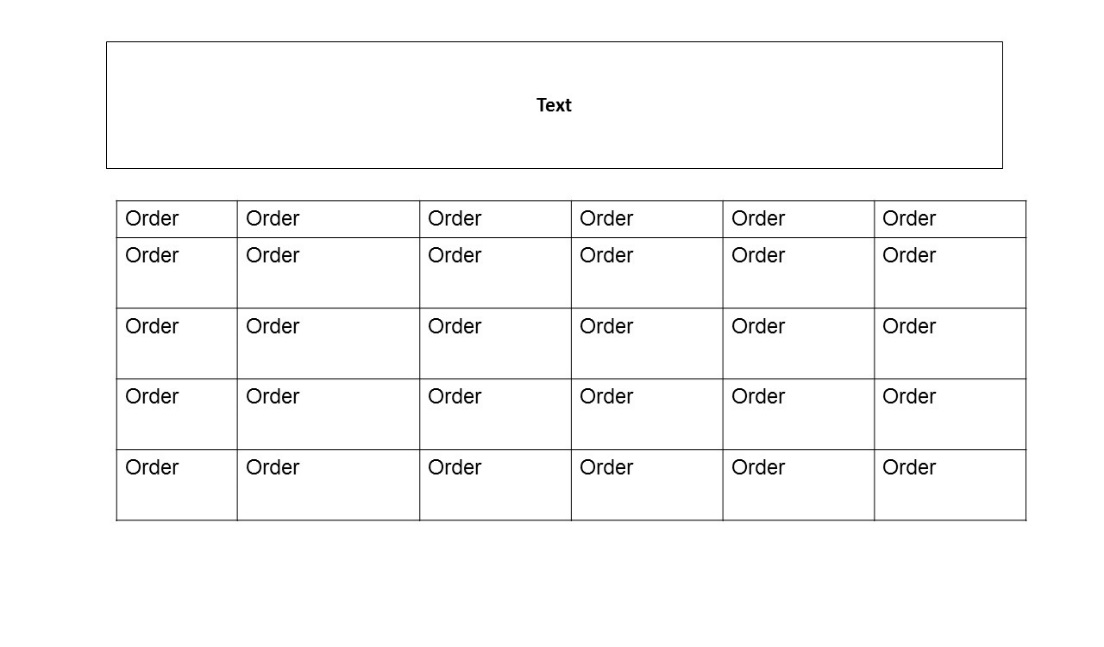


Figure 27 view previous orders pages

The above diagram will show what the view previous orders page of the application will look like. As shown in the diagram a table will be displayed to the user showing all the user’s previous orders.



Figure 28 add new customer page

The above diagram shows the add new customer page. This will ask the user to enter the customer’s information. All the form inputs will need to be entered in text format, this will be checked in the application

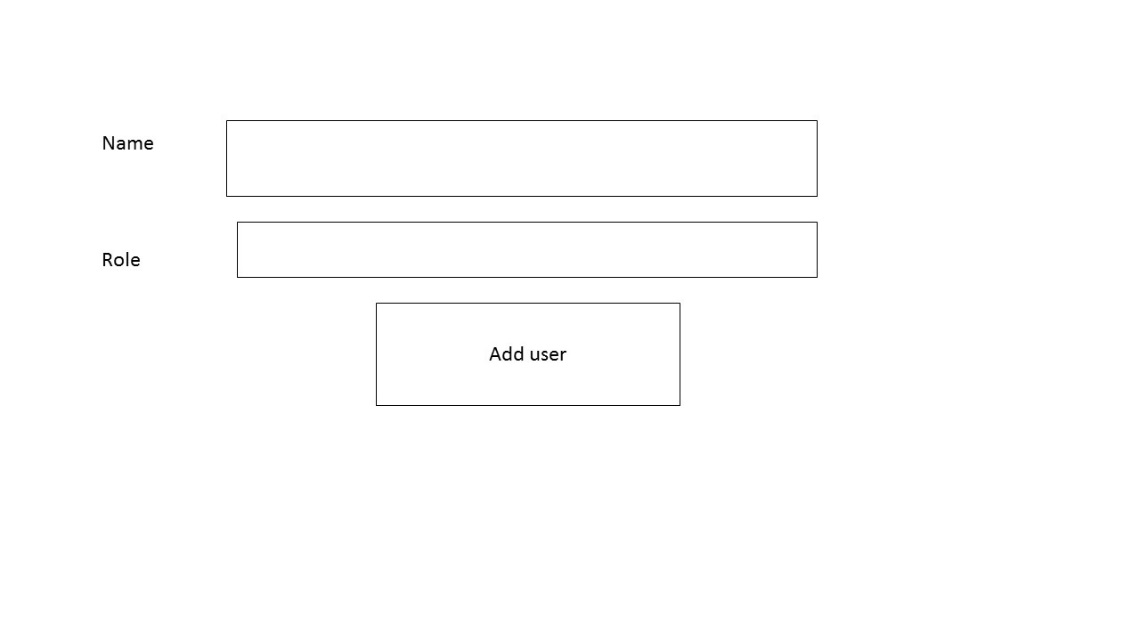


Figure 29 Add new customer page

The above diagram gives an overview of the add new user page. As shown the in the diagram the users will be asked to enter the user’s name and role. Both form inputs will have to be entered in a text format and there will be checks for this within this application.

