

FINAL REPORT

Reengineering business processes for a local business

Name: Arsalan Sadeghpour

Candidate Number: 105977

Degree: BSc Computing for Business and Management

Department: Informatics

Year: 2017

Supervisor: Dr Natalia Beloff



University of Sussex

STATEMENT OF ORIGINALITY

This report is submitted as part of the requirement for the degree of BSc Computing for Business and Management at the University of Sussex. It is the product of my own labour except where indicated in the text. The report may be freely copied and distributed provided the source is acknowledged.

ACKNOWLEDGEMENTS

I would like to acknowledge my project supervisor Natalia Beloff for her significant amount of assistance throughout the undertaking of this project. I would also like to thank the restaurant manager at CouCou Food for giving me the opportunity to undertake this project and for acting as my client whilst professionally communicating his requirements and indicating his preferred design requirements. I would also like to thank for the staff at CouCou Food for contributing to the development of this project and the individuals who provided useful feedback during the project's design stage.

TABLE OF CONTENTS

Introduction -----	-4
Problem Area -----	-4
Project Objectives -----	-5
Professional Considerations-----	-6
Requirements Analysis-----	-7
Interview With Restaurant Manager -----	-7
Evaluation of Existing Mobile Applications -----	-9
The Importance of Business Intelligence -----	-19
Evaluation of Existing Restaurant Analytics Software -----	-20
Use Case Diagram -----	-22
Requirements Specification-----	-23
Functional Requirements -----	-23
Non Functional Requirements -----	-27
Software Design-----	-28
Model View Controller Architectural Design Pattern-----	-28
Model -----	-29
Controller -----	-30
View -----	-34
Prototype Design Feedback -----	-36
Implementation-----	-37
Model - MySQL -----	-37
View - Bootstrap-----	-37
Controller - PHP -----	-38
Client Side Validation -----	-38
Server Side Validation -----	-39
Password Security-----	-39
Session Management -----	-41
PHP and MySQL Database Connectivity-----	-43
AJAX and Document Object Model Manipulation -----	-47
Testing-----	-53
The Testing Process -----	-53
Unit and Integration Testing Results Summary -----	-53
User Acceptance Testing Results Summary -----	-53
Conclusion -----	-54
Process and Product Objective Evaluation -----	-54
Project Improvements -----	-54
Skills Demonstrated and Acquired -----	-55
Bibliography -----	-56
Appendix A - Project Log-----	-59
Appendix B - Additional Sequence Diagrams -----	-64
Appendix C - Additional Prototype Designs -----	-73
Appendix D - Questionnaire Results -----	-88
Appendix E - Test Plan Results -----	-96
Appendix F - Ethical Compliance Form-----	-109

INTRODUCTION

Problem Area

Business processes as defined by Davenport and Short (1990, p.12) are '*a set of logically related tasks performed to achieve a defined business outcome for a particular customer or market*'. Business process reengineering is therefore concerned with the redesign of these processes. Gunasekaran and Nath (1997,p.92) state "*Re-engineering is not about fine tuning or marginal increases*" rather it is about transforming processes to "*improve performance significantly*" therefore a business process reengineering strategy should involve redesigning ineffective business processes to better incorporate strategic alignment through tackling objectives such as increasing competitiveness, efficiency, customer service and brand loyalty whilst also reducing costs. In the journal 'The role of information technology in business process reengineering' Gunasekaran and Nath (1997,p.94-95) state "*any reengineering program must take account of the tremendous advantage offered by technologies*" and that technologies can be used to "*remove much of the human error in inherent and incomplete complex and repetitive tasks*".

This project involves developing a business process reengineering strategy for CouCou Food (a local restaurant business in Brighton and Hove) who currently offer their own website allowing customers to order for delivery and collection whilst also utilising the Just Eat platform, who are an online restaurant delivery service provider acting as a third party between customers and restaurants. Just Eat allows restaurants to promote themselves to customers and offer a delivery and collection service via their web and mobile application infrastructure. CouCou's use of Just Eat has led to increases in brand awareness and sales through gaining a larger customer base alongside continuous quality improvements through using Just Eat's customer review system, however the utilisation of Just Eat's platform is proving costly due to the 14% commission taken on sales as stated by Just Eat (2016) which may hinder their long term plan for growth as they plan on opening a 2nd branch within the area.

CouCou faces a further problem of increased competition from market leaders such as Domino's Pizza and Papa John's who operate in the same road and are leveraging their own mobile application infrastructures integrating live order tracking features alongside a loyalty programme and the provision of special offers. Domino's' Annual Report & Accounts (2015) consisted of '*e-commerce sales representing 77.7% of all delivered sales*', whereas Business Insider (2016) stated Papa John's also announced in the first quarter of 2016 '*55% of its total sales now come digitally along with 60% of those digital orders come from mobile*'. Evan Bakker a research analyst for Business Insider's premium research service BI Intelligence produced a report detailing the mobile app industries future growth and states "*mobile ordering platforms have been proven to intensify customer loyalty and increase purchase frequency*". Just Eat do not provide a highly detailed live order tracking feature in their app compared with Domino's but are also successful at using this platform for revenue generation, according to Just Eat plc's Annual Report & Accounts (2015) their total revenues were '*£247.6 million (up 65% from the previous year) along with 13.4 million active users (up 65% from the previous year)*'. Overall the growth of mobile applications is proving to be a successful model for increasing sales and satisfying customers by the provision of an enjoyable and rewarding experience and has the potential to form a basis for developing loyalty and repeat business.

Project Objectives

Mobile Application

Provide a mobile application that allows customers to order for delivery and collection whilst also providing a live order tracking feature, an integrated loyalty programme and a customer review system allowing customers to review their order and receive feedback from the business. The aim is to meet objectives of providing a more enjoyable and rewarding experience to compete with market leaders such as Domino's, Just Eat and Papa John's alongside continuously improving product quality through gaining customer feedback from the review system and reducing costs through withdrawing from Just Eat's platform.

Business Interface

In terms of order management the aim is to provide restaurant staff with the ability to update the status of customer order's via a business interface which is then updated and displayed to the customer via the mobile application's live track order page.

The ability to respond to customer reviews will form the Customer Relationship Management component of the system. The purpose of Customer Relationship Management systems according to Jeston and Nelis (2014, p. 11) is to have "*extensive focus on the customer view and customer experience*" which the ability for customers to communicate their food quality experience and for the staff to respond to their experience brings, as it will allow CouCou to identify whether the food quality leads to a higher quality customer experience or not.

CouCou is a growing small business and thus will acquire a lot of customers and accumulate a lot of order data and these customers may also submit reviews which also leads to an accumulation of review data. A long term objective would be to plan for the system to integrate a business intelligence component into the interface that will allow staff at CouCou to view the demand of popular menu items and base this performance data on their ingredient reordering process to prevent wastage in terms of oversupplied/undersupplied ingredients, reduce costs and increase the efficiency of internal business process which will then lead to an increase in customer satisfaction through the constant availability of products as CouCou's customer base increases.

The reason for developing an app relying on CouCou's infrastructure is due to the fact it is easier to gather analytics data in the long term from their own infrastructure as opposed to integrating the system with Just Eat's platform and extracting analytics data which could prove difficult. The reason for developing a Review and Order Management system is that given CouCou wants to leave Just Eat's infrastructure they wish to keep the benefits realised from the Just Eat service such as Review Management.

PROFESSIONAL CONSIDERATIONS

British Computer Society are a professional body providing a code of conduct outlining the required professional standards of IT professionals. The aspects of the code applicable to this project are specified below.

1a) "have due regard for the public health, privacy, security and wellbeing of others" (British Computer Society, 2016)

The storage of sensitive data such as names, email addresses, contact numbers, delivery addresses and payment information gives rise to security concerns such as the possible unauthorised access of this data by malicious third parties compromising privacy rights. Customers must be reassured that their data will not be passed onto third parties and will be securely stored. Customers must be notified that their order history will be used for analytics purposes. If customers deregister from the mobile application then their personal data shall be deleted from the database.

1d) "promote equal access to the benefits of IT and seek to promote the inclusion of all sectors in society wherever opportunities rise" (British Computer Society, 2016)

Customers with disabilities such as blindness, difficulties with their eyesight and users who are not as proficient with technology may use the application, giving rise to accessibility concerns. Suitable colours and text sizes along with a simple tutorial should be included to prevent these users from being excluded.

2a) "only undertake to do work or provide a service within your professional competence" (British Computer Society, 2016)

The project shall be implemented using familiar technology in order to produce a high quality system. If unknown technologies are to be considered, then they shall be thoroughly researched before implementation to prevent unexpected consequences.

2d) "ensure that you have the knowledge and understanding of legislation and that you comply with such legislation in carrying out your professional responsibilities (British Computer Society, 2016)

Due to the storage of sensitive customer data it is necessary to comply with the Data Protection Act of 1998. (Gov.UK,2016) indicates data should be "kept safe and secure", "used fairly and lawfully" and "kept for no longer than is absolutely necessary".

2e) "respect and value alternative viewpoints and seek, accept and offer honest criticisms of work" (British Computer Society, 2016)

The client may criticise the suitability of the system produced so far and offer alternative ideas which should always be respected and be given a fair assessment in terms of criticism and acceptability.

3a) "carry out your professional responsibilities with due care and diligence in accordance with the Relevant Authority's requirements whilst exercising your professional judgement at all times" (British Computer Society, 2016)

The highest level of professionalism shall be maintained during communication with the client.

4f) "encourage and support fellow members their professional development" (British Computer Society, 2016)

Upon completion, the client will be given a clear understanding of the system in terms of training and be offered continuous support when required.

REQUIREMENTS ANALYSIS

Interview With Restaurant Manager

The first stage of the requirements elicitation process is to conduct an unstructured interview consisting of open ended questions in order to clarify the software requirements that will satisfy internal business needs and external user needs.

Date: Monday October 10th 2016 Time: 1:00 PM Location: CouCou Restaurant

Q: In terms of external business process improvement what are the areas of technology you wish to utilise?

A: We would like our own cross platform mobile app for customers which is built on our own infrastructure allowing us to provide a more enjoyable, personal and rewarding experience over using Just Eat's infrastructure alongside being able to reduce costs through the current 14% commission we pay to Just Eat for using their mobile app and as a result we will be able to afford to offer a loyalty programme as we currently cannot do so.

Q: What do you want to achieve through deploying your own mobile app in relation to what you currently gain from existing business processes by using Just Eat?

A: We would like to add value by providing customers with a live order tracking feature. Our 2nd goal is for the app to market our brand effectively and to gain feedback on the quality of orders for the purpose of continuous quality improvement through which Just Eat has helped us with already by expanding our customer base and including a review system. We would finally like to satisfy customers by ensuring the loyalty programme provides a rewarding experience as well as enjoyable one.

Q: Are there any internal business improvements that you require?

A: The core requirement we have is to have a business system which interacts with them mobile app on our end which allows us to manage orders and manage reviews. In the future as we grow larger we would also like to increase efficiency by being able to base our ingredient re-ordering process on the demand for our menu items through customer orders which prevent demanding menu items from being unavailable to order at critically demanding hours and reduces waste through maximising the usage of all our ingredients. At the moment this is not crucial as we are a small business with very few customers, this is more of a long term plan.

Q: Based on existing data from Just Eat who do you consider to be the intended users of the mobile app?

A: We want to target students, large families and people who are too busy to cook at home or find it infeasible to cook regularly and are more likely to order food for takeaway as opposed to pensioners or people who have time and enjoy cooking by providing student promotions and an enjoyable, simple and rewarding experience.

Interview Evaluation

The initial understanding of requirements which should be taken into account when formulating the requirements specification is as follows:

- Enjoyable and rewarding cross platform app built on company infrastructure to reengineer customer facing business processes and reduce costs through using existing Just Eat infrastructure
- The aim of the app is to add value for customers by adding live tracking and loyalty features along with utilising it as a platform for quality control through implementation of a review system
- Order and Review management component for the business interface which communicates with the customer facing mobile application
- Ensure the business interface is extensible by planning to support increasing efficiency and customer satisfaction in the future by reengineering internal business processes by basing the ingredient reordering process on the demand for popular menu items to maintain the availability of popular menu items and to reduce waste through unused ingredients.

Evaluation of Existing Mobile Applications

Upon conducting primary research by conducting a requirements interview with the client, the next stage of the requirements analysis process involves conducting secondary research by evaluating existing market leading mobile applications with a focus on human computer interaction for the purpose of identifying key strengths and weaknesses and an opportunity to exploit these weaknesses in order to increase competitiveness.

HCI Design Principle Comparison

Design Principles	Domino's	Papa Johns	Just Eat	Comments
Visibility	Dark theme may not appeal to target audience, good use of icons aiding interaction process, insufficient use of spacing on pizza tracker page, good use of separators for menu items, good positioning of recommendations on order page. 7/10	Very simple and basic layout, font size and images overly enlarged, excellent use of colours, no visual pizza tracker process as it only relies on text, continuous scrolling during order progression. Benefits are the increased size prevents order errors. 4/10	Clear layout of interface with excellent colour scheme, choice of font size and colour. Simple and effective use of separators for menu items, no images for menu items as it only relies on text. 9/10	Deploy consistent font size and colour scheme. Images should be of a relevant size. Create an aesthetically pleasing UI for an enjoyable interaction process. Use easily recognisable icons for directing the interaction process. Design for object recognition and recall for ease of reusability.
Feedback	On selection, colour changes on link area and screen slides right. Pizza tracker colour and text changes by fading in and out. Use of updating titles indicate status of interaction process, use of checkout and continue buttons to indicate future interaction. 8/10	Button links don't change colour on selection, pizza tracker refresh process is less dynamic than Domino's. No indication of current status of interaction process. No use of push notifications for order feedback. Use of current reward status and future status. 5/10	Clear indication of current status through selectors below screen. Good use of sliders on restaurant pages indicating menu/reviews/info status. Uses push notifications on order confirmation. 9/10	Notify and visually present changes. Provide colour/text changes upon link selection. Add dynamic updating of pizza tracking.

Design Principles	Domino's	Papa Johns	Just Eat	Comments
Constraints	No text search feature for menu items. 7/10	Inability to edit pizza toppings and create half and half pizzas. No search feature. Back button is squashed in the corner. 7/10	Lack of communication during order delays. Occasional app crashes/errors. Doesn't display live estimated delivery time. 8/10	Build a highly responsive app. Provide ability to search through menu items. Provide order customisation ability.
Mapping	Content is separated by tabs, food menu is presented with sub sections for item selection and customisation. Easily identifiable buttons. 9/10	Home screen presented with evenly spread out buttons. Difficult to identify user's interaction trail, repetitive use of back button to go to the home screen. 5/10	Tab separation included, restaurant menu items are not separated by category, user trail easy to recognise and go back on. 8/10	Introduce tabs to separate app content. Create easy to navigate user trail.
Consistency	Extremely consistent colour scheme, font size and style along with the layout and size of content appropriate for the context. 10/10	Poor use of overly large images in menu's. Different colours used from home screen to ordering process. 6/10	Very consistent colour scheme, font size and style usage for menu items and ordering. 10/10	Provide consistent colour schemes, font style, image sizes and layout, forming a uniform design and structure.
Affordance	Tabs are distinct, user trail isn't specified, however use of arrows identifiable for proceeding. Good use of icons for adding deals and menu items. 8/10	Links easily identifiable using clear colours and text sizing, refreshing of pizza tracker difficult to understand. 5/10	Every button and tab has a specific action, good use of arrows and boxes indicating selection of items. 9/10	Clear presentation of buttons for action needed. Ensure every button has a purpose and is easily understandable by the user.
Total	49/50	32/60	53/60	

HCI Usability Comparison

Usability Goal	Domino's	Papa Johns	Just Eat	Comments
Effectiveness	Explains offers and usual orders for fast re-ordering well. Decent explanation of menu items whilst providing allergy information. Good use of tempting orders section on checkout. Good use of pizza tracker. Use of stock images doesn't provide accurate representation of products. 8/10	More detailed explanation of offers. Rewards concept is emphasised more than re-ordering process, very detailed information about quality. 8/10	Limited information on restaurant food menu items, focused on simplicity. Excellent use of customer review system. App clearly describes minimum value for delivery before check-out. 7/10	Emphasis re-ordering and loyalty processes to focus on repeat business. Moderately detailed description and realistic photos of menu items helps clarify users and prevent errors in ordering process.
Efficiency	Extremely efficient ordering process as 1 button takes user to menu. Clear identification of deals, menu items and customisation options along with clean native design. 9/10	Re-ordering process highly inefficient. Required to progress through repetitive menu's. 'Web app' rather than a native application. Some pages have endless scrolling to proceed. 4/10	Fast ordering process. use of different text styling and sizing clearly indicates restaurant name/menu items. Banners showing special offers easily identifiable. Occasional payment difficulties. 9/10	Minimise Re-ordering process. Ensure app pages load quickly and content is conveniently organised
Safety	Processing of payments is secure. Clean design and order confirmation minimises accidental orders. Displaying allergy information helps prevent issues with customers. 10/10	Secure payments. Doesn't display allergy information. Design prevents accidental orders. Restrictive customisation prevents input of data. 8/10	Secure payment processing. Allergy requirements displayed at checkout. Minimal design prevents order errors and duplicate items being added. 10/10	Ensure sensitive customer and payment data is secure. Ensure organisation of content prevents errors with ordering.

Usability Goal	Domino's	Papa Johns	Just Eat	Comments
Utility	Customers can view allergy information on menu items. Live pizza tracker gives customer an expected duration of order. Informs customers of offers. Store locator provides opening hours. Minimum cost for delivery not shown. Provides Apple Watch app for live pizza tracking. Opportunity for gluten free bases. 9/10	Customers can immediately see available points balance and offers. Provides information about values of company and live pizza tracker. Push notifications on special offers. Opening hours and minimum order cost for delivery hidden. 7/10	Provides Apple Watch app for estimation of order arrival. Easily shows opening hours and minimum order required for delivery along with delivery costs per restaurant. Customers can easily search through menu's using a search function along with a reviewing functionality. 10/10	Ensure opening hours and minimum orders for delivery are easily identifiable. Provide allergy information and order customisation options per menu item and include a text search function for menu. Include live order tracking functionality.
Learnability	Simple interface with easy to identify options due to clean colour scheme. 10/10	Order process contains a long trail of selections. Help icon provided. Better Ingredients and Quality guarantee link purposes are unclear. 7/10.	Minimal design requiring minimal input to complete an order. Organised and consistent layout for menu ensures simplicity. Each link/button serves a clear purpose. 10/10.	Customers should find that a repeatedly in use app has minimal learning curve. Each link and button should serve a clear purpose. Ensure colour scheme, icons and other graphics guide user through ordering process with minimal complexity required.
Memorability	Extremely memorable due to use of a consistent colour scheme, font size and font style, however dark colour scheme may make it less memorable than a brighter colour scheme. 9/10	Fairly memorable, however colour scheme and inefficient use of menu item images allows customers to get confused and forget about the layout due to the poor structure of the menu. Long trail of selections for the ordering process. 5/10	Simple and clear design with standardised restaurant pages and consistent layout of menu items. Ensures customers find it easy to locate menu items, review orders and find useful information such as opening hours and minimum order required. 10/10	Ensure use of aesthetically pleasing colour scheme along with a standardised layout for menu items to allow customers to remember app structure which aids ease of use for repeatability purposes.
Total	55/60	39/60	56/60	

Nielsens Heuristic Evaluation Comparison

Heuristic Principle	Domino's	Papa Johns	Just Eat	Comments
Visibility of System Status	Labelled tabs using text and icons gives a clear indication to the user where they are along with the use of tab highlighting. Process is defined by viewing the menu first and then order. 9/10	Poor indication of where user is in ordering process, icons only provide user to go back and give no indication of content of next page, however home page icon design gives user good indication of where to browse to. Poor visibility of help icon as it is located in the top right using a dark colour scheme. Pizza tracker is static and uses no colour to indicate order status. Image carousel on 'Better Ingredients' page is not clearly visible. 4/10	Labelled tabs with text and icons allows easy visibility of status. Lack of live visibility of order once complete, only gives estimated delivery time. Restaurant view clearly indicates minimum delivery required, cost of delivery and proximity to location. Restaurant availability for ordering is also shown. 10/10	<p>Users should easily be able to identify which section of the app they are currently on and the content on future sections they wish to browse to using labelled tabs, icons and intelligent use of colours.</p> <p>Mobile applications should have a clearly identifiable search hierarchy to aid ease of navigation.</p>
System/Real World Match	Use of language is entertaining and applicable to intended users, provides no technical jargon, however foreigners would find the app difficult to fully understand if english is not their first language. 9/10	No entertaining language but no technical jargon. Use of 'Better Ingredients' and 'Papa's Quality Guarantee' icons can show meaningless purpose to the user. 6/10	No technical jargon and information is clear and unambiguous. 10/10	<p>Unnecessary technical words should not be used, only simple english words to emphasise ease and simplicity to broaden the target audience.</p> <p>Possibility to include language options in the mobile app as a further extension in order to broaden the target audience.</p>
User Control and Freedom	Users can easily go back from the order page and add more items from the menu tab. 10/10	If user is deep in the process of ordering but wants to go back and order further items this requires a long series of back icon selections due to lack of tab usage for app content. 3/10	Broad range of search options for menu by text/category along with cuisine search by text and category. Clear icons and tabs allows the user to easily navigate and go back regardless of depth of ordering process. 10/10	Users should have an easy route to navigate through and be able to return back from app sections without frustration and complexity in the interaction trail.

Heuristic Principle	Domino's	Papa Johns	Just Eat	Comments
Consistency and Standards	Extremely consistent colour scheme, font size and style along with standardised description structure for all menu items. 8/10	Colour scheme, font size and style is consistent, however poor structure and organisation of images along with lack of use of native resolution when browsing the menu results in poor standards. 4/10	Excellent choice of colour scheme, font size and style which is highly consistent. Standardised restaurant pages along with menu layout for restaurants and review system results in high standards. 10/10	Consistency in terms of layout, images, colour schemes, font size and font style will increase memorability leading to an increased chance in repeated use.
Error Recognition, Recovery	Accidentally selecting a deal from the main page can be easily recovered using the back arrow selector. No use of error recognition warnings before payment as its unnecessary. 10/10	Very difficult to recover from accidentally selecting undesired menu item as it requires a long selection of back icon clicks. Unclear whether recovery from accidental deal selection is possible due to small icon size. 3/10	When users input invalid search criteria for menu items there is no recognition of this entry, very easy to recover from accidental menu item selections due to excellent layout, otherwise no other errors can be made. 8/10	Validation techniques should be deployed to notify and assist users of incorrect data entry or confirming selections where necessary.
Error Prevention	Differentiating quantity selection and add to order button results prevents duplicate order errors along with links to pizza descriptions before additions to order prevent incorrect pizza being added. Use of clear font colour for pizza customisation options easily visible to user before order confirmation, however deals being on main page could be unintentionally selected. 9/10	Crammed order interface due to enlarged images, text and inefficient scrolling make errors easily achievable, therefore preventing errors during order process is unlikely. It is increasingly possible to accidentally add undesired deals to the orders due to the icon size. 2/10	Use of icon validation for cuisine category selection helps prevent searching undesired cuisines. Autocomplete for each character when searching for restaurants and menu items helps simplify search process and prevent typing errors. 8/10	User Interface error minimisation techniques should be included to minimise the chance of errors occurring for the user such as the appropriate spacing, sizing and organisation of app content to prevent unintentional selections.

Heuristic Principle	Domino's	Papa Johns	Just Eat	Comments
Recognition not Recall	Descriptive icons ensure users know where the menu is, descriptions of pizza's and how to add items to their order. Every pizza has standardised customisation, "size, crust, topping" easy to recognise. 9/10	Home page icons are fairly descriptive, however user does not get the opportunity to view their user trail to go back to a certain page that is not a direct descendant of the page they came from. User sometimes has to scroll endlessly to find the continue button to proceed. 6/10	Text descriptions on search fields and consistently labelled buttons alongside the use of helpful icons ensure users recognise where to search for and navigate through content. 9/10	Ensure app design and navigation includes easily recognisable icons and searchable content is identified clearly to the user. App design should require minimal effort from the user to progress through sections within app.
Flexibility and Efficiency	Simple and easy to use navigation system as users can quickly get back to the menu, read pizza descriptions and add more pizza's to their order and return to the confirmation page without any difficulty. Flexibility with payments would make ordering process easier e.g. providing Apple Pay. Payment processing errors cause app crashes sometimes. 8/10	Inefficient scrolling as app can sometimes crash. Users are not provided with opportunity to customise order. Adjusting order is slightly complicated as it involves a long series of clicks to deselect and select another menu item. 4/10	Use of Facebook login integration increases efficiency of ordering as minimal effort is required to sign up. Order customisation/special requirement options for menu items are not available. App speed is highly efficient and does not crash. Highly informative app detailing availability and location of restaurants. 9/10	Make sure app contains highly informative information that is useful to the user such as opening hours and minimum order amount for delivery. Ensure users can easily edit an order through flexible and uncomplicated navigation. Allow users to customise their order using text input for increased flexibility. Prevent information overload of app content which could provoke app crashes.

Heuristic Principle	Domino's	Papa Johns	Just Eat	Comments
Aesthetic Design	Good use of white space utilisation along with efficient sizing, location of images and menu item descriptions suitable without overloading the user interface. Dark colour scheme and transparent text for menu descriptions may be a little tricky to read for some users. Attractive and entertaining design of pizza tracker. 8/10	Extremely poor aesthetic design because the app is rendering the desktop version of website as opposed to a streamlined mobile app. App information is crammed causing navigation difficulties. Pizza images as backgrounds can make text difficult to read and feel less emphasised particularly on order tracking process. 3/10	Fantastic aesthetic design, makes clever use of white space to increase clarity within the app as colour palette is emphasised clearly. Menu items consistently and clearly sectioned by category. 10/10	Ensure colour palette, font size and style in use aids visual clarity of app content. Excessive and oversized Images and text should be avoided as it leads to overloading the interface causing navigation issues to arise.
Help and Documentation	No help icon or initial tutorial on entry of app, however simplicity of app does not require extensive documentation. 4/10	Help icon exists in the top right corner roughly outlining functionality of buttons, however does not appear upon first time opening of app which is pointless given it is mainly needed at this time. 4/10	No user manual or introductory tutorial on first opening of app, however interface is intuitive and clean therefore users shouldn't run into trouble. 2/10	Provide a short introductory tutorial for the user detailing how to navigate within the app upon first opening and make this available in case it is required by users later on.
Total	84/100	39/100	86/100	

User Experience Comparison

User Experience Goal	Domino's	Papa Johns	Just Eat	Comments
Satisfying	Very satisfying, watching the live pizza tracker, users can browse the menu with ease and get information about the content of menu items. Re-ordering previous order is extremely simple with a visually satisfying order button. 7/10	Unsatisfying experience with unbalanced sections involving increasingly large images and text and poor use of background images. 6/10	Satisfying as users can get thorough restaurant information such as minimum amount for delivery, delivery cost and allergy information. Estimated time for order is useful, however not live per restaurant. Users get total control through cuisine search and filtering restaurants by reviews. 8/10	Navigating through the app and ordering should be a simple and satisfying experience which can be achieved by use of an enjoyable to watch pizza tracker and simplified re-ordering process. Include ability to filter through menu items based on reviews to satisfy user searches.
Enjoyable and Rewarding	Very enjoyable experience watching the live pizza tracker. Plenty of rewarding opportunities such as various deals integrated into orders. 9/10	A stressful experience due to the complicated and fiddly process of ordering items, however points system aids repeat business opportunity to do reward points gained. 5/10	Interface is enjoyable to navigate and proceed through orders, however ETA of order arrival is not very enjoyable to watch as it is not live. There is a shortage of restaurant offers/deal sections and the app does not reward users to return after order. Review system does not reward users for their contributions. 8/10	Ensure pizza tracker and ordering process entertains the user by providing clean transitions between app sections and live pizza tracker statuses, whilst also making the rewards process for the user an enjoyable journey rather than simply just updating the points balance.
Fun and Entertaining	Only fun aspects are the live pizza tracker and enjoyable order button design. 8/10	Pizza tracker refresh process is static and not entertaining to watch. 4/10	Does not offer visually entertaining transitions throughout ordering process. Restaurant information pages provide users with a more detailed insight which can be entertaining. Apple Watch app is extremely fun to use, however ETA of order arrival is static. 7/10	Possibility to include visually pleasing and relevant transition animations to entertain the user while they order. Ensure pizza tracker dynamically updates and is fun to watch.

User Experience Goal	Domino's	Papa Johns	Just Eat	Comments
Helpful and Motivating	Menu items provide reasonable level of information such as the pizza size in inches and number of slices, along with allergy information. Pizza tracker interaction motivates users to use the app again. 9/10	Provides reasonably helpful information about the values of the company, but does nothing to necessarily motivate the user to order. 6/10	Provides an extremely helpful ordering process with customisation options to narrow down by cuisine type, refine restaurants by review and sorting options which can motivate the users to order as the search process satisfies their needs effectively, however not all restaurant pages offer additional notes area for orders. 8/10	Include a brief summary of company values to notify customers and persuade customers to order again. Include useful information such as nutritional information and provide users with allergy information on menu items.
Aesthetically Pleasing	Pizza tracker has a clean design along with the general app design and a particular satisfying order button. 9/10	Navigating through the app is unenjoyable due to the poor resolution. Inefficient use of background images and overly sized images sometimes cause text to be less emphasised. 5/10	Incredibly clean and pleasing user interface due to excellent content layout, consistent colour scheme, button and tab design and appropriate font sizes and style. No images provided for menu items, however text is clearly visible to the user. 10/10	Provide clear and visually appealing design of buttons, pizza tracker and links along with balanced sizing of images and organisation of content.
Total	42/50	20/50	33/50	

Evaluation Conclusion

The following conclusions can be made which can be taken into account for the requirements specification.

- Use clean transitions and utilise colour and text changes to provide an enjoyable status updates for customer orders
- Utilise a consistent font and colour scheme for the purpose of providing an aesthetically pleasing and easy to use interface
- Ensure layout of information is standardised to enhance memorability for repeat usage
- Image, text and menu item description sizes should be standardised preventing an overload of content resulting in navigation difficulties
- Provide flexible usage through order customisation and search functionality
- Ensure the application is informative providing opening hour, menu item and nutritional information descriptions along with a brief overview of company values
- Emphasis loyalty points and provide an easy re-ordering process on opening of the mobile application

The Importance of Business Intelligence

The client has requested in the interview that viewing the demand for popular menu items and basing this performance data on their ingredient reordering process is a feature which they would like to include, as they are a growing business. I have conducted some research to determine whether or not there is a substantial benefit that business intelligence could bring to CouCou.

Business Analytics as defined by Jeston and Nelis (2014, p.65) is “*the collection and examination of performance related process information*”. The provision of business analytics will provide benefits including according to Jeston and Nelis (2014, p.65) allowing for “*monitoring processes in real time resulting in less problems and lower costs*”, in this case it will lead to better ingredient reordering decisions in relation to the demand for popular menu items which will prevent wastage in terms of oversupplied or undersupplied ingredients, reduce costs and ultimately increase the efficiency of business processes which will then lead to an increase in customer satisfaction through the constant availability of products.

Delaget are a provider of restaurant analytics software and outline several important metrics to restaurant owners in their eBook ‘*Metrics that matter to restaurant owners*’ The most relevant metric for the undertaking of this project is the process of managing food costs outlined by Delaget (2016,p.3) which can be problematic and can be improved by analysing the amount of food waste and portion sizes determined by the current usage of ingredients for menu items by restaurant staff for the purposes of controlling efficiency and gaining an understanding into what drives these costs. Sales forecasting is another metric which can accurately predict the quantity of menu items needed at a point in time thus reducing wastage through excess stock and leading to customer satisfaction through meeting demand.

