

Requirements Specification

Group two

N.Allen, A.Azeez, X.Chen, M.Grabara, M.Lee, W.Nujum, J.Tong, A.Wood.

May 13, 2019

Project name: CLUK Inventory Management

Prepared By: Group two

Email/Phone: m.lee10@newcastle.ac.uk

Document Version No: 01

Preparation Date: 13/12/2018

Document Version Date: 15/02/2019}

Ver. No.	Ver. Date	Revised By	Description
01	25/11/2018	M.Lee	Initial template and scope section.
02	29/11/2018	N.Allen	Assumptions section added.
03	30/11/2018	A.Wood	Personas section added.
04	04/12/2018	W.Nujum	Solution requirements section added.
05	06/12/2018	N.Allen	Constraints section added.
06	06/12/2018	M.Grabara	Hardware and software section added.
07	07/12/2018	W.Nujum, J.Tong, X.Chen	Initial UI design section added.
08	08/12/2018	A.Azeez	Background and analysis section added.
09	09/12/2018	A.Wood	Glossary section added.
10	13/12/2018	M.Lee	Corrections and first submitted draft.
11	05/02/2019	M.Lee	Formatting improvements.
12	07/02/2019	A.Azeez	Delivery systems updated.
13	08/02/2019	N.Allen	UML activity diagram added.
14	08/02/2019	W.Nujum	UML class diagram added.
15	08/02/2019	A.Wood	Use cases added.
16	12/02/2019	W.Nujum	Solution requirements updated.
17	14/02/2019	M.Grabara	Hardware and software updated.
18	15/02/2019	N.Allen	Further activity diagrams added.
19	20/02/2019	M.Lee	Updating and formatting document.

Contents

1	Executive Summary	3
1.1	Section summary	3
2	Domain Analysis	4
2.1	Delivery systems	4
2.2	Stock managements systems	4
3	Scope	6
3.1	In scope	6
3.2	Out of scope	6
4	Hardware and software platforms	7
5	Solution requirements	8
5.1	Functional Requirements	8
5.2	Non-Functional Requirements	8
6	Assumptions	9
7	Constraints and Dependencies	9
8	Project management and testing methodology	9
9	Personas	10
10	ER Diagram	10
11	Use Cases	11
12	UML Activity Diagrams	12
13	UML Class Diagram	15
14	Ideas for GUI	16
14.1	Main Login Interface	16
14.2	Manager Interface	16
14.3	New Order Interface	17
14.4	Reports Interface	17
14.5	Delivery Map	18
15	Gantt Chart	19
16	Glossary	20

1 Executive Summary

This project requires the design, implementation and testing of a stock management and delivery system for the local restaurant chain “Chicken Lovers United Kingdom” (CLUK). The project brief provided outlines a number of specific functions the system must perform, as well as operational conditions which must be satisfied. This document outlines how our system will be developed to satisfy each of these client requests.

1.1 Section summary

Below is a brief description of each section included in this document.

- **2 Domain Analysis:**
Details the domain analysis and research conducted by the group.
- **3 Scope:**
Specifies what is, and is not, relevant to the work of this project.
- **4 Hardware and software platforms:**
Outlines the hardware and software systems required for the system.
- **5 Solution requirements:**
Lays out the functional requirements and non-functional requirements of the system.
- **6 Assumptions:**
Summarises the assumptions the group has made with respect to the design brief.
- **7 Constraints and Dependencies:**
Explains the risks involved with the project and the conditions the group must work to.
- **8 Project management and testing methodology:**
Discusses how the project will be managed and tested.
- **9 Personas:**
Examines the requirements of potential users.
- **10 ER Diagram:**
ER diagram.
- **11 Use Cases:**
Use case diagrams for a range of potential users.
- **12 UML Activity Diagrams:**
Details the workings of specific functions.
- **13 UML Class Diagram:**
Outlines the class structure of the system.
- **14 Ideas for GUI:**
Shows the preliminary designs for the UI of the system, including mock-ups of various interfaces.
- **15 Gantt Chart:**
Outlining the timeline for the project.
- **16 Glossary:**
Defines terms used throughout the document.

2 Domain Analysis

To get a better understanding of what can be achieved with a delivery and stock management systems research was conducted into the commercial systems currently available on the market.

2.1 Delivery systems

One example of a commercial delivery system is Samsara Software which is a cloud-based fleet management solution that offers some similar features to our ideas such as GPS tracking, vehicle tracking, dashboard camera, routing and dispatch. A GPS tracker is installed in the driver's vehicle which provides real-time traffic updates and displays the best available route to drivers. It collects data on vehicle performance, breakdowns and fuel consumption to the back-office managers. The solution combines 'Hours of Service' with the GPS tracker and other sensors installed in the vehicle to record the driver's working hours for HoS logging [1].

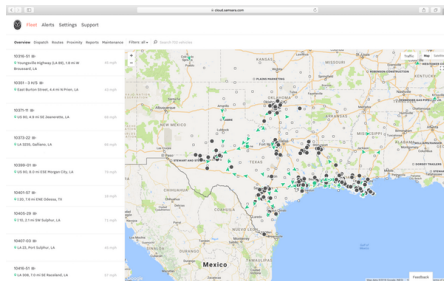


Figure 1: GPS Tracking

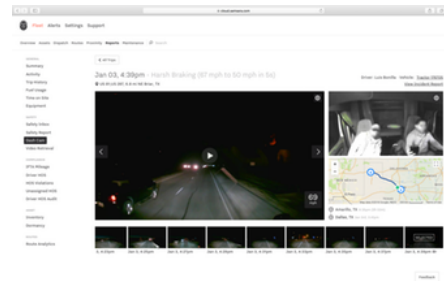


Figure 2: Dash-cam footage

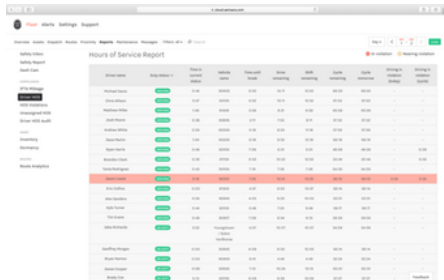


Figure 3: Hours of Service Reports

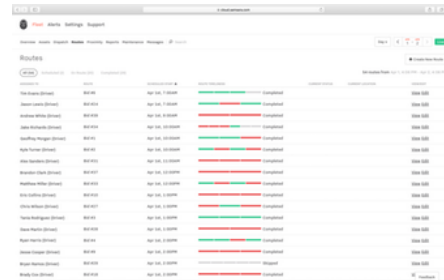


Figure 4: Route Scheduling

From our domain analysis, it was concluded that the formal route planning software market is heavily saturated. Options range from the most basic a to z planner all the way up to full fleet management solutions which allow for thousands of vehicles to be managed simultaneously. For this project, the system developed will be on the lightweight end of this spectrum. The client Chicken Lovers United Kingdom has a total of five locations within a 50-mile radius, it therefore would not benefit from the complex functionality of the larger packages.