## 

Figure 30 Create CoA page

The above diagram shows the create COA page. This will allow the Administrator to enter information regarding the product and its pass status. The product information will be required to be in text format and the pass status will be a drop down box. In order to gather exactly what information is entered in this page it will it will need to confirmed what data a CoA takes

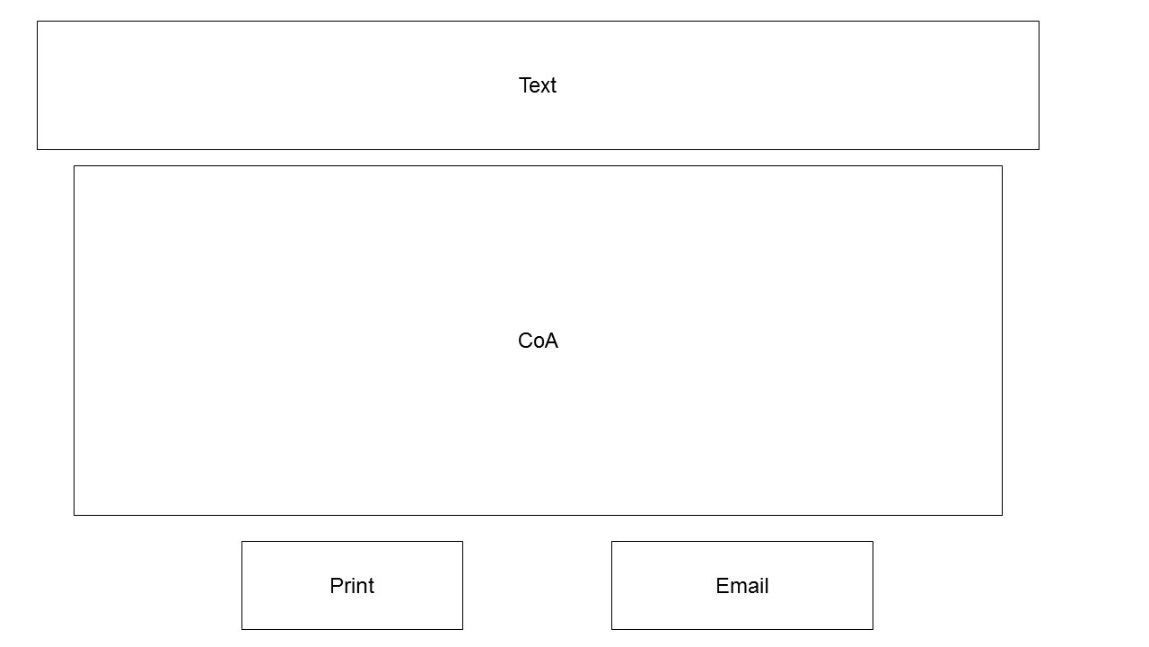


Figure 31 view CoA page

The above diagram shows what the view CoA page will look like. It will display a certificate of analysis for a particular product to the customer. This will be displayed in PDF format. This will be for the View previous order feature

## 

Figure create quote page

The above diagram shows the create quote page. It will allow the administrator to create a quote for customers who have requested a quote.

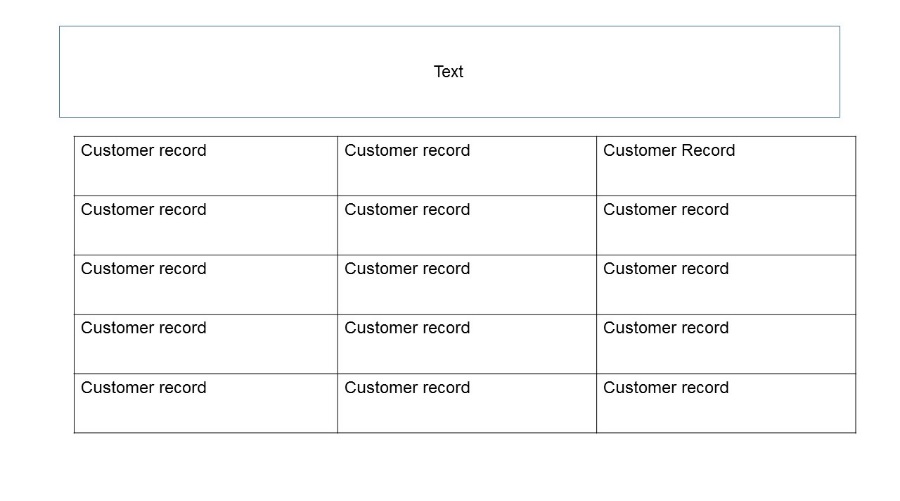


Figure customer record page

The above diagram shows the client logs page, this will show information from discussions with clients and will also allow an administrator to view to records about the customer

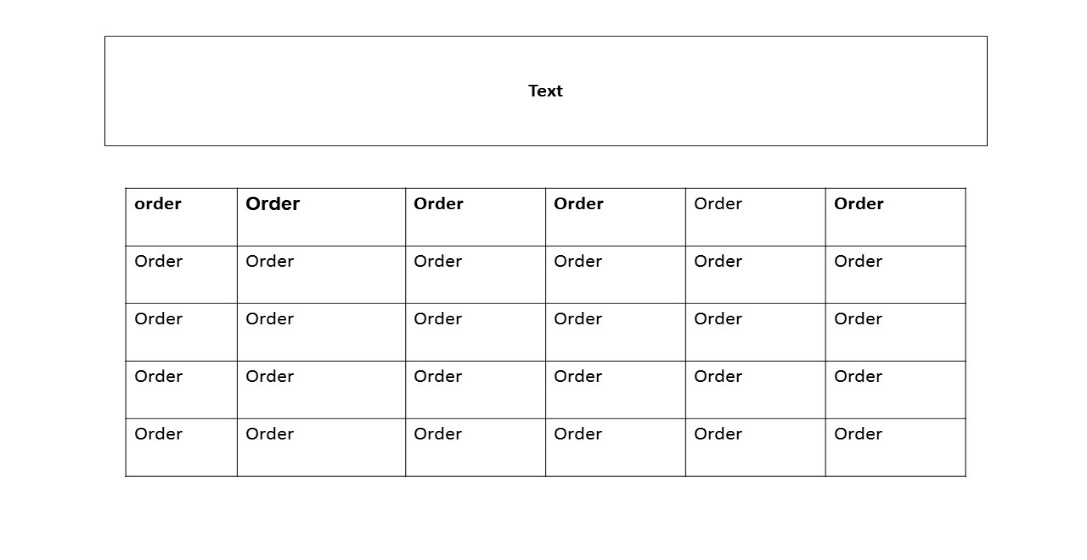


Figure previous orders page.