Pixentia who also provide restaurant analytics software helped to increase sales by 50% on a yearly basis for a restaurant chain by 50% by utilising predictive analytics to identify the order preferences and order frequency of repeating customers. The restaurant chain previously struggled to identify customer preferences in all of its locations and wanted a solution that would accurately identify behavioural trends. Pixentia’s solution allowed the chain to ‘*redefine its menu exactly according to consumer preferences*’ Pixentia (2016,p.2) allowing for the potential of increasing customer satisfaction which also led to an ‘*increase in customer loyalty in all locations*’ along with ‘*the frequency of customer visits increasing*’ Pixentia (2016,p.2).

Fishbowl who are another provider researched the demographics of customers for a particular dining brand who were struggling to deliver a more personalised service and were able to draw conclusions on the dining behaviour of customers. The analysed data according to a case study by (Fishbowl,2016) resulted in ‘*72% of its guest falling into personalised personas as a result of analysing order metrics*’ allowing the company to engage in more personalised marketing based on the order frequency and menu item preferences of customers. The result was that these customers were more interested in these personalised marketing campaigns based on their behaviours as opposed to more general forms of marketing. In the dining brands case ‘*open rates for families increased from 18 to 24%*’. These open rates are ‘*an indication of how many people view commercial email that marketers send out*’.

Evaluation of Existing Restaurant Analytics Software

The next step upon the realisation that analytics software is essential involves researching useful features in existing analytics solutions for the purpose of deciding whether or not to plan for integrating these features in the future.



Figure 2: Fishbowl Logo

Fishbowl's data analytics platform is divided into three main areas, however the two areas which are relevant to this project are Guest Analytics and Menu Analytics. Fishbowl has been chosen because they currently serve market leading companies such as Burger King and Planet Hollywood.

Guest Analytics features involve measuring and predicting customer behaviour based on sales data extracted from member databases on mobile applications and in the restaurant via POS transactions. The types of data they are able to analyse are the demographics of customers such as their age, their frequency of orders and how to increase sales by engaging with this data for decision support purposes to provide targeted special offers to these customers. Existing loyalty programmes are also analysed in terms of whether their utilisation is effective in making a significant contribution towards driving customer sales.

Menu Analytics features provide data on each individual menu item in terms of its contribution towards increasing sales, how specific menu item price fluctuations affect sales and whether the inclusion of a menu item is a major factor towards strategic objectives in terms of increasing the frequency of orders and therefore increasing sales.



Figure 3: Upserve Logo

Upserve's analytics solution is specifically integrated within POS (Point of Sale) systems in that data is largely coming from customer visits in store rather than digitally via a mobile application and has been chosen purely to evaluate the scope of internal systems for the future.

Upserve also provide customer focused analytics such as favourite dishes, frequency of visits and average spending along with menu optimisation features similar to Fishbowl's functionality such as identifying popular menu items. Upserve's system works by providing a 'daily digest' outlining key performance data for that specific day along with 'Shift prep' notifying important data that should be known for decision support the night before the next business day.

Upserve's platform differs in that it offers a companion mobile application where restaurant managers can view analytics data via a smartphone as well as via the web whilst also providing the measuring of staff productivity in relation to how significantly they have contributed towards sales.



Figure 4: Posist Logo

Posist's analytics solution is purely browser based and therefore works in the cloud therefore providing interoperability along with offline access to static analytics data. The features provided involve a dedicated customer relationship management featuring detailing comprehensive reports about customer behaviour, integrated offers module to allow the generation of custom offers for specific customers with specific order frequencies, integrated loyalty program tool that can be customised in relation to customer order frequency and menu item behaviour. The CRM feature also provides SMS integration to allow customers to be notified of their order status and email as well as SMS features for marketing purposes such as special offer notifications.

The features that make Posist different in relation to Fishbowl and Upserve are that they provide a delivery module for live tracking orders which can be integrated with online ordering apps included within their sales analytics solution along with delivery driver analytics data useful for identifying the performance levels of delivery drivers in terms of time taken to deliver.

Posist also differs in that it provides an Inventory and Stock Control component providing features such as 'Stock-In and Stock-Out' revealing ingredients available at the start of business along with the total ingredients used at the end of business which is scalable and can provide ingredients data for more than one restaurant. Ingredient usage can also be forecasted based on previous sales data and if necessary reorder alerts for ingredients can be utilised for the purpose of decision support to prevent out of stock ingredients and dissatisfied customers as a result of this previous sales data. Perishable Ingredients can also be managed, however recommended actions based on sales data are not provided, and finally Posist allows for recipe alterations for menu items.

Evaluation Conclusion

Upon comparing the three analytics solutions the following conclusions can be made which can be taken into account for the design of the analytics system.

- CRM features such as identifying frequency of orders, preferred menu items, targeted offers are relevant to CouCou and are a key feature, however average spending may not be as useful
- SMS and email marketing is not a major feature for providing offers, rather social media should be implemented instead as it seems more relevant in this digital age
- A loyalty programme is due to be implemented, however there is not much scope for customisation necessary, thus a fixed loyalty programme won't require detailed analytics
- Sales Analytics data on specific menu items is useful towards identifying its contribution to total sales
- Daily and Evening Summary forecasts similar to Upserve's feature for sales may prove useful
- POS integration similar to Upserve's offering is not required as emphasis is being placed on deliveries
- Mobile access could later be implemented, however not strongly important
- Inventory and Stock Control features similar to Posist's offering is essential to reengineer internal operations, however will prove difficult to manage the ingredients required per item in day to day operations

Use Case Diagram

Upon conducting primary and secondary research to better capture the complete system requirements a use case diagram can now be created which represents the possible actions implementable by users of the mobile app and users of the business interface. It also represents the sequence of possible actions along with identifying how mobile app interactions such as live tracking of orders are integrated with business interface interactions.

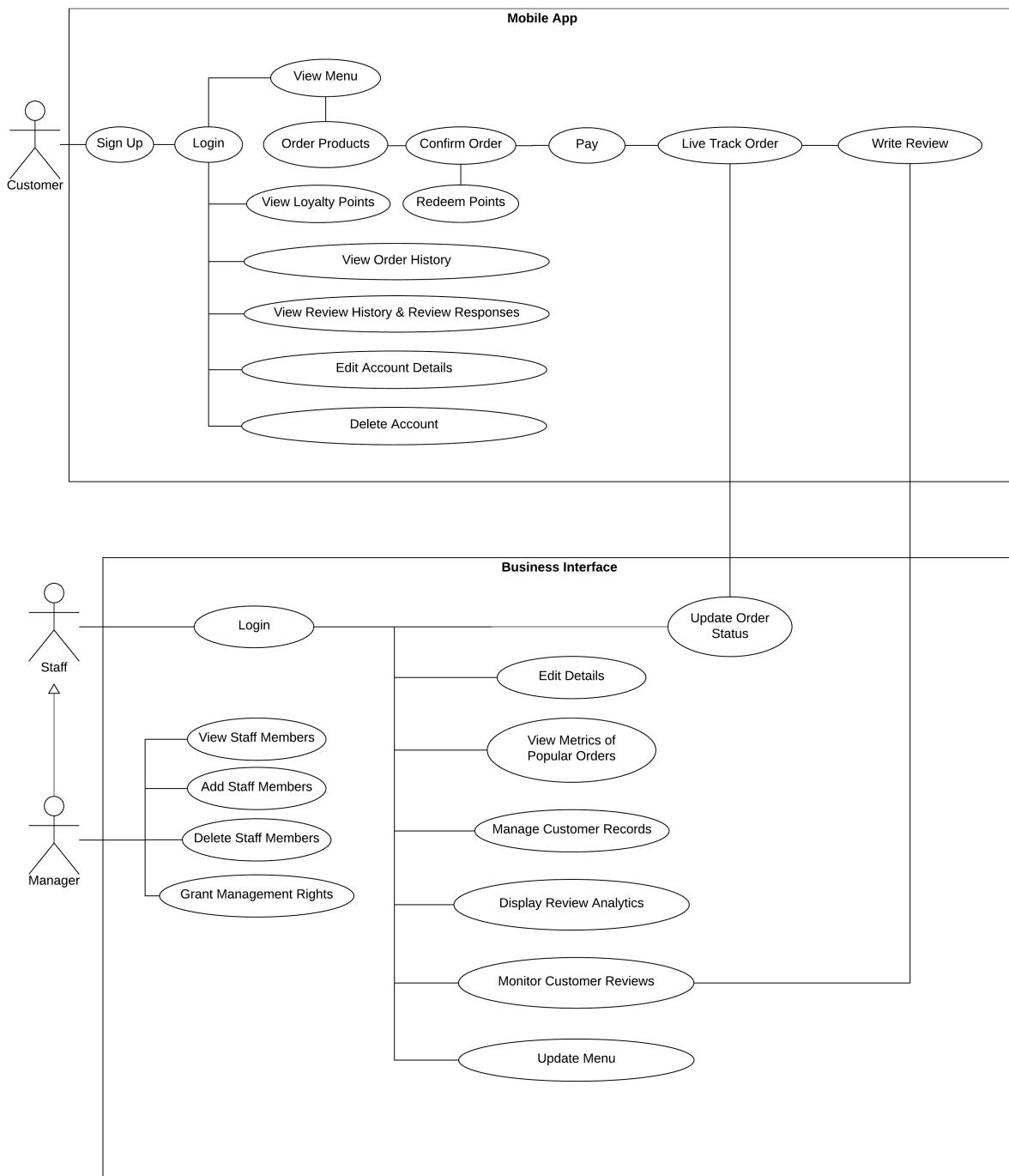


Figure 5: Use Case Diagram

REQUIREMENTS SPECIFICATION

Upon conducting an initial interview with the client, analysing human computer interaction aspects of existing mobile applications, conducting research on the benefits of restaurant analytics systems and designing an initial use case diagram, the complete requirements specification can now be formed using the use case diagram as there is now a clear understanding of the complete system. The use case diagram represents the core functionality of the complete system and includes extended functionality such as payment and analytics features, which are included because the long term implementation of the system is being considered, however in the implemented system these two features are left out.

Functional Requirements

Core

Mobile App

1. The mobile app shall allow customers to sign up to order menu items

- 1.1. The customer shall be able to signup if they provide valid data
- 1.2. The customer shall not be able to signup if they provide invalid data

2. The mobile app shall allow customers to login to order menu items

- 2.1. The customer shall be able to login if they provide valid data
- 2.2. The customer is not able to login if they provide invalid data

3. The mobile app shall display a welcome screen

- 3.1. The welcome screen shall display the customer's available loyalty points
- 3.2. The welcome screen should update the available loyalty points after the customer gains points
- 3.3. The welcome screen should update the available loyalty points after the customer uses points
- 3.4. The live track button should only be accessible if a customer has an order waiting to be tracked
- 3.5. The write review button should only be accessible if a customer has not submitted a review for their latest order
- 3.6. Viewing review responses should only be accessible if the customer has submitted a review and is waiting for a response

4. The mobile app shall allow customers to order menu items

- 4.1. The customer shall be able to browse the menu
- 4.2. The customer is able to customise their menu items by increasing the size of the menu item
- 4.3. The customer is able to customise their menu items by adding extra ingredients to menu items
- 4.4. The customer is able to add menu items to their order cart
- 4.5. The customer is able to remove menu items from their order cart
- 4.6. The customer is able to add additional requirements to their order
- 4.7. The mobile app should generate the customer's order details in the database upon the customer confirming their order

5. The mobile app shall allow customers to use 10 loyalty points on qualifying orders

- 5.1. The mobile app shall provide a Use Points button on the order confirmation page if the customer has ten available loyalty points
- 5.2. The database should correctly update with the customer's points balance
- 5.3. The mobile app should not display a Use Points button on the order confirmation page if the customer does not have ten loyalty points

6. The mobile app shall allow customers to live track their order

- 6.1. Upon confirming their order the customer should be redirected to a live track order page
- 6.2. The mobile app shall automatically update and display the status of the customer's order
- 6.3. When the order status is delivered the customer shall be presented with a write review button

7. The mobile app shall allow customers to review their order

- 7.1. The mobile app should display all the menu items from the latest order that can be reviewed and not already reviewed
- 7.2. The mobile app should display interactive star rating selectors
- 7.3. The mobile app should display a text area where the customer can input their review
- 7.4. The mobile app should generate a review in the database upon the customer submitting their review

8. The mobile app shall allow customers to view their review history and review staff responses

- 8.1. The mobile app should not display review history buttons if the customer has not reviewed their order yet
- 8.2. The mobile app should present the correct number of review history buttons if the customer has reviewed their order
- 8.3. The mobile app should display all details of the selected reviewed order
- 8.4. The mobile app should display staff responses if staff have responded to the customer's review
- 8.5. The mobile app should display a blank text area if staff have not yet responded to the customer's review

9. The mobile app shall allow customers to view their order history

- 9.1. The mobile app should not display any order history buttons if the customer has not made an order yet
- 9.2. The mobile app should present the correct number of order history buttons if the customer has made orders
- 9.3. The mobile app should correctly display all the details of the selected past order

10. The mobile app shall allow customers to edit their account details

- 10.1. The mobile app should display the customer's details
- 10.2. The database should keep the customer's details unchanged if the customer saves their unchanged details
- 10.3. The mobile app shall prevent the customer updating their details with invalid data
- 10.4. The mobile app shall prevent the customer from updating their email address to one that is already in use
- 10.5. The database should correctly update the customer's details

11. The mobile app shall allow customers to delete their accounts

- 11.1. The database correctly deletes the customer's account upon the customer deleting their account
- 11.2. The customer is redirected to the index page of the mobile application

Business Interface

12. The web interface will provide the ability to update the status of customer orders

- 12.1. The web interface shall correctly display all customer orders
- 12.2. The web interface shall correctly display the selected order's details
- 12.3. The web interface shall automatically load newly generated orders without a page refresh
- 12.4. The database shall update the selected order's status upon updating the order status

13. The web interface will provide the ability to monitor customer reviews

- 13.1. The web interface shall correctly display all customer reviews for menu items
- 13.2. The web interface shall automatically load newly generated reviews without a page refresh
- 13.3. The web interface will be able to delete unhelpful reviews
- 13.4. The selected review to be deleted is removed from the database
- 13.5. The web interface will allow staff to respond to customer reviews
- 13.6. The staff response for the selected customer review shall be saved in the database

Extensions

Mobile App

1. The ability to incorporate a payment feature to allow customers to pay via PayPal or Touch ID when deploying an iPhone version.
2. The ability to reward customers in terms of providing promotional discounts or special offers tailored to their favourite items which is based on the number of reviews they make which can encourage repeat business.
3. The ability to deploy push notifications encouraging customers to use the app during a special promotion
4. The ability to reorder a customers last order from the welcome page instantly
5. The ability to display the average reviews for a particular menu item selected by the customer

Business Interface

6. The web interface could have the ability for restaurant managers to update the menu on the mobile app
7. The web interface could allow restaurant managers to add staff members
8. The web interface could allow restaurant managers to delete staff members
9. The web interface could allow staff to edit their own details
10. The web interface could allow restaurant managers to grant management rights to staff members
11. The web interface could allow staff to view, edit and delete customer records
12. The web interface could in future produce analytics data on popular menu items per day, month and year
13. The web interface could display the total revenue generated by the business per day
14. The web interface could in future allow staff to view all customer reviews for a particular menu item and display 1-5 star rating summaries for the top 3 menu items
15. The web interface could provide the measurement of staff speed in preparing the food based on the time of the order
16. The web interface could provide the ability to reward staff based on the number of good reviews they receive
17. The web interface could provide the ability for staff to view reviews only for menu items prepared by them indicating specific staff who generate good reviews for specific menu items
18. The web interface could provide the ability for staff to view their review response history
19. The web interface could provide the ability to manage the ingredient inventory such as managing expiration dates
20. The web interface could provide the ability to suggest ingredient replenishments for popular items where the quantity of ingredients of these popular items are running low.

Non Functional Requirements

Product Usability

1. The mobile app and web interface should have an aesthetically pleasing user interface design
2. The mobile app and web interface should consist of a simple and enjoyable interaction process
3. The web interface should have a minimal learning curve and only require 1 day's training to be completely usable

Product Dependability

4. In the event of a crash the data in the web interface and mobile app shall remain intact
5. The mobile app should ensure maintainability can be achieved through the provision of adding and deleting staff members and through the ability for staff to edit their own details
6. The mobile app and web interface should ensure scalability to handle additional branches and additional staff members

Product Efficiency

7. The mobile app shall be as responsive as possible to all user actions
8. The web interface customer status updates shall be instantly updated on the mobile app
9. The web interface review responses shall be instantly displayed on the mobile app

Product Interoperability

10. The web interface shall run on the latest versions of Google Chrome, Safari and Firefox to ensure interoperability

Product Security

11. Data stored for the purpose of analytics shall maintain data integrity to retain its beneficial purpose
12. Sensitive customer data such as passwords shall be encrypted to ensure security

SOFTWARE DESIGN

Upon generating a clear set of requirements for the complete system it is now appropriate to consider the software architecture of the system along with how the system will be represented visually. The use of a software design pattern is important to utilise at this stage.

Model View Controller Architectural Design Pattern

Model View Controller will be used for implementation as it separates the software application into three components, each of which have their own dedicated task thereby providing cohesion. This design pattern provides abstraction in that the implementation details are hidden between components. The model represents the underlying logical data structures and the methods for manipulating the data. The view represents the visual representation i.e. the user interface of the application and the controller manages the exchange of information between the model and the view.

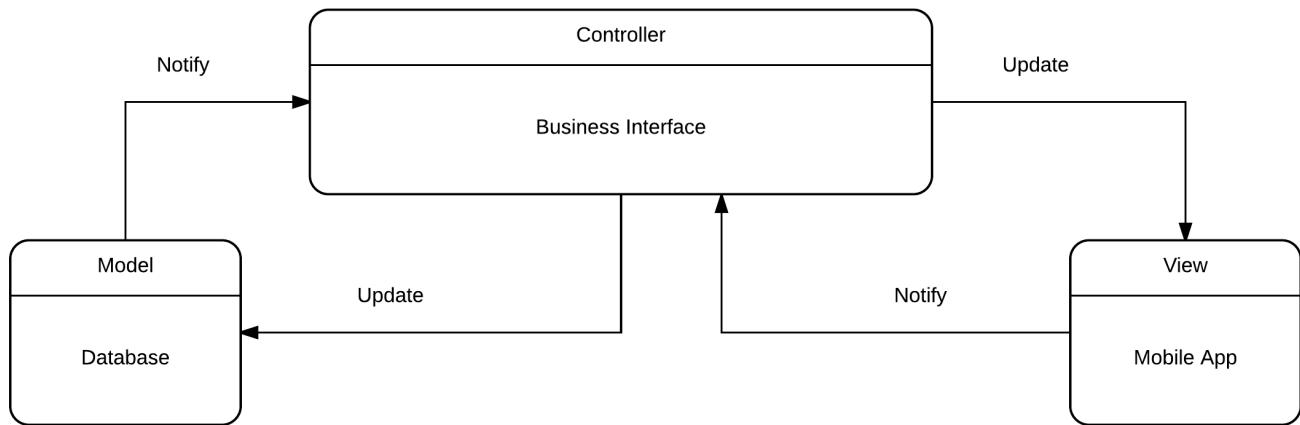


Figure 6: Model View Controller Architecture Diagram

Model

The model deals with the storage of customer's and staff's personal data, order data, review data and menu item data. In terms of storage the controller will directly store menu item data into the model. In terms of updates the controller will retrieve customer order data and update the status of orders directly into the model.

View

The view deals with the visual representation of model data and notifies the controller of new customer orders, customer reviews, updates to customer and staff's personal data to be stored in the model along with the removal of a customer or staff account. In terms of updates, the controller will send the updated order data to the view which will display the updated status of customer orders.

Controller

The controller deals with storing customers personal data, order data, review data and menu item data into the model. The controller will also retrieve customer order data and extract menu items, dates and times, and customer data for the purpose of providing business analytics. The controller will also retrieve updated customer order status data and will update the visual representation of the view.

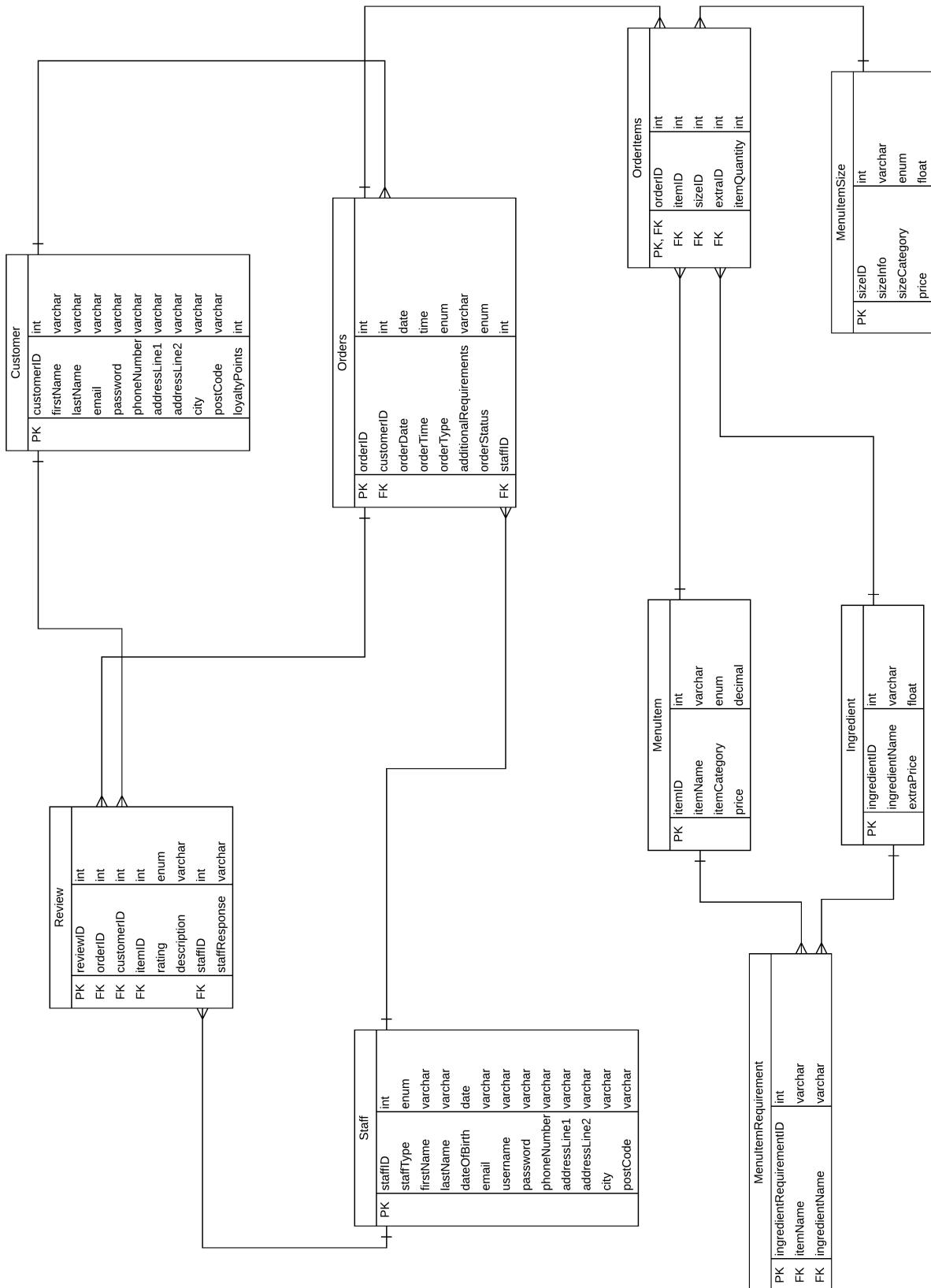
Model


Figure 7: Entity Relationship Diagram

Controller

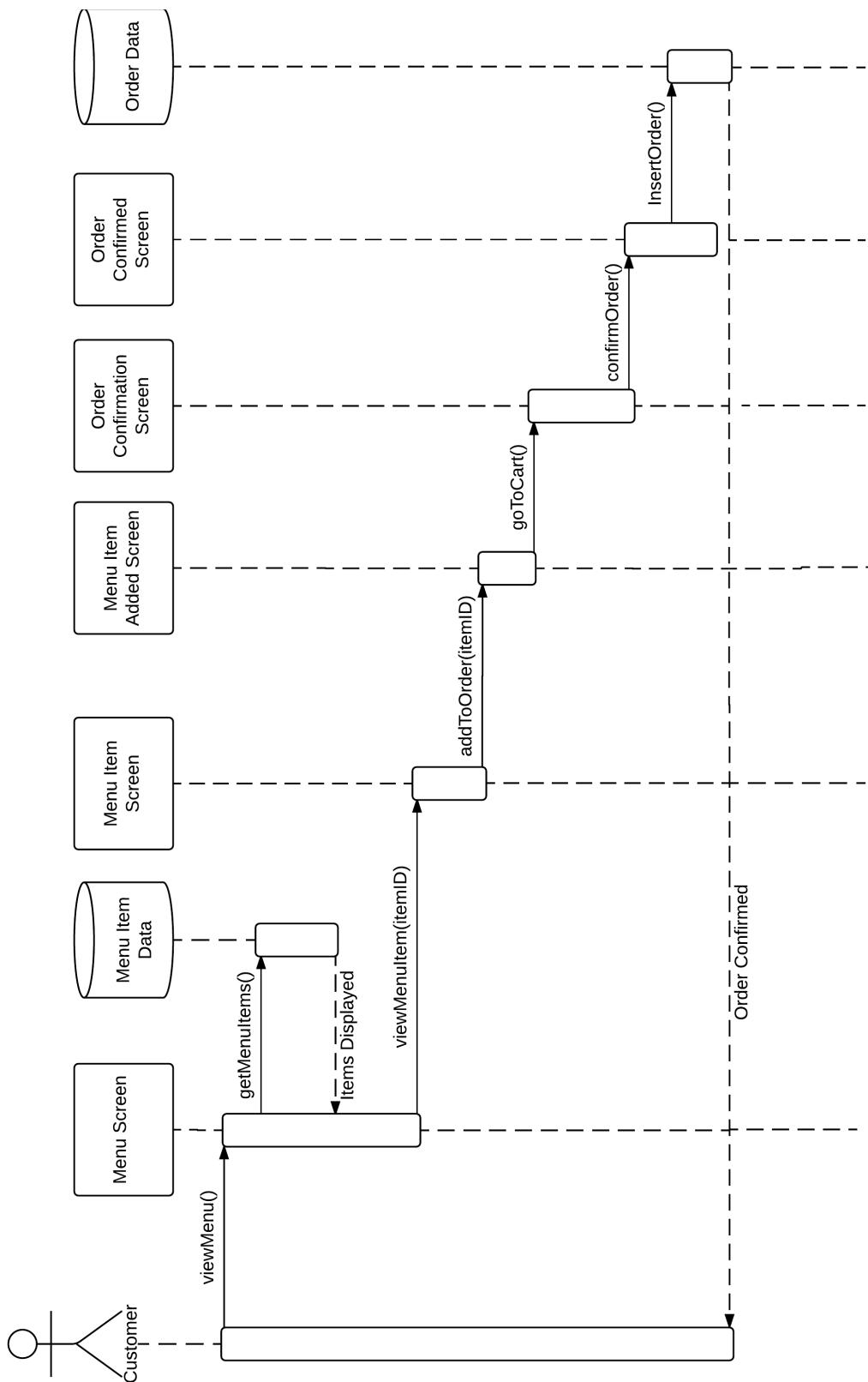
Order Products - Mobile App


Figure 8: Order Products Sequence Diagram

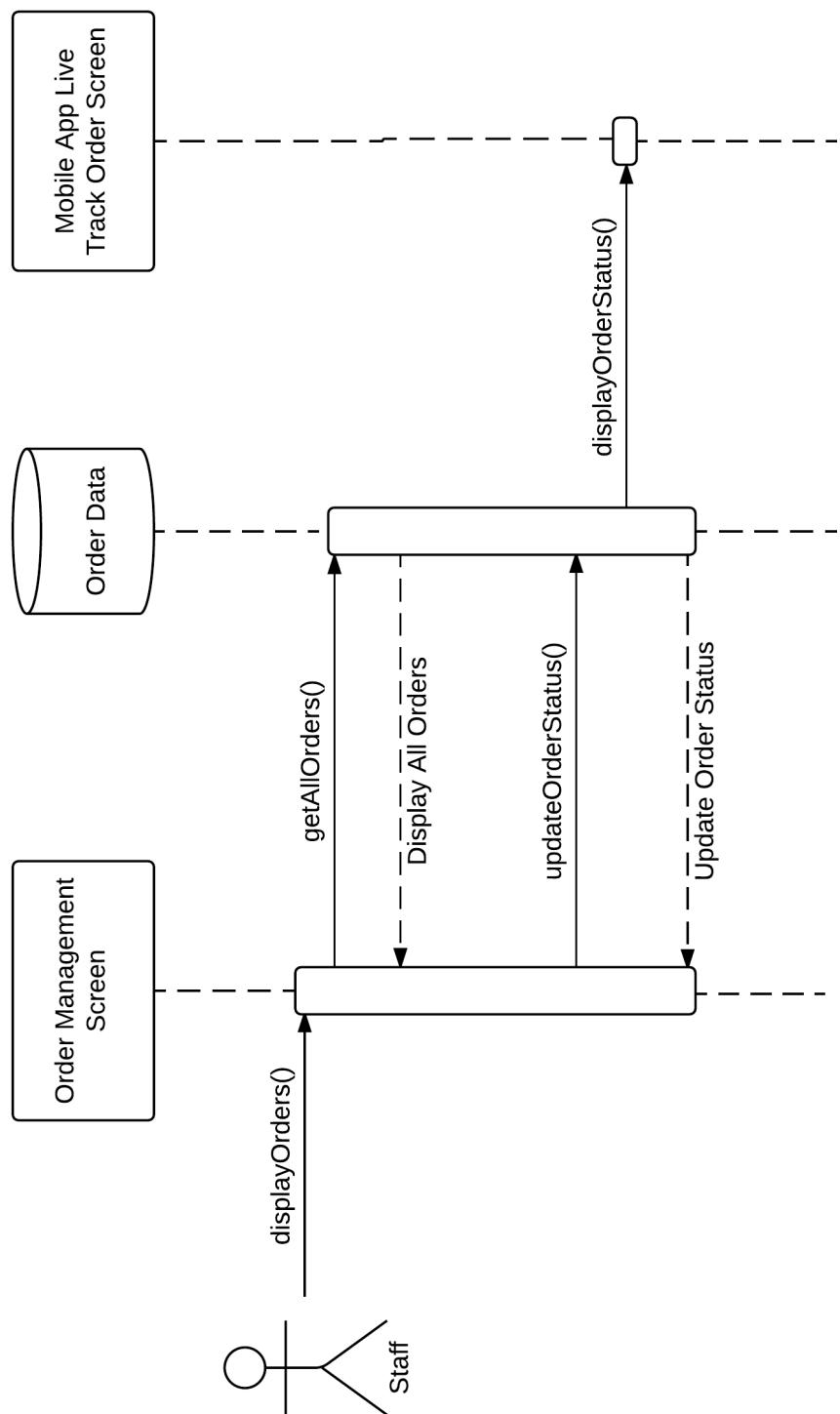
Update Order Status - Business Interface