2.2 Stock managements systems

Fishbowl InventoryTM is a commercially available stock management program that offers some of the functionality outlined in the project specification, and would be a direct competitor to our system. The

software's key features can be categorised into three sections: manufacturing, warehouse and asset tracking as can be seen in Figure 5.

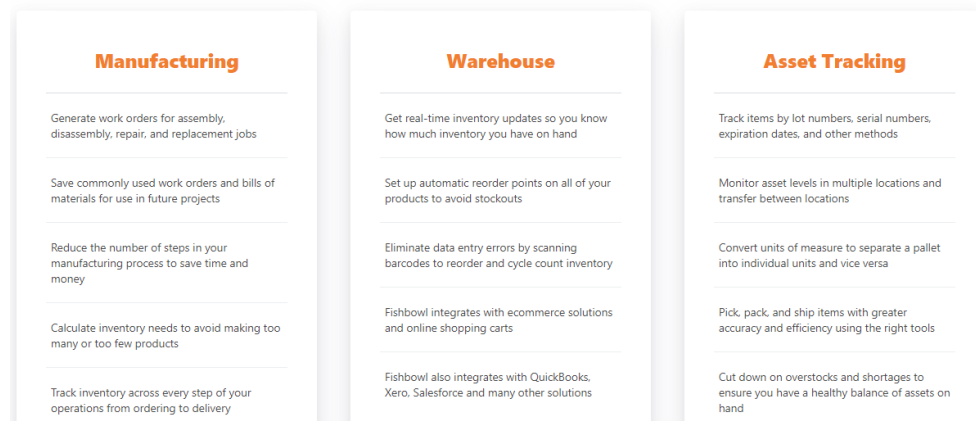


Figure 5: Fishbowl Inventory features [2]

Below are a number of screenshots from the Fishbowl inventory management system. [3] These will be used for inspiration and reference when designing our system.

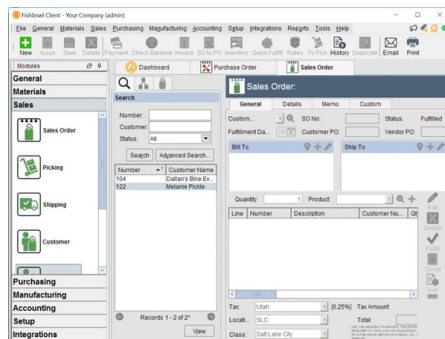


Figure 6: Admin

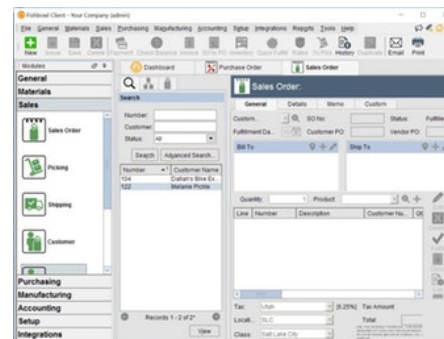


Figure 7: Sales Order

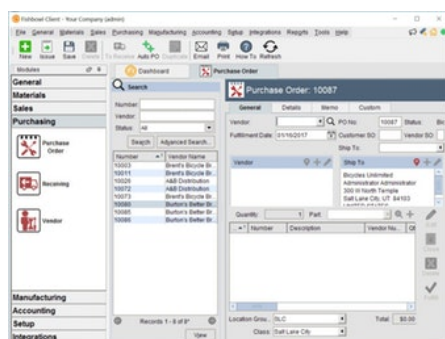


Figure 8: Manufacture Order

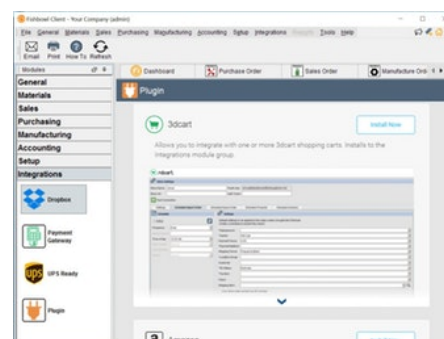


Figure 9: Shopping Cart plug-in

Another available software is Goods Order Inventory System (GOIS Pro) Software. GOIS Pro is a cloud-based inventory management solution that provides users with order management functionalities. It delivers to businesses of all sizes across various industry verticals.

GOIS Pro allows users to custom the rear camera of their mobile device to scan product barcodes and perform quick actions like stock check-in, stock check-out, ordering, transfers and adjustments. Inventory counts sync in real time, so they remain up-to-date in online and offline mode. It also provides functionality for users to receive notifications when stock runs low. Stock can also be automatically reconciled for damaged, missing or incorrectly entered items.

Sales and order management capabilities include customizable receipts, packing slips, order templates, invoice templates and status tracking. GOIS Pro also offers reporting and analytics, such as sales reports, profit and loss reports, stock projections and more [4].

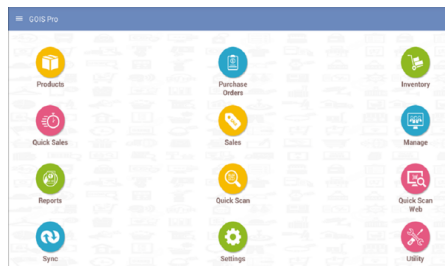


Figure 10: Home Dashboard

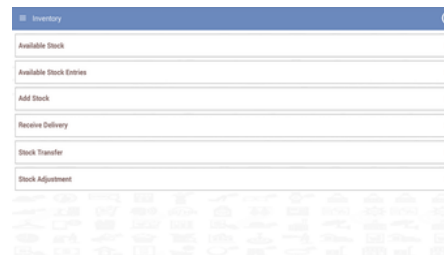


Figure 11: Inventory

3 Scope

3.1 In scope

Below is the project scope. This outlines the core functionality of the stock management system which must be present upon project completion. This is stated here to ensure that the project deliverables and boundaries are clearly understood before coding work begins.