The above diagram shows the previous orders page. It will allow a user to view their previous orders which will be displayed in a table. These orders will be displayed as links and users to will be able to click in order to view the specific order.

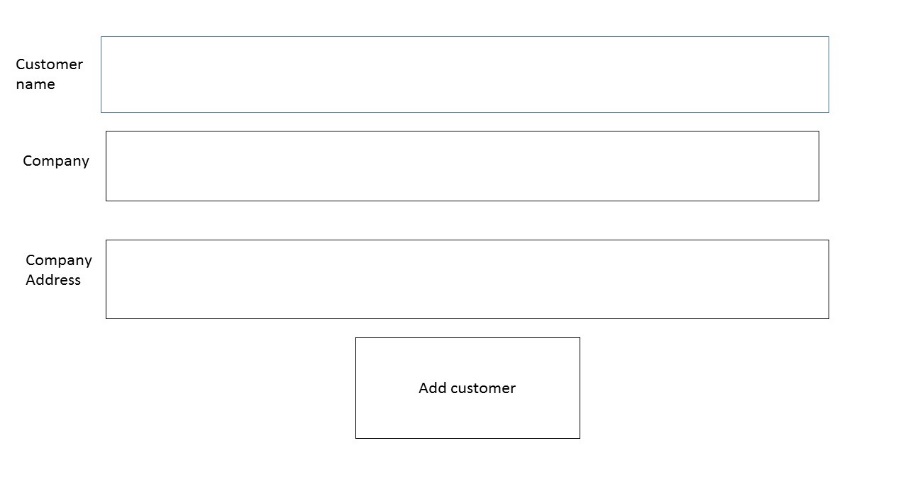


Figure 35 add new customer page

The above diagram shows the add new customer shows it will allows the accounts user to enter details about. All forum must be entered into text format which will be checked by the application. Which will be checked. This will be page for the add new customer feature

## 2.2System features

### 3.2.1 Feature 1

Table 3 login use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use case Name: Login | | Id: 1 | | Importance level: High | |
|  | | | | | |
| Primary Actor: user, | | | | | |
| Short description: process taken when a user logs in. | | | | | |
| Trigger: User submits log in form. | | | | | |
| **Major Inputs**  Username  Password | **Source**  User  User | | **Major outputs.**  Login time | | **Destination**  Database, administrator |
| **Major steps**   1. Determine log in details.   1.1If log details correct Redirect to appropriate page.   * 1. If details incorrect show error message to the user.  1. Determine what type of user 2. .    * 1. If customer. Redirect to customer homepage.      2. If Administrator , Redirect to administrator homepage,      3. If accounts administrator page. Redirect to account administrator homepage.   2.1.4 If production manager Redirect to production manager homepage. | | | **Information for major for steps.**  Username, password,    Role of user form database    Login in time | | |

The above table describes how the login feature works. It describes the process from when the user logs in to the system. The reason why the importance level is high is that without the login feature users could not use any of the other functions as they require a specific user to use them. The data input in this feature are the username and password. The first step in the process is verify the log in details and if they are incorrect show an error message to the user, the information taken for this step is username and password. The second step is to determine which type of user has logged in and redirect to the appropriate page.

### 3.2.1 **Feature 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case name**: create order. | | Id: 2 | | **Importance level**: high | |
| **Primary actor**: Customer, | | | | | |
| **Short description**. Tiger how the customer creates orders. | | | | | |
| **Trigger:** customer submits order form.  **Type:** External | | | | | |
| **Major inputs**  Customer name.  Customer come company address,  Name of products  Quantity of products. | **Source**  Customer  Customer,  Customer  Customer  Customer | | **Major outputs.**  Cost of each Item.  Total cost. | | **Destination**  Invoice  Invoice. |
| **Major steps preformed.**   1. Verify data input.    1. If not valid. Show error message. 2. Determine what items have been ordered    1. check items are in stock   2.1.1 if items not in stock display out of stock message  2.1 work out cost of each item,  2.2Calculate total cost. | | | **Information for major steps**  Name of products      Cost of each item,  Total cost | | |

The above table explains the create order function. It explains the process from when the customer submits the order form and what steps are taken by the application. The importance level is high as the create order feature is one of the main features of this site as it allows the customer to order products online. The inputs that this feature takes are customer name, customer address, name of products, and quantity of products. The source for all these inputs is the customer. The output data from this feature are cost of each item and total cost, these are outputted to the invoice. The first step is to verify data input, if the data is invalid for example numeric content is used instead of text, an error message would be shown to the user. After this, the next step is to determine what items have been ordered. The third step is to determine the quantity of items. The last step is to calculate the total cost

