



**Software Engineering and Testing. BSC Year 2, 2020/2021
(Assignment 2 - 20%)**

Assessment 2: Requirements Document

**Submitted by: Jake Callow B00152286 / Lukas
Stoskus B00152862 / Callum Breen B00152224**

Submission date: 14/02/2024

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Ordinary Degree in Computing in the Institute of Technology Blanchardstown, is entirely my own work except where otherwise stated.

Author: Jake Callow

Dated: 07/02/2024

Author: Lukas Stoskus

Dated: 07/02/2024

Author: Callum Breen

Dated: 07/02/2024

Table of Contents

Contents

Title.....	3
Client.....	3
1. Project Overview (1 paragraph).....	3
2. Document Revision.....	4
3. Scope (max 1/2 page)	4
5.1 User Requirements	6
5.2 System Requirements	6
5.3 Non-functional Requirements:.....	6
6. Graphical User Interface Design.....	6
7. Technical Requirements and Feasibility:.....	7
8. Conclusion (1-2 paragraphs)	7

Title

Coffee Shop Online Supply Store.

Client

The client is Beans and Machines, a coffee shop that is looking for an online store to branch out their products and audience.

1. Project Overview (1 paragraph)

The project definition:

- The project is an online coffee supply store.
- The end product will be a website that delivers coffee supplies to customers.
- The main components of the product are a main screen for viewing and ordering products, a delivery section, and a payment page.
- The site will be used by customers who will view and order products from the main page. The customer will then choose their delivery address and options between recurring and non-recurring delivery. Finally, the customer will pay using the payment page and optionally apply any discounts they might have if they have a membership to the site.

2. Document Revision

Rev. 1.0 09/02/2024 – initial draft of requirements document.

Rev. 2.0 12/02/2024 – revised requirements document and added first draft of use case diagram.

Rev. 3.0 14/02/2024 – completed requirements document with new use case diagram.

3. Scope (max 1/2 page)

The product will include a product viewing page, a shopping cart, a login function, a delivery system, and payment processing.

Optional features of the product include a membership option for customers that will include discounts and recurring deliveries.

The product will not include two factor authentication, a returns or warranties system, or loyalty points.

4. Walkthrough Scenarios

The customer will access the home page where they can view the product page without needing to login. The customer can then add products to their cart, which will be stored in a separate page. When the customer accesses the shopping cart, they will be prompted to log in to the site or create an account. The customer will then access the payment processing page, where they will be prompted to enter their address and if they want to set up recurring deliveries. The customer will then get confirmation of their order.

5. Software Requirements Analysis:

Functional Requirements:

These are statements of services the system should provide – how the system should react to particular inputs and how it should behave in particular situations. Explicitly state what the system should do. Every major scenario should be represented by a use case. Diagrams are encouraged. UML Use case diagrams, Use case specifications (as legible screen dumps, typed listings or activity diagrams)

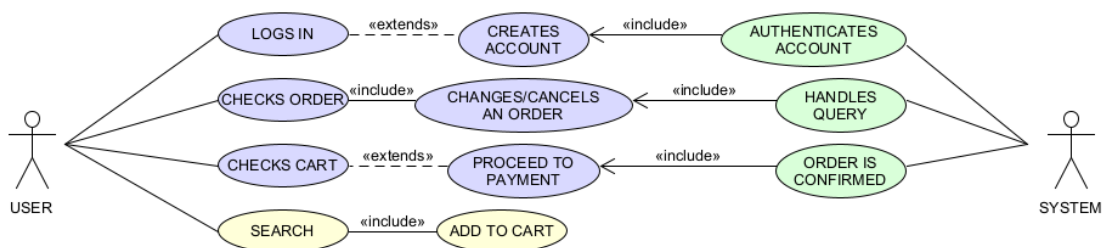
Can approach them from a *user* and *system* point of view.

User – high level abstract requirements, readable by someone with no detailed technical knowledge.

System – detailed description of what the system should do. Targeted at technical staff and project managers...

e.g.

Login:	<ul style="list-style-type: none"> - User logs in to database using an email and a password. - Can create an account if they do not have one. - Stores users login information in the database. - If the user logs in the database will check to make sure the credentials exist. - If the user creates an account the database will check to make sure the credentials do not exist.
Products Page:	<ul style="list-style-type: none"> - User can view products and add to cart. - Products that are not in stock should not be addable to the cart. - User can edit quantity before adding.
Shopping Cart:	<ul style="list-style-type: none"> - Shopping cart will show list of products user has added. - Accessible from any page from a pinned button at the top. - User must log in here if they have not already. - If user logs in it requests info from the database to ensure the user exists. - If not the user must create an account which will save to the database.
Payment Page:	<ul style="list-style-type: none"> - Payment page is accessed through the shopping cart if the user is logged in. - User will enter payment details. - If payment is successful, user will be taken to a confirmation page summarising their order and prompting them to continue shopping. - A request will be sent to the database to remove stock from the products the user purchased.