- Design and implement a cloud based database to store the quantity of stock held at the main warehouse and restaurant venues.
- Develop a process to allow for the request of more/less stock to accommodate for the fluctuating sales of each restaurant.
- Create functionality that allows for the simulation of "using" stock as items on the menu are sold.
- Generate a delivery schedule as specified by constraints in the project brief.
- Develop a delivery route mapping system. This must include fuel optimisation to reduce drivers fuel consumption.

3.2 Out of scope

Below are a number of possible extensions to the stock management system. Although they are related to the project, they do not fall within its requirements. They are outlined here to clearly define the boundaries of this work.

- The project is concerned with the development of a stock management system. It doesn't cover the inner workings of the fast food business and so market research into this sector will not be conducted.

- The Stock management is an internal system for CLUK staff. It therefore will not include systems to allow for customers to purchase CLUK goods directly.
- Although the system will include functionality to allow for the simulation of "using" menu items it will not allow for automatic real time stock updates as food is sold over the counter.
- The database will be designed for a small fast food chain with five total locations as specified in the project brief. The system will be designed with a certain degree of expansion in mind, but it will not be suitable for a large scale chain.

4 Hardware and software platforms

Our solution will be built using open-source web technologies. Data will be stored in MySQL database hosted serverless in the cloud. This allows to build scalable application without the need to purchase on-premise server hardware. This means that the customer does not need to arrange additional secure, air-conditioned space to accommodate server hardware [5], neither needs to upgrade their broadband internet package to adjust for increased data transmission between all the restaurants and the central warehouse.

The Warehouse and the restaurants need, if they do not have it already, broadband internet access. It is estimated using of the system should not require higher synchronous bandwidth than 15 Mb. If any of the venues use internet connection for other purposes, e.g. online marketing and everyday office tasks, higher bandwidth might be required.

Cloud hosting services guarantee uptime and provide redundancy unachievable for any on-premise data storage and hosting solution targeted for small and medium enterprises [6]. Power or internet outage at the central warehouse will not cause interruption of the services provided to the restaurants, thus will not result in downtime for any point of sales. Backup tasks can be automated in the cloud environment, thus reducing the potential downtime after and effects of data corruption due to application errors or and external factor.

Software itself will be provided as a web service built on Angular 7 framework and Node.js 8 LTS platform. This way, platform-independence can be guaranteed, allowing for significant savings on the client devices and its maintenance. Updates to the system could be rolled-out instantly without the need to access client devices.

The warehouse and each of the venues that will use the software will need to be equipped with devices capable of running the application. Because of the above-described nature of the proposed solution, any device powerful enough to run a Chromium-based web browser and capable of scheduled unattended system and web browser updates will meet the compliance criteria. This includes, but is not limited to, all personal computers, vast majority of tablets and smartphones offered on the European market.

Minimum hardware required to support the system is a personal computer with at least 2 GHz Intel Core i3 CPU, 4 GB of RAM, integrated graphics and standard HD-ready screen. Input can be supported by mouse and keyboard or a touchscreen. Alternatively, a 10-inch Android or iOS tablet with HD-ready screen, quad-core 1.8 GHz ARM-compatible CPU and 2 GB of RAM can be used.

The user interface will be designed using Angular Material library. It helps designing aesthetic user interfaces in no time and is deeply integrated with Angular framework [6].

5 Solution requirements

Below are the functional and non functional requirements for our system. Each requirement has its priority listed. H indicates High priority, M medium priority, and L low priority.

5.1 Functional Requirements

	Requirement	Priority
01	Keep track of all stock held at the central warehouse.	H
02	Keep track of all stock held at each restaurant location.	H
03	Restaurants must not receive more than 2 deliveries per week.	H
04	Restaurants must have at least a 3 day gap between deliveries.	H
05	Update restaurant stock when it has received a delivery.	H
06	Retrieve meal details from the database.	M
07	Allow restaurants to request standard and custom orders.	H
08	Allow warehouse and restaurant to set and update a minimum stock level.	M
09	Allow warehouse and restaurant to display a warning if stock drops below minimum level.	H
10	Create meal items from the restaurant menu and reduce stock accordingly.	H
11	Retrieve price of meal items from the database.	M
12	Display today's orders to the restaurants and drivers.	M
13	Retrieve stock item details from the database.	M
14	Allow warehouse to approve and decline custom orders.	L
15	Send orders to restaurant and update warehouse stock accordingly.	H
16	Generate a graphical display for stock sent to each restaurant.	L
17	Display menu items on a graphical point of sale interface.	L
18	Display currently pending orders.	L
19	Plot a route for the day's deliveries.	H
20	The delivery route must be the most efficient route taking into account fuel consumption.	H
21	Routes over 4 hours must include a 45 minute break. Routes cannot be over 9 hours.	H
22	Create and delete system accounts.	M
23	Set, update and check account permissions.	L
24	Display staff information.	M
25	Login and logout.	H
26	Connect to the MySQL database.	H

5.2 Non-Functional Requirements

	Requirement	Priority
01	Accessibility and usability: The system will be designed with respect to Nielsen's Heuristics in user interface design. The system shall be accessible to people with disabilities such as colour blindness.	M
02	Confidentiality and Security: The system will have a permissions system for function access. i.e only drivers will be displayed route maps. The access permissions for the system may only be changed by accounts with manager permissions.	H
03	Availability and flexibility: Using both layout and platforms that ensure the App is flexible to use on different Web and Mobile cross-platforms such as Google Chrome or IOS.	M
04	Safety: The system will separate the frontend actions from the database. This will protect against potential attacks such as SQL injection.	H
05	Quality and Accuracy: The system must behave exactly with the user input and modulate its states precisely.	M
06	Safety: The system must be created, implemented and tested and documented by 7th May 2019.	H

6 Assumptions

A summary of the assumptions we have made for the CLUCK Stock Management are as follows:

- All deliveries of stock come from one central warehouse in Chester le Street.
- All deliveries of stock go to restaurants within a Restaurants in 50 mile radius.
- The stock items only include items listed in the project brief document (for example, the only beverage is Cola).
- Each restaurant only receives 2 deliveries per week.
- Deliveries only occur between 07:00-20:00 during the working week (Monday-Friday), with a minimum of 3 days between each delivery.
- The Chester-Le-Street central warehouse receives one delivery per week from CLUK's suppliers, and only keeps enough stock in the warehouse to supply each restaurant for the two specified deliveries per week.
- The warehouse is open to delivery drivers from 05:00 –21:00.
- Restaurants in the town centres will require more stock, due to larger sales volume.
- There are enough delivery drivers to deliver to all the stores.
- The drivers can only work for a max 10 hours per day, and need to take a 45-minute break after driving for 4.5 hours.