## 3.3.3 Feature 3

Table 5 append order use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case name**: append order | | Id: 3 | | Importance level: high | |
| **Short description** : : how the append order function works, | | | | | |
| **Primary actor**: accounts users  **Type**: external | | | | | |
| Trigger: accounts user updates order form. | | | | | |
| **Major inputs**  Order  Items to add. | **Source**  Accounts user  Accounts user**.** | | **Major output**  Cost of each item added.  Updated total cost**.** | | **Destination**  Invoice  Invoice |
| **Major steps performed.**   1. Check if order has been dispatched   1.1 If order has been dispatched show message to user to say order cannot be updated.  2. Determine any added items.  2.1 Check Items are in stock  2.1 .1 if items are not in stock show out of stock message.  2.1 calculate cost of each item added.   * 1. calculate updated total cost   3.Update invoice | | | **Information for major steps**  Order  Items to add.  Cost of each item added  Updated total cost | | |

The above table shows the append order feature. It describes the process taken when an accounts user appends an order. The datum input in this feature is items to add. The output data in this feature are cost of each item are cost of each item added and updated total cost. The first step is to determine any added items, the information is used for this step is items to add. The next step is to update the invoice.

**3.2.4 Feature 4**

Table 6 view invoice use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case name:** view invoice | | **Id:** 4 | | **Importance level:** medium. | |
| **Short description:** how the accounts user views an invoice. | | | | | |
| **Primary Actor:** Accounts user | | | | | |
| **Trigger:** Order is submitted.  **Type:** external | | | | | |
| **Major input**  Cost of each  Total cost | **Source**  Order  Order | | **Major outputs**  Invoice | | **Destination**  Accounts user**.** |
| **Major steps performed**  **1.** Shows invoice.  1.1 locate invoice  1.2 determine format to displayed | | | Cost of each, total cost.  Invoice | | |

The above table shows the process taken when an accounts user views an invoice. The input data in this feature are cost of each item and total cost. This feature involves one main step which is show invoice. Whiten this step the invoice is located and it is determined what format,

### 3.2.5 Feature 5

Table 7 add new customer use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case name:** add new customer | | **Id**: 5 | | **Importance level:** high | |
| **Short description** : The account administrator adding new customers to the system | | | | | |
| **Primary actor**: accounts administrator: | | | | | |
| **Trigger:** account administrator adding new customer information  **Type**: external | | | | | |
| **Major inputs**  Customer name    Company the of the customer, | **Source**  Accounts Administrator  Accounts Administrator | | **Major outputs**  Customer Id | | **Destination**  Customer, database, |
| Major Steps Preformed.   1. Determine what format information is entered    1. If not valid show error message 2. Update customer table in database.   2.2 check customer does not already exist  2.2.1 if customer already exists show error message to user,  2.2.2 if customer does not exist update, customer information in database. | | | **Information for major steps**  Customer name, company of customer.    Customer ID | | |

The above table describes the add new customer feature. It describes the process from when the accounts administrator adds a new customer to the system. The input data in this feature are customer name and company of the customer.

The first step is to determine if the information is valid and if not show an error message to the user. The second step is to update the customer table in the database.

### 3.3.6 Feature 6

Table 8 add new user use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case name**: add new user | | **Id**: 6 | | **Importance level**: high | |
| **Short description** : administrator adding new user | | | | | |
| **Primary actor**: Administrator | | | | | |
| **Trigger:** administrator add new user form  **Type:** external | | | | | |
| **Major Inputs**  name of user  role of user | **Source**  Administrator  Administrator | | **Major outputs**  Username  password | | **Destination**  Database  Database |
| **Major steps preformed**   1. Verify user details    1. check input is in right format,   1.1.1 if user input is not in right format   1. Generate username 2. Generate password | | | **Information for major steps.**  Name of user, role of user  Username, password | | |

The above table gives an overview of the add new customer feature its shows process taken when an Administratoradd a user, the inputs taken in this the user’s name and user’s role

### 3.3.7 Feature 7

Table 9 prepare current orders use

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case Name**: prepare orders | | **Id**: 7 | | **Impotence level**: High. | |
| **Short description**: when a production manger the prepares the current orders. | | | | | |
| **Trigger:** protection manger prepares orders.  **Type:** external | | | | | |
| **Primary Actor** : production manager | | | | | |
| **Major inputs**  Orders | **Source**  Accounts users | | **Major outputs**  Order information, | | **Destination**  Production manger |
| **Major steps preformed**   1. Show orders    1. Locate orders made    2. Display in table 2. Show order status | | | **Information for major steps**  Orders,  Order information | | |

The above table describes how the prepare order functions works, it shows how the process from when the production manager prepares orders. The input datum this feature is orders the information that is outputted from this feature is order information this gets passed to the enter despatch details function. This feature will consists of two steps shows and show order status.

### 3.3.8 Feature 8

Table 10 enter despatch details use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use case name: enter despatch details | | Id: 8 | | Importance level: medium | |
| Short description : product manager entering the dispatch details | | | | | |
| Primary actor: protection manager | | | | | |
| Trigger : protection manger filling out despatch details:  Type: external type | | | | | |
| **Major Inputs**  Order information  Dispatch details. | **Source**  Prepare orders feature  Production manager | | **Major Outputs** | | **Destination**  Customer |
| **Major steps perform**   1. Determine whether product has been dispatched    1. if product has been dispatched set dispatch status to despatched    2. If product has not been despatch if product has not been dispatched set dispatch status to awaiting dispatch | | | **Information for major steps**  Order information from prepare orders use case  Despatch details.    Despatch status | | |

The above table describes the process of enter despatch details. It describes the step taken when the production manager enters the dispatch details. The reason why the importance level is medium is that it seems that it would be more important to implement other features first such as the login and create order. The information that will be inputted into this feature will be order information and despatch detail. The Information that will be outputted is dispatch status. This feature consists of one main step which is to determine whether the product has been despatched and if it has, update despatch details in the system.