5.1 User Requirements

The user needs to be able to navigate the pages easily through a pinned menu. They should also be able to access a search bar to search for products. The quantity of product should be easily viewable and adjustable on the product page and then adding to cart should take one click. The shopping cart must be accessible from any other page on the site. The cart should clearly redirect to the payment page which will request the user's payment information. The user should receive confirmation that their order was successful.

5.2 System Requirements

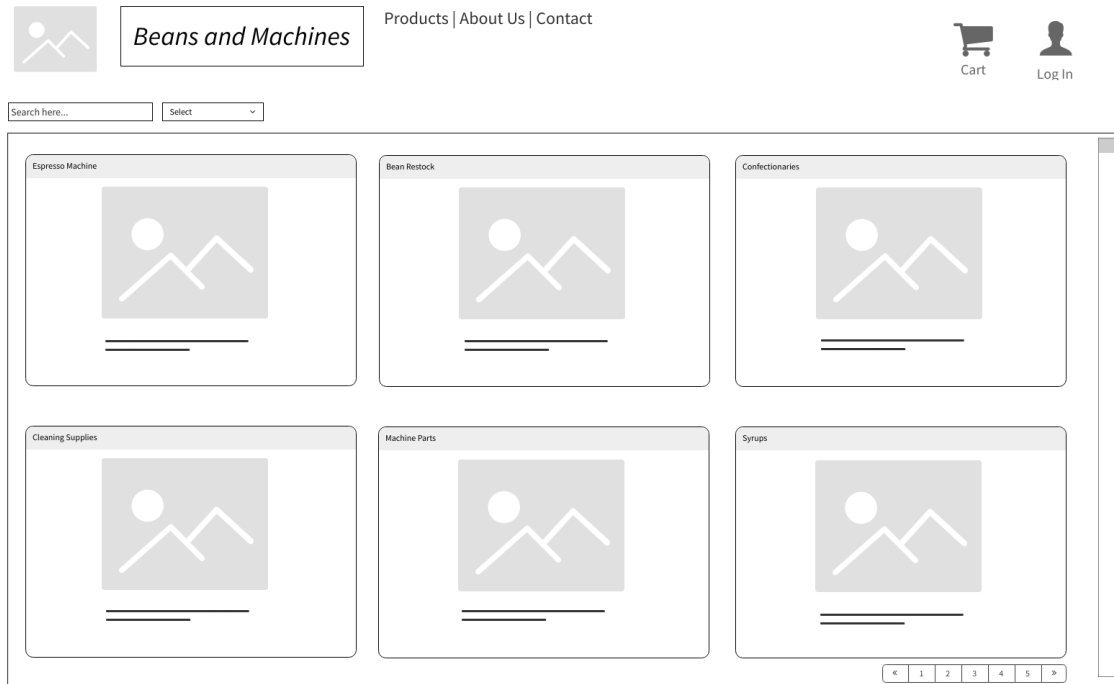
The system requires a functional database to store the data for all of the products as well as the delivery addresses for the customers. It requires a delivery system to be able to deliver products to customers. The system will also need a products page connected to said database which will display the products to the customer. It will require a login function for the user to store the info per customer. The database will store this login information.

5.3 Non-functional Requirements:

An optional requirement as an addition to deliveries would be the option for recurring deliveries. The system could also extend upon the login system to add a membership option for customers which will give discount offers.

6. Graphical User Interface Design

Our focus for the GUI Design is learnability and consistency. There will be a main page to the website where a list of products will be displayed. This page will also have an option to login or access the shopping cart, which are static and visible on every page. Clicking on a product will redirect to that product page where the details are displayed and the user can edit the quantity and add to cart. The shopping cart page will display a list of all products as well as the option to remove from the cart. If the user chooses to checkout they will be prompted to enter their payment information. Once payment information is entered the user will see a confirmation page summarising their order.



7. Technical Requirements and Feasibility:

We will be using Agile methodology to manage our project because it allows us to iterate our project, letting us break it into smaller pieces and test each piece individually. We will be using PHP as our scripting language as we can more easily interface with the database and display information to the web page. We will use MySQL Database for the database as it is easy to set up and migrate to a more scalable database in the future.

8. Conclusion (1-2 paragraphs)

In conclusion the required components are clear and quite feasible and are easily completable. The website will be easily usable by the customer and connected to a database to store information. Our optional requirements are also feasible and should be completed in time.

There is a consistent level of detail given towards each aspect of the project and all team members are confident in being able to complete the project in an organised and timely manner.

Checklist: Is your document complete and correct?

Content:

- Do the requirements state the customers' needs
- Are you satisfied with all parts of the document
- Do you believe all parts are possible to implement
- Is each part of the document in agreement with all other parts
- Do the requirements avoid specifying a solution
- Do the requirements avoid specifying a design

Completeness:

- Are all the necessary interfaces specified – this includes input and output
- Are the specifications precise enough
- Are all sections from the document template included – if changed, why?

Clarity:

- Are all requirements reasonable?
- Is the level of details for each requirements appropriate?
- Are the requirements written in a language appropriate to the reader?
- Are all items clear and unambiguous?