7 Constraints and Dependencies

- The group project is due for completion in May 2019. The specification document will be used to constrain the scope of the project, and every effort will be made to ensure all the non-functional and functional specification elements are delivered within the required time frame. However, if unanticipated delays do occur, then sections of the project may need to be revised based upon meeting the core functionality required.
- The development phase of the project will involve ongoing testing and evaluation by the project team. It is also our ambition to include a broader analysis of the system's performance by using an independent focus group of users to ensure objective testing of the system.
- As the project is dependent on hardware that is not owned or maintained by ourselves, then the team cannot be held responsible for technical issues that may arise during the lifespan of this project. However, the risk of a significant and or critical failure of the IT infrastructure is unlikely.
- Another potential risk is through absence and illness of team members, and the impact that can have on work flows and timescales. The group aim to develop a work breakdown structure (WBS) and a Gantt chart of major tasks and milestones to be achieved during the development and testing phase of the project.

8 Project management and testing methodology

Throughout this project we will be using the agile method for project management. The system requires a large number of modular components that can each be developed, implemented and tested individually. The agile method works well for such systems. This approach will involve periods of testing after each piece of functionality has been developed. The testing will be conducted using black box testing.

9 Personas

It is important to outline which users will have access permissions to specific functionality of the system.

The warehouse staff will need to have knowledge about the incoming and outgoing of stock from the warehouse. Therefore, they will have access to the warehouse and the supplies within it and the organisation of shipments from the warehouse.

The only access that the drivers need is information on what stock type and quantity to pick up, where to deliver them and the time period they need to deliver the stock by.

The restaurant staff can be split into smaller groups including the management staff and cashier staff for each restaurant separately. The management staff have knowledge of the delivery times of stock, as well as information on when the stock is running low. The chef will have access to information regarding the incoming stock, including time of delivery and quantity. They will also have access to the order ID and times that the meal needs to be produced by. The cashier staff will have minimal access, only having knowledge of what the order ID is. They are unable to make any changes and will have read only permission.

The finance/admin team and the IT staff are included in the head office staff. The admin team have access to all operations but will have read only permission. They are not allowed to execute any changes. The IT staff have full access to the system and are able to execute changes to any operation. They will overview all the procedures.

10 ER Diagram

Below is the ER diagram that will be used in the development of the system database.

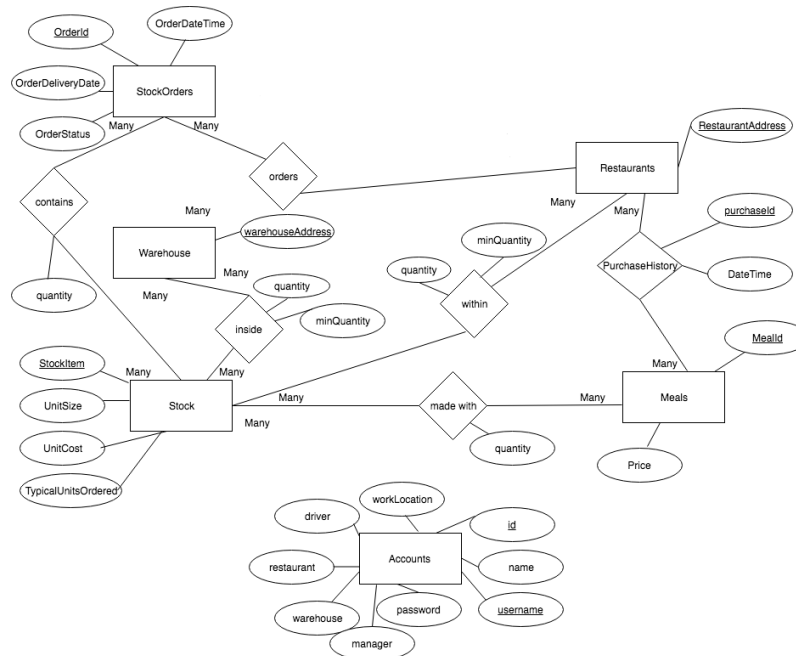


Figure 12: ER Diagram

11 Use Cases

Below are a number of use case diagrams. Each shows how a certain user will interact with the system.