Figure 9: Update Order Status Sequence Diagram

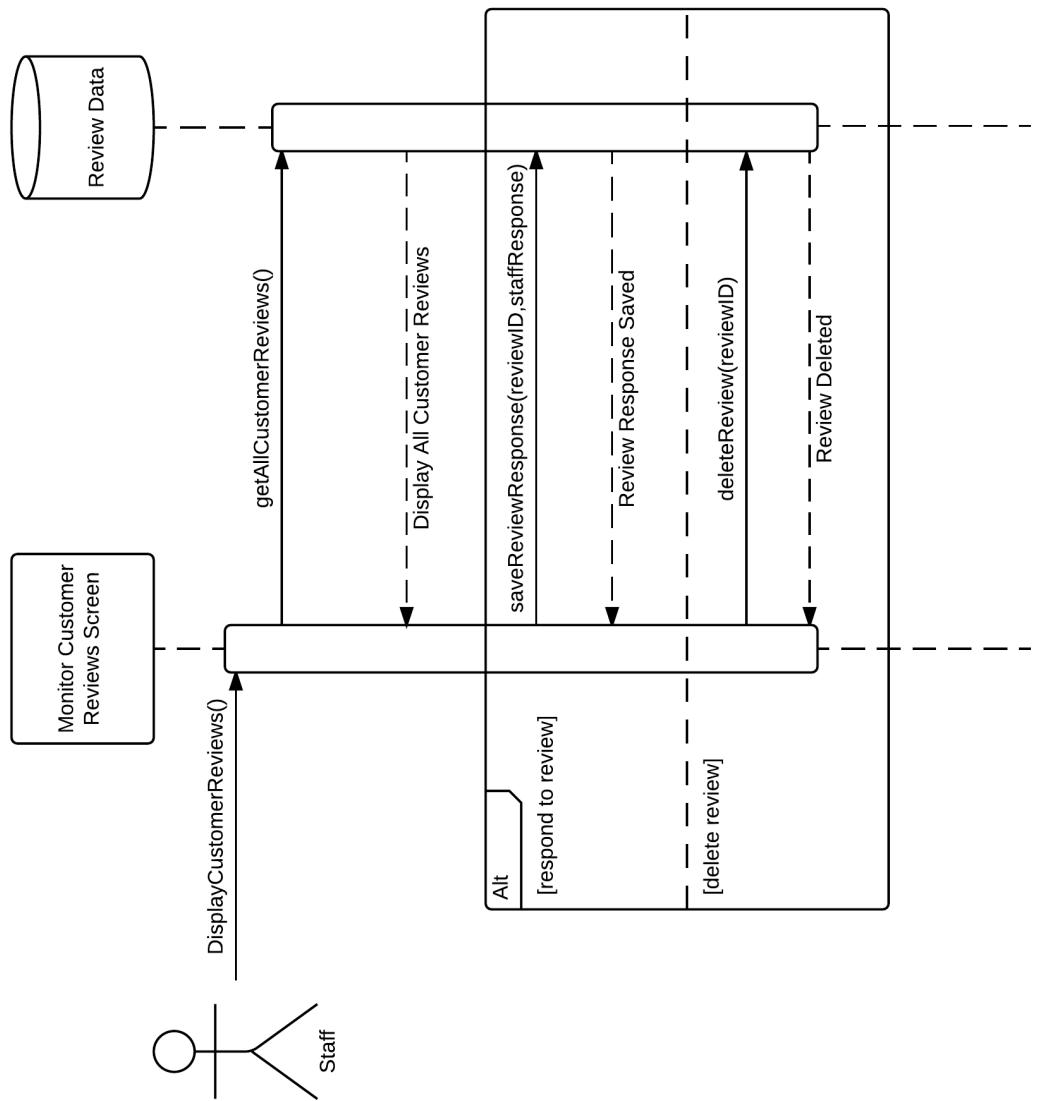
Monitor Customer Reviews - Business Interface

Figure 10: Monitor Customer Reviews Sequence Diagram

View Review History & Responses - Mobile App

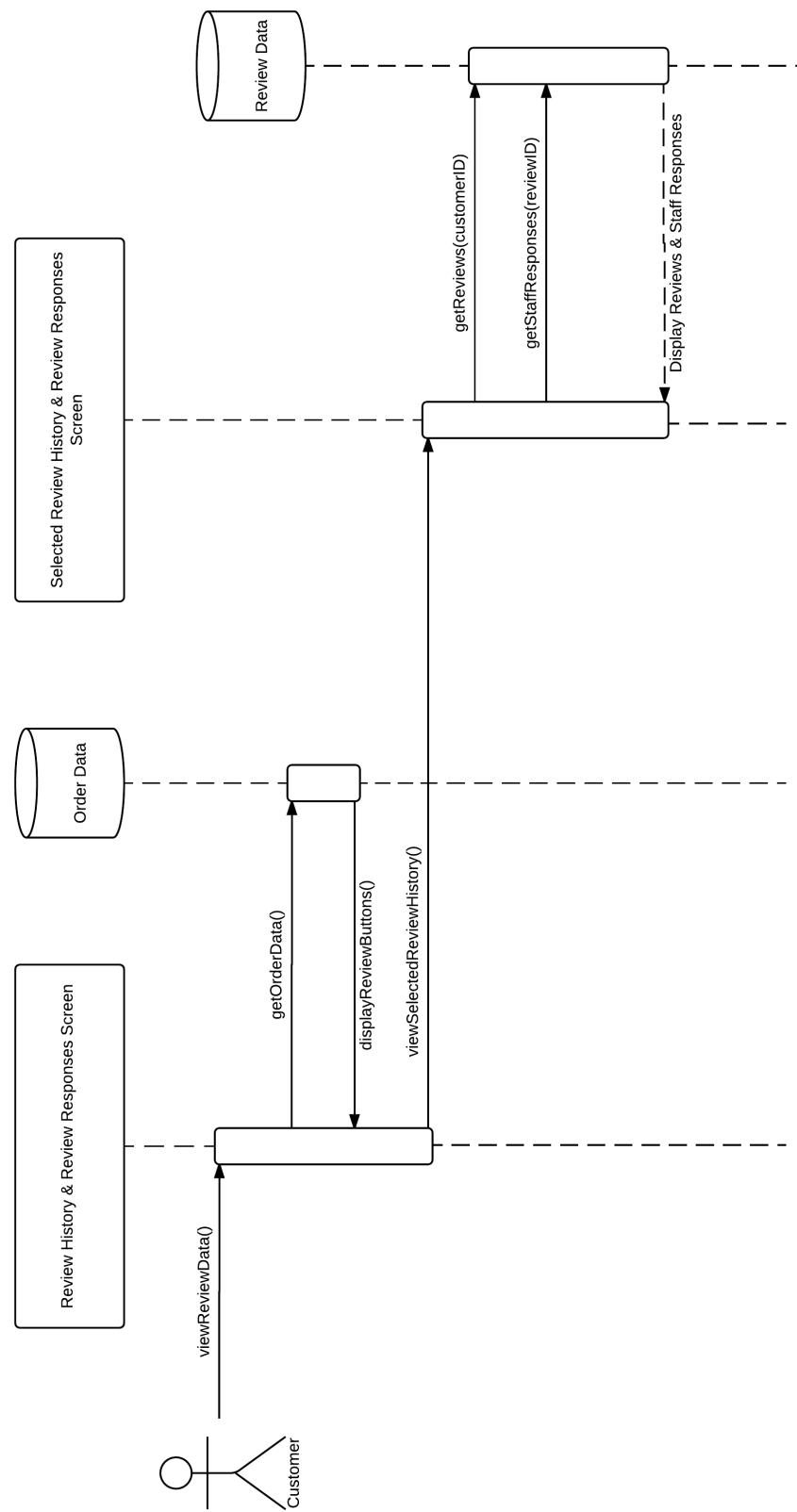


Figure 11: View Review History & Responses Sequence Diagram

View

Upon gaining a clear understanding of the data required for the system and understanding how the data is to be processed across various screens, it is now appropriate to design a visual prototype to get a clearer representation of the user interface of the system using [proto.io](#).

The prototype simulations can be accessed via the links below which allows users to trial the software in order to obtain a feel for the user interface. The prototypes were demonstrated to the client to see a visual representation of their requirements in order to clarify that everything was understood correctly and whether or not to make any necessary changes.

Mobile App Prototype Simulation Link - <https://pr.to/RQUTY5/>

Business Interface Prototype Simulation Link - <https://pr.to/CATR8N/>

Mobile App

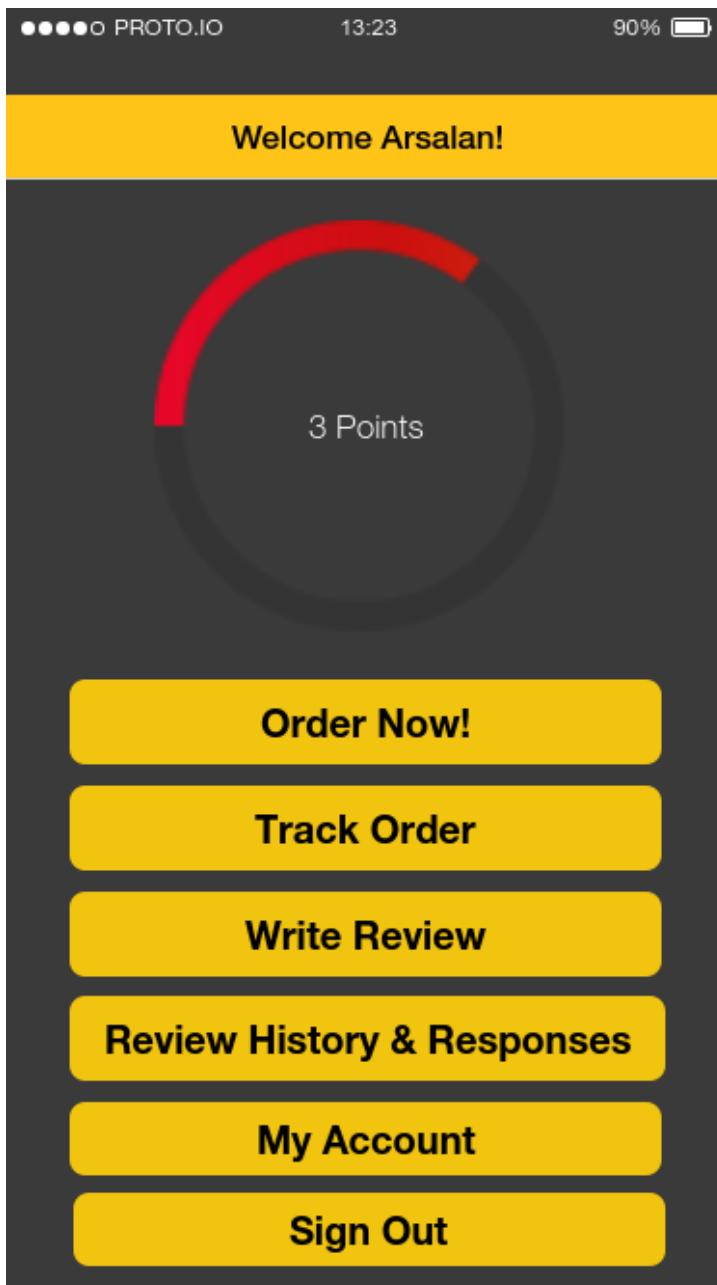


Figure 12: Mobile App Welcome Screen

Business Interface

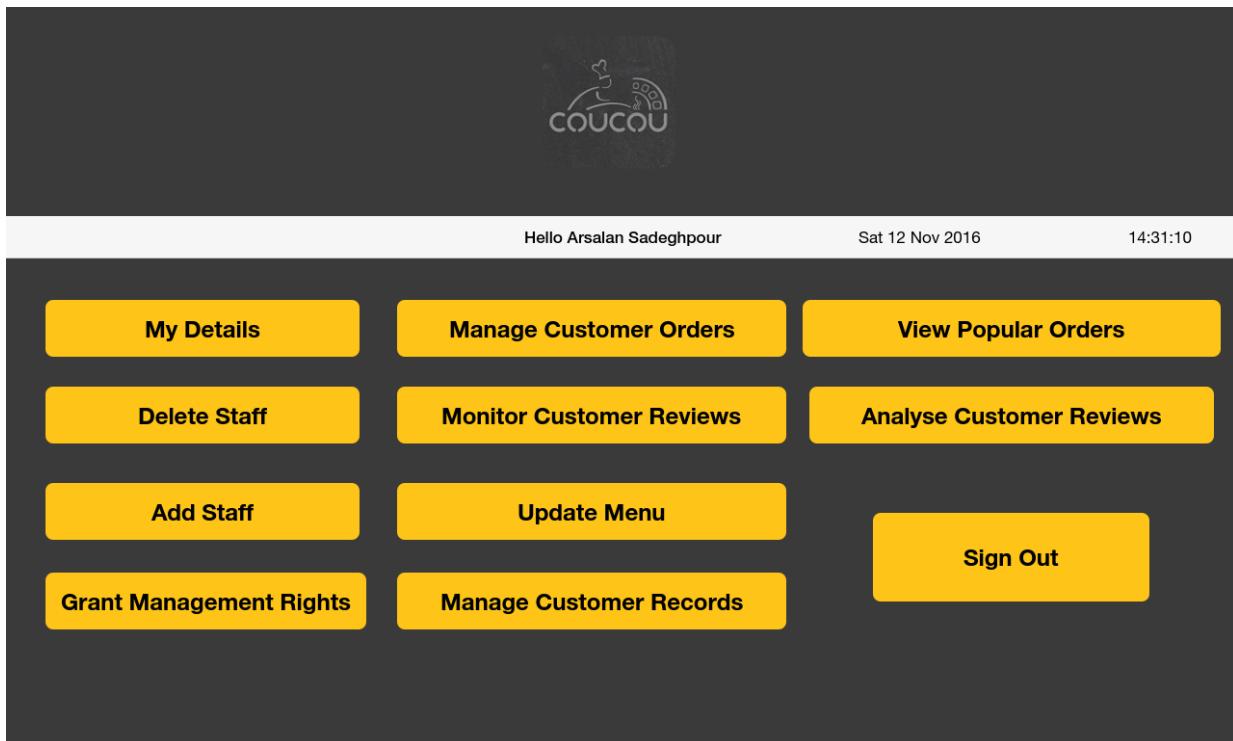


Figure 13: Business Interface Welcome Screen

Order Number	First Name	Surname	Mobile Number	Order Time	Order Type	Order Status		
1	Arsalan	Sadeghpour	07719292876	13:12	Delivery	Ordered	View Order Details	Update Order Status
2	Natalia	Beloff	07729292876	14:05	Collection	Ordered	View Order Details	Update Order Status

Figure 14: Business Interface Manage Customer Orders Screen

Prototype Design Feedback

The next step is to gain feedback from prospective users via a questionnaire from the interactive prototype which allowed users to simulate navigations and view the user interface in order to gain an understanding as to what they like, what needs improving and whether they enjoy using both interfaces. 11 users aged 18-24 responded to the app questionnaire and 5 members of staff aged 25-40 responded to the business interface questionnaire, therefore the sample and variety was not particularly large and this will be taken into account when analysing feedback. A detailed analysis of the questionnaire results can be found in Appendix D.

Questionnaire Results Summary

Mobile App Design

Overall the majority of customers liked the design with the exception of a few customers who stated the design was plain, tedious and preferred more interesting screen transitions. Customers suggested that to improve the design a map showing the order journey would have been more exciting. Overall the majority of customers would use the mobile app again, given some customers who stated it's design was plain.

Feature Requests

Overall customers prefer a personalised and convenient experience as sharing their orders to social networks was a useful feature request along with customisable order points to be used with any item, not just pizzas. Customers stated including suggestions to what they like based on their previous orders makes them feel special and individualised.

Brand Image

CouCou wish their brand image to be perceived as higher quality, fresher and more authentic than their competitors through their app. Overall customers felt there was no singular appeal in that it appears neither better or worse than their competition, however some customers admitted the food looks a lot more wholesome than traditional companies, implying a high quality perception. Some customers felt to improve their perception of freshness, images of ingredients incorporated into the design would have helped.

Business Interface Design

The majority of staff members liked the design, with a minority stating the date and time doesn't add any value to the system. The majority of staff like the way the order status can be updated within tables and that all data within these tables is useful to them. In contrast some staff members felt viewing the selected order details in a pop up box may be more pleasant than viewing the order details at the bottom. The majority of staff felt upon completion of an order they prefer to have the ability to work on any order they choose as some orders are completed faster than others, which is an important suggestion given the importance of efficiency in this industry.

In relation to the extensions ,staff were shown a screen detailing what the analytics feature would look like and the majority preferred to have today, this month and this year as previous years were not useful as the business cannot react to this data.

IMPLEMENTATION

Upon designing the complete system's architecture and receiving feedback on the prototype, it is now appropriate to discuss the technologies suitable to conduct the implementation of the system. Upon completing the implementation a discussion of how key features were implemented will also occur.

Model - MySQL

MySQL was chosen to represent the data shared by both the mobile application and the business interface. The main reasons MySQL was selected over XML to represent the data model was because the data for both parts of the system is largely intended for automatic processing due to the purpose of order and review data and has a fixed structure, therefore the data tends towards being data-centric. In contrast XML would not have been beneficial as the performance of XML processing is not as efficient as MySQL for example when querying the data, and an XML structure is intended for when the data has a less regular structure prone to change.

The relational nature of MySQL means it can provide persistent and efficient storage of shared and organised data into tables, data integrity and can represent the large and complex relationships between different parts of the data effectively. MySQL allows a schema to be defined which aims to eradicate data redundancy by designing the structure of the table whilst also providing integrity constraints. This is beneficial as it enforces the data to be organised in a particular way.

MySQL is also more advantageous than NoSQL databases such as MongoDB for this particular system because the system requirements were clear and the data structure is clearly defined as opposed to NoSQL which is more appropriate for processing a large amount of unstructured data, and the greater flexibility offered in terms of the scalability of data by NoSQL databases is not a benefit at this time because the data won't be classified as large enough for its benefits to be realised, however when analytics features are to be implemented, it may be beneficial to migrate to a NoSQL database.

View - Bootstrap

Bootstrap is a responsive framework for HTML, CSS and JavaScript and was implemented to represent the user interface of both the mobile application and the business interface because of its responsive nature which adapts the interface depending on each device, whilst also providing clean, pre-styled and consistent interface components such as form components, buttons and themes. The fact the components are pre-styled made the development process faster as less time was needed to design the interface as Bootstrap's reusable components were easy to implement. If the business interface were to be adapted to support mobile devices, the flexibility of Bootstrap makes it easy to migrate between different devices.

Bootstrap was preferred over jQuery Mobile because jQuery Mobile was slow and the components looked outdated and a lot less modern than Bootstrap which would have affected customer satisfaction as usability and user experience are important. The components provided by jQuery Mobile were also a lot more difficult to customise to provide a unique feel to the application.

Controller - PHP

PHP was used as the scripting language that will connect to the model, query the database and output the relevant HTML to display to the user on the mobile application. The main reasons why PHP was selected over JSP and JDBC to interact with the database is because PHP it is relatively simple to learn and given JSP is based on Java it requires a more complex structure and also has strict data type restrictions whilst PHP is less strict on the type of variables. JSP is also better used for larger enterprise level applications such as banking systems due to the complexity.

Client Side Validation

The ‘Validator’ JavaScript plugin was used because the design of the plugin integrated well with the design of the Bootstrap components. The plugin was implemented during the signup and login process of the mobile application, whilst also being implemented during the staff login page in the business interface. Client Side Validation provided visual feedback conforming to the design of Bootstrap by highlighting invalid fields, for example when the user did not enter a valid email address or a valid name whilst also providing support for confirming the email address and password fields by checking to see if both the original email address and password fields matched with their confirm counterparts.

Client Side Validation was performed to prevent extra burden on the server to ensure the data is well formatted before it is inserted into the database. The Validator JavaScript plugin is also used on the write review page to prevent the submission of empty reviews by disabling the ‘Submit Review’ button when no rating or description is given.

```
<div class="form-group">
    <label for="firstName">First Name</label>
    <input type="text" class="form-control" id="firstName" name="firstName" placeholder="First Name" pattern="^A-z{1,}$" required>
</div>
```

Figure 15: Example Validation - Regular Expression

In this example the use of the ‘pattern’ attribute allowed the ability to define a regular expression pattern to match the contents of the field with to ensure the name fields only consisted of upper and lowercase letters and not special characters or numbers.

Server Side Validation

The use of PHP scripts to connect to the MySQL database were implemented to provide validation so that for example a customer could not signup with an existing email address. If the email address was already existent in the database then the page would display an error message notifying the customer. If the email address was non existent in the database then an SQL query would be generated using PHP to insert the customer's details into the database and notifying the customer of a successful signup. During the login process the use of PHP scripts also provides validation by checking to see if the inputted password matches the stored password of that account or if the email address exists in the database and displays the relevant message.

```
// Get the email from the signup form
$email = $_POST["email"];

// check if email exists in DB
$sql = "SELECT email FROM Customer WHERE email='$email'";
$result = mysqli_query($conn, $sql);

// If email exists in DB display error and back button to signup form
if (mysqli_num_rows($result) > 0)
{
    echo "<div class=\"animstion\">";
    echo "<div class=\"appHeader\">";
    echo "<p id=\"headerText\">User Already Exists!</p>";
    echo "</div>";
}
```

Figure 16: Example Email Address In Use Server Side Validation

Password Security

In order to prevent customer passwords being stored into the database as plain text and having their accounts vulnerable to being compromised, passwords are hashed before they are inserted into the database using a hashing algorithm which makes determining the original plain text password difficult to obtain, therefore this hash can be used for comparing it with the original password.

PHP provides a password hashing API, specifically the function `password_hash(string password,bcrypt algorithm)` takes the inputted password and applies a hashing algorithm, specifically the bcrypt algorithm to the plain text password and returns the resulting hash. The resulting hash is then stored in the database instead of the plain text password, given the resulting hash's length can change over time, the resulting hash is stored in the database with a length of 255 characters to be on the safe side. The `password_verify(string password, string hash)` function takes a string representing a password and a string representing a hash, which is obtained from the database and verifies that the particular hash matches the inputted password and returns true if the password and hash match and false otherwise.

Figure 17: Example of Hashing Plain Text Password Entry

```
// add signup form details to DB
$firstName = $_POST["firstName"];
$lastName = $_POST["surname"];
$email = $_POST["email"];

$password = $_POST["password"];

$hash = password_hash($password,PASSWORD_DEFAULT);

$phone = $_POST["phone"];
$addressLine1 = $_POST["addressLine1"];
$addressLine2 = $_POST["addressLine2"];
$city = $_POST["city"];
$postcode= $_POST["postcode"];

$sql = "INSERT INTO Customer (firstName, lastName, email,password,phoneNumber,addressLine1,addressLine2,city,postCode)
VALUES ('$firstName', '$lastName', '$email','$hash','$phone','$addressLine1','$addressLine2','$city','$postcode')";
```

Figure 18: Example of Verifying Hashed Password Matches Entered Password

```
// Get the email and password from the login form
$email = $_POST["email"];
$pass = $_POST["password"];

// check if email exists in DB
$sql = "SELECT email FROM Customer WHERE email='$email'";
$result = mysqli_query($conn, $sql);

// if email exists in DB
if (mysqli_num_rows($result) > 0)
{
    $qry = "SELECT password FROM Customer WHERE email='$email'";
    $res = mysqli_query($conn, $qry);

    // if password exists for username
    if (mysqli_num_rows($res) > 0)
    {
        while($row = $res->fetch_assoc())
        {
            // check if entered password matches stored password hash
            $passwordMatch = password_verify($pass,$row["password"]);

            // if the password matches
            if ($passwordMatch == true)
            {
                // display welcome page
            }
        }
    }
}
```

Figure 19: Example of Hashed Password in Database

password

\$2y\$10\$QsdtrYqh6kx0ty.KPozsO.41UyJ24yzX1y.IXGnUHn.1Xe7gFm.3y

This figure shows how a customer's plain text password entered during the signup process such as 'Hello1234' is used to generate a resulting hash of the password which is then stored in the database representing the customer's password.

Session Management

PHP session variables were used in both the mobile application and the business interface to store data which will be retrieved by another PHP file at a later time. The reason for using sessions is due to the HTTP protocol being stateless, state is required to be kept between pages e.g. to keep track of the current items in the shopping cart when the user visits another page. The second usage of sessions is to store the customer's email address and staff username in session variables and use these to query the database when retrieving account details or inserting data. When the customer signs out of the mobile application/when a staff member signs out of the business interface all session variables are cleared and the session is destroyed using `session_unset()` and `session_destroy()`.

Session Creation - Mobile Application

```
// Store Logged In Customer's Email Address
$_SESSION["emailAddress"] = $email;

// setup array holding items in order cart
$addedItems = array();

$_SESSION["addedItems"] = $addedItems;
```

Figure 20: Session Creation for Email Address and Order Cart

In this code snippet of the mobile application a session is used to store the customers email address after they have logged in. This is used as part of an SQL query to extract the logged in customer's first name and surname to display on the welcome screen and to load their points. This code snippet shows the creation of an array which will store the customer's potential order cart items to keep their order cart history whilst the customer navigates the app.

Database Connectivity

```
// get the variable from the login handler page
$user= $_SESSION["username"];

// get the first name from the DB using the email address
$sql = "SELECT firstName, lastName FROM Staff WHERE username= '$user'";
$result = mysqli_query($conn, $sql);

// display welcome Staff Member!
if (mysqli_num_rows($result) > 0) {
    while($row = $result->fetch_assoc())
    {
        echo " ";
        echo $row["firstName"];
        echo " ";
        echo $row["lastName"];
    }
}
```

Figure 21: Utilising Session Variables To Extract Account Details

In this snippet the staff's username stored in a session variable is used as part of an SQL query to retrieve the first name and surname as part of the welcome page, similarly this approach is used for the welcome screen of the mobile application.

Managing The Order Cart

```
// Get the item data
$extra = $_POST["toppings"];
$size = $_POST["size"];
$price = $_POST["price"];
$quantity = $_POST["quantity"];

// Create an array from the item data

$_SESSION["itemName"] = array($_SESSION["itemName"], $extra, $size, $price, $quantity);

$newarray = $_SESSION["addedItems"];

// insert that item array into the array representing the cart items

$newarray[count($newarray)+1] = $_SESSION["itemName"];

$_SESSION["addedItems"] = $newarray;
```

Figure 22: Adding an Item To The Order Cart

In this snippet once the customer has added an item to their cart a POST request is sent to this page where the details of the item added are extracted and stored within an array which is then inserted into the session array holding the cart items. This cart array will be used to later display the added items to the customer on the order confirmation page.

```
// get the array of cart items where each index holds another array of item details
$addedItems = $_SESSION["addedItems"];

foreach ($addedItems as $value)
{
    echo "<tr>";
    echo "<td>";
    echo $value[0];
    echo "</td>";
    echo "<td>";
    echo $value[1];
    echo "</td>";
    echo "<td>";
    echo $value[2];
    echo "</td>";
    echo "<td>";
    echo $value[3];
    echo "</td>";
    echo "<td>";
    echo $value[4];
    echo "</td>";
    echo "<td>";
    echo "<button type=\\"button\\" class=\\"btn btn-danger\\" onclick=\\"removeItem(this)\\\">Remove</button>";
    echo "</td>";
    echo "</tr>";
}
```

Figure 23: Displaying Cart Items in The Order Confirmation Screen

In this snippet the contents of the session array representing the customer's cart items holds menu item objects, these objects are accessed iteratively and their details such as their name, price, size are rendered to the customer within an HTML table.

PHP and MySQL Database Connectivity

Dynamically Displaying Menu Data

```
$itemName = $_POST['menuItem'];

$sql = "SELECT * FROM MenuItem WHERE itemCategory='$itemName'";

$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0)
{
    $index = 1;

    while($row = mysqli_fetch_assoc($result))
    {
        $addedItems = $_SESSION["addedItems"];

        if (empty($addedItems))
        {
            echo "<a href=\"menu-item.php?itemName=\"";
            echo $row["itemName"];
            echo "\\"";
            echo "class=\"list-link\"><span class=\"list-link-label\">". $row["itemName"]. " </span><span class=\"right-arrow\">&gt; </span></a>";
        }
        else
        {
            $itemDetails = $addedItems[$index];

            $arraySize = count($addedItems);

            if ($itemDetails[0] != $row["itemName"] && $index <= $arraySize)
            {
                echo "<a href=\"menu-item.php?itemName=\"";
                echo $row["itemName"];
                echo "\\"";
                echo "class=\"list-link\"><span class=\"list-link-label\">". $row["itemName"]. " </span><span class=\"right-arrow\">&gt; </span></a>";
            }
            else
            {
                if ($arraySize > $index)
                {
                    $index++;
                }
            }
        }
    }
}
```

Figure 24: Dynamic Generation of Menu Data

In this snippet an SQL query is written within PHP where all menu items are retrieved from the database for a specific category of items e.g. pizza and are then displayed on the menu page, however only those menu items that are not in the customer's current order cart will be displayed. This is achieved by checking each item's name and determining whether or not it exists in the session array representing the customers order cart. This was performed to prevent the possibility of customers ordering duplicate items, instead they should use the quantity selectors for a specific menu item.

Order Creation

```
// get the data to insert into the order
$email = $_SESSION["emailAddress"];
$customerID = "";
$date = date("Y-m-d");
$time = date("h:i:s");
$orderType = $_POST["orderType"];
$requirements = $_POST["additionalRequirements"];
$points = $_POST["currentPoints"];
$gained = $_POST["pointsGained"];
```

Figure 25: Order Data

```
$addedItems = $_SESSION["addedItems"];

foreach ($addedItems as $value)
{
    $itemID = "";
    $extraID = NULL;
    $sizeID = "";
    $itemQuantity = "";

    // get the item id
    $abc = "SELECT itemID FROM MenuItem WHERE itemName = '$value[0]' ";
    $res= mysqli_query($conn, $abc);

    if (mysqli_num_rows($res) > 0)
    {
        while($row = mysqli_fetch_assoc($res))
        {
            $itemID = $row["itemID"];
        }
    }
}
```

Figure 26: Example of Extracting The Item ID of a Menu Item

Firstly data is sent from the order confirmation page via a POST request to a PHP file where the data in Figure 25 is received. The customer ID will be retrieved via an SQL query using the session variable storing the customer's email address and The date and time are generated fields. The data in Figure 25 is then inserted into the Order table via an SQL query.

The next step as shown in Figure 26 is to extract all menu items from the session array storing the customer's order cart and insert each item into the database by getting the itemID and extraID information if available, the sizeID of the item and the quantity of that item respectively via separate SQL queries and inserting each result of these queries into the OrderItems table via another SQL query.