### 3.3.9 Feature 9

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use case name: cancel order | | Id: 9 | | Importance level: high | |
| Short description: process of when the accounts administrator cancels an order. | | | | | |
| Primary Actor: Accounts administrator | | | | | |
| Trigger: The Accounts Administrator submits cancelation of order  Type: External | | | | | |
| Major Inputs  Order information | Source  Accounts administrator | | Major outputs  Cancel order request | | Destination  Administrator |
| Major steps performed   1. Determine the order cancelled    1. check if order has been dispatched       1. if dispatched show error    2. check if order has already been cancelled   1.2.1 If order has been cancelled show error message.   1. Send cancel order request. | | | Information for major steps  Order information,    Cancel order request | | |

The above table shows how the cancel order feature works. It describes the process of when an Accounts Administrator cancels an order. The information that is inputted into this feature is order information. The information that will be outputted from this feature will be cancel order request as the administrator has to approve all order cancelations. The first step will be determine to the order cancelled. The next step is to send a cancel order request to the administrator.

### 3.3.10 Feature 10

Table 11 Approve Cancellation Use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case Name**: Approve cancelation. | | **Id**: 10 | | **Importance level**: high | |
| **Short description**: process taken when an account administer approves a cancelation | | | | | |
| **Primary actor**: Administrator. | | | | | |
| **Trigger**: Administrator submits approval for cancelation  Type: External | | | | | |
| **Major Inputs**  Order to approve | **Source**  Account administrator | | **Major outputs**  Approval status | | **Destination**  Database**.** |
| Major steps performed   1. Determine whether the order has been approved for cancellation    1. if approved cancel order    2. if not approved not send to notice to account administer 2. Update Approval status | | | **Information for major steps**  Order to approve  Approval status | | |

The above table describes how the approve cancelation function works. It describes the processes from when the administrator submits approval of cancelation. The information that is inputted into this feature is order to approve. The information that is outputted from this feature is approval status. There are two steps to this feature as shown in the table. The first step is to determine whether the order has been approved for cancelation, if it is approved then cancel order and if has not been approved send notice to the account Administrator. The next step is to update the approval status of the order.

### 3.2 11 Feature 11

Table 12 Create CoA use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case Name:** create CoA | | Id number: 11 | | Impotence level: high | |
| Short description: process taken when Administer enters information for CoA | | | | | |
| Primary actor: Administrator | | | | | |
| Trigger: Administer enters information for CoA  Type: external | | | | | |
| **Major Inputs**  Information about the product | **Source**  **Administer** | | Major outputs  CoA | | Destination  Customer |
| **Major steps preformed**   1. Gather information about product.    1. Determine the pass status.   1.1.1 if not passed show message.  1.1. 2 If passed create certificate | | | **Information for manger steps**  **I**nformation about product  CoA | | |

The above describes the function of create order CoA function, CoA stands for certificate of Analysis. This describes the process when an administrator enters information for a CoA, The input datum taken for this feature is information about. The output datum form this feature is COA. This step continents two steps the first step is gather information about the product. The next step is create CoA is feature

### 3.2.12 Feature 12

Table 13 View CoA use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Use case name: view CoA | | Id: 12 | | Impotence level: high. | |
| Short description: the process taken when Customer views A CoA | | | | | |
| Primary actor: Customer. | | | | | |
| Trigger: customer views CoA  Type: external | | | | | |
| Major inputs  CoA | Source  Create feature | | **Major outputs**  CoA | | Destination  Customer |
| **Major steps performed**   1. Show CoA    1. locate certificate    2. Determine format it needs to display. | | | **Information for major steps**  CoA  CoA | | |
|  | | | | | |

The above table gives an overview of the view CoA function, it describes the process taken when customers Views CoA. Both the input and output datum for this feature is CoA. This feature involves one main step which is Show CoA, whiten in the step the certificate is located and it is determined the format it needs to display

### 3.2.13 Feature 13

Table 14 build quote use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case name**: create quote | | **Id:** 13 | | **Impotence level**: high | |
| Description : process taken when an Administrator enters quote information | | | | | |
| Primary actor: administrator: | | | | | |
| Trigger: Administrator enters quote information  Type: external | | | | | |
| **Major inputs**  Product information  Price  Discounts | **Source.**  Administrator  Administrator  Administrator | | **Major outputs**  Price for each  Item  Discounts  Total price | | **Destination.**  Customer  Customer  Customer |
| **Major steps preformed**   1. Determine products.    1. check products are in stock   1.1.1 if products not in stock show out of stock message,   1. Determine price. 2. Determine discounts 3. Determine total price | | | Product information, price dis | | |

The above table describes the build a quote a feature. It describes the process taken from when an administrator enters information for a quote. The input data for this feature are product information, price and discount. The output data form this feature are price for each item, discounts and total price. This feature consist of four main steps. The first step is to determine the products. The next step is determine the products. The next step is to detrained price. The third step is to determine discounts. The final step is to determine total price.

### 3.2.14 Feature 14

Table 15 request quote use case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Use case**: request quote | | **Id**: 14 | | **Importance level:** high | |
| **Short description**: process taken when a customer requests quote. | | | | | |
| **Primary actor**: customer | | | | | |
| **Trigger**: customer requests a quote.  **Type**: external | | | | | |
| **Major Inputs**  Product information | **Source**  Customer | | **Major outputs**  Production information | | **Destination**  Administrator |
| Major steps performed.   1. Determine product information    1. check if valid input       1. if not valid show error message.   1.2 check what products have been requested,   1. Send request to administrator. | | | Information for major steps  Product information | | |

The above table explains the request quote feature, it describes the process form when the customer requests a quote. Both the input data and output data in this feature are product information. This feature consists of two main steps the first step is to determine product information. Whitten this step it is checked if the user has enter valid input showed an error message if information is not valid. Additionally it is checked what products have been requested second step is to send the request to the Administrator.