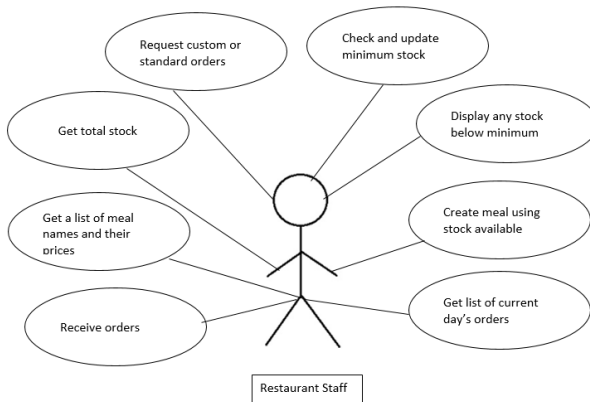


Figure 13: Restaurant Staff

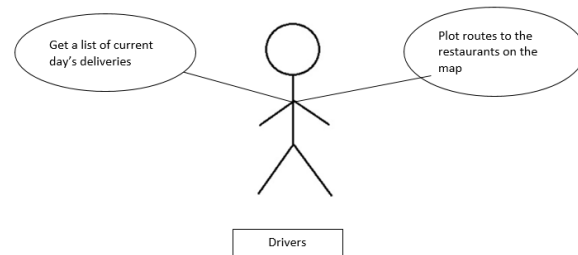


Figure 14: Drivers

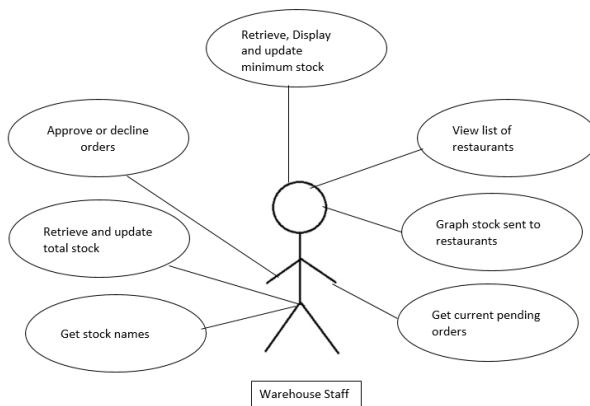


Figure 15: Warehouse Staff

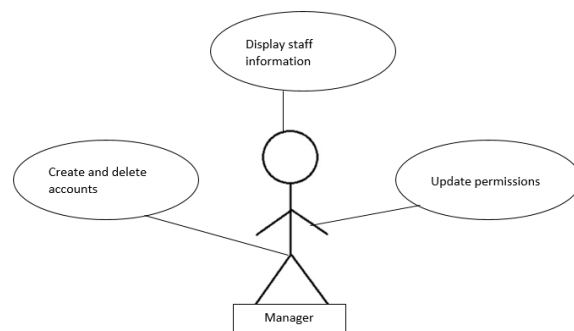


Figure 16: Manager

12 UML Activity Diagrams

Below are UML activity diagrams outlining how a number of our systems functional requirements will work.

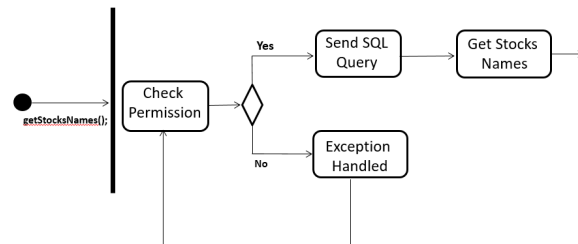


Figure 17: Get Stock Names

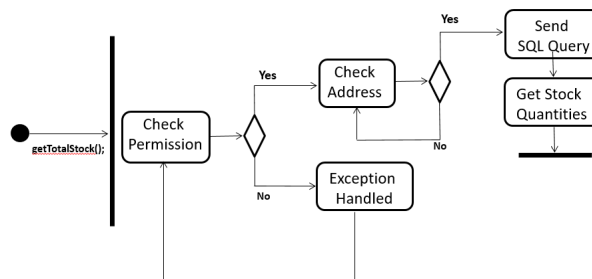


Figure 18: Get the total Stock of central Warehouse Address

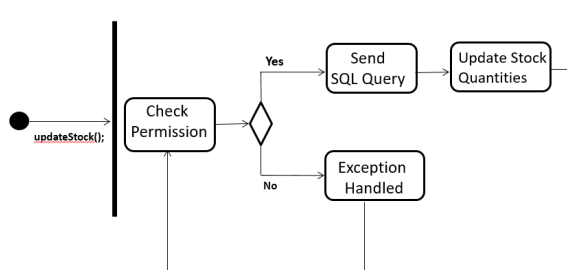


Figure 19: Update Stock

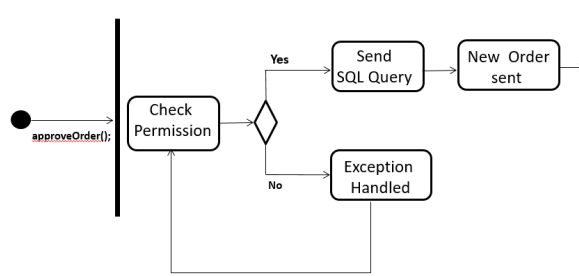


Figure 20: Approve Order

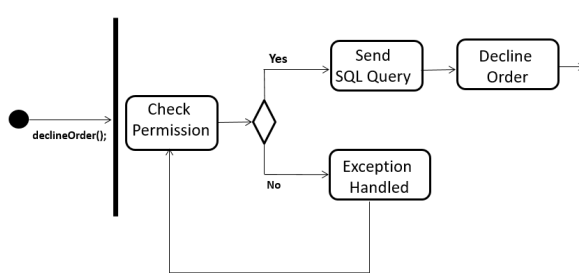


Figure 21: Decline Order

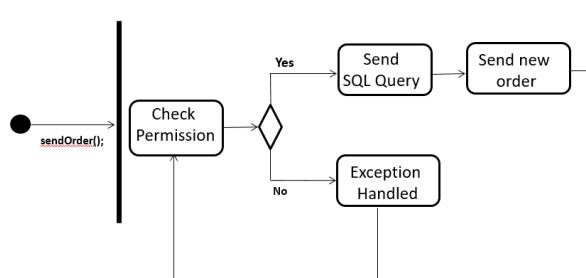


Figure 22: Send Order

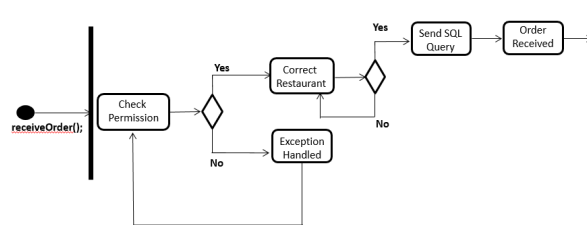


Figure 23: Receive Order

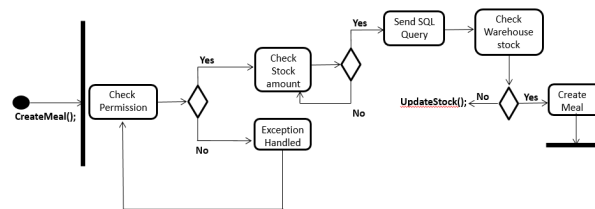


Figure 24: Create Meal

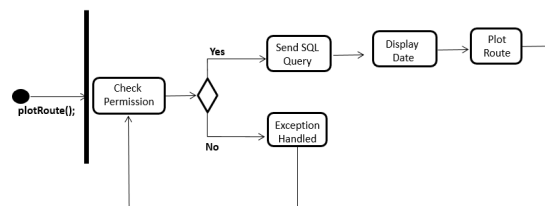


Figure 25: Plot Route

13 UML Class Diagram

Below is a UML class diagrams for our proposed system.