Review Submission

```
$qry = "SELECT MAX(orderID) FROM Review WHERE customerID = '$customerID'";
$res = mysqli_query($conn, $qry);

if (mysqli_num_rows($res) > 0)
{
    while($row = mysqli_fetch_assoc($res))
    {
        $orderID = $row["MAX(orderID)"];
        $_SESSION["currentOrderID"] = $orderID;
        $_SESSION["reviewItemNames"] = array();
        $index = 0;

        $dta = "SELECT MenuItem.itemName FROM MenuItem INNER JOIN Review ON MenuItem.itemID = Review.itemID WHERE orderID = '$orderID' AND rating IS NULL";
        $ans = mysqli_query($conn, $dta);

        if (mysqli_num_rows($ans) > 0)
        {
            while($row = mysqli_fetch_assoc($ans))
            {
                $itemName = $row["itemName"];
                $_SESSION["reviewItemNames"][$index] = $row["itemName"];
            }
        }
    }
}
```

Figure 27: Extracting The Order ID of Latest Customer Order & Storing All Reviewed Menu Items

Firstly to display the menu items available for review by the customer, the latest order by the customer is retrieved and then those menu item names which have not been reviewed by the customer are placed into a session array.

```
$ratings = $_POST['itemRating'];
$descriptions = $_POST['reviewDescription'];
$orderID = $_SESSION["currentOrderID"];
$itemNames = $_SESSION["reviewItemNames"];

$count = count($itemNames);

for ($x = 0; $x <= $count; $x++)
{
    $itemRating = $ratings[$x];
    $reviewDescription = $descriptions[$x];
    $theItem = $itemNames[$x];
    $theItemID = "";

    // get the id for the name
    $qry = "SELECT itemID FROM MenuItem WHERE itemName = '$theItem'";
    $result = mysqli_query($conn, $qry);

    if (mysqli_num_rows($result) > 0)
    {
        while($row = mysqli_fetch_assoc($result))
        {
            $theItemID = $row["itemID"];
        }
    }

    // use the id to update the review for the item and the order id
    $sql = "UPDATE Review SET rating = '$itemRating', description= '$reviewDescription' WHERE orderID = '$orderID' AND itemID = '$theItemID'";
}
```

Figure 28: Review Submission Into Database

In this snippet the Item ID's of all the menu item names which have been reviewed are extracted from the session array storing these names. The ratings for these menu items and their respective review descriptions are stored in respective arrays which are accessed and used as part of an SQL update query to store the ratings and descriptions into the database.

Viewing Review Data and Staff Responses

```
$orderIDText = $_POST['orderIDButton'];
$theString = substr($orderIDText,6,2);
$orderID = trim($theString,":");

$qry = "SELECT MenuItem.itemName, Review.rating, Review.description, Review.staffResponse FROM MenuItem INNER JOIN Review ON MenuItem.itemID = Review.itemID WHERE orderID = '$orderID';"
```

Figure 29: Extracting Review Data and Staff Responses

When a customer wishes to view review data and staff responses for a specific order, they are presented with a list of buttons representing the order id. When the customer selects this, the code snippet in Figure 29 is run where it extracts the order id from the button and uses it to perform an SQL query which extracts all the review information for the order including staff responses (if any), otherwise staff responses are left blank.

AJAX and Document Object Model Manipulation

Live Tracking Orders

```
setInterval(function(){

    // value of num is determined by AJAX request to DB to get order status from PHP file
    // if status is ordered then num = 0.2
    // if status is preparing then num = 0.4
    // if status is cooking then num = 0.6
    // if status is out for delivery then num = 0.8
    // if status is delivered then num = 1.0

    var XMLHttpRequest = new XMLHttpRequest();
    XMLHttpRequest.onreadystatechange = function()
    {
        if (XMLHttpRequest.readyState == 4 && XMLHttpRequest.status == 200)
        {
            var num = XMLHttpRequest.responseText;
            bar.animate(num);    // Number from 0.0 to 1.0

            if (num == 0.2)
            {
                document.getElementById("message").innerHTML = "Order Recieved!";
                document.getElementById('trackingImage').src='images/tick.png';
            }

            if (num == 0.4)
            {
                document.getElementById("message").innerHTML = "Preparing Your Order!";
                document.getElementById('trackingImage').src='images/preparing.jpg';
            }
        }
    }
})
```

Figure 29: AJAX Request To Server Fetching Order Status

The implementation of AJAX leading to asynchronous requests meant that the customer did not need to refresh the page each time their order status is updated. In order to achieve this an AJAX request is created and is sent to the server. The server then starts an SQL query to fetch the order status of the latest order by that particular customer and responds with either 0.2, 0.4, 0.6, 0.8 or 1.0 and these numbers represent the different order statuses and are used to update the page by updating the progress bar to the relevant percentage which displays the relevant message and image, this is achieved using the Document Object Model.

Displaying Customer Orders and Customer Reviews

```
$orderStatus = "Delivered";

// Retrieve All Customer Orders
// Where Their Order Status Is Not Delivered Yet

$sql = "SELECT Orders.orderID, Customer.firstName, Customer.lastName, Customer.phoneNumber, orderTime, orderType, orderStatus FROM Orders INNER JOIN Customer ON
Orders.customerID = Customer.customerID WHERE orderStatus != '$orderStatus' GROUP BY orderID";

$result = mysqli_query($conn, $sql);

// Output Data Separated By | Symbol Back To Client

if (mysqli_num_rows($result) > 0)
{
    while($row = mysqli_fetch_assoc($result))
    {
        echo $row["orderID"];
        echo "|";
        echo $row["firstName"];
        echo "|";
        echo $row["lastName"];
        echo "|";
        echo $row["phoneNumber"];
        echo "|";
        echo $row["orderTime"];
        echo "|";
        echo $row["orderType"];
        echo "|";
        echo $row["orderStatus"];
        echo "|";
    }
}
```

Figure 30: AJAX Request Fetching Order Data

An AJAX request is sent to the server and the PHP file in Figure 30 which then queries the database via an SQL query to extract each row of orders where the order status is not delivered. The data is concatenated as a single string separated by a | symbol and sent back to the client for processing. The same procedure follows for review data as shown in Figure 31 where an AJAX request is sent to the server and the review data is retrieved via an SQL query obtaining review data which the staff have not responded to, concatenated by a | symbol and sent back to the client.

```
// Retrieve All Review Data
// Where Staff Have Not Responded Yet

$sql = "SELECT reviewID, Customer.firstName, Customer.lastName, rating, MenuItem.itemName, description FROM Review INNER JOIN Customer ON Customer.customerID =
Review.customerID INNER JOIN MenuItem ON Review.itemID = MenuItem.itemID WHERE staffResponse IS NULL";

$result = mysqli_query($conn, $sql);

// Output Review Data Separated By | Symbol

if (mysqli_num_rows($result) > 0)
{
    while($row = mysqli_fetch_assoc($result))
    {
        echo $row["reviewID"];
        echo "|";
        echo $row["firstName"];
        echo "|";
        echo $row["lastName"];
        echo "|";
        echo $row["rating"];
        echo "|";
        echo $row["itemName"];
        echo "|";
        echo $row["description"];
        echo "|";
    }
}
```

Figure 31: AJAX Request Fetching Review Data

```

var XMLHttpRequest = new XMLHttpRequest();

XMLHttpRequest.onreadystatechange = function()
{
    if (XMLHttpRequest.readyState == 4 && XMLHttpRequest.status == 200)
    {
        /* Retrieved Order Data as a complete string separated by the | symbol */
        var orderData = XMLHttpRequest.responseText;

        /* Extract Each Order Data Component */
        var theData = orderData.split("|");

        var table = document.getElementById("orderTable").getElementsByTagName('tbody')[0];

        /* Row Number Which Increments After Inserting Each Order Row Into Table */
        var rowNum = 0;

        /* Iterate Through Order Table */
        /* Update Table with Order Data */

        for (var i = 0; i < theData.length-1; i++)
        {
            var row = table.insertRow(rowNum);

            row.insertCell(0).innerHTML = theData[i];
            i++;
            row.insertCell(1).innerHTML = theData[i];
            i++;
            row.insertCell(2).innerHTML = theData[i];
            i++;
            row.insertCell(3).innerHTML = theData[i];
            i++;
            row.insertCell(4).innerHTML = theData[i];
            i++;
            row.insertCell(5).innerHTML = theData[i];
            i++;
            row.insertCell(6).innerHTML = theData[i];

            row.insertCell(7).innerHTML = "<button type=\"button\" onclick=\"updateStatus(this)\" class=\"btn btn-lg btn-default interfaceButton\">Update Status</button>";
            row.insertCell(8).innerHTML = "<button type=\"button\" id=\"a\" onclick=\"getOrderDetails(this)\" class=\"btn btn-lg btn-default interfaceButton\">View Order Details</button>";

            rowNum++;
        }
    }
}

```

Figure 32: Receiving Response From AJAX Request and Displaying Order Data

On receipt of the order data the client splits the concatenated data based on the | symbol and updates the table body by iterating through the data and placing each component of the separated data in the appropriate cell.

```

var XMLHttpRequest = new XMLHttpRequest();

XMLHttpRequest.onreadystatechange = function()
{
    if (XMLHttpRequest.readyState == 4 && XMLHttpRequest.status == 200)
    {
        /* Retrieved Review Data as a complete string separated by the | symbol */
        var reviewData = XMLHttpRequest.responseText;

        /* Extract Each Review Data Component */
        var theData = reviewData.split("|");
    }
}

```

The same procedure is applied when obtaining review data.

Figure 33: Receiving Response From AJAX Request and Displaying Review Data

Updating Order Status

```

/* Get Index of Selected Order From Table */
var index = i.parentNode.parentNode.rowIndex;

var request= new XMLHttpRequest();

/* Get Cell Displaying The Current Order Status */
var orderStatus = document.getElementById("orderTable").getElementsByName("tbody")[0].rows[index-1].cells[5].innerHTML;

// first name of selected row
var fname = document.getElementById("orderTable").getElementsByName("tbody")[0].rows[index-1].cells[0].innerHTML;

// time of selected row
var time = document.getElementById("orderTable").getElementsByName("tbody")[0].rows[index-1].cells[3].innerHTML;

request.onreadystatechange = function()
{
    if (request.readyState == 4 && request.status == 200)
    {
        displayUpdatedStatus(index);
    }
}

request.open("POST","update-status.php",true);
request.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
request.send("status="+orderStatus+"&theFName="+fname+"&theTime="+time);

```

Figure 34: AJAX Request For Selected Order To Update

When a staff member updates an order status: the row number of that order, the current order status and the first name and time of the order is obtained and sent to the server as parameters to the AJAX POST request. This is so the order to update can be uniquely identified by the first name and time of the order, the status is for the purpose of determine the next order status to set the order to. The row of the order is used to determine the order ID when querying the database.

```

// First Name Of Customer Who Ordered The Selected Order
$fname = $_POST['theName'];
// Order Time of Selected Order To Update Status For
$time = $_POST['theTime'];

// If Order Status Is 'Ordered'
// Update Order Status Of Selected Order To Next Status e.g. 'Preparing'
// Where First Name and Order Time Of The Order Matches First Name and Order Time of Selected Order
// Output Preparing Back To Client

if ($_POST['status'] == "Ordered")
{
    $updatedStatus = "Preparing";

    $sql = "UPDATE Orders SET orderStatus = '$updatedStatus' WHERE customerID = (SELECT customerID FROM Customer WHERE firstName='$fname') AND orderTime= '$time'";

    if (mysqli_query($conn, $sql))
    {
        echo "Preparing";
    }
}

```

Figure 35: PHP Response For Selected Order To Update

The first name and order time variables to the POST request are then used to update the status of that selected order to the appropriate status. The displayUpdatedStatus() JavaScript method then simply uses the row number belonging to that order to update the order status cell of that order using the Document Object Model.

Submitting Review Responses

```
/* Get The Selected Review Number */
var reviewID = document.getElementById("responseNumber").innerHTML;

var theID = reviewID.substring(7,8);

/* Get The Staff's Response */
var reviewDesc = document.getElementById("response").value;

/* If No Response Was Entered - Alert Staff Member! */
if (reviewDesc == "") {
    alert("Error: No Response Entered!");
}
else {
    var responseRequest = new XMLHttpRequest();

    responseRequest.onreadystatechange = function()
    {
        if (responseRequest.readyState == 4 && responseRequest.status == 200)
        {
            /* Delete The Review From The Database */
            /* Alert Staff Member That Response Was Saved */
            /* Reset Selected Review To First Review */
            /* Clear Response Text Area */
            alert("Response Saved!");
            document.getElementById("orderTable").deleteRow(theID);
            var reviewNum = document.getElementById("reviewTable").rows[1].cells[0].innerHTML;
            document.getElementById("responseNumber").innerHTML = "Review " + reviewNum + ": Response";
            document.getElementById("response").value="";
        }
    }

    responseRequest.open("POST","save-response.php",true);
    responseRequest.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
    responseRequest.send("reviewID="+theID+"&reviewDesc="+reviewDesc);
}
```

Figure 36: AJAX Post Request To Save Review Response

The selected review's review number is obtained along with the contents of the text field used to respond to the review and is sent to the server via parameters to the POST request. Validation is performed to determine whether or not a staff member is submitting an empty review. On submission the staff member is notified that the review was saved and the table is then updated by deleting the review and setting the review response area to the next available review.

```

// Get Selected Review ID
// Get Staff's Response
// Get Staff's Username

$reviewID = $_POST['reviewID'];
$reviewDesc = $_POST['reviewDesc'];
$staffUsername = $_SESSION["username"];

// Retrieve Staff ID of Staff Member Who Responded To The Review
// Used to Update Review Table Later

$qry = "SELECT staffID FROM Staff WHERE username = '$staffUsername'";
$result = mysqli_query($conn, $qry);

if (mysqli_num_rows($result) > 0)
{
    while($row = mysqli_fetch_assoc($result))
    {
        $staffID = $row["staffID"];

        // Update The Review Response for The Selected Review
        // Based On ID of Review Responded To

        $sql = "UPDATE Review SET staffResponse = '$reviewDesc', staffID = '$staffID' WHERE reviewID = '$reviewID'";
    }
}

```

Figure 37: Server Updates Review With Staff Response

The server uses the review id, staff response and the staff's username sent to the server via a POST request to update the review by setting the response to the entered staff response and assigning the staff member's staff id to the selected review via a SELECT query. The selected review to update is determined by the review id that was sent to the server.

```

// Get Review ID of Review to Delete
$reviewID = $_POST['reviewID'];

// Delete Selected Review
$sql = "DELETE FROM Review WHERE reviewID = '$reviewID'";

```

Figure 38: Deleting a Review

When a review is deleted the server simply runs a delete query on the server after the client has sent an AJAX POST request to delete the selected review determined by the review id sent by the client.

TESTING

The Testing Process

The complete system will now be tested after implementation to ensure it meets the functional and non functional requirements as specified in the requirements specification. The testing process consisted of:

- Unit Testing which involves testing individual components of the complete system which happens in parallel with development.
- Integration Testing to see if the mobile application and the business interface interact together, this is fundamentally important given the complete system consists of 2 core components which are required to integrate together.
- Acceptance Testing to see if both the mobile application meets the client's expectations and customers user experience expectations, and the business interface meets the requirements of the client and is usable by the business.

Unit and Integration Testing Results Summary

An overall test plan was developed which tests whether the final system has met all of the specified functional requirements plus other developmental requirements such as validation along with non functional requirements such as efficiency and enjoyability. The detailed results of the test plan can be found in Appendix E. Overall 53 test cases were created representing the functional requirements, in total the system has passed all 53/53 of these test cases.

User Acceptance Testing Results Summary

Acceptance Testing will require a selection of individuals to test the system. Customers are needed to test the mobile application and the form of testing will be Beta Testing where the developer is not present. The aim is to have an unstructured test allowing the customer to interact with the mobile application freely. The business will also beta test the business interface to ensure the user experience and usability is to their satisfaction and they will also be involved with testing the mobile application to see if it meets their requirements. 5 customers will be selected to test the mobile application and 2 staff members will be selected to test the web interface.

All 5 customers had no issues whilst testing all functionality of the mobile app. The 2 staff members had no issues logging into the system, navigating and updating customer orders and responding to customer reviews.

Overall 12 test cases were created representing the non functional requirements, in total the system has passed 9/12 of these test cases. The detailed results can also be found in Appendix E. The failed test cases were due to a lack of analytics data to test requirement 12 with because this has not been implemented, along with password hashing for staff data being non existent for requirement 13. It was not possible to test whether the system could handle additional branches so requirement 6 could not be tested. Finally some staff members reported the manage customer orders screen facing minor usability issues with the loading time of orders.

CONCLUSION

Process and Product Objective Evaluation

Overall the project process was successful on a personal level because the interaction with a real world business in order to provide a software solution to a real world problem helped me to develop essential skills crucial for my career in software engineering. I became exposed to developing software for the mobile sector which is highly valuable in this industry.

The final product was overall an important and useful system, which therefore provided success for the business as it has met the client's objectives by helping them to become more competitive with dominant firms in the market, whilst also creating a more enjoyable and rewarding experience for their customers alongside reducing costs from being able to rely on their own infrastructure. The finished product met 61 out of 65 total requirements, with the unsuccessful requirements being minor usability issues and other untested requirements such as scalability testing.

Overall the features such as a payment and analytics features which were not supposed to be implemented in the time frame of this project were still included in the original design so I could test the usefulness of those features with respective users (members of staff and customers) and because I intend to develop this system in the future. Overall I designed a much larger scale system and implemented a part of this system and this process allowed me to gain experience planning and developing an extensible software solution.

Project Improvements

If I was to carry out the project again I would have implemented the mobile application using a particular mobile application framework such as PhoneGap to make it easier to deploy the mobile application to various application stores instead of creating a mobile site.

In terms of the implementation process I would have aimed to follow the Model View Controller pattern a bit more closely by choosing to deploy server code via a PHP framework as throughout development content was generated manually using PHP by outputting HTML code, this made the application less maintainable as model code was tightly coupled with the view code.

I would also have chosen to implement both components of the system using a well known framework such as J2EE to separate the application into several tiers which would have allowed the database to become more flexible to change, which would be beneficial when the client wishes to move away from a relational structure due to business needs.

Skills Demonstrated and Acquired

In terms of skills demonstrated, I utilised project management skills such as time management as I had to implement the project whilst working on other university modules. I ensured the project was on track using project management software, whilst also developing communication skills during my interactions in the form of progress meetings with my client.

During the project process I was required to learn PHP during the implementation stage which was problematic at first due to learning the new syntax, however this proved to be a suitable choice as it was easy to learn and use when interacting with my database via writing SQL queries. I was also able to demonstrate technical skills I have acquired over other modules at university such as Software Engineering and Databases, particularly during the design and implementation phases.

Overall I was able to demonstrate that as a software professional I am able to develop a complex and extensible piece of software by breaking down the solution into smaller manageable components that can be implemented within a specific time window. I also gained experience integrating and testing its components, along with gaining experience working with a hybrid of the waterfall and agile software engineering model and a specific software design pattern, specifically the Model View Controller pattern. I was able to conduct research, design and develop my solution for extensibility in relation to the analytics features that were planned after successfully carrying out research.

During the development stage I learned that working with a real client results in challenges to do with continuously changing requirements that are needed to meet continuously changing business needs. I also learnt that as a result of this, not all functionality can be implemented effectively in time, therefore it is better to implement a high quality solution with fewer features than a solution which has plenty of features with bugs.

Overall I was able to demonstrate project planning skills because I planned to complete the project for the duration of 6 months, however I knew that after designing the complete system it was unrealistic to expect both the payment and analytics features to be implemented in the time allocated, therefore the complete large scale system would take 9 months to implement, however I included the extended features at the design stage so I could see and test the usefulness of those features with users. I completed 6 months of implementation work, but overall the design was planned for the complete system.

BIBLIOGRAPHY

1. Davenport, T., Short, J. (1990) 'The New Industrial Engineering: Information Technology and Business Process Redesign', *Sloan Management Review*, pp. 11-25, [Online] Available at: [http://is.ieis.tue.nl/education/bpmcourse/papers/Davenport%20\(1990\)%20-%20The%20New%20Industrial%20Engineering.pdf](http://is.ieis.tue.nl/education/bpmcourse/papers/Davenport%20(1990)%20-%20The%20New%20Industrial%20Engineering.pdf) [Accessed 3rd October 2016].
2. Gunasekaran, A., Nath, B. (1997) 'The role of information technology in business process reengineering', Elsevier, pp. 91-104 [Online] Available at: https://www.researchgate.net/profile/Angappa_Gunasekaran/publication/222493466_The_role_of_information_technology_in_business_process_reengineering/links/0c9605395cfb5a386c000000.pdf [Accessed 3rd October 2016]
3. Just Eat plc (2016) 'Benefits', [Online] Available at: <https://restaurants.just-eat.co.uk/benefits.html> [Accessed 3rd October 2016]
4. Domino's Pizza Group plc (2015) 'Annual Report & Accounts', [Online] Available at: <http://investors.dominos.co.uk/system/files/uploads/financialdocs/dominos-ar-2015.pdf> [Accessed 3rd October 2016]
5. Business Insider (2016) 'Papa John's thrives after embracing this hot tech trend', [Online] Available at: <http://www.businessinsider.com/papa-johns-grows-thanks-to-digital-and-mobile-app-ordering-2016-5?IR=T> [Accessed 3rd October 2016]
6. Just Eat plc (2016) 'Annual Report & Accounts 2015' [Online] Available at: <http://je-ict-live-corpse-site-assets-eu-west-1.s3.amazonaws.com/wp-content/uploads/2016/03/JUST-EAT-ARA-201516.pdf> [Accessed 3rd October 2016]
7. Jeston, J. and Nelis, J. (2014) *Business Process Management*. 3rd edn. London, United Kingdom: Routledge. [Accessed 4th October 2016]
8. British Computer Society (2016) *BCS Code of Conduct* [Online] Available at: <http://www.bcs.org/category/6030> [Accessed 5th October 2016]
9. Gov.UK (2016) 'The Data Protection Act' [Online] Available at: <https://www.gov.uk/data-protection/the-data-protection-act> [Accessed 5th October 2016]
10. Delaget (2016) 'Metrics That Matter to Restaurant Owners' [Online] Available at: https://www.delaget.com/hubfs/pdf/DG_MK_0009_Metrics_That_Matter_to_Restaurant_Owners_03.pdf?hssc=219873286.2.1478043891103&hstc=219873286.f63128873b53328730790afda3072cd1.1478043891103.1478043891103.1478043891103.1&hsfp=65780237&hsCtaTracking=25dc621c-f097-41ec-ad61-91ce096e61c3%7Ce2113c58-3a49-4571-b7a8-4a57bb7d6e79 [Accessed 6th October 2016]

11. Pixentia (2016) '*Predictive Analytics work to grow 50% sales year on year for a restaurant chain*' [Online] Available at: https://cdn2.hubspot.net/hubfs/1751195/Resources/Case_Studies/Case_Study_PDFs/CS12-Predictive_analytics_drives_restaurant_chain_sales_by_50_year_on_year.pdf?utm_source=hs_automation&utm_medium=email&utm_content=24834595&hsenc=p2ANqtz-FL06mreifCUFJX5e-yJ7lJO-sU33cNE7spkO84-e9GU2Lq19AJ9Zp0VJzj0ApIQBJqOwGGeqxN9DaMVq5vVduNgd_w&hs_mi=24834595 [Accessed 6th October 2016]
12. Fishbowl (2016) '*Targeting with Detailed Guest Personas*' [Online] Available at: <https://www.fishbowl.com/case-studies-targeting-with-detailed-guest-personas/> [Accessed 6th October 2016]
13. Fishbowl Restaurant Analytics (2016) [Online] Available at: <http://fishbowl.com> [Accessed 8th October 2016]
14. Upserve Restaurant Analytics (2016) [Online] Available at: <https://upserve.com> [Accessed 8th October 2016]
15. Posist Restaurant Management Platform (2016) [Online] Available at: <http://www.posist.com> [Accessed 8th October 2016]
16. Henman, R. (2016) *Designing Platform Independent Mobile Apps and Services*, Wiley
17. Panhale, M. (2015) *Beginning Hybrid Mobile Application Development*, Apress
18. Fling, B. (2009) *Mobile Design and Development*, O'Reilly
19. Bootstrap (2017) [Online] Available at: <http://getbootstrap.com> (Accessed 12th December 2016)
20. Flat UI Colours (2017) [Online] Available at: <http://flatuicolors.com> (Accessed 16th December 2016)
21. Flat Icon '*Back Icon*' (2017) [Online] Available at: http://www.flaticon.com/free-icon/left-arrow_160760#term=back%20icon&page=1&position=32 (Accessed 17th December 2016)
22. Flat Icon '*Shopping Cart*' (2017) [Online] Available at: http://www.flaticon.com/free-icon/shopping-cart_60992#term=Shopping%20Cart&page=1&position=24 (Accessed 17th December 2016)
23. Flat Icon '*Plus Icon*' (2017) [Online] Available at: http://www.flaticon.com/free-icon/plus_148764#term=plus&page=1&position=3 (Accessed 17th December 2016)
24. Flat Icon '*Error Icon*' (2017) [Online] Available at: http://www.flaticon.com/free-icon/error_148766#term=cross&page=1&position=1 (Accessed 17th December 2016)
25. Flat Icon '*Add Icon*' (2017) [Online] Available at: http://www.flaticon.com/free-icon/plus_148764#term=add&page=1&position=4 (Accessed 17th December 2016)
26. Flat Icon '*Subtract Icon*' (2017) [Online] Available at: http://www.flaticon.com/free-icon-minus_148765#term=subtraction&page=1&position=2 (Accessed 17th December 2016)

27. Google Fonts (2016) [Online] Available at: <https://fonts.google.com/specimen/Source+Sans+Pro> (Accessed 18th December 2016)
28. Animsition (2016) [Online] Available at: <http://git.blivesta.com/animstion/> (Accessed 19th December 2016)
29. Validator (2016) [Online] Available at: <https://1000hz.github.io/bootstrap-validator/> (Accessed 19th December 2016)
30. Progress Bar (2016) [Online] Available at: <https://kimmobrunfeldt.github.io/progressbar.js/> (Accessed 19th December 2016)
31. jQuery Bar Rating (2016) [Online] Available at: <http://antenna.io/demo/jquery-bar-rating/examples/> (Accessed 19th December 2016)
32. iOS Menu Navigation (2016) [Online] Available at: <https://codepen.io/ifthenelse/pen/pvRVP?editors=1000> (Accessed 20th December 2016)
33. MySQL (2016) [Online] Available at: <https://www.mysql.com> (Accessed 21st December 2016)
34. PHP (2016) [Online] Available at: <http://php.net> (Accessed 21st December 2016)
35. PHP Password Hashing API (2016) [Online] Available at: <http://php.net/manual/en/book.password.php> (Accessed 22nd December 2016)

APPENDIX A - PROJECT LOG

Date	Discussions	Work to be Implemented
6th October 2016	<ul style="list-style-type: none"> • Project Proposal Accepted. • Provided outline for Project Plan and HCI Evaluation Example for 3 Websites. It is suggested to implement a similar document for 3 Mobile Apps. • Conduct an interview with the client to gain a further understanding of the requirements for the project. 	<ul style="list-style-type: none"> • Digitise Project Plan into Gantt Chart. • Draft of Interim Report Introduction • Draft HCI Evaluation for Requirements Analysis. • Conduct requirements elicitation interview with CouCou Food Restaurant Manager.
13th October 2016	<ul style="list-style-type: none"> • Introduction detailing reason for BPR, Sales Analytics suitable. • Include a brief overview of client's current operations and overview of competitors such as Domino's, Papa John's and Just Eat in the Introduction. • HCI Evaluation was suitable. • Expand use case diagram to include sales analytics backend. • Object Orientation may not be necessary, class diagram should be implemented as an ERD Diagram. • Provide a Model View Controller overview in design section for a better understanding of the system architecture. 	<ul style="list-style-type: none"> • Include brief overview of competitor history in Introduction. • Add Sales Analytics functionality to Use Case Diagram. • Develop an ERD Diagram representing data storage. • Add a description detailing the use of the MVC design pattern.
20th October 2016	<ul style="list-style-type: none"> • Suitable approach to the Professional Considerations section was discussed including the need to reference the BSC Code of Conduct in relation to project and a good example was provided. • Expand and Show Sequence of Steps in Use Case Diagram. • Number and Categorise Requirements to easily refer back to for testing purposes • Provide descriptions for design diagrams in relation to MVC. • Research existing sales analytics software features, strengths and weaknesses and how this should be taken into account for the project 	<ul style="list-style-type: none"> • Write professional considerations section referencing BSC code of conduct. • Expand Sequence of steps in Use Case Diagram. • Categorise and Number Requirements Spec. • Provide short descriptions for design diagrams referencing the component of MVC represented.
27th October 2016	<ul style="list-style-type: none"> • Improve flow of report by including a short statement before the requirements specification and a short statement after the interview transcript for reflection purposes • Improve formatting of Use Case Diagram increasing font size and positioning • Move software analysis to design section 	<ul style="list-style-type: none"> • Begin sequence diagrams taking each branch of use case diagram • Begin to wireframe mobile app, obtain client feedback and make required changes • Develop wireframe questionnaire for 5 users
3rd November 2016	<ul style="list-style-type: none"> • Interim Report is ready for submission 	N/A

Date	Discussions	Work to be Implemented
10th November 2016	<ul style="list-style-type: none"> Developed mobile app questionnaire for customers and would like some feedback Developed mobile app and analytics interface prototypes in proto.io and would like some feedback 	<ul style="list-style-type: none"> Designs look good but add alternative colours for table rows to ease readability. Change questionnaire format to reduce open questions to bullet point answers as this will motivate customers to answer
17th November 2016	<ul style="list-style-type: none"> Deployed customer questionnaire and received feedback, would like to review the most appropriate way to present this feedback Designed analytics questionnaire for staff and would like some feedback 	<ul style="list-style-type: none"> Do not disregard responses with only 1 customer, just say "Due to time concerns I won't be implementing this now" Staff questionnaire looks good
24th November 2016	<ul style="list-style-type: none"> Sequence Diagrams, ERD and Use Case Diagram Final Designs Should I explain requirements changes in the report? What is the best way to describe my designs? Should I have a risk assessment section What happens if some requirements do not get implemented due to time? Would I be able to simulate payment processing, due to Stripe being too complicated to learn? Received staff responses to analytics questionnaire and would like some feedback on the most appropriate way to present this feedback 	<ul style="list-style-type: none"> Sequence Diagrams layout need readjusting Use Case Diagram is suitable The ERD requires a staff table No need for a risk assessment section No need to explain requirements changes in the report Provide an extensions section outlining the requirements which you won't have time for We will put a few complex designs in the body of the report and others go in the appendix Look at the PayPal Sandbox for payment processing Presentation of staff responses to analytics questionnaire are suitable
1st December 2016	<ul style="list-style-type: none"> Updated designs and ER diagram following the recent requests from my client to adapt the requirements slightly to include staff-customer review response functionality I have begun working on the implementation of the mobile app user interface for logging in, signing up and the welcome screen for the customer I have begun to write the implementation section of my report detailing the technologies I have chosen to complete my implementation 	<ul style="list-style-type: none"> The staff table in the ERD should link to the reviews table The prototypes look suitable Implementation section should detail why you have chosen a particular technology over another Implementation is suitable. Begin to develop the database functionality for next week
10th December 2016	<ul style="list-style-type: none"> I have now added 'View Customer Records' and 'Monitor Reviews' in the Use Case Diagram as requested along with changing 'Review Order' to 'Write Review' I have added a relationship from the staff table to the review table I have begun to implement the dynamic loading of menu items from the database in the mobile app 	<ul style="list-style-type: none"> Designs now look suitable Allow staff to edit and delete customer records Continue to work on implementation Carry on working on implementation section of report

Date	Discussions	Work to be Implemented
2nd February 2017	<p>The Mobile Application is now complete, since there is a little time try to complete as much of the business interface as possible prioritising order management and review responses as they are critical for the app's functionality.</p> <p>The implementation section looks good and concise descriptions are added, however try to add some more images which help explain the descriptions</p>	<ul style="list-style-type: none"> Implement as much of the analytics interface as possible - focus on core requirements of order management and review responses Aim to add more images explaining the implementation section Aim to clean up the application code by adding comments Start to look at preparing for the poster presentation by thinking about how to structure the poster
9th February 2017	<p>The draft poster presentation is a little bit too crammed. Suggested a SWOT structure for the poster to make it more effective.</p> <p>Reduce the design diagrams to only the important diagrams, the rest can go in the appendix.</p> <p>Reorganise the software requirements - suggested primary and secondary functional requirements</p>	<ul style="list-style-type: none"> Work on redraft of poster presentation using SWOT analysis Continue working on the Business Interface Restructure software requirements document
16th February 2017	<p>The draft poster quadrants should be labelled and it was suggested to use a circular presentation for the 5 stage process instead of bullet points.</p> <p>Ensure coding is finished by next week</p>	<ul style="list-style-type: none"> Make changes to poster Finish Business Interface
23rd February 2017	<p>Coding is now finished.</p> <p>Ensure requirements are separated into Core and Extensions</p> <p>Make the Poster's extensibility chart more appropriate by using a real example of a chart that would be used in the system.</p>	<ul style="list-style-type: none"> Separate software requirements into Core and Extensions Add a real chart to use in the poster for the analytics section Begin to Test the final system and document the test results into a test plan

Date	Discussions	Work to be Implemented
2nd March 2017	<p>Software Requirements have been separated into Core and Extensions</p> <p>Poster now includes screenshots of both analytics charts but is a little bland, poster could be more exciting.</p> <p>Began writing testing section and have documented results into a test plan</p> <p>Ensure Word count meets limit of 12000 words. Ideally have 10000 words, perhaps show a writing specialist</p>	<ul style="list-style-type: none"> Finish Testing section by next week Redraft of Poster by next week with additional screenshots and enhanced design Meet with writing specialist to discuss writing style to ensure every word is meaningful
9th March 2017	<p>Test Results have now been moved into the Appendix and a brief summary of results has been written.</p> <p>Questionnaire results for the prototype design have also been moved into the Appendix with a brief summary of results written.</p> <p>In total there are 50 pages of main content, totalling around 10,000 words. Looking to cut down on word count as the Project Review section is yet to be completed.</p> <p>Wrote a quick summary just before implementation glueing the design and implementation section together as discussed</p> <p>Poster now looks good, if possible make the 5 stage process colours be different as this adds more excitement to the poster.</p>	<ul style="list-style-type: none"> Start thinking about writing a rough draft of the Project Review by next week Ensure 5 stage process has different colours for each stage in the poster. Aim to submit poster before the deadline on 16th March 2017

Date	Discussions	Work to be Implemented
16th March 2017	<p>Move the Use Case Diagram as part of the Requirements Analysis as it's part of the initial requirements capturing process.</p> <p>Move the research conducted on analytics system also in the Requirements Analysis phase as it is research that will aid the requirements process.</p> <p>Rename the headings in the design section to reflect MVC pattern.</p> <p>Separate testing section into Unit and Integration testing where discussion of requirement results are conducted, whilst moving the description of customer and staff feedback to tests into the User Acceptance Testing section.</p> <p>Incorporate time management and ability to break down complex system into manageable chunks and the challenges of working with a client having constantly changing requirements into the conclusion</p>	<ul style="list-style-type: none"> Restructure the requirements analysis section by including the Use Case Diagram and the sales analytics research conducted. Rename headings to reflect MVC pattern. Make testing section clearer to understand by separating results summaries from the different types of testing conducted. Incorporate more project planning skills learned and challenges which I overcame whilst communicating with a client in the conclusion section
30th March 2017	<p>Final Report changes have been made</p> <p>After each section include a short summary of the previous section to improve the flow of the report</p> <p>How best to represent mentioning the lack of payments and analytics features in the implementation but it's existence in the use case diagram was discussed.</p> <p>Structure conclusion section to improve how it ends by moving project improvements to the middle and finishing the conclusion section on the skills demonstrated and acquired section.</p>	<ul style="list-style-type: none"> Mention that analytics and payment features are included as the system represents a long term implementation Start to think about planning for the short talk on 12th May. Have a draft of the presentation ready by 4th May.