## 3.3 Performance requirements

This system will transferring data into a database and the data input of what is stored in the database will be the data type that is required for any form input.

Any incomplete data submission will not be entered into the database and dialysed the user to fix any errors that are found before transferring into the data. The information.

When an order is made, this must update stock information in the database to keep this information up to date. When an order has been despatched this must update the database.

## 3.4 Design constraints

This application must use the already developed fount end of the application and must not alter that. The application must be developed in PHP. The logo and colour of the logo must be marinated during the development. The site must be developed with a mobile first approach to cater for all deceives,

## 3.5 Software system attributes

## 3.5.1 Reliability.

The reliability of this application will be depend on access software such as the user’s browser and the reliability of the server. It assumed that as long the server is reliable then the site is also reality.

### 3.5.2 Availability

This application will be hosted on server. Therefore, the application will be running when the server is available.

### 3.5.3 Security

As this system is involves personal details, it will need to be secure. In order to achieve this as stated earlier it will need a security certificates. The reason for using HTTPS instead of the standard HTTP is that HTTP is written in plain text leaving it open for attackers to use. Also HTTPS does not hide rough HTTP request or response sizes, traffic direction and timing patents in a browsing session which means attackers can still get some information. Therefore a further method to hide this information will need to be implemented in this application. Zalewski (2012)

There ways for keeping web applications secure, this includes percent encode all information expect for numeric information when user data, according to W3C schools percent encoding also known as URL encoding replaces all invalid ASCII Characters with a % followed by two hexadecimal did gets. Therefore, following this suggestion any user data that is entered into cookies will be URL encoded. Zalewski (2012)

All data in the database will be encrypted to minimise risk of unauthorized accessed

It is suggested is that with all HTML content always display inform valid and browser supported connect. This will be ensured within the application. (Zalewski 2012)

There are common vulnerabilities to be aware of when developing web applications .When developing this application these will need to be thought about. (Zalewski 2012) This includes

* Cookie forcing: cookie forcing is the risk of unknowing injecting cookies into the content of a web application because of issues with the way the method is developed within a modem web browser. This will be handled by having correct cookie sectary rules, , there are security rules that must be developed in order to stop cookies belonging to one site being altered by sites that are not related. Cookies must be linked to the domain and not reduced easily to a single hostname. This application will have a way of implementing this security rule. (Zalewski 2012)
* HTTP downgrade: This is when an attacker is able is to stop a user form reaching a version of HTTPS of a specific site or downgrading a HTTPS season to HTTP. This will need to be consideration within this application. Zalewski (2012) point out a way of dealing with this is strict transport security to ensure this there is a need to use secure cookies. (Zalewski 2012)
* Mixed content: This is putting non HTTPS content onto HTTPS pages which leaves the page vulnerable to attackers. Zalewski (2012) suggests in this case requiring all resources to be sent over HTTPS. This will be ensured within the application.
* Click jacking: Click jacking is when an attacker hides to behind a site making the user unaware that they are interacting with the site. While developing the site this will need to be considered. (Zalewski 2012)

To ensure information that is held in the database of this application is secure, there will be Checks for SQL injection attacks to prevent any unauthorised access to the data in the database. An SQL injection attack is where someone enters a partial/incomplete SQL query into an input filed in order to gain access to information inside a database

### Maintainability,

In order for application to be easily maintained. It will be written with clean code. One way of ensuring this is to make sure code is clearly commented when needed. The code will have will spaces between separate functions variables and functions will clearly name.

### Portability.

This site should be able to run in any web browser and should be able to use on any operating system. As this application will developed in PHP which is an interrupted language use of any compliers in this application are not needed.

## Logical database requirements

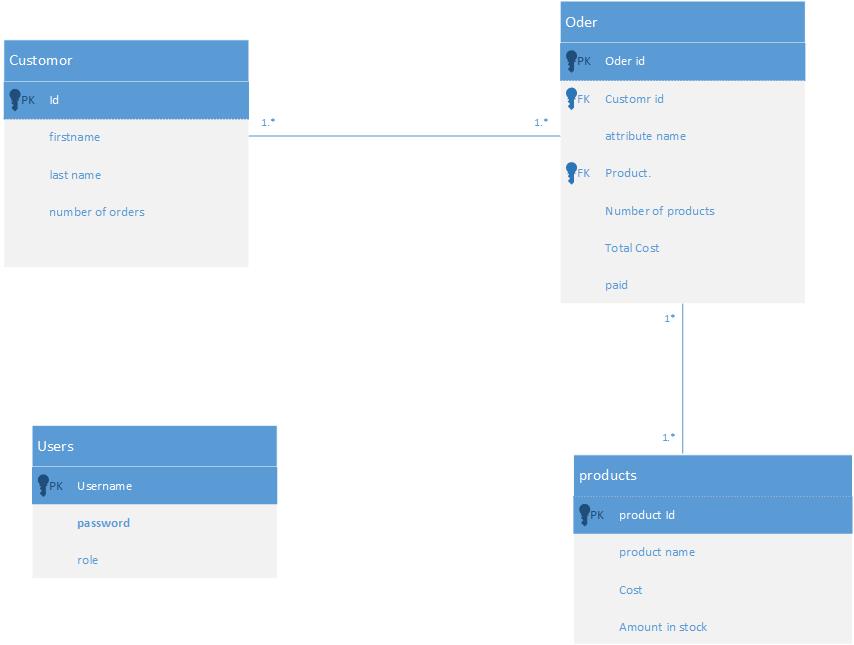


Figure 36 entry relationship diagram

The above diagram shows the tables that will in the database relating to this application of the will application. It consist of four tables which are users. Customers, Orders, The customers and orders have a relationships that says many customers can have many orders, the primary key for the users table is username. The primary key for the for customer table is customer Id, the primary key for the products table is product Id, These will enable records to uniquely identified. The order table has a foreign key of customer ID which comes from the customer table. Additionally, there is a father foreign key of product id in the order table this comes from the product table. All fields in all tables will be have a Constraint of not null. This means that all fields need to have a value when records are entered.