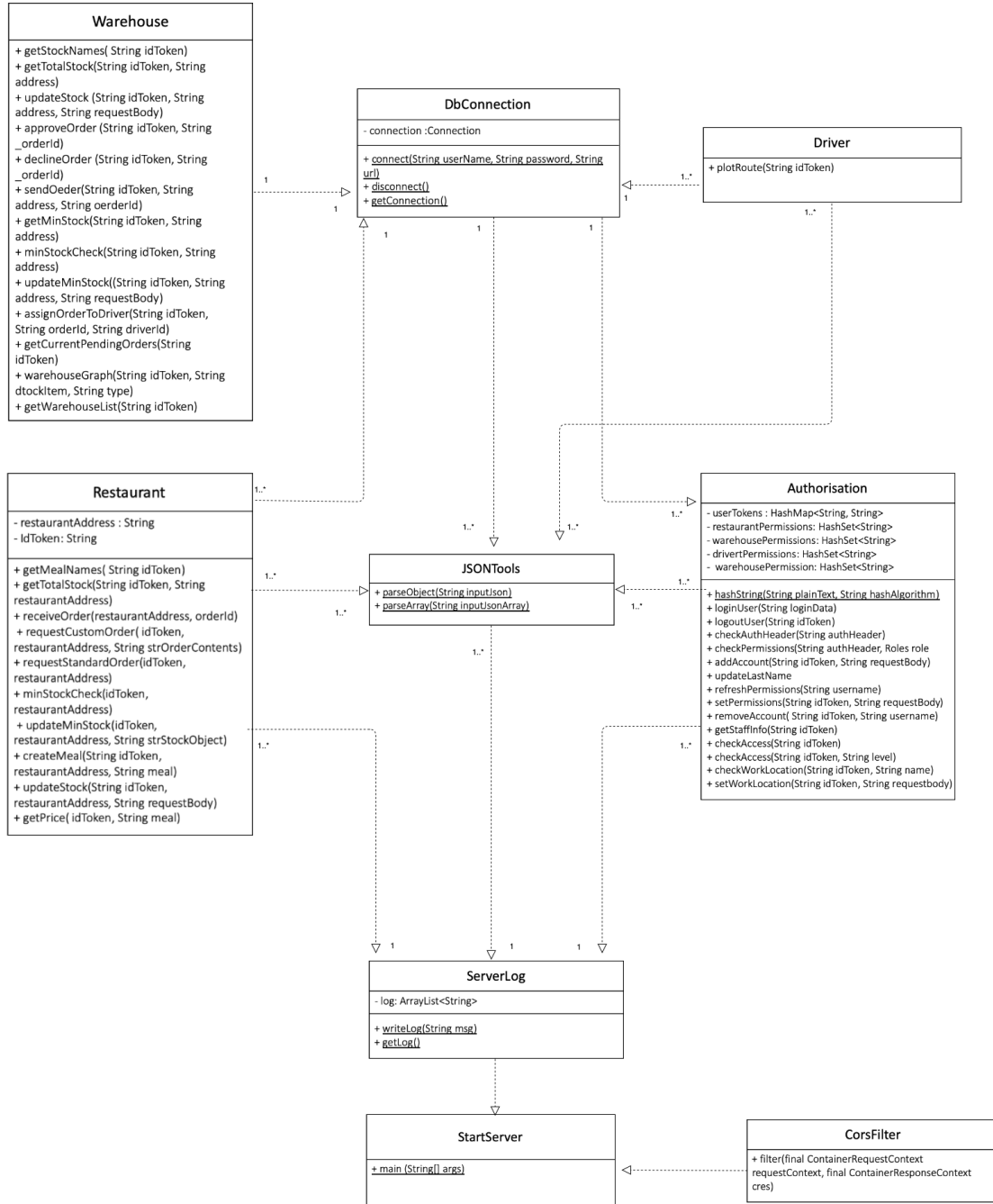


Figure 26: UML Class Diagram

14 Ideas for GUI

14.1 Main Login Interface

Below is a mock up of the main login page for the system. After a user enters their login, the system will display an interface showing options available to the user according to their access rights.

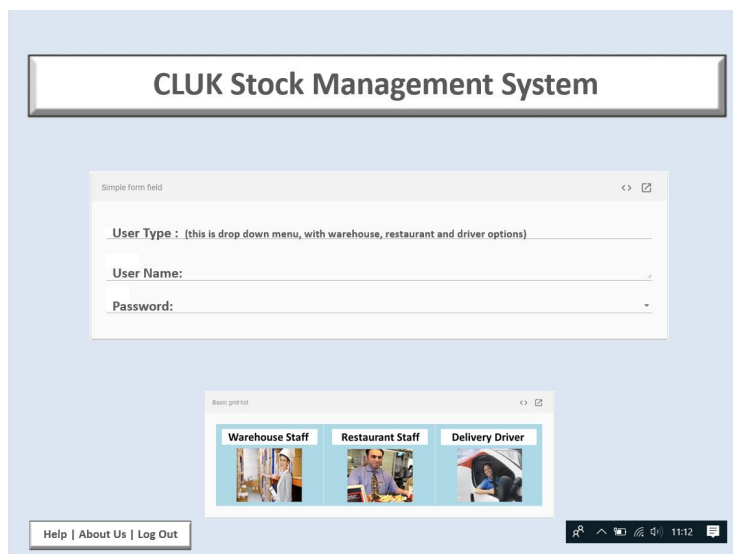


Figure 27: Main Login Interface

14.2 Manager Interface

Below is a mock up of a managers main interface. The left hand side displays the options available to this user.