APPENDIX B - ADDITIONAL SEQUENCE DIAGRAMS

Live Track Order - Mobile App

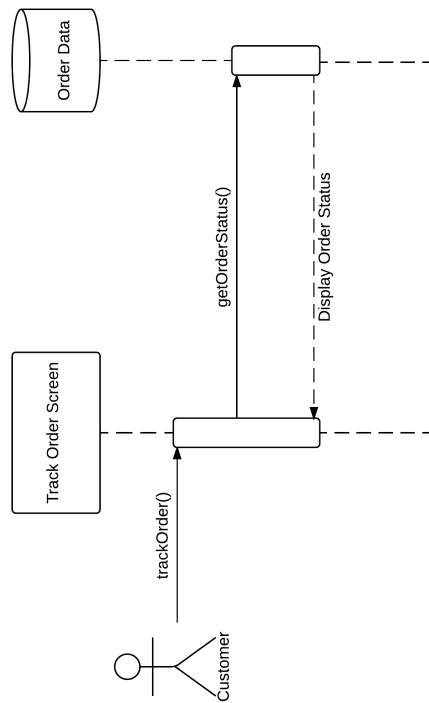


Figure B1: Live Track Order Sequence Diagram

View Loyalty Points - Mobile App

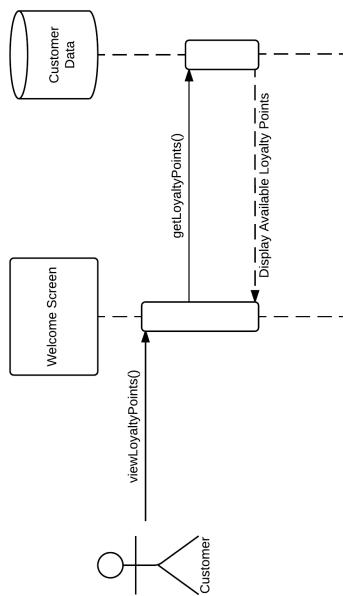
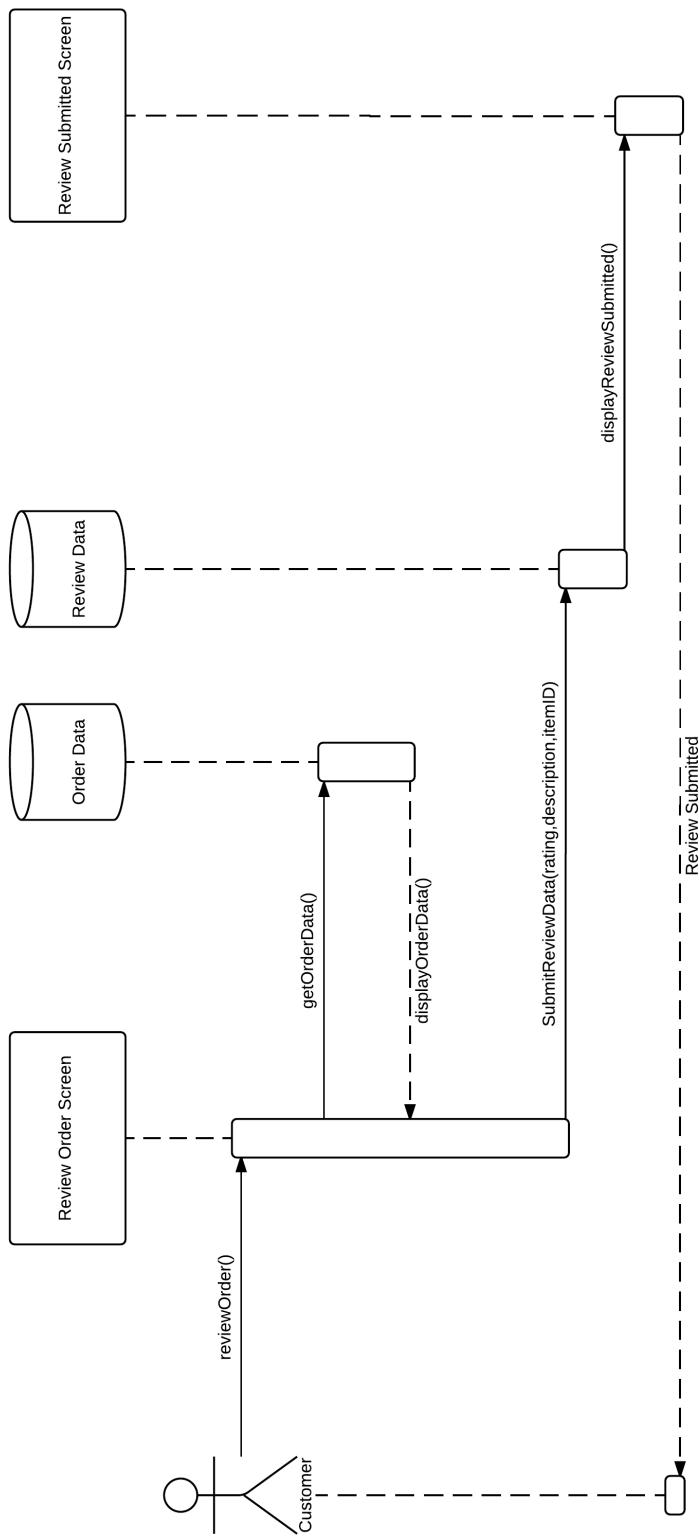
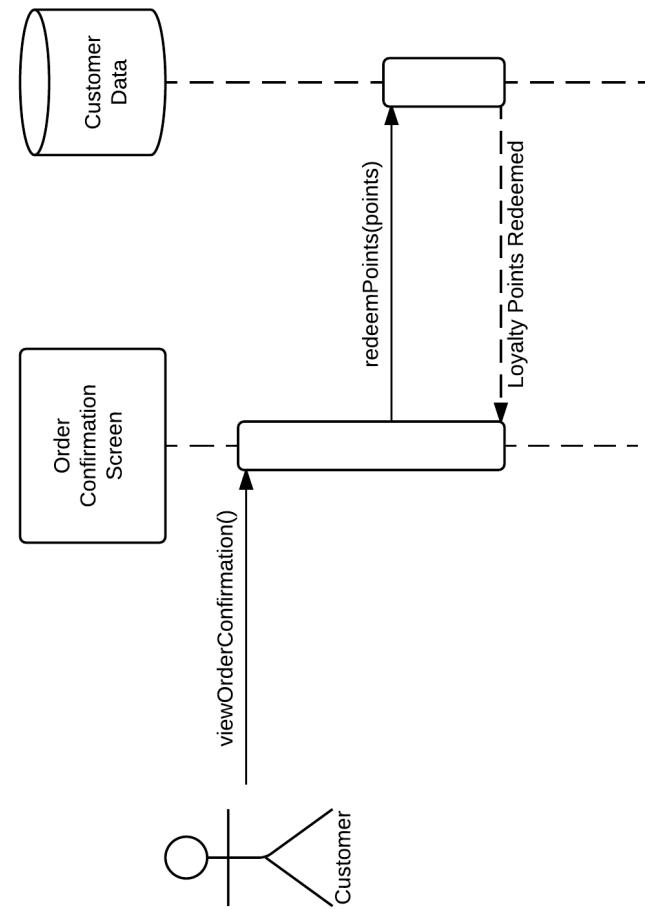


Figure B2: View Loyalty Points Sequence Diagram

Review Order - Mobile App*Figure B3: View Loyalty Points Sequence Diagram*

Redeem Loyalty Points- Mobile App*Figure B4: Redeem Loyalty Points Sequence Diagram*

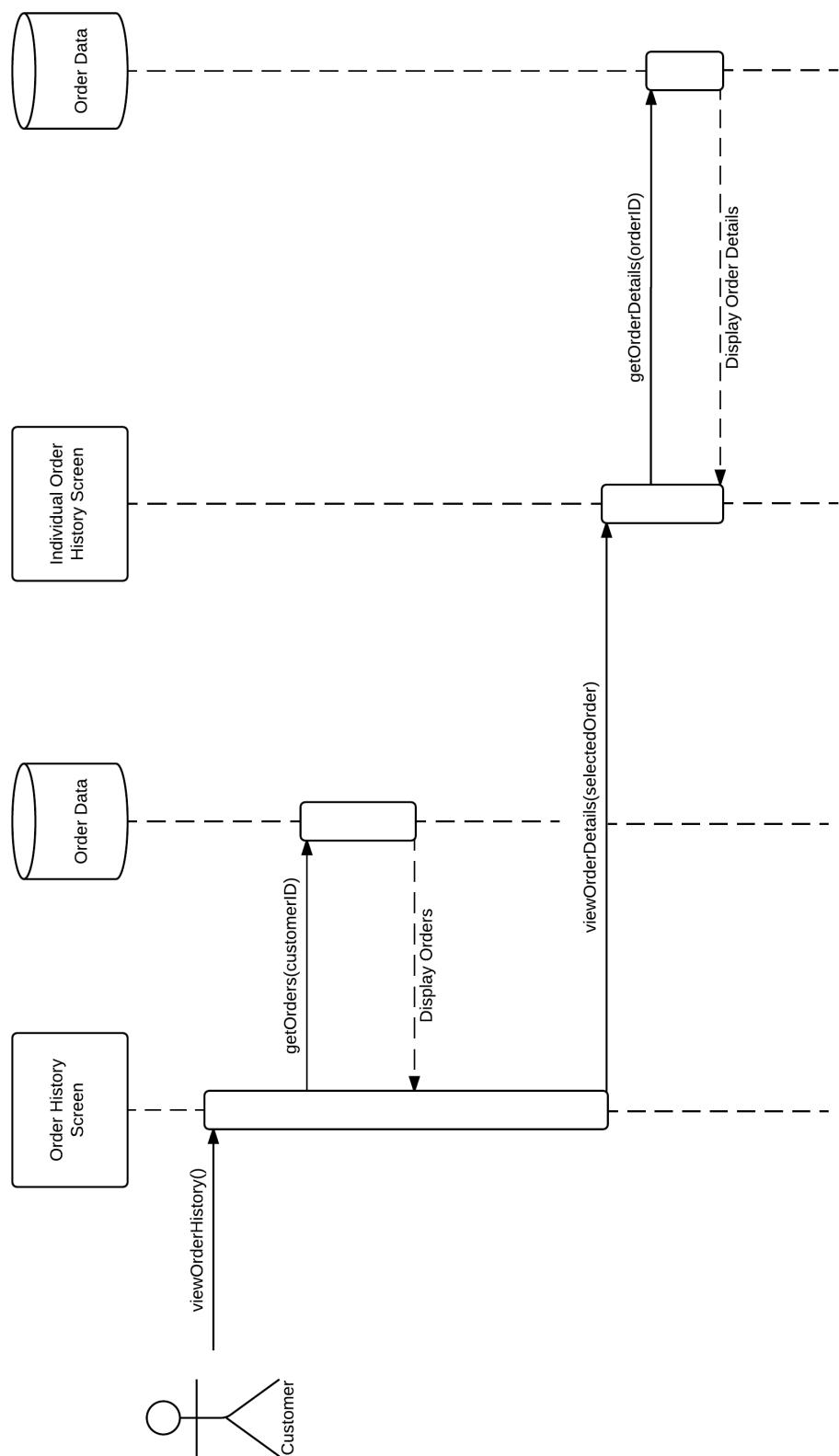
View Order History - Mobile App

Figure B5: View Order History Sequence Diagram

View Metrics of Popular Orders- Business Interface

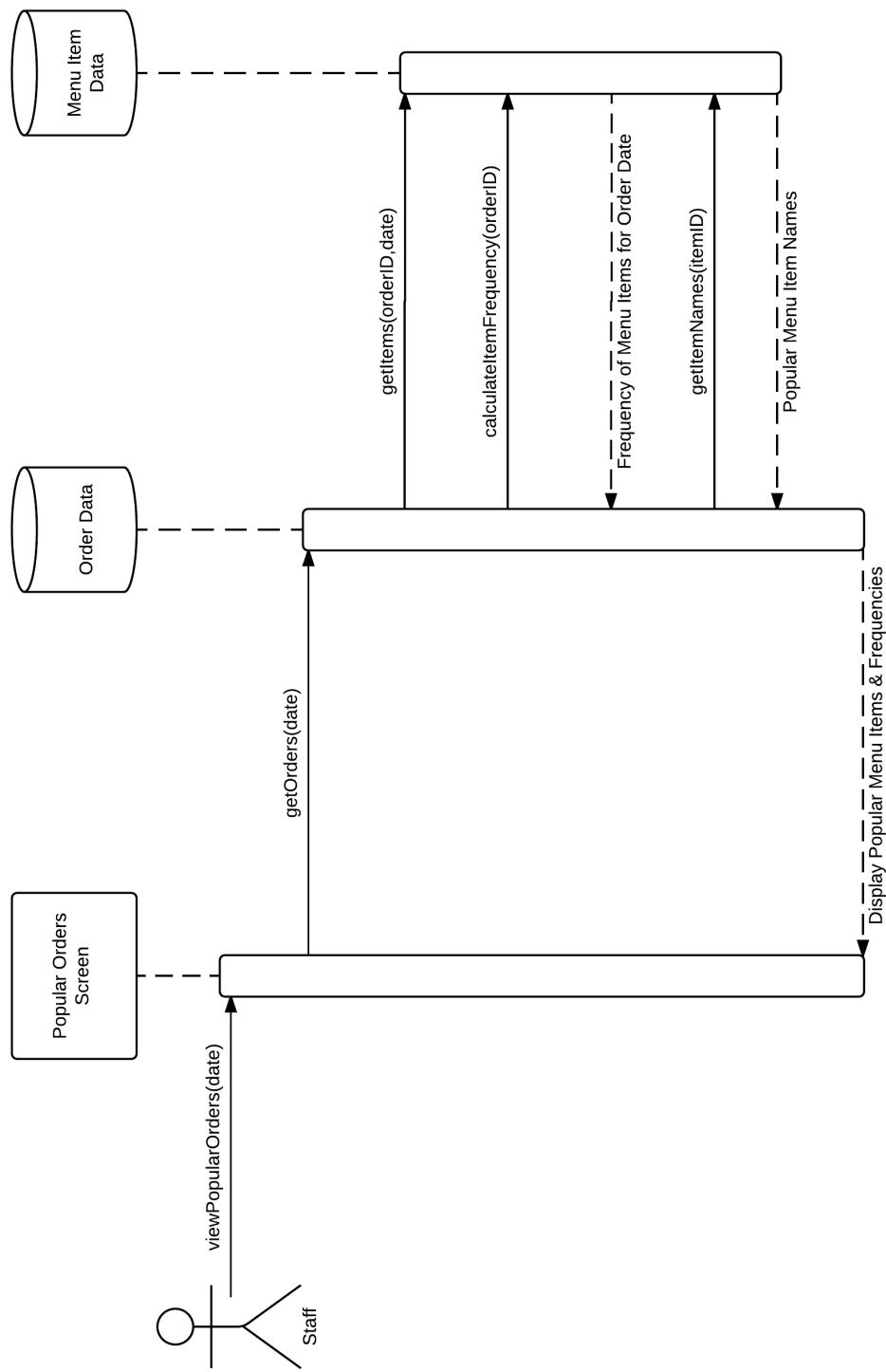


Figure B6: View Metrics of Popular Orders Sequence Diagram

Update Menu - Business Interface

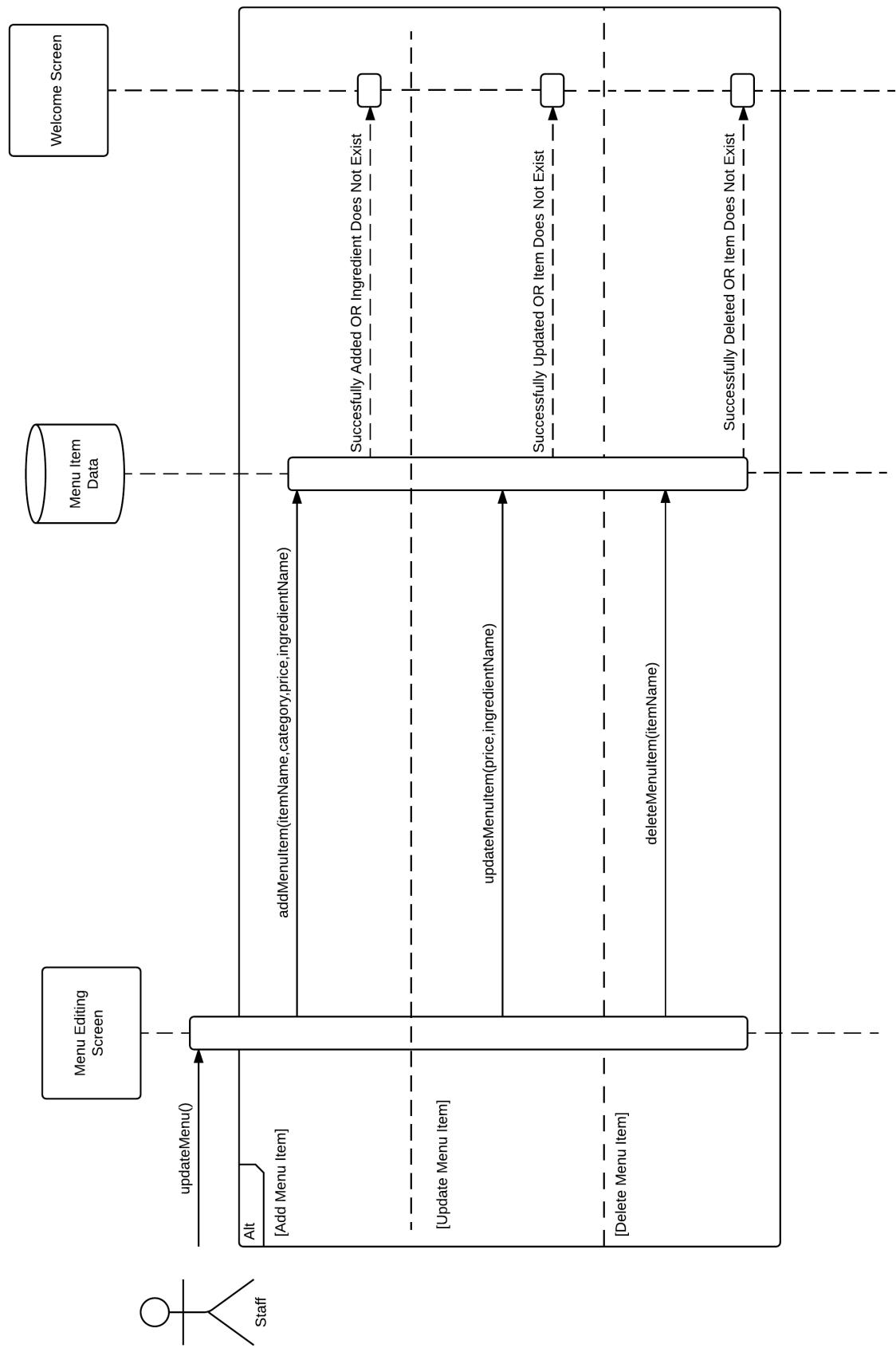


Figure B7: Update Menu Sequence Diagram

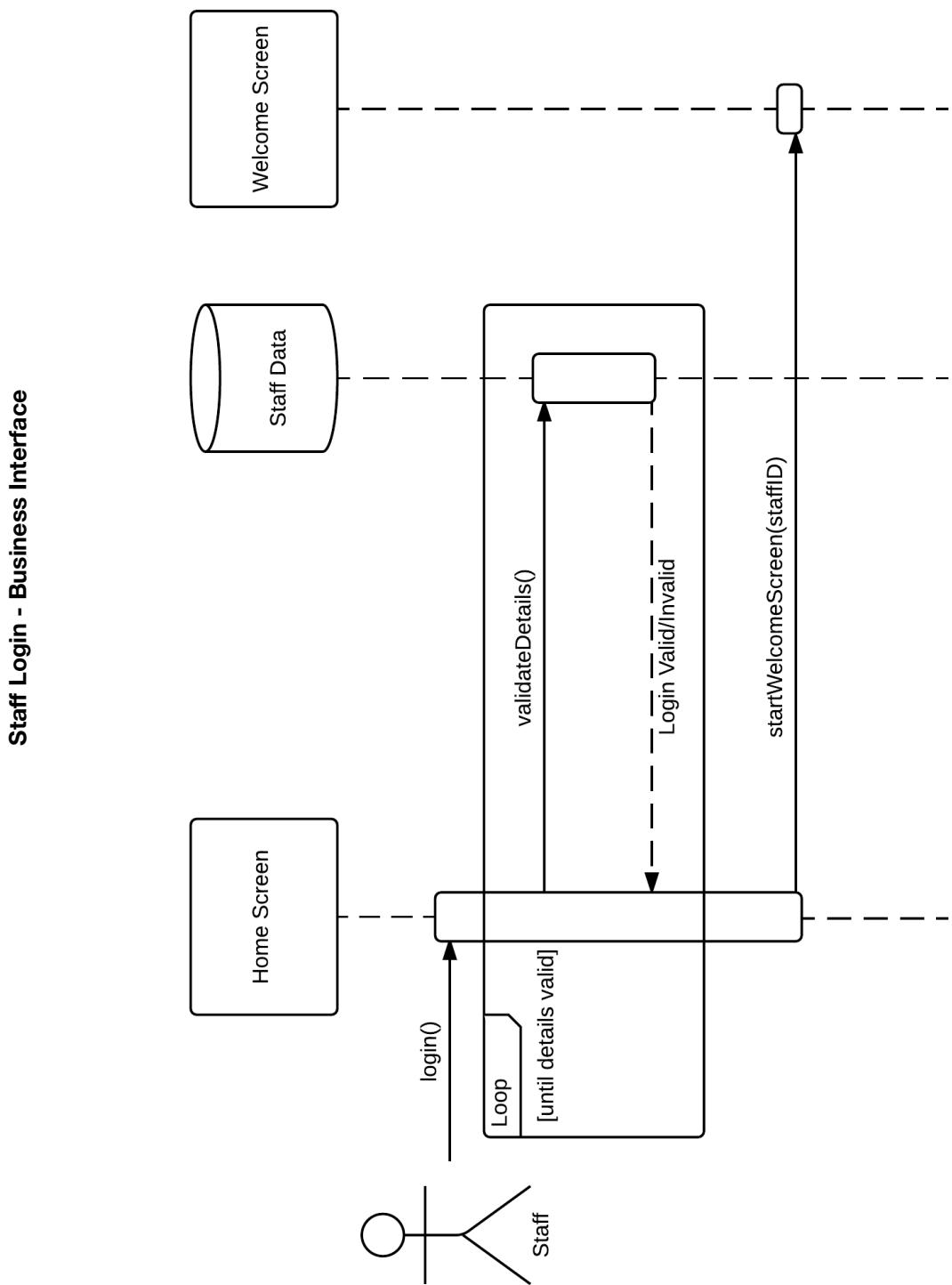


Figure B8: Staff Login Sequence Diagram

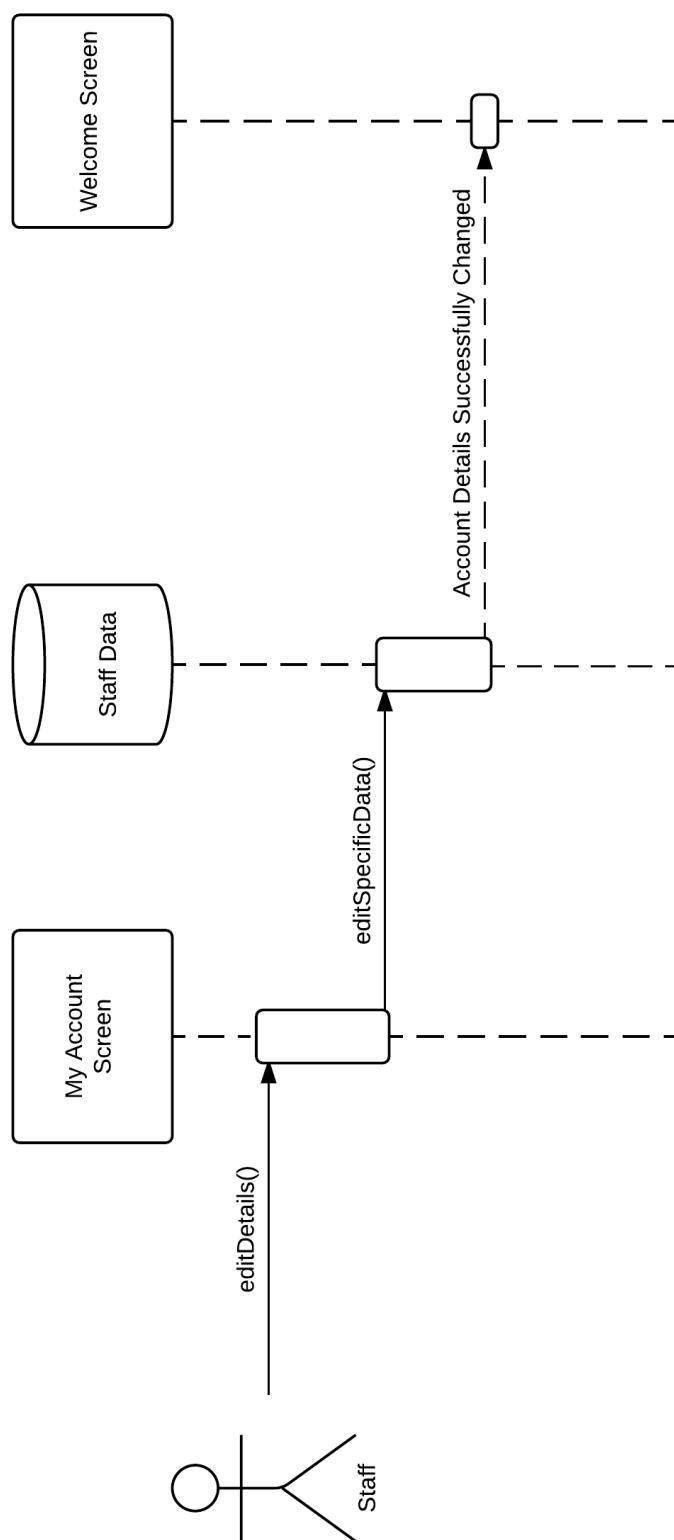
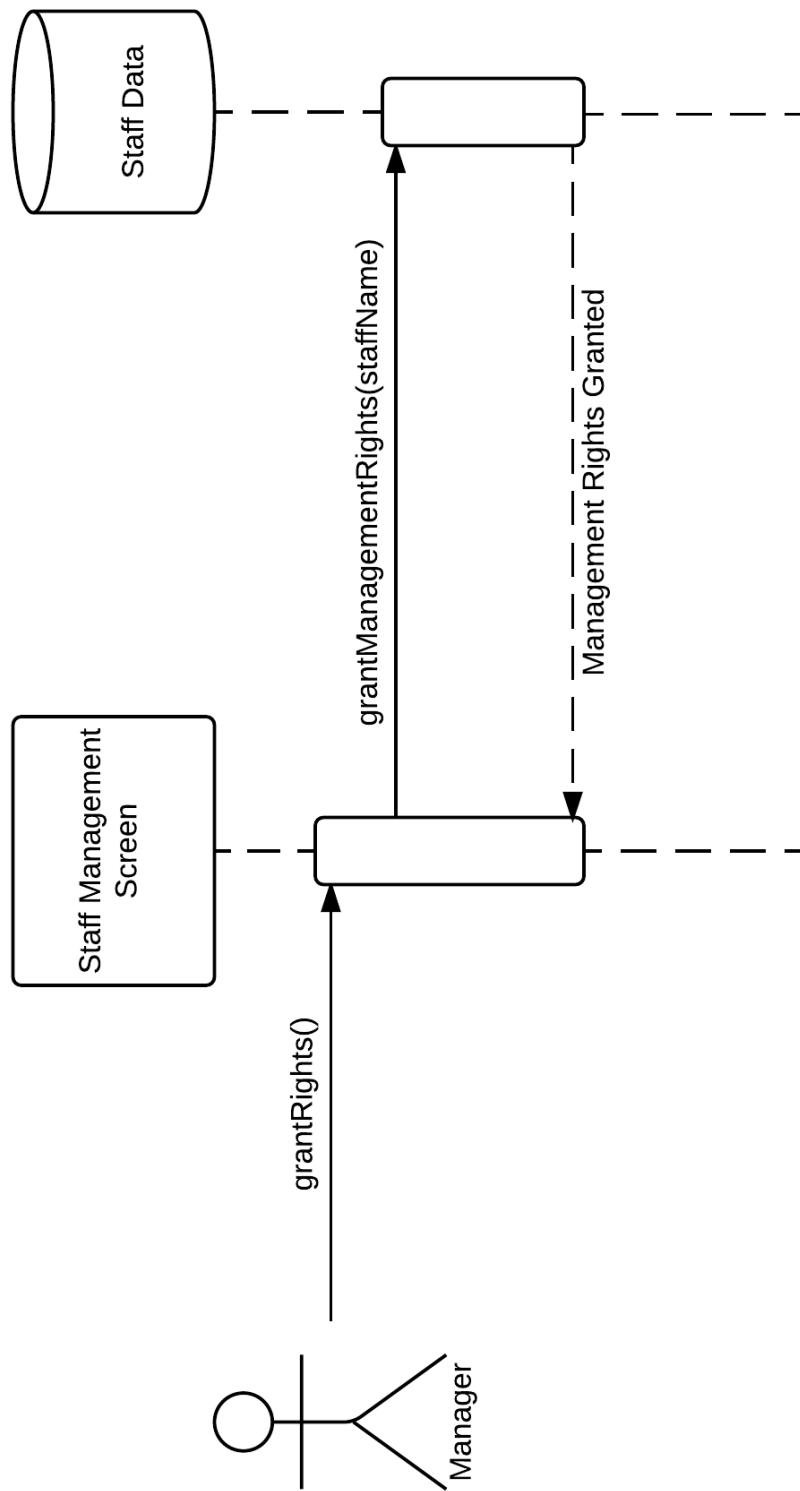
Edit Staff Details - Business Interface