## Standards and compliance.

This application will need to comply with the W3C accessibility guidelines. These guidelines are a legal requirement. As stated in the Web Content Accessibility(WCAG) guidelines (2008) these guidelines are made up of four principles probable, operable, understandable and robust, the four guidelines are explained below,

* Predicable . This guideline says that content must be presentable in ways users can interpret as stated in the WCAG content accessibility guidelines 2.0(2008). This principle is made up of four areas, non-text content, time based media, time based media, adjustable and disable
  + Guideline 1.1 of the web Content accessibility guideline 2.0 non text content: states that all non-text content must have a text based alternative to achieve this all form input control will have clear names this is said in the WCAG Content accessibility guidelines . This will be done though HTML name tags.
  + Guideline 1.2 of the web Content accessibility guidelines time based media is not relative in this case as there will be no pre-recorded content.
  + Guideline 1.3 of the web content accessibility guidelines 2.0: adaptable states to make content easy to see, this will be achieved by allowing users to enter their resize the text.
  + Guideline 1.4 of the web content of web content of accessibility guidelines 2.0(2008) states the included with the certain
* Operable: this guideline of the web content accessibility guidelines 2.0 states that that functionality must be useable by all users
  + Guideline 2.1 of the web content accessibility guidelines keyboarded accessibility says that all content must be accessible through the use of a keyboarded the site will be able to narrated though the use of keyboarded. In addition to this any controlled functionality such as form inputs will be able to be controlled by a keyboard.
  + Guideline 2.2 says enough time states to give users enough time to read content. They will be no timed content in this application
  + Guideline 2.3 of web content accessibility guidelines sizzles states that content must not be displayed in a way that can cause tizzies they will not be any content will not be cause squeezes there will no flashing content on this web application therefore this is not relative to this application to help user navigate to provide, this includes ways for skipping text that is repented on server pages and included in ma
  + Guideline 2,4 of web content accessibility navigable states there must be ways for the user to navigate locate this includes an way to skip any content that is repeated on several pages, having clear titles and making it clear in the name a link what the purpose of each link is.
* Understandable: this guideline of the content accessibly guidelines 2.0 states to portray and information in an understandable way. This includes of areas
  + Guideline 3.1 of web content accessibility guidelines states that it will be that content must be readable.
  + Guideline 3.2 of web content accessibility guidelines predictable says that content must be displayed in a predictable way. This includes narration appear on server pages to displayed in the same way and proving a way to propene ways, in order to achieve this they will be an option to display how to provoke words that are difficult.
  + Guideline 3.3 user input assistance says to provide help for user input this includes provide instructions when user input is needed and prove details of an error when atomically detected as well as provide suggestions. In order to achieve this in order to achieve this it’s this will the format of any form input will made clear and any errors that are ducted will be displayed to the user with suggestions on how to fix them.
* Robust: this guideline of the Web content accessibility guideline states that content must be perceived reliably by variety of users.
  + Guideline 4.1 of the web accessibility guidelines sates to make applications fully compile including with any assertive technologies , this includes ensuring any content that is developed with a mark-up language has appropriate start and end tags an content that is not developed by a mark-up language. To achieve this all HTML content within this application will have appropriative start and end tags.

Each page of this application will be uploaded to an accessibility checker to ensure the application is fully accessible. If it is found in the testing that anything is not accessible then appropriate changes will be made

This application will also need to apply with the data protection act. In order to do this the application will make it clear to users how their data will be stored.. In this guideline will make it clear to users what will happen with their data. This data will only be accessible to those that authorized to access, this will be password protected to prevent an unauthorised access to the system.

Cookies will be used to help this application manage logins. Users will be warned about the use of cookies and asked for consent, there for will be clear information on what cookies are, to ensure that all users are aware to what is happening with their information and that data is being stored on their devices. No cookies will be used for tracking purposes.

http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6547130http://moodle2.bolton.ac.uk/pluginfile.php/422475/mod\_resource/content/0/IEEE830%20SRS.pdf