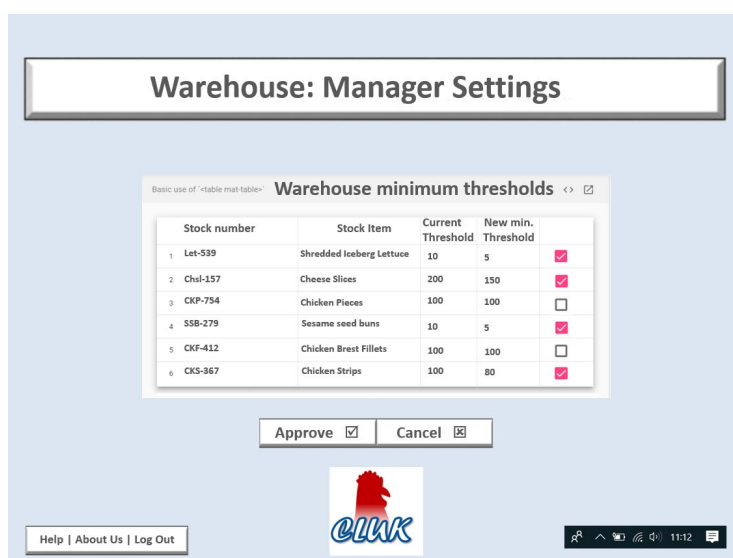
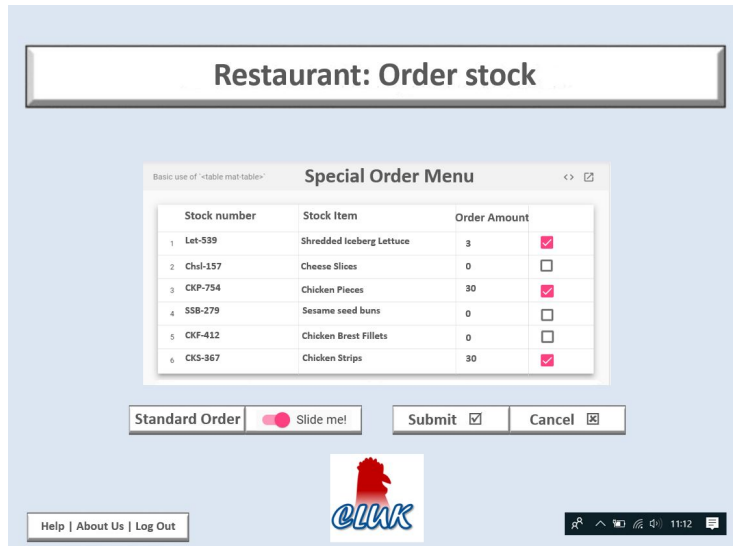


Figure 28: Manager Interface

14.3 New Order Interface

Clicking on the new order button will bring up a screen where the details of the order can be filed.



Stock number	Stock Item	Order Amount	
1 Let-539	Shredded Iceberg Lettuce	3	<input checked="" type="checkbox"/>
2 Chsl-157	Cheese Slices	0	<input type="checkbox"/>
3 CKP-754	Chicken Pieces	30	<input checked="" type="checkbox"/>
4 SSB-279	Sesame seed buns	0	<input type="checkbox"/>
5 CKF-412	Chicken Breast Fillets	0	<input type="checkbox"/>
6 CKS-367	Chicken Strips	30	<input checked="" type="checkbox"/>

Figure 29: New Order Interface

14.4 Reports Interface

The reports screen allows a user to select from a range of reporting options.

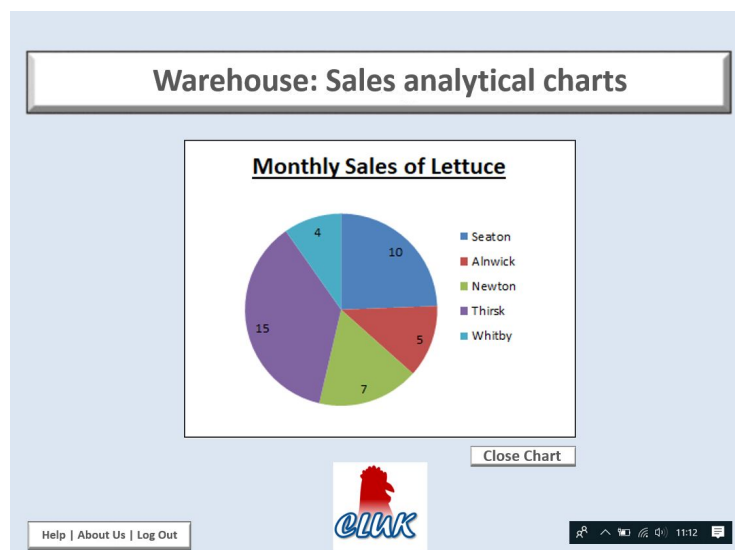


Figure 30: Reports Interface

14.5 Delivery Map

The Locations interface displays a map of current store locations.