Figure B9: Edit Staff Details Sequence Diagram

Grant Management Rights - Business Interface*Figure B10: Grant Management Rights Sequence Diagram*

APPENDIX C - ADDITIONAL PROTOTYPE DESIGNS

Mobile App

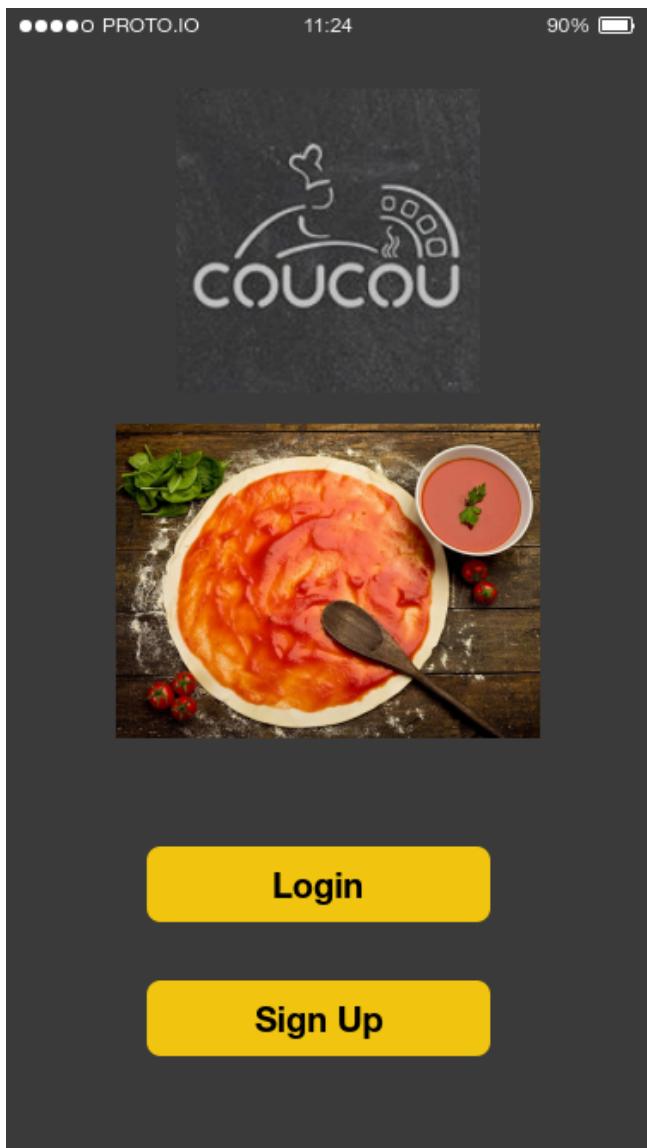


Figure C1: Home Screen

The Signup Screen of the COUCOU mobile application. The top bar shows five dots, "PROTO.IO", the time "13:18", and a battery icon showing 90%. A back arrow and the text "Sign Up" are at the top. The screen contains several input fields with placeholder text: "Email" (with a red error message "Email is required"), "Confirm Email Address" (empty), "Password" (empty), "Confirm Password" (empty), "Phone Number" (empty), "Address" (empty), and "Address Line 1" (empty).

Figure C2: Signup Screen

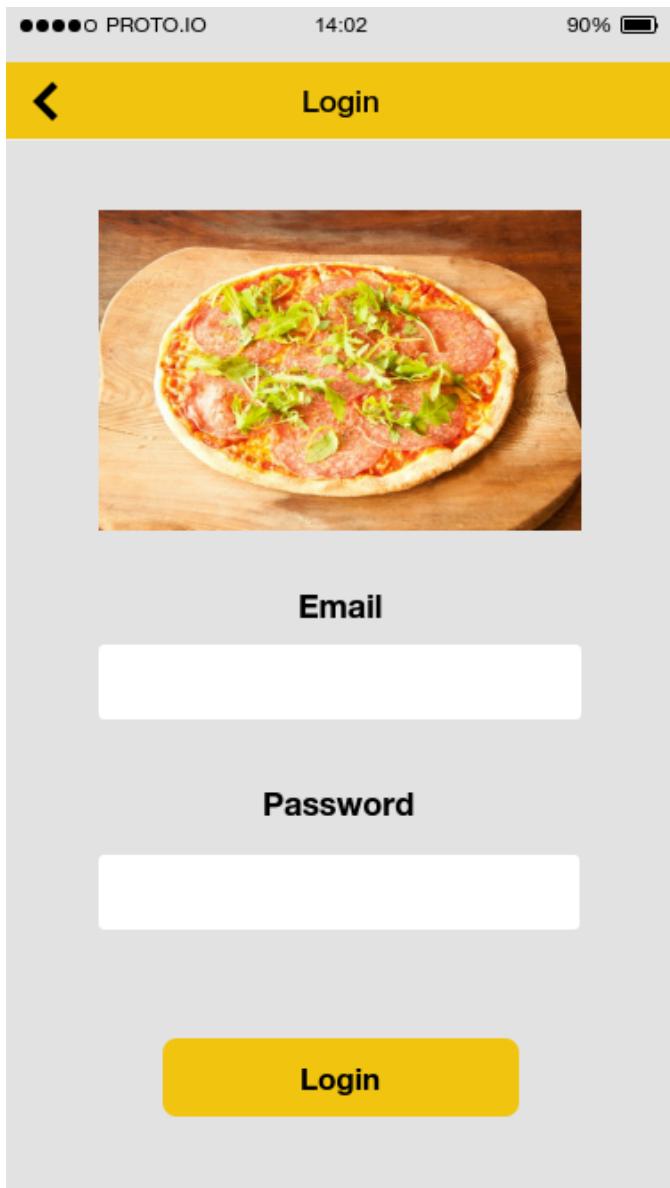


Figure C3: Login Screen

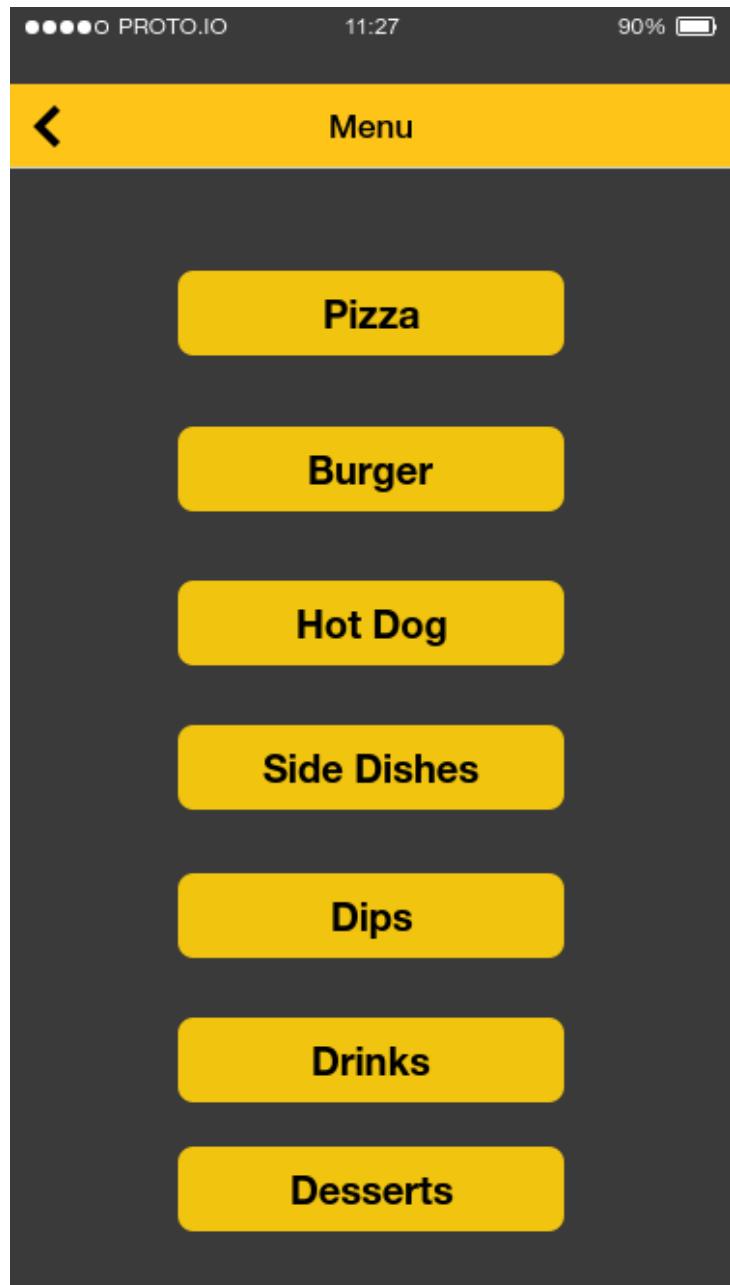


Figure C4: Menu Screen

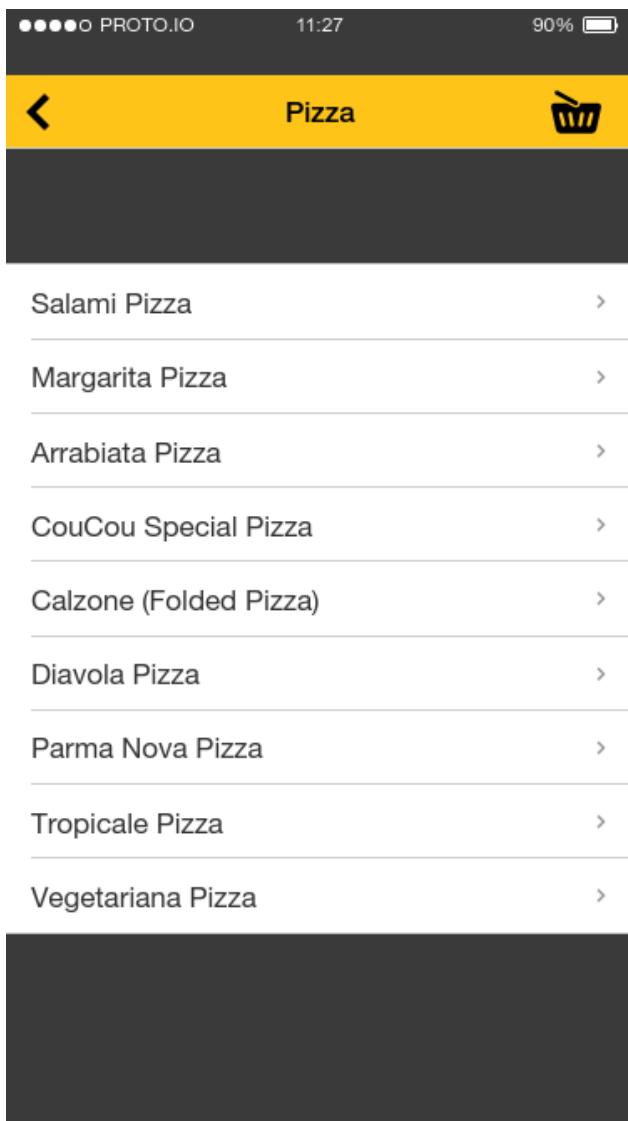


Figure C5: Pizza Category Menu Screen

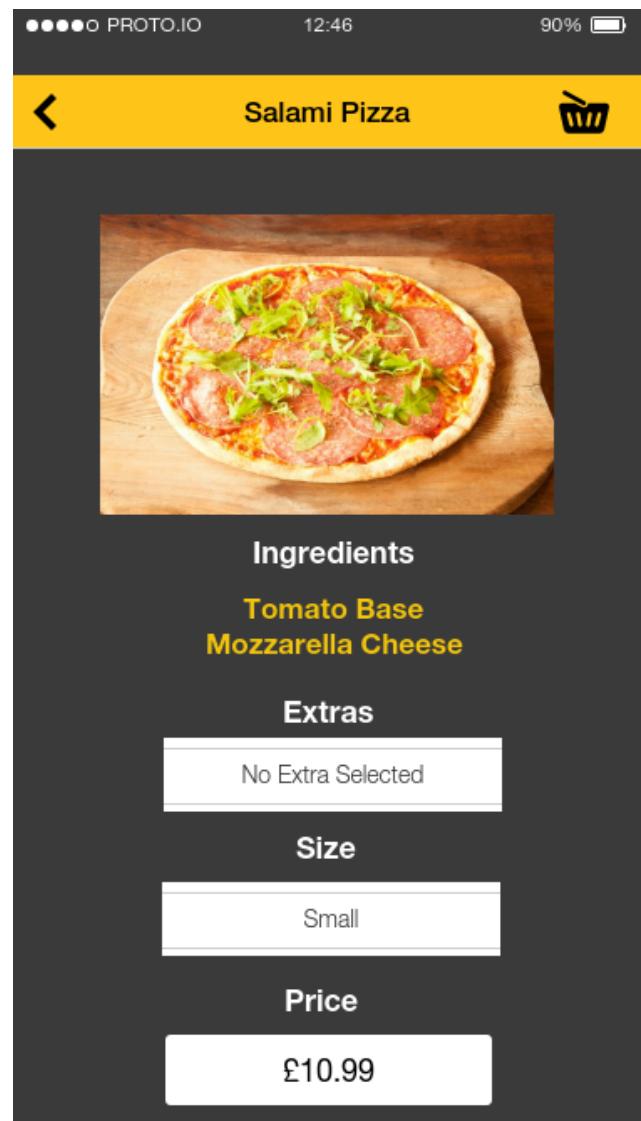


Figure C6: Pizza Menu Item Screen

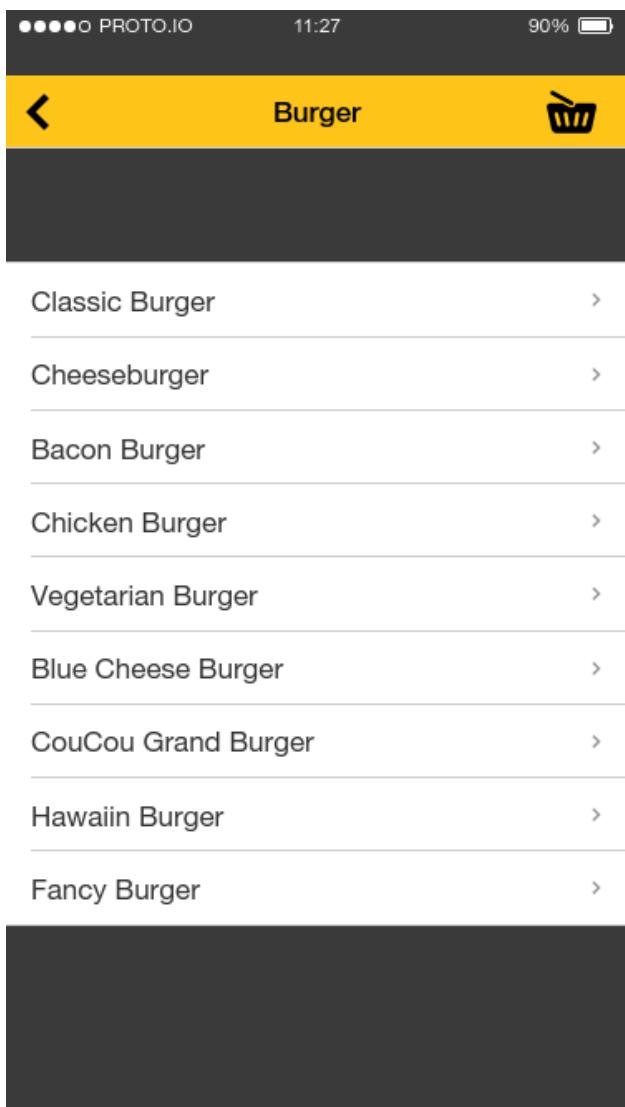


Figure C7: Burger Category Menu Screen

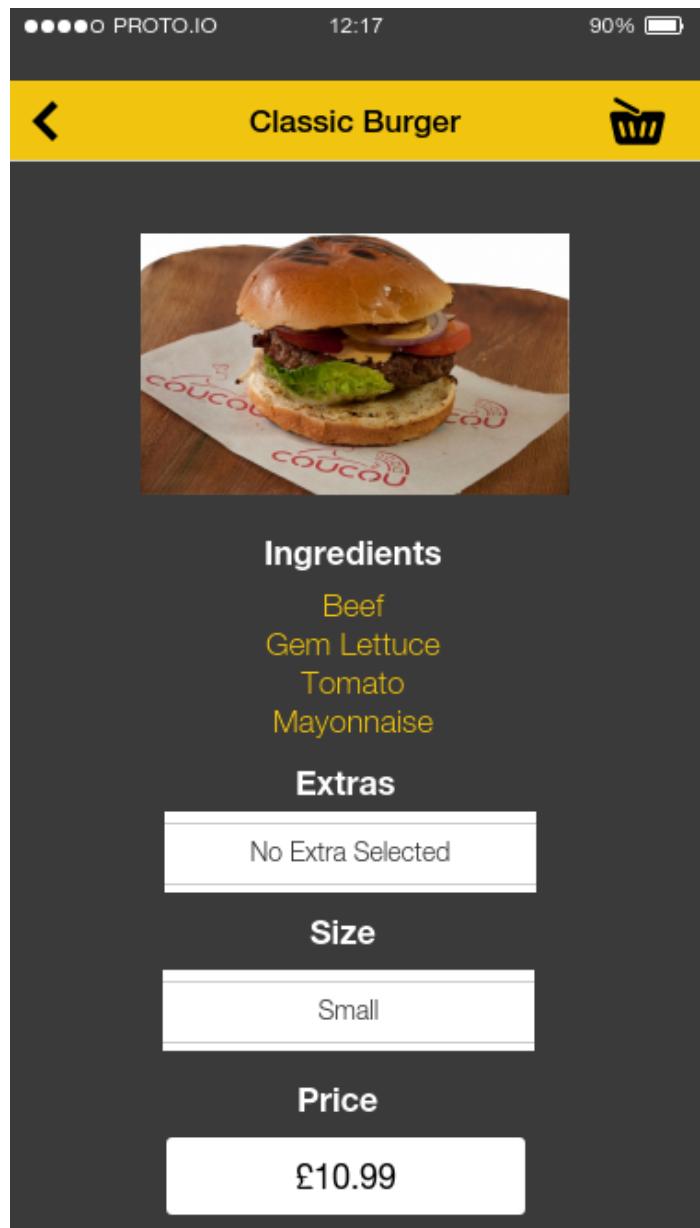


Figure C8: Burger Menu Item Screen

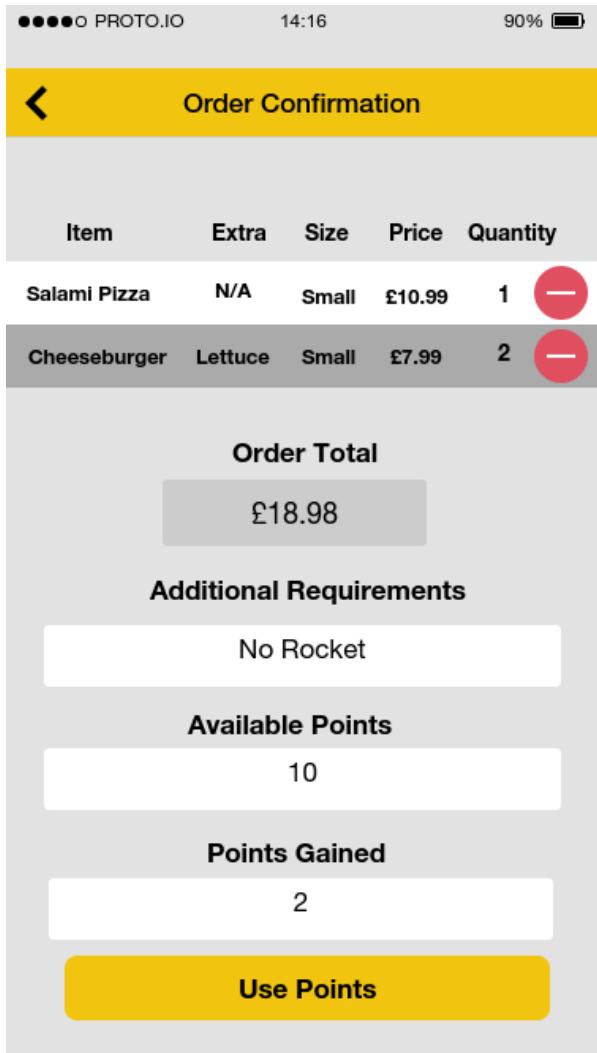


Figure C9: Order Confirmation Screen

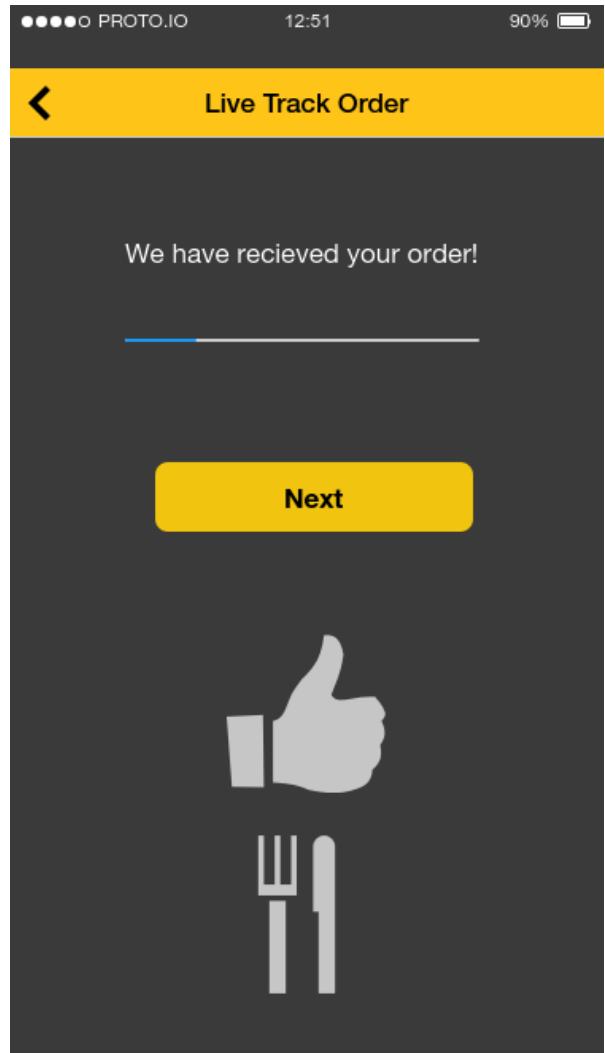


Figure C10: Live Track Order Screen

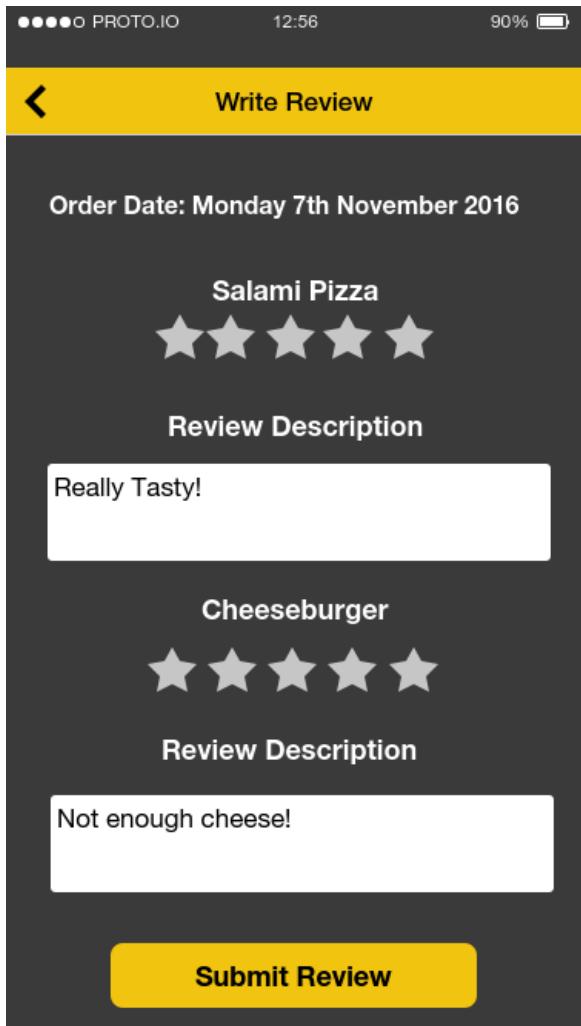


Figure C11: Write Review Screen

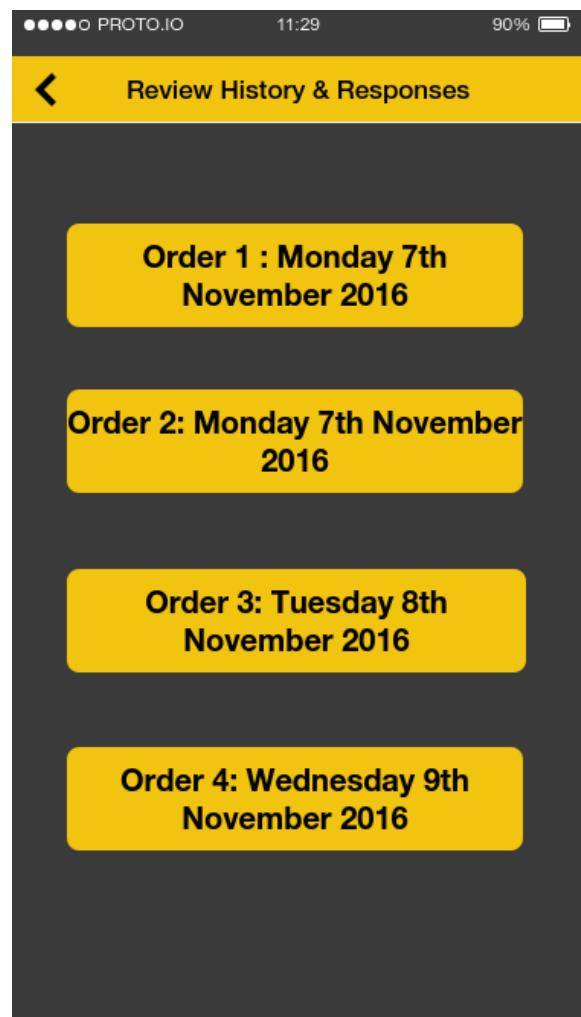


Figure C12: Write Review Screen

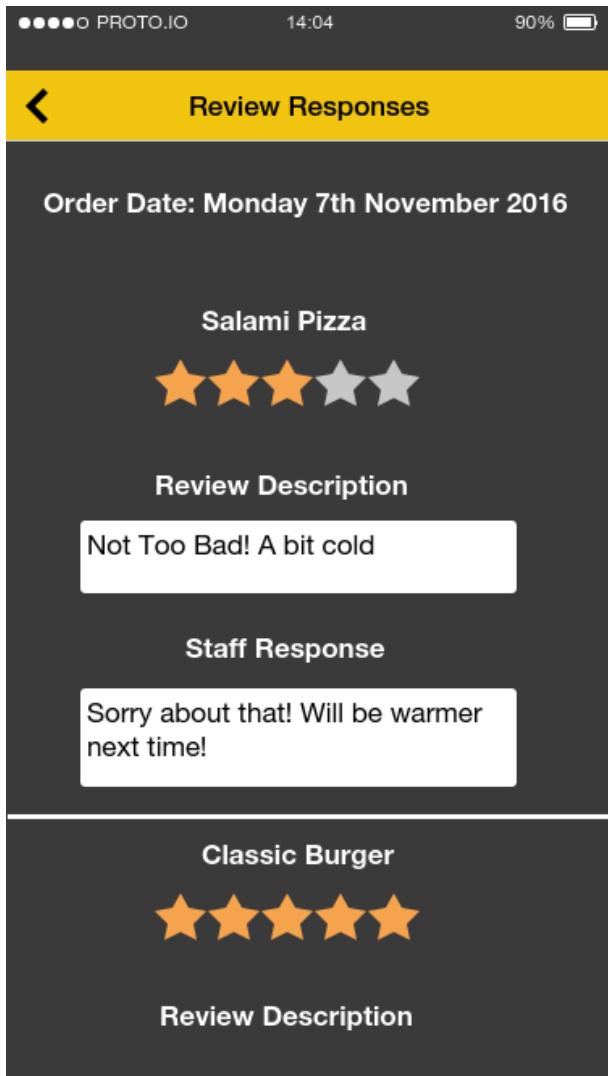


Figure C13: Review Responses Screen

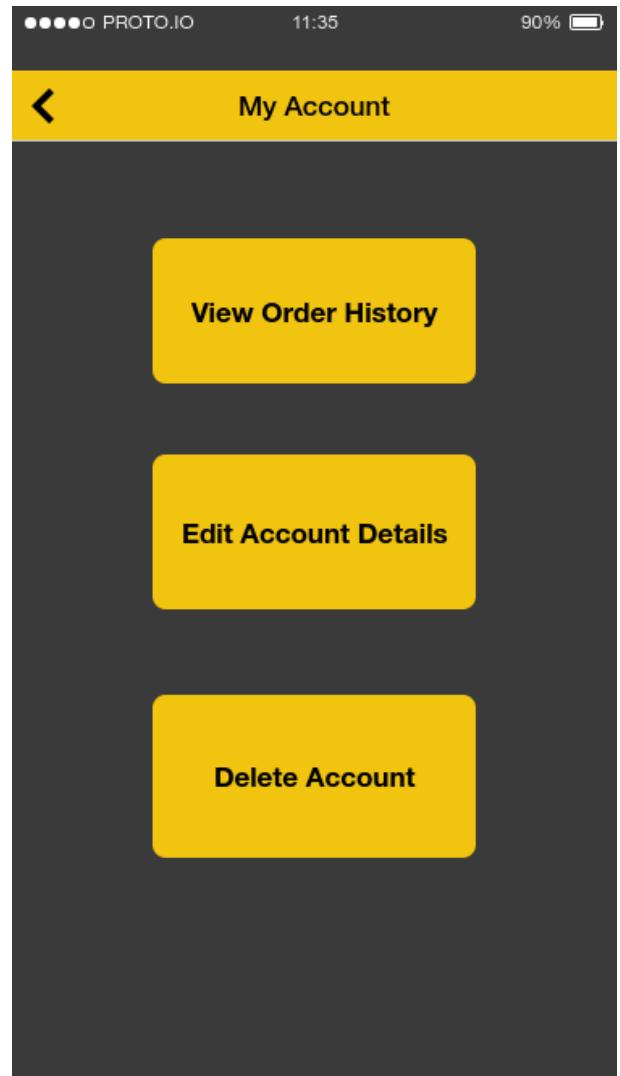


Figure C14: My Account Screen

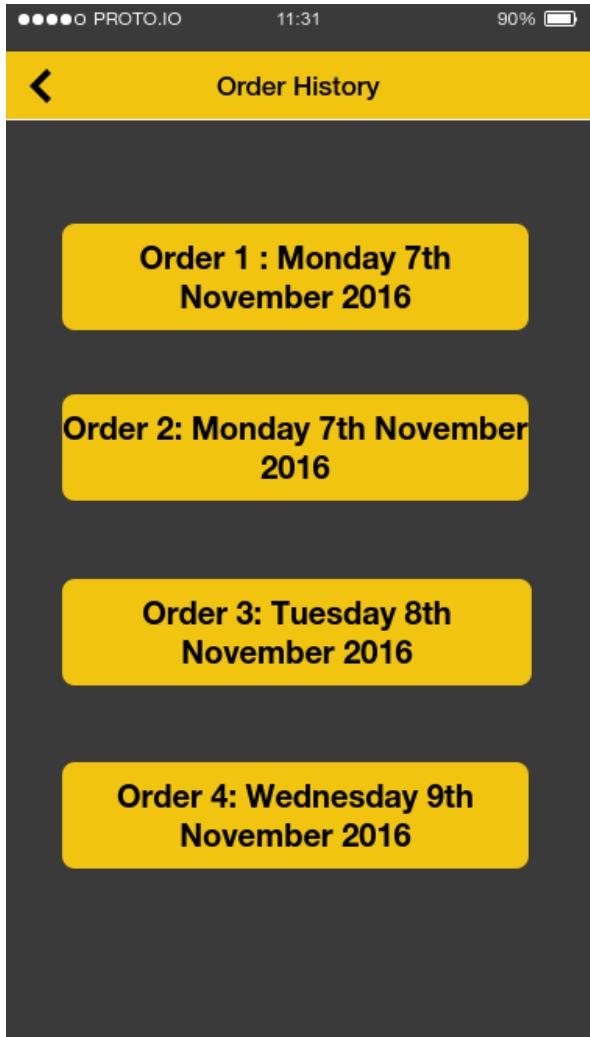


Figure C15: Order History Screen

Item	Extra	Size	Price	Quantity
Salami Pizza		Small	£10.99	1
Cheeseburger	Lettuce	Small	£7.99	2
Order Total				
£18.98				
Additional Requirements				
Extra Salami				