# Appendix A BPNM.

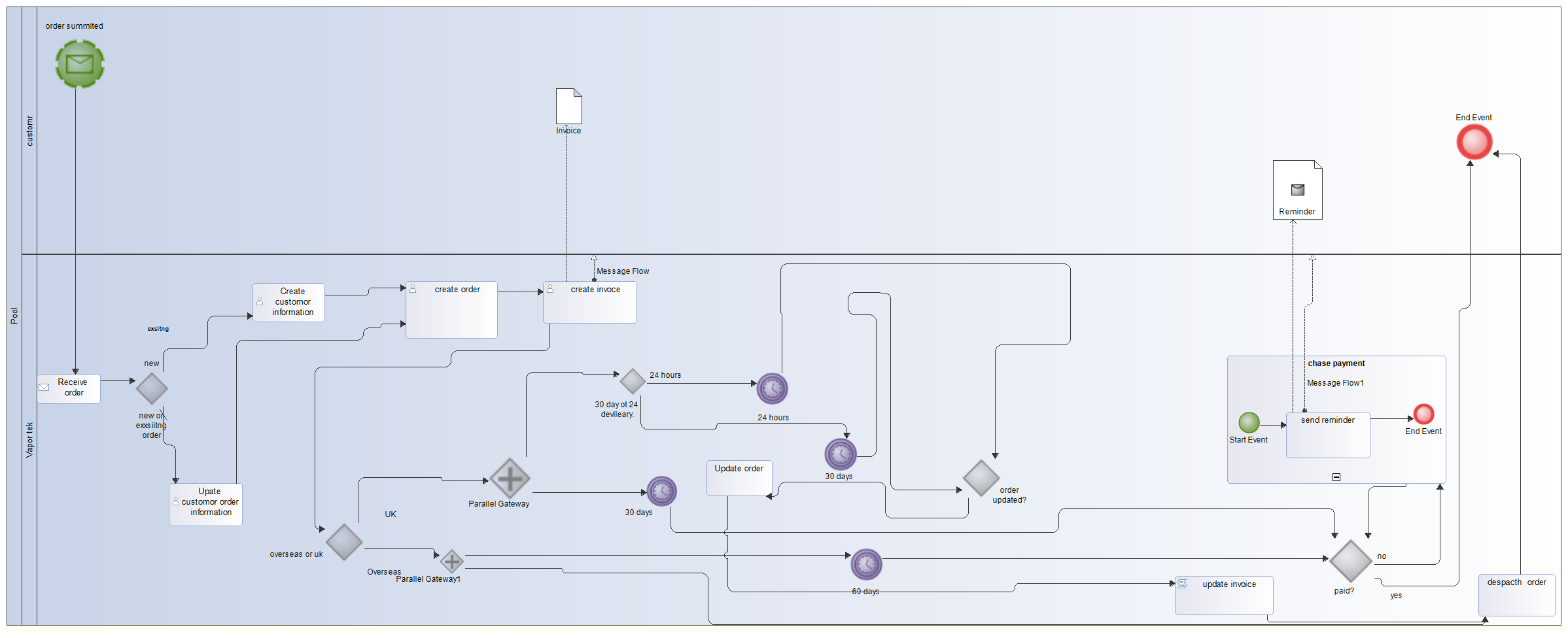


Figure BPMN of the Vapor-tek purchasing process

This shows that the process of the pushing process. There are two areas in this process the customer and vapor-tek. The event that starts the process of is order summited the order is made the customer the summiting of the order is a message event. This then goes to vapor-tek who receive the order this a receive task. One the order is received it is then checked whether the order is form a new or existing customer. If the order is form an existing customer then the customer order information is update. If the order is from a new customer then customer information is created. After this an order is created followed by creating and invoice. Once this is done it is checked whether the order is for within the UK or overseas. If the order is form overseas is then a timer of 60 days starts. Simultaneously with this the order is despatched. If the order is within the UK then a timer a timer of 30 days is attached. Concurrently with this it is checked whether the is 30 days or 24 hours, if the there is a 24 delivery then a timer of 24 hours starts, if there is a 30 day delivery then a timer of 30 days starts. After the timers are finished there is a check to see whether the order has been updated. If the order has not been updated then the order is despatched. If the order has been updated then there is task of updating the order followed by updating the invoice. After this the order is despatched. After the timers of 30 and 60 days following the check of UK or overseas are finished there is check of weather the order has been paid the process ends. If the order has not been paid for a sub process of chase payments starts. Within this process a reminder is sent to the customer. Once this process is finished it loops back to the payment check.

# Appendix B confirmation and expected mark.

This is all my own work. This is probably a C-B grade as it has described all the requirements that requirements that are needed for the system and there are use case provided showing the steps. However it does not contain a deployment diagram,. The BPMN provides an overview of the system, However could have gone into more detail. The entity relationship diagram is very simple and could have included more detail.

# Appendix C feature as described within the SRS.

|  |  |
| --- | --- |
| **Feature** | **Distribution** |
| Login. | Allows the user to log in. |
| Create order. | Allows an accounts user to create an order |
| Append order | Allows an accounts to update and order with more products |
| View previous orders. | Allows to accounts use to view previous orders and information regarding them. |
| Add new Customer | Allows a accounts administer to add a new customer to system |
| Add new user | Allows Administrators add a new user to the system |
| Prepare current orders | Allows the production manager to prepare current orders. |
| Enter dispatch details | Allows the production manager to enter despatch details of a Specific order |
| Cancel order. | Allows an accounts administrator to request a cancellation of an order. |
| Approve cancellation | Allows the administrator to approve an order cancellation |
| View invoice | Allows and account user to view an invoice |
| Request quote | Allows a customer to request a quote for a product. |
| Create quote | Allows an administrator to create a quote when one is requested. |
| Create CoA | Allows an administrator to create an certificate of analysis |
| View CoA | Allows a customer to view a certificate of analysis |

Figure 38 Overview of all features.

**Appendix D input/output format**

All inputs will be entered into a form, the forms will include mainly include text fields. In some areas the output will be in PDF format such the invoice.

**Appendix E problems solved by the SRS.**

The Vapour-tek pushing process is current not online. Meaning that invoices have to be sent out by post. The problem with not having an online system is that orders could take a long time to be processed, where with on online system part of the processing can be done automatically. As invoice has to be posted, it can take a long time to reach the customer and there is a risk of it getting lost. This problem will be solved by generating invoices in PDF format in order for the user to print, store and email them. Therefore, customer will be able to view the invoices as soon as they have made the orders made.