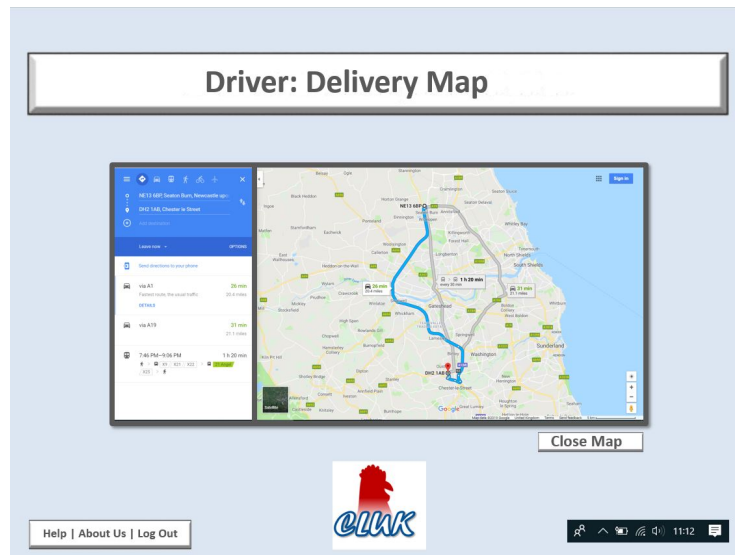
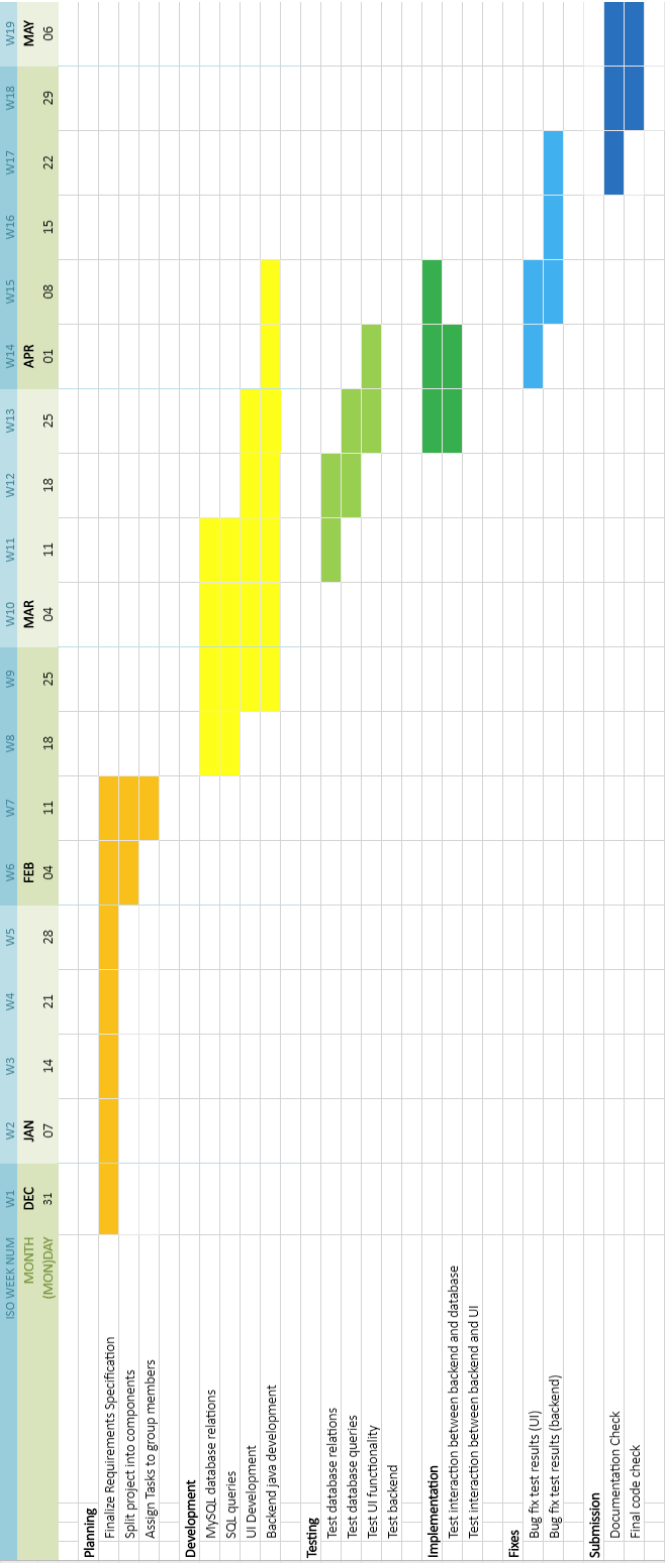


Figure 31: Delivery Map

15 Gantt Chart



16 Glossary

- **Angular 7 framework:** JavaScript-based open-source front-end web application framework mainly maintained by Google and by a community of individuals and corporations to address many of the challenges encountered in developing single-page applications.
- **Angular Material library:** A mature and stable product that is ready for production use.
- **Broadband internet package:** Internet package that allows uninterrupted and unlimited internet access without data transfer limits at guaranteed minimum or average speed.
- **Chromium-based (web browser):** An open source program for accessing the World Wide Web and running Web-based applications.
- **Client devices:** Devices used to access the application, installed in the warehouse and restaurants.
- **Cloud hosting:** Hosting service provided on virtual servers sharing their computing resources between data centres in various locations within the region (e.g. European Union) or globally.
- **Database:** A structured collection of data organized in a way to support accessing this data.
- **Gantt chart:** A chart in which a series of horizontal lines shows the amount of work done or production completed in certain periods of time in relation to the amount planned for those periods.
- **Graphical User Interface (GUI):** A user interface that allows users to interact with the system using images rather than text commands.
- **Interface:** A device or program enabling a user to communicate with a computer.
- **JavaScript:** High-level, interpreted programming language designed to enhance websites and currently also used to build web applications implementing ECMAScript international standard.
- **MySQL:** Open source relational database management system that runs as a server providing access to a number of databases.
- **Node.js:** An open-source, cross-platform JavaScript run-time environment that executes JavaScript code outside of a browser.
- **Open-source:** The program's source code is freely available to the public.
- **Platform-independence:** Feature of a software solution that allows to use the solution on any compatible device, regardless of its operating system and hardware architecture.
- **Personas:** Fictional characters created to represent a user type that might have access and use the system in a specific way.
- **Scalable application:** Application capable of dynamic adjustment of computing and memory resources, depending on the changing flow of incoming data.
- **Server-side:** Executed on the side of the server, so that the end-user has no access to the executed code or processed data and can only see result of the computation as received from the server.
- **Stock:** The goods or merchandise kept on the premises of a shop or warehouse and available for sale or distribution.
- **Web browser** Software solution used to access the World Wide Web.
- **Web service** Online service built using web technologies.
- **Work Breakdown Structure (WBS):** A deliverable-oriented breakdown of a project into smaller components.

References

- [1] Retrieved February 15, 2019, from <https://www.softwareadvice.com/fleet-management/samsara-profile/>
- [2] Retrieved February 15, 2019, from <https://www.fishbowlinventory.com/>
- [3] Retrieved February 15, 2019, from <https://www.softwareadvice.com/uk/inventory-management/fishbowl-inventory-manufacturing-profile/>
- [4] Retrieved February 15, 2019, from <https://www.softwareadvice.com/uk/inventory-management/gois-pro-profile/>
- [5] Mike U. (2015). Checklist For Designing a New Server Room - IT Peer Network. Retrieved December 12, 2018, from <https://itpeernetwork.intel.com/checklist-for-designing-a-new-server-room/>
- [6] Benefits of cloud computing — IBM Cloud. (n.d.). Retrieved December 12, 2018, from <https://www.ibm.com/cloud/learn/benefits-of-cloud-computing>