Figure C16: Past Order Screen

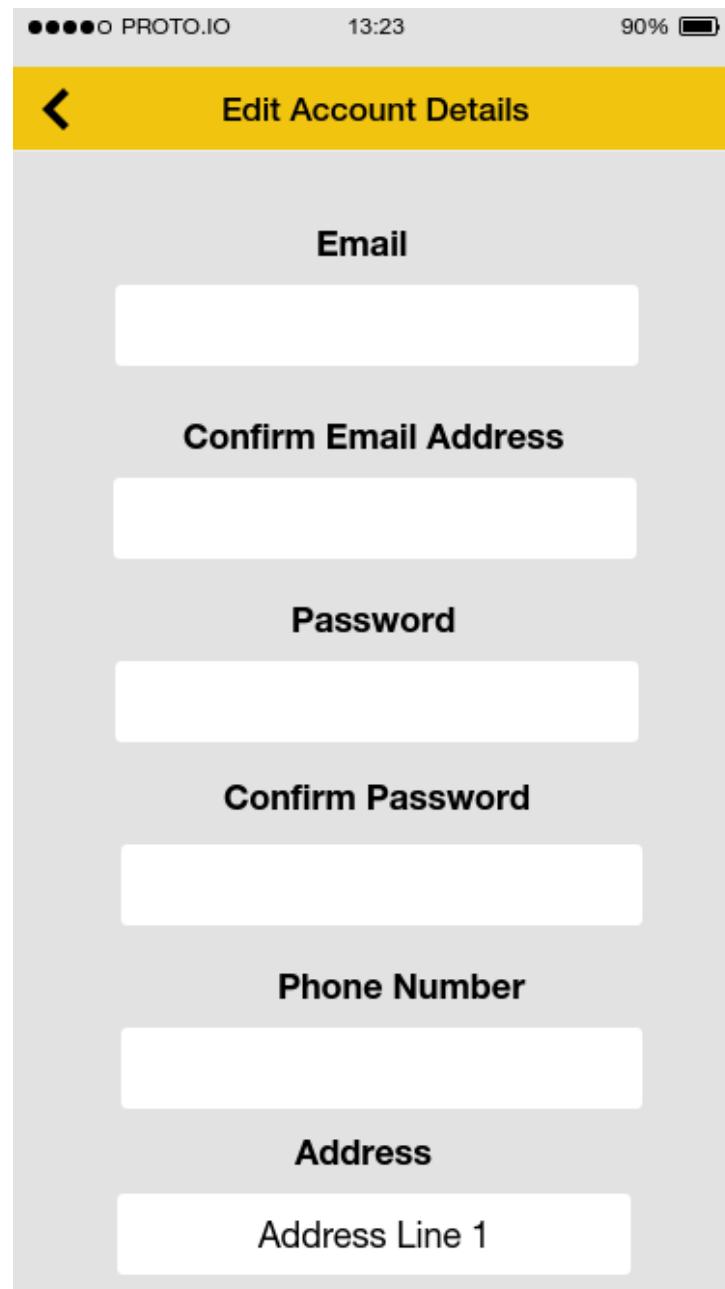


Figure C17: Edit Account Details Screen

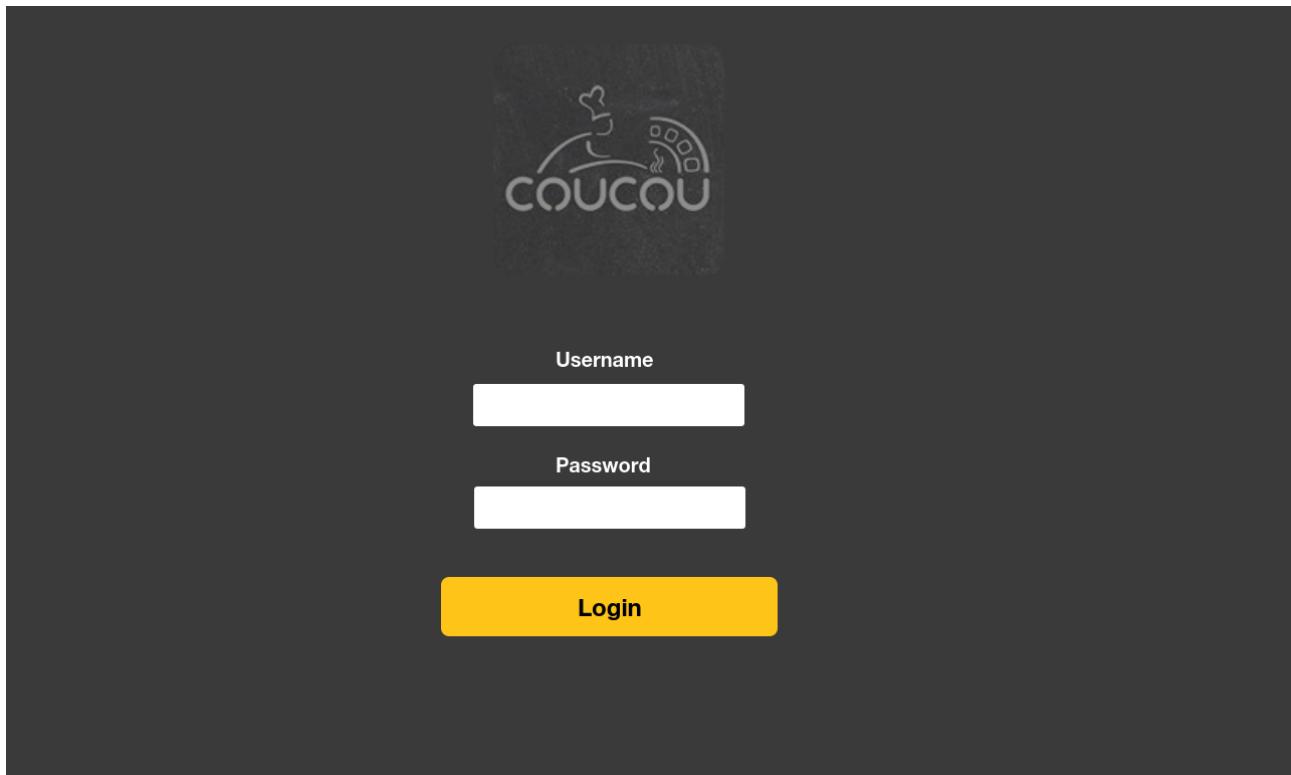
Business Interface

Figure C18: Business Interface Login Screen

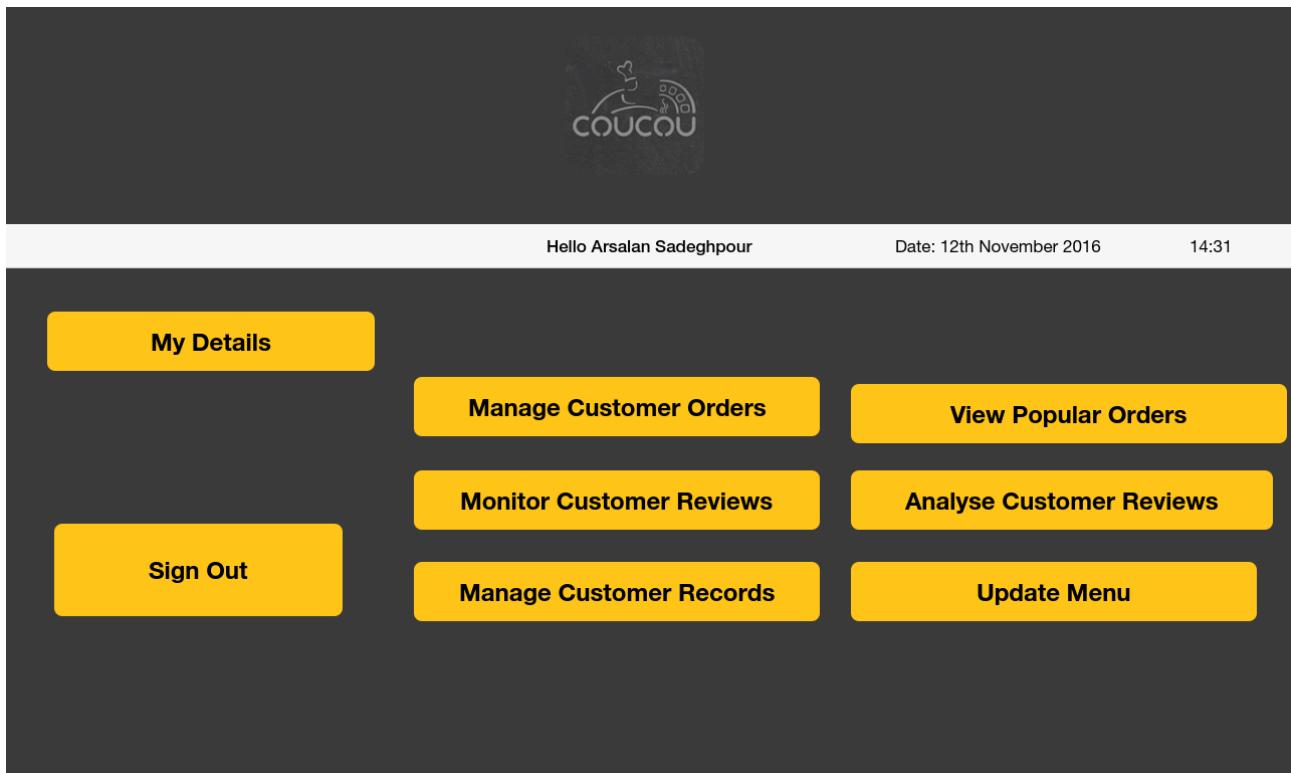


Figure C19: Business Interface Welcome Screen



Logged In: Arsalan Sadeghpour My Details Sat 12 Nov 2016 14:31:10

First Name	Username	Phone Number
First Name	Username	Phone Number
Surname	Password	Address
Surname	Password	Address Line 1
Email Address	Confirm Password	Address Line 2
Email Address	Password	Postcode

Save

Figure C20: Business Interface Staff Details Screen



Logged In: Arsalan Sadeghpour Delete Staff Sat 12 Nov 2016 14:31:10

First Name	Surname	Mobile Number	Date of Birth	Staff Type	Phone Number	Action
Arsalan	Sadeghpour	07719292876	08/03/1994	Manager	07789086318	Delete
Natalia	Beloff	07729292876	10/02/1955	Worker	07798600054	Delete

Home

Figure C21: Business Interface Delete Staff Screen



Back Logged In: Arsalan Sadeghpour Staff Members Sat 12 Nov 2016 14:31:10

Staff Name: Arsalan Sadeghpour

First Name	Surname	Staff Type	
Arsalan	Sadeghpour	Manager	View Details
Natalia	Beloff	Worker	View Details

Address Line 1	Address Line 2	City	Postcode
148 Downsway	Southwick	Brighton	BN42 4WF

Phone Number	Email
07789086318	arsalan@gmail.com

Home

Figure C22: Business Interface Staff Members Screen



Back Logged In: Arsalan Sadeghpour Manage Customer Records Sat 12 Nov 2016 14:31:10

Customer: Arsalan Sadeghpour

First Name	Surname		
Arsalan	Sadeghpour	View Details	Delete
Natalia	Beloff	View Details	Delete

Address Line 1	Address Line 2	City	Postcode
148 Downsway	Southwick	Brighton	BN42 4WF

Phone Number	Email
07789086318	arsalan@gmail.com

Home **Save**

Figure C23: Business Interface Manage Customer Records Screen

The screenshot shows a mobile application interface for managing staff grants. At the top, there is a header bar with a back arrow, the text "Logged In: Arsalan Sadeghpour", the title "Grant Management Rights", the date "Sat 12 Nov 2016", and the time "14:31:10". Below the header is a search bar with the placeholder "Search". The main content area contains a table with the following data:

First Name	Surname	Mobile Number	Date of Birth	Staff Type	Grant Rights
Arsalan	Sadeghpour	07719292876	08/03/1994	Worker	Grant Rights
Natalia	Beloff	07729292876	10/02/1955	Worker	Grant Rights

At the bottom of the screen is a yellow "Home" button.

Figure C24: Business Interface Grant Management Rights Screen

The screenshot shows a mobile application interface for analyzing customer reviews. At the top, there is a header bar with a back arrow, the text "Logged In: Arsalan Sadeghpour", the title "Analyse Customer Reviews", the date "Date: 12th November 2016", and the time "14:31". Below the header is a search bar with the placeholder "Search By Menu Item" and a yellow "Summarise Star Ratings" button. The main content area features a horizontal bar chart with the following data:

Rating	Count
5 Stars	200
4 Stars	100
3 Stars	50
2 Stars	0
1 Star	0

The chart shows the distribution of ratings for three menu items: Salami Pizza (200), Cheeseburger (100), and Classic Hot Dog (50).

Figure C25: Business Interface Analyse Customer Reviews Screen



Back Logged In: Arsalan Sadeghpour Customer Review Analytics Sat 12 Nov 2016 14:31:10

Search By Menu Item Summarise Star Ratings

Salami Pizza

First Name	Surname	Rating	Review Date	Review Description
Arsalan	Sadeghpour	★★★★★	11/11/16	Very tasty but was not that warm
Natalia	Beloff	★★★	1/12/16	Not enough salami! Not warm

Figure C26: Business Interface Customer Review Analytics Screen



Back Logged In: Arsalan Sadeghpour Popular Orders Sat 12 Nov 2016 14:31:10

Today This Month This Year

Number of Orders

Menu Items	Number of Orders
Salami Pizza	40
Cheeseburger	30
Classic Dog	20

Figure C27: Business Interface Popular Orders Screen



Back Logged In: Arsalan Sadeghpour Monitor Customer Reviews Sat 12 Nov 2016 14:31:10

Review Number	First Name	Surname	Rating	Menu Item	Review Description	Select Review	Delete Review
1	Arsalan	Sadeghpour	★★★★★	Bacon Burger	Very tasty but was not that warm	<button>Select Review</button>	<button>Delete Review</button>
2	Natalia	Beloff	★★★★	Salami Pizza	Not enough salami! Not warm	<button>Select Review</button>	<button>Delete Review</button>

Review 1: Response

No problem Arsalan! We will make sure it's warm next time!

Save Response

Figure C28: Business Interface Monitor Customer Reviews Screen



Back Logged In: Arsalan Sadeghpour Update Menu Date: 12th November 2016 14:31

Add Menu Item	Edit Menu Item	Delete Menu Item
Menu Item Name: <input type="text" value="Item Name"/> Price: <input type="text" value="Price"/> Ingredients: <input checked="" type="checkbox" value="Cheese"/> Cheese, <input checked="" type="checkbox" value="Tomato"/> Tomato, <input type="checkbox" value="Mushrooms"/> Mushrooms	Item Category: <input type="text" value="Pizza"/> <input type="text" value="Burger"/>	Quantity: <input type="text" value="Quantity"/> Add Menu Item

Figure C29: Business Interface Update Menu Screen

APPENDIX D - QUESTIONNAIRE RESULTS

Mobile App Questionnaire Results Analysis

Is the mobile app well designed?

Answered: 11 Skipped: 0

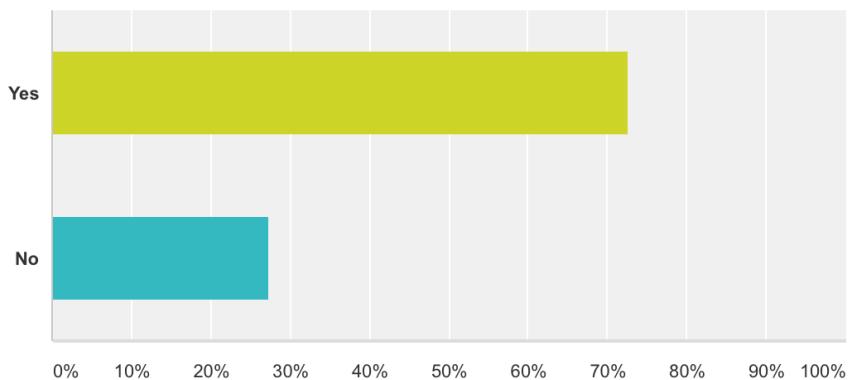


Figure D1: App Design Questionnaire Results

Do you think the app is enjoyable to use?

Answered: 11 Skipped: 0

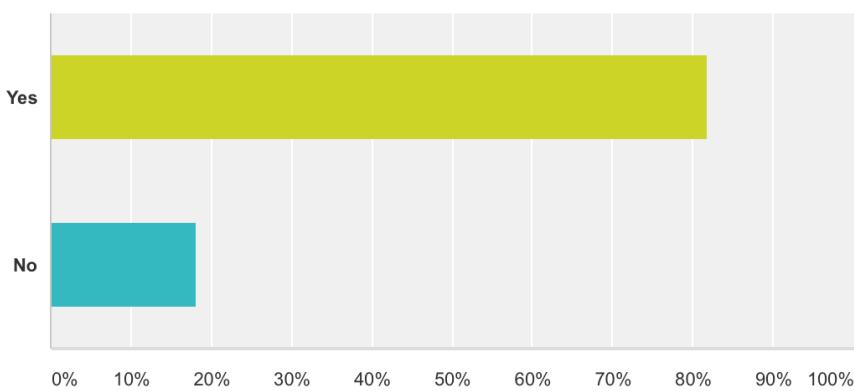


Figure D2: App Enjoyability Questionnaire Results

The results suggest the design of the app could be kept the same given 8 users liked the design and 9 enjoyed using the app adding further comments such as the app is “*simple*”, “*easy to use*” and it “*takes no time to get used to*” along with the style “*appealing to many types of user*” which is who CouCou intend to target. There are 3 users however, who don’t like the design and 2 users who don’t find it enjoyable adding further comments such as the design is “*plain*” and has a “*tedious feel about it*” suggesting the design should be refined to make it more enjoyable, they also indicated they preferred “*icons on the buttons*”, “*more interesting screen transitions*” and “*more appealing images*”, therefore these changes shall be implemented as they are not time consuming.

How would you improve the design of the app?

Answered: 10 Skipped: 1

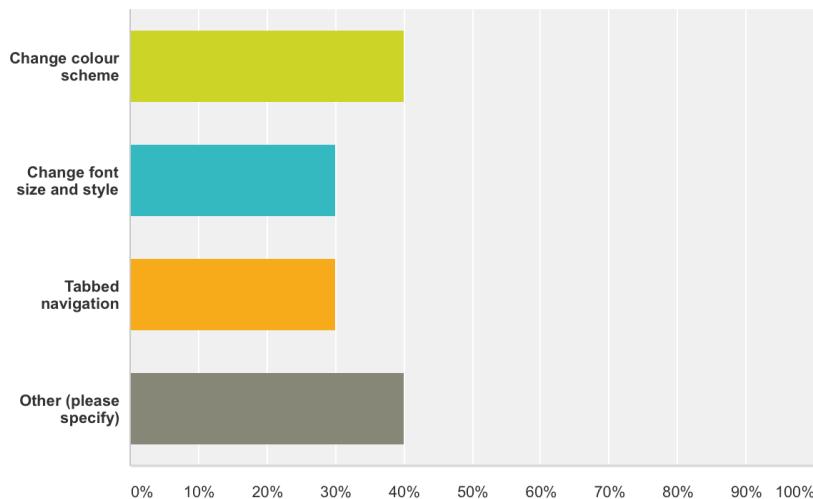


Figure D3: App Design Improvements Questionnaire Results

The results suggested that even the users who previously stated they liked the apps design felt it could be further improved. 4 users wanted the colour scheme to be changed, 3 users preferred a change in the font size and style, 3 users wanted the app's design to incorporate tabs instead of buttons and 4 users stated comments such as “visual feedback when they perform actions” would increase the usability, “videos when clicking on content would make it more enjoyable” followed by improvements to the live tracker such as “Consider the virtual tracker..its always cool to watch!” and the possibility of including a “map image for the tracker showing the food journey for delivery” meaning users would like more interactivity and more visual appeal for the tracker. The timeframe of this particular application doesn't allow me to accommodate important suggestions such as a virtual tracker or GPS tracking for deliveries, however these could be implemented in future versions based on feedback from a wider variety of customers.

Would you use the app again?

Answered: 11 Skipped: 0

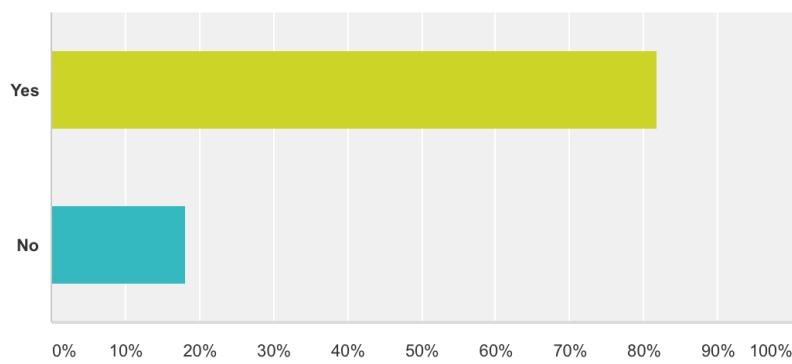


Figure D4: App Repeat Usage Questionnaire Results

Overall given the design improvements indicated, 9 users felt they would use the app again if it were in the market right now with only 2 not using it, such as the same person that felt the app was “plain” and “tedious”. The timeframe of this particular application will not allow me to respond to the important suggestion of adapting the design even further to make it more enjoyable, however the design could be altered in future versions based on feedback from a wider variety of customers .

Which of these possible features are the most helpful and important to you?

Answered: 11 Skipped: 0

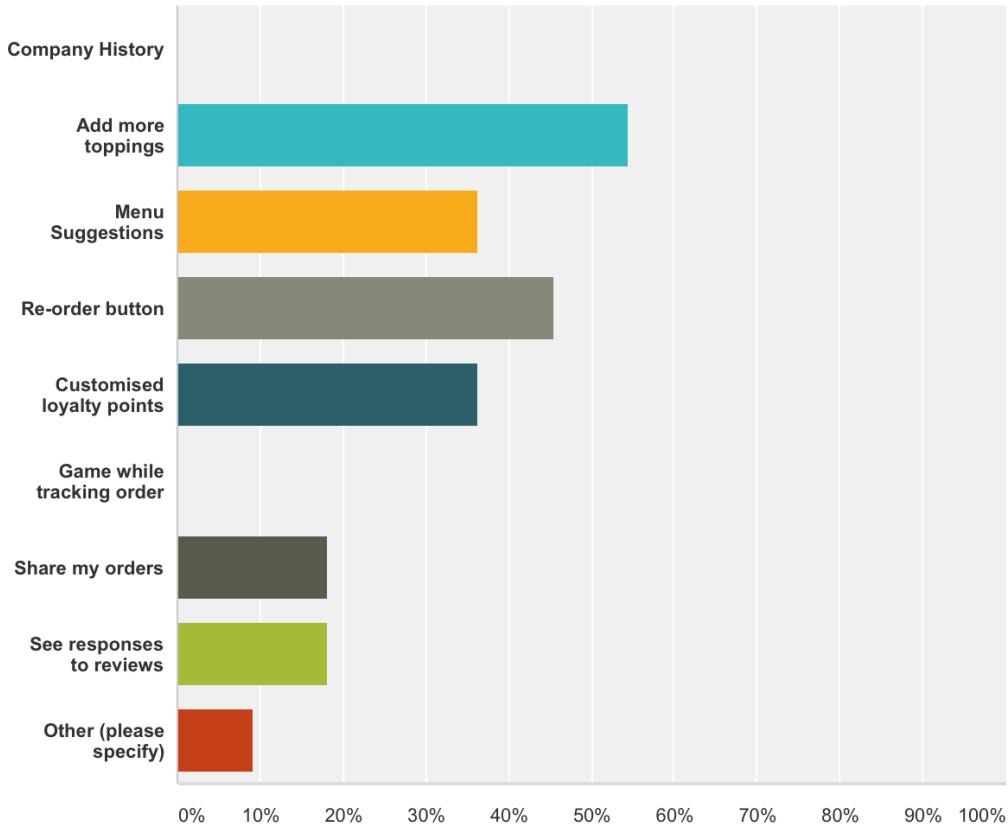


Figure D5: App Feature Importance Questionnaire Results

The results suggest users preferred to have the ability to add more toppings to menu items rather than an overview of the brand. 6 users wanted the ability to add more toppings, 5 users also wanted a quick re-order button for their last order and 4 users wanted menu item suggestions tailored to their order. 4 users also wanted the ability to customise loyalty points according to a menu item category e.g. pizza points, burger points and hot dog points. 2 users wanted to share their orders to social media platforms and 2 users wanted to see the business respond to reviews. 1 user stated “*the customisation of loyalty points stands out as it makes the customer feel special and individualised*” along with “*suggesting what items I may like is useful as I can quickly order it*”.

In summary users would like a more personalised, convenient and customisable experience along with the ability to share their orders and this may also create further brand exposure for marketing purposes along with creating a reactive and responsive brand that values their customers opinions.

Due to the timeframe of the application, suggestions such as the re-order button, customised loyalty points, social media sharing, menu suggestions for customers and responding to reviews are important, however will be considered in future versions of the app as they are quite complex. The addition of more toppings is not a complex change, however will be implemented in a future version due to the client's business requirements.

What impression does the app's design give you about the brand?

Answered: 11 Skipped: 1

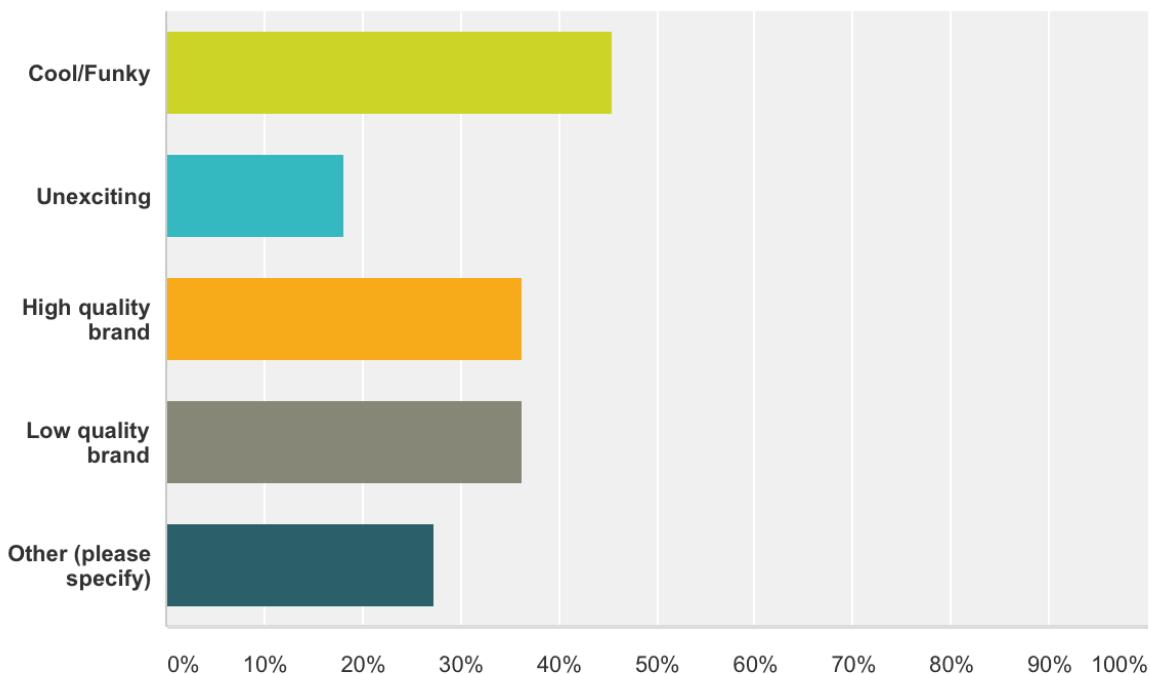


Figure D6: Brand Image Questionnaire Results

Overall CouCou wish their brand image to be perceived as providing authentic Italian pizza using the freshest ingredients and a home cooked feel towards their products.

The results suggest mixed views about the brand image with 5 users feeling it is cool and funky, 2 who felt it doesn't represent an exciting brand. 4 users who feel that it may be higher quality than competitors such as Dominoes and Papa Johns and 4 users who felt it may be of lower quality than those competitors.

3 users stated further comments such as "*there is no singular appeal*", and positive comments praising the inclusion of images "*I also like the fact you include images of the food. As this is something some cheaper fast food companies lack and it really bugs me when trying to make a decision!*", however the images should be utilised in a different way in that "*images of individual ingredients thrown together to make menu items*" would make the brand stand out as being of high quality as there is "*emphasis on ingredients*". There were also encouraging comments about the appearance of the food "*the food looks a lot more wholesome than traditional chains*".

Overall given the timeframe of the application, suggestions such as adding images of individual ingredients, tailoring the design to appeal to a specific audience and improving the brands perception in terms of quality are important suggestions which will be implemented in future versions of the app upon receiving feedback from a wider variety of customers.

Business Interface Questionnaire Results Analysis

If you could improve the design, which would you improve?

Answered: 5 Skipped: 0

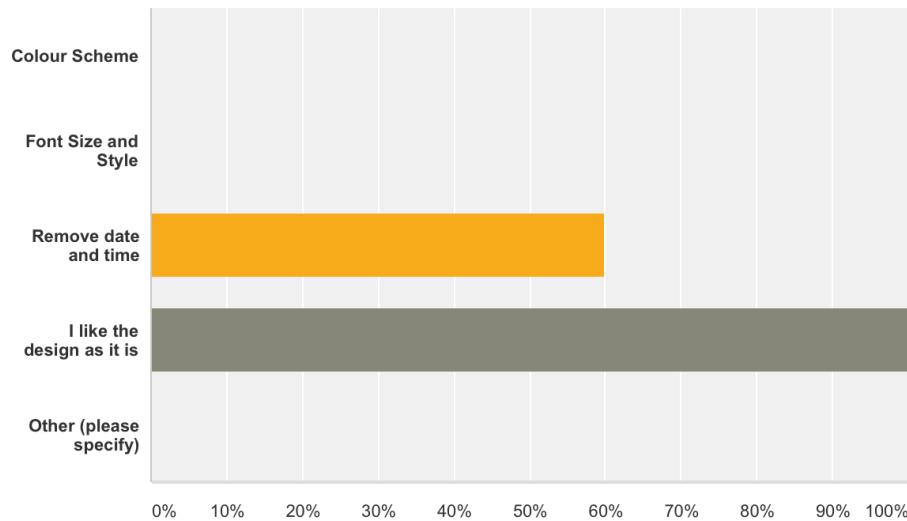


Figure D7: Business Interface Design Improvements Questionnaire Results

The results suggest staff like the theme of the user interface as all 5 members of staff like the design at is, however 3 members of staff stated they would prefer if the date and time be removed from the system. Upon consulting with those staff they stated “*it does not add any value to the system*” and that there is “*no use for it*”, however all those staff liked the design as it is. Upon consulting with those 2 members of staff that did not want the date and time removed they stated they “*don’t mind if it is removed or not*”.

Which way would you prefer to update the status of customer orders?

Answered: 5 Skipped: 0

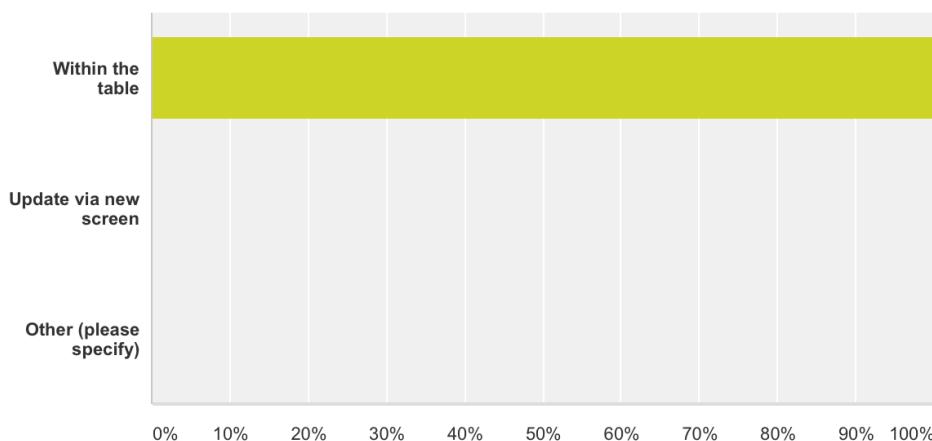


Figure D8: Status Update Preference Questionnaire Results

The results suggest all members of staff prefer to update the status of customer orders within the order management table itself, therefore no changes to the design shall be made.

**Is there data in the customer order table
that is not useful for you?**

Answered: 5 Skipped: 0

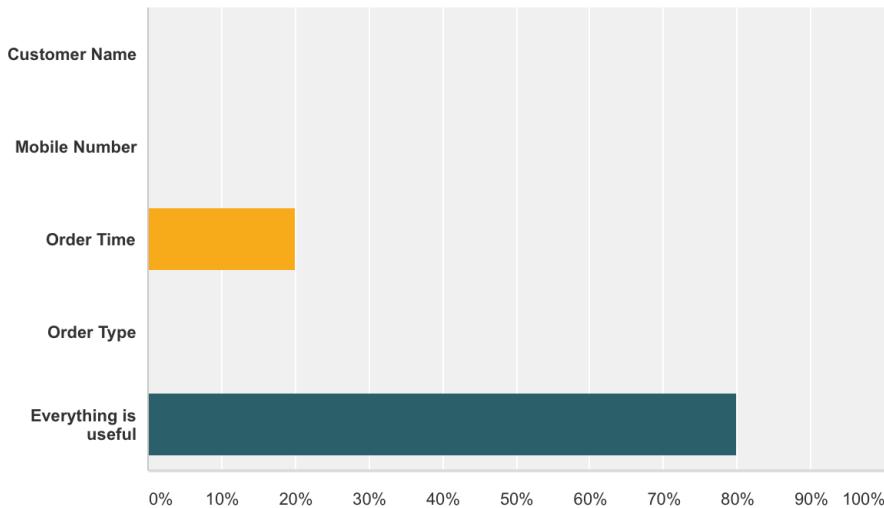


Figure D9: Usefulness Of Customer Order Data Questionnaire Results

The results suggest that all the data in the customer table is helpful with the exception of 1 member of staff who stated the order time was not useful. In this case given the member of staff's suggestion is equally important the system could perhaps have the ability to hide/show certain fields in the table to tailor it to different staff's viewing preferences, however due to the time frame of the system, this enhancement may have to be implemented as an upgrade.

**Is the button that triggers viewing order
details in a separate box for customer
orders useful?**

Answered: 5 Skipped: 0

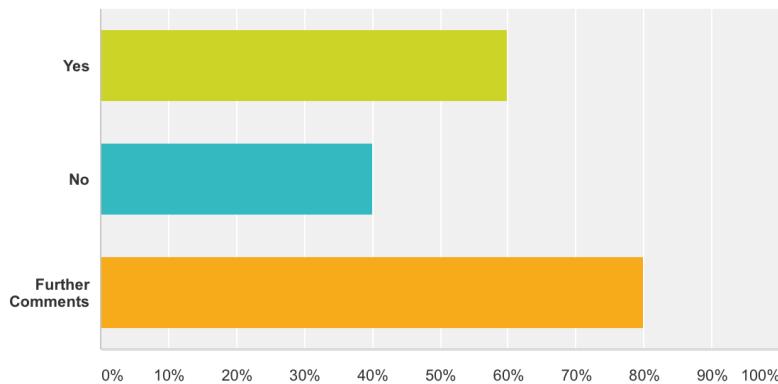


Figure D10: Order Details User Interface Questionnaire Results

Overall 3 members of staff found that viewing order details in a separate area was useful , however 2 members of staff do not find it useful. 2 members of staff stated further comments such as "*it would be better if the order details were shown in a popup window*" and the 3 members of staff who did find it useful stated "*It is not clear which order, the order details section is referring to, so It would be better if you made it clearer*". Overall enhancements shall be made to the clarity of viewing order details to make it clearer which order it refers to, for example by using an indicator referring to the selected order. Due to the timeframe of the application, the important suggestion of restructuring the design to include a popup window instead of a separate area for viewing details may not be completed due to the time constraints of this project, however it shall be implemented in a later version of the system.

When the status of an order is 'complete', would you prefer the next order below to be moved up?

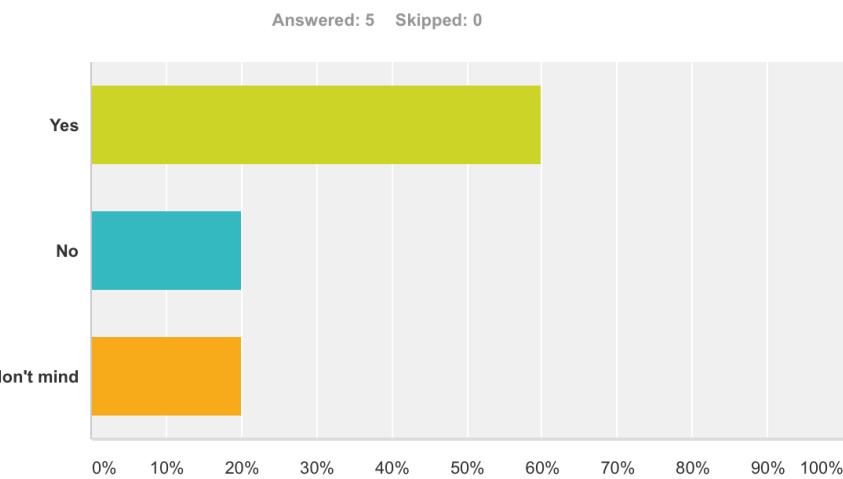


Figure D11: Order Status Movement Preference Questionnaire Results

The results suggest the majority of members of staff would like the next incoming order to be dynamically moved to the top of the order management table with 3 members of staff stating yes, however 1 member of staff stated they would not want this functionality and 1 member of staff stated they did not mind. Upon consulting with that member of staff, they preferred to have the ability to work on “any order I want” because “some orders are faster to complete than others” which is an important concern for efficiency given it is one of the project’s key objectives. Overall the design shall be adjusted to allow members of staff to work on any order they like irrespective of its place in the table.

How many orders on average would you be preparing at once?

Answered: 5 Skipped: 0

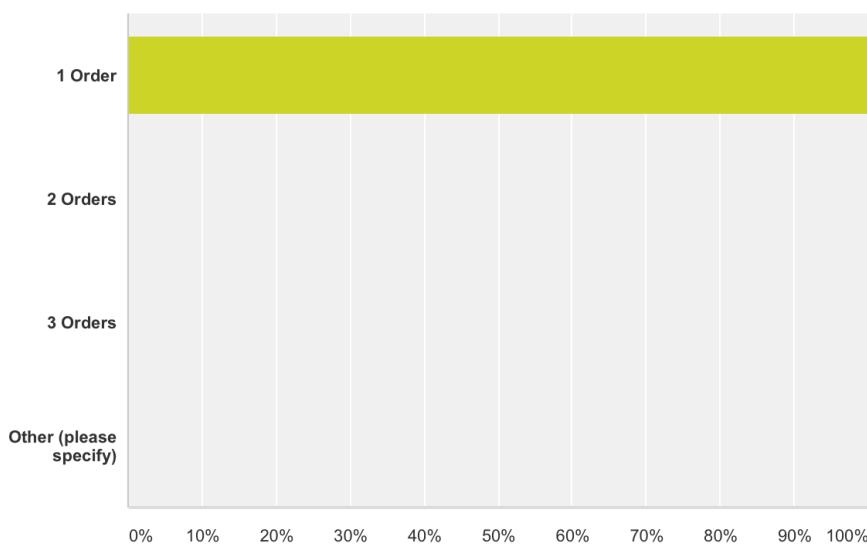


Figure D12: Number of Staff Orders at Once Questionnaire Results

The results suggest that all members of staff will only be working on 1 order at a time, therefore the system shall keep the functionality of only allowing the ability to view the order details of 1 order at a time.

When viewing popular orders which historic data would be more useful?

Answered: 5 Skipped: 0

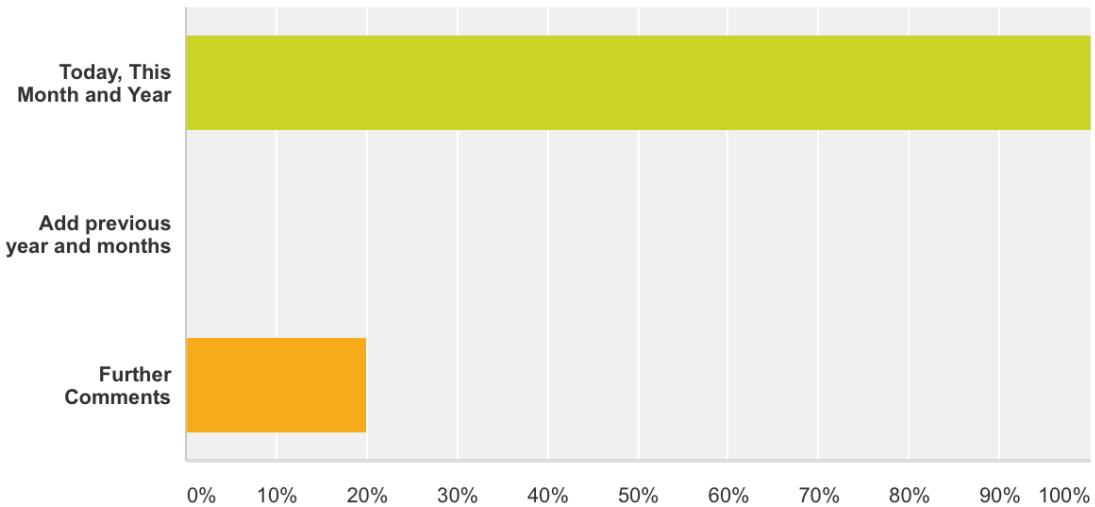


Figure D13: Viewing Historic Data in Popular Orders Questionnaire Results

The results suggest that staff prefer to view the popular orders of the current day, month and year with 1 member of staff adding a further comment justifying themselves by stating "*the previous years are not helpful because we cannot react to this information*". Overall the design of the system shall be kept the same which is to only state the current day, month and year of order analytics.

APPENDIX E - TEST PLAN RESULTS

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
1.1	Functional	Valid signup data without special characters	Successful signup	Tests whether it is possible to signup using a valid email address and other valid data such as names without special characters	Success
1.2	Functional	Invalid signup data including special characters and existing email addresses	Unsuccessful signup	Tests whether signup fails when invalid data entered and whether an error is generated when using an existing email address	Success
2.1	Functional	Login with existing username and password	Successful login	Tests whether email and password combination exists in the database and customer is redirected correctly	Success
2.2	Functional	Login with an invalid email address/ password	Unsuccessful login	Error message is displayed for either email doesn't exist or incorrect password entered	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
3.1	Functional	Welcome page Accessed	Successfully displays customers points	Tests whether the customers points are accessed from the database and displayed	Success
3.2	Functional	Generate an order and return to the Welcome page	Successfully updates the display of customers points	Tests whether points are increased after a qualifying order	Success
3.3	Functional	Set customer's points to 10, add items to order and go to confirm order page	Use Points button is displayed and updates the customers available points	Tests whether the customers points are redeemed	Success
3.4	Functional	1) Generate an order and return to the Welcome page. 2) Login with a customer without an order	Live track button should be available to select for case 1 and disabled for case 2	A test order is created for the customer to test case 1, then a new customer is created without an order to test case 2	Success
3.5	Functional	1) Generate an order and return to the Welcome page. 2) Login with a customer without an order	Write Review button should be available to select for case 1 and disabled for case 2	A test order is created for the customer to test case 1, then a new customer is created without an order to test case 2	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
3.6	Functional	1) After generating an order submit a review and return to the Welcome page. 2) Login with a customer with an order who has not submitted a review	View review history and responses button should be available to select for case 1 and disabled for case 2.	A test order is created for the customer to test case 1, then an order is generated for another customer to test case 2	Success
4.1	Functional	Menu Page, Menu page for an item category	Menu items are dynamically displayed from the database	Menu items should reflect items in the database	Success
4.2	Functional	Menu Item Page	Price dynamically changes based on an increase in size options	Document Object Model correctly updates the price field	Success
4.3	Functional	Menu Item Page	Price dynamically changes based on extras added	Document Object Model correctly updates the price field	Success
4.4	Functional	Menu Item Page, Order Confirmation Page	After adding item to cart, the item details are displayed on the Order confirmation page	Session variable used to hold cart items functions correctly	Success
4.5	Functional	Order Confirmation Page	After removing an item from the cart, the item details are removed from the page	Item details should also be removed from the session variable used to hold cart items	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
4.6	Functional	Order Confirmation Page	The customer's additional requirements should display in the database	Additional requirements should display for the selected order in the database after an order is generated	Success
4.7	Functional	Order Confirmed Page	The customer's order should be generated in the database and a confirmation message should display	Check database to determine if order was successfully generated	Success
5.1	Functional	Order Confirmation Page, Customer with 10 loyalty points	The Use Points button should display on the page	Database should be contacted and the Document Object Model used to update the page	Success
5.2	Functional	Order Confirmation Page, Customer with 10 loyalty points	Use Points button should update the customer's available loyalty points	Database should be contacted to update the logged in customer's loyalty points	Success
5.3	Functional	Customer without 10 loyalty points	Use Points button should not display	Database should be contacted to determine if Use Points button should be displayed or not	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
6.1	Functional	Order Confirmed Page	Customer redirected to Live Track Order page after confirming order		Success
6.2	Functional	Live Track Order Page	Customer's order status should display and update the status automatically	AJAX call to PHP script should check the latest order generated by the customer and check the order status constantly. Status is updated manually using the Database for testing purposes for now.	Success
6.3	Functional	Live Track Order Page	Write Review button should display when Order Status is Delivered	Update the order status to delivered using the Database to check if the button is displayed	Success
7.1	Functional	Write Review Page	Displays menu items from the latest order which can be reviewed	An order for the customer must have been generated previously	Success
7.2	Functional	Write Review Page	Page should display interactive rating selectors		Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
7.3	Functional	Write Review Page	Text area for writing reviews is accessible by the customer		Success
7.4	Functional	Review Confirmed Page	Review details are successfully updated for that Order in the database		Success
8.1	Functional	View Review History and Responses Page	Page should not display any review history buttons	Requires logged in customer who has ordered but not submitted a review yet	Success
8.2	Functional	View Review History and Responses Page	Page should display correct number of buttons reflecting number of reviews made by the customer in the database	Requires logged in customer who has ordered and made at least 1 review	Success
8.3	Functional	Selected Review History and Responses Page	Review details for the selected review should display	Requires customer who has ordered and made at least 1 review	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
8.4	Functional	Selected Review History and Responses Page	Staff responses should display for selected review	Requires customer who has ordered and made at least 1 review and received at least 1 review response from a staff member	Success
8.5	Functional	Selected Review History and Responses Page	Blank text area for staff response should display for selected review with no staff responses just yet	Requires customer who has ordered, made at least 1 review and has received no staff responses just yet	Success
9.1	Functional	View Order History Page	No order history buttons should display	Requires customer who has no orders	Success
9.2	Functional	View Order History Page	Correct number of order history buttons should display for logged in customer	Requires customer with at least 1 order	Success
9.3	Functional	Selected Past Order Page	Order details of selected past order should display		Success
10.1	Functional	Edit Account Details Page	The customer's current details should display		Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
10.2	Functional	Edit Account Details Page	The customers details should remain unchanged if customer saves details without changing them	Update database query should not be sent to increase efficiency	Success
10.3	Functional	Edit Account Details Page	Prevent updating customer details with invalid data		Success
10.4	Functional	Edit Account Details Page	Prevent updating customer email address to an email address which is already existent for another customer		Success
10.5	Functional	Account Details Saved	Database correctly updates the customer's account details		Success
11.1	Functional	Delete Account Button	Database correctly deletes customer account	Attempt login to the account afterwards to determine if account was deleted	Success
11.2	Functional	Delete Account Button	Customer redirected to Index page after deleting their account	Index page allows customers to signup to a new account or login to an existing account	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
12.1	Functional	Manage Customer Orders Screen	All customer orders should display correctly		Success
12.2	Functional	View Order Details Button	Selected customer order's details should display correctly		Success
12.3	Functional	Generate a new order for a customer	Newly generated order should display automatically without the page refreshing	Tests whether AJAX script checking customer orders in database functions correctly	Success
12.4	Functional	A customer order to update the status	The order status should be updated and displayed for that order. The order status should also be updated in the database along with the logged in Staff member's ID assigned to that order	Tests whether PHP script used to generate SQL Update Query functions correctly	Success
13.1	Functional	Monitor Customer Reviews Screen	All Customer reviews should display correctly		Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
13.2	Functional	Generated a new review for a customer	Newly generated reviews should display automatically without the page refreshing	Tests whether AJAX script checking customer reviews in database functions correctly	Success
13.3	Functional	Selected Review to delete	Review is deleted from the web page table		Success
13.4	Functional	Selected Review to delete	Review is deleted from the database back end		Success
13.5	Functional	Select Review Button	Text area with a heading for the selected review is displayed along with a text box to allow staff members to submit review responses		Success
13.6	Functional	Save Response Button	The staff response for the selected review is saved into the database	Check that staff response is correctly assigned to the selected review in the database	Success
1	Non Functional	Test User Interface	Customers should find the User Interface aesthetically pleasing	Customer feels user interface is modern and looks fun	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
2	Non Functional	Test App Simplicity and Enjoyability	Customers should gain enjoyment from using the application and should find it simple to use	Customer feels app is self explanatory and finds the tracker enjoyable to watch	Success
3	Non Functional	Test Web Interface learning curve	Staff should be knowledgeable enough to fully interact with interface within 1 day	Staff members can fully use the web interface without any supervision	Success
4	Non Functional	Crash the mobile app and web interface to see if order and review data remains intact and state can be resumed from before the crash occurred	Mobile App should keep customer logged in with all review and order data intact. Web interface should keep staff member logged in with all order data and review data intact.	State is resumed for both the mobile app and web interface from before the crash occurred with no data loss	Success
5	Non Functional	Large amount of customer and order data is inputted into the database (approx 100) to ensure app is maintainable.	Mobile App, Business Interface and Database should still run efficiently	Mobile App, Business Interface and Database continue to run efficiently as normal	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
6	Non Functional	Large amount of staff members added (approx 100)	Mobile App, Business Interface and Database should still run efficiently	Mobile App, Business Interface and Database continue to run efficiently as normal. Not possible to test additional branches. Consider having separate staff databases for each branch in the future	Fail
7	Non Functional	Customer navigates all Mobile App screens	Mobile App should remain responsive and should not become slow	Mobile App remains efficient and transitions between screens remain quick and smooth	Success
8	Non Functional	Updating a customer's order status on the business interface	The order status should instantly update on the mobile app. The maximum it should take is 1 second to update	Updating the order status via the web interface instantly updates the mobile app's live track order page	Success
9	Non Functional	Submitting a review response on the business interface	The review response should instantly display on the mobile app. The maximum it should take is 1 second to update.	Submitting a review response via the business interface instantly updates the mobile app's view review history and responses screen	Success

Requirement	Requirement Type	Test Input	Expected Result	Comment	Success/Fail
10	Non Functional	Viewing the mobile app on Chrome and Safari mobile browsers	The mobile app should correctly render on both the Chrome and Safari mobile browsers	The mobile app correctly renders on both the Chrome and Safari and browsers The mobile app runs on both Android and iOS	Success
11	Non Functional	No Analytics test data	Analytics data should maintain data integrity	No test results available	Fail
12	Non Functional	Password data should be encrypted for both staff and customers	Customer and Staff passwords should be encrypted	Only customer passwords are encrypted. Staff passwords are stored as plain text	Fail

APPENDIX F - ETHICAL COMPLIANCE FORM

Ethical Compliance Form for UG and PGT Projects* School of Engineering and Informatics University of Sussex

This form should be used in conjunction with the document entitled "Research Ethics Guidance for UG and PGT Projects".

Prior to conducting your project, you and your supervisor will have discussed the ethical implications of your research. If it was determined that your proposed project would comply with **all** of the points in this form, then both you and your supervisor should complete and sign the form on page 3, and submit the signed copy with your final project report/dissertation.

If this is not the case, you should refer back to the "Research Ethics Guidance for UG and PGT Projects" document for further guidance.

-
1. Participants were not exposed to any risks greater than those encountered in their normal working life.

Investigators have a responsibility to protect participants from physical, mental and emotional harm during the investigation. The risk of harm must be no greater than in ordinary life. Areas of potential risk that require ethical approval include, but are not limited to, investigations that require participant mobility (e.g. walking, running, use of public transport), unusual or repetitive activity or movement, physical hazards or discomfort, emotional distress, use of sensory deprivation (e.g. ear plugs or blindfolds), sensitive topics (e.g. sexual activity, drug use, political behaviour, ethnicity) or those which might induce discomfort, stress or anxiety (e.g. violent video games), bright or flashing lights, loud or disorienting noises, smell, taste, vibration, or force feedback.

2. The study materials were paper-based, or comprised software running on standard hardware.

Participants should not be exposed to any risks associated with the use of non-standard equipment: anything other than pen-and-paper, standard PCs, mobile phones, and tablet computers is considered non-standard.

3. All participants explicitly stated that they agreed to take part, and that their data could be used in the project.

Participants cannot take part in the study without their knowledge or consent (i.e. no covert observation). Covert observation, deception or withholding information are deemed to be high risk and require ethical approval through the relevant C-REC.

*This checklist was originally developed by Professor Steven Brewster at the University of Glasgow, and modified by Dr Judith Good for use at the University of Sussex with his permission.

If the results of the evaluation are likely to be used beyond the term of the project (for example, the software is to be deployed, the data is to be published or there are future secondary uses of the data), then it will be necessary to obtain signed consent from each participant. Otherwise, verbal consent is sufficient, and should be explicitly requested in the introductory script (see Appendix 1).

4. No incentives were offered to the participants.

The payment of participants must not be used to induce them to risk harm beyond that which they risk without payment in their normal lifestyle. People volunteering to participate in research may be compensated financially e.g. for reasonable travel expenses. Payments made to individuals must not be so large as to induce individuals to risk harm beyond that which they would usually undertake.

5. No information about the evaluation or materials was intentionally withheld from the participants.

Withholding information from participants or misleading them is unacceptable without justifiable reasons for doing so. Any projects requiring deception (for example, only telling participants of the true purpose of the study afterwards so as not to influence their behaviour) are deemed high risk and require approval from the relevant C-REC.

6. No participant was under the age of 18.

Any studies involving children or young people are deemed to be high risk and require ethical approval through the relevant C-REC.

7. No participant had a disability or impairment that may have limited their understanding or communication or capacity to consent.

Projects involving participants with disabilities are deemed to be high risk and require ethical approval from the relevant C-REC.

8. Neither I nor my supervisor are in a position of authority or influence over any of the participants.

A position of authority or influence over any participant must not be allowed to pressurise participants to take part in, or remain in, any study.

9. All participants were informed that they could withdraw at any time.

All participants have the right to withdraw at any time during the investigation. They should be told this in the introductory script (see Appendix 1).

10. All participants have been informed of my contact details, and the contact details of my supervisor.

All participants must be able to contact the investigator and/or the supervisor after the investigation. They should be given contact details for both student and supervisor as part of the debriefing.

11. The evaluation was described in detail with all of the participants at the beginning of the session, and participants were fully debriefed at the end of the session. All participants were given the opportunity to ask questions at both the beginning and end of the session.

Participants must be provided with sufficient information prior to starting the session, and in the debriefing, to enable them to understand the nature of the investigation.

12. All the data collected from the participants is stored securely, and in an anonymous form.

All participant data (hard-copy and soft-copy) should be stored securely (i.e. locked filing cabinets for hard copy, password protected computer for electronic data), and in an anonymised form.

Project title: Reengineering business processes for a local business

Student's Name: Arsalan Sadeghpour

Student's Registration Number: 21201414

Student's Signature: AS

Date: 20/10/16

Supervisor's Name: Dr. Natasha BELOFF

Supervisor's Signature: N. Beloff,

Date: 20/10/16

Ethical Compliance Form for UG and PGT Projects· Appendix 1: Introduction and Debriefing Scripts

If you intend to obtain verbal consent from your participants, you need to create an introduction script, to read out at the start of the study, and a debriefing script, to read out at the end of the study. You should get your supervisor's approval for your introduction and debriefing scripts before commencing your study. **NB: When submitting the signed Ethical Compliance form, you do not need to include this Appendix.**

Introduction Script

The introduction script must:

- state the general aim of the study
- explain why you need the involvement of other people
- describe what will happen in the study
- describe what data will be collected
- reassure the participant that any data collected will be stored securely, and in an anonymous format
- explain what interaction the participant may have with you during the study
- reassure the participant that this is not a test of ability
- state that the participant may withdraw at any time
- seek explicit consent
- allow the participant to ask questions

An example introductory script (italics indicate required information, some of which will be specific to your project):

**Web Interface Investigation
Final Year Project, 2014-15
Jane Student**

The aim of this study is to investigate the suitability of a new website [*state the general aim of the experiment*]. We cannot tell how good this website is unless we ask those people who are likely to be using it, which is why we need to run studies like these [*explain why you need the involvement of other people*]. I will give you some time to browse the website, before asking you to answer some questions [*describe what will happen in the study*].

I will be observing you while you perform the tasks [*describe what data will be collected*]. If you have any questions, please ask me, and please let me know when you are finished [*explain what interaction the participant may have with you during the study*].

I will ask you some questions at the end of the study [describe what will happen in the study]. Please remember that it is the system, not you, that is being evaluated [reassure the participant that this is not a test of ability].

You are welcome to withdraw from the study at any time [state that the participant may withdraw at any time]. Please be assured that any data collected will be stored securely and in an anonymous form [describe how the data will be anonymised and stored].

Do you agree to take part in this evaluation? [seek explicit consent]. Do you have any questions before we start? [allow the participant to ask questions].

Debriefing Script

The debriefing script that is used at the end of the study must:

- restate the main aim of the experiment
- if applicable, explain any other, related aims of the experiment, and any particular data collected
- allow the participant to make comments or ask questions
- give contact details for the student and supervisor
- thank the participant

An example debriefing script (italics indicate required information, some of which will be specific to your project):

**Web Interface Investigation
Final Year Project. 2014-15
Jane Student**

The main aim of the experiment was to investigate the suitability of this website [restate the main aim of the experiment]. In particular, I was looking to see whether you made use of the site map, and whether any of the links on the website seemed to be confusing. I wanted to know how easy the pages were to navigate [if applicable, explain any other, related aims of the experiment, and any particular data collected].

Do you have any comments or questions about the experiment? [allow the participant to make comments or ask questions]. Here are my contact details, and those of my supervisor, and please let us know if you have any further questions about this study [give contact details of student and supervisor to participant]. Thank you for your help [thank the participant].