Assessment 3 – Doughnut Ordering System (Master)

Henry Fallows, Kaleem Ashraf Parveen, Lukman Dad and Ivo Margarido Freire

# Requirements Analysis

## Stakeholders:

* Customer
  + They will be who is using the customer site which allows them to place orders and view previous orders.
* Chef
  + Will use the staff site to view the current orders to allow them to be prepared for collection or delivery
* Driver/Checkout staff
  + Will use the staff site to verify identity of customer and mark orders as completed.
* Admin
  + Will use the staff site to amend menu items and reorder if needed.

## Non-functional Requirements:

* Product
  + Website needs to be fast because “A website user’s attention span lies somewhere around eight seconds” (K, Howard) and after that the potential customer will be likely to stop using the site.
  + Easily navigable both on customer and staff site to encourage the use by potential customers and make the use by staff more efficient.
  + Website needs to be functional on both desktop and mobile views because “when a website is not optimized for mobile viewing… they (the users) will often get frustrated and leave” (M, Tungate). This means no orders will be processed through the system and it will be redundant.
* Organisational
  + Must be designed with company Typeface and Colour Scheme. This will relate their website to their stores and will be increasing their digital presence, with the website serving a dual purpose of advertisement for the store.
  + It is not law to provide a receipt of purchase however it is company policy that one is offered so there needs to be a system in place that can either print a physical one or generate e-receipts and send them by email.
* External
  + GDPR requires information about customers and their payment details to be kept secure and if that is neglected the company is liable. To prevent this, measures need to be put in place to minimise this risk and legally protect them.

## Functional Requirements:

From these user requirements we have determined that the functional requirements of the proposed system are:

* Customer Site

1. The system will allow the customer to order their food from the website. For the customer, this means that the customer will be able to see images of the food that they will be ordering and all the menu options during the ordering process.
2. The system will store accounts of the customers saving their preferences e.g., payment information, delivery location. This information will be used in the ordering process. The result of this is eliminating a point of miscommunication with the order and the delivery location in comparison to ordering by phone.
3. The system will allow payment for the order. This opens up the payment options for the food order contrasting to cash only as in the existing system.
4. The system will allow the customer to view their previous orders. This means that the customer will be able to track orders made on their account to view to monitor unusual activity.

* Store Site

1. The System will allow the chefs to view the orders that need to be prepared. This means that they can see exactly what needs to be made and not from a paper list that could possibly be wrong which is the case in the existing system.
2. The system will produce customer details for the order. This will allow the cashier or Driver to verify that the order is being handed to the correct person.
3. The database can be changed by the Driver/Cashier to confirm that the order was handed over. This will prevent orders being made and handed over more than once.
4. Menu items need to be amendable by admin to keep the product pricing up to date.
5. Menu items need to be able to be added by admin as new items may be added to the store menu at anytime
6. Menu items need to be able to be removed at any time as they may be discontinued at any time.

## Use Cases:

Customer:

* Order Food:

Actors: Customer, website, database

Inputs: Food Choice, Customer’s address, Customers Details, Customer Card Details

Outputs: Receipt for Customer

Normal Operation: The customer will select their desired food on the website. Once all items are selected, they will select the “proceed to checkout “button that will re-navigate the customer to a form to enter their name, address and card details. After entering this information, they will select a button saying, “pay for food”. Upon selecting this button, the value of the food is transferred from the customers bank account into the companies account and an e-receipt is generated and sent by email to customer.

Exception: Insufficient Funds – return to payment details screen and prompt a warning of insufficient funds.

Invalid Card – return to payment details screen and prompted to re-enter card details.

Invalid Address (Postcode) – Prompt invalid postcode and advise to check and re-enter.

Chef:

* View Orders:

Actors: Chef, Staff site, database

Inputs: Button Click

Outputs: List of current orders

Normal Operation: The chef will first click a button on the staff website. This will pull a list from the database of all orders that are to be prepared. From this the chef is able to prepare the food for the orders.

Exception: No orders that are incomplete - will return a message saying that there are no orders that need to be prepared.

Driver/Checkout staff:

* Complete Orders:

Actors: Customer, Driver/Checkout Staff, Staff site, database

Inputs: Order details, Button click

Normal Operation: Customer will tell driver or checkout staff their order details. The order details will be imputed into the staff site and then the identity of the customer can be confirmed. Once the food is handed over, the staff will mark the order as completed by clicking a button that updates the database.

Exception: Invalid order details – returns a message saying the order details are not valid.

Admin:

* Amend item:

Actors: Admin, Staff site, Database

Inputs: Admin login, Item id, New Price

Normal Operation: The Admin will login to the staff site, they will then navigate to the amend page where they will enter the ID value of the item, they want to amend the price of and the price they would like to change it too and press the change button. Upon pressing this button, the food record item (with the id that is the same as the one entered by the item) is updated with the price field now having the value of the price entered by the admin.

Exception: Invalid ID – prompt sent to admin stating that the food item with that id does not exist and is asked to check and re-enter.

Extreme Price – If a price is entered that is an extreme value compared to the existing then they are prompted to check and re-enter. After re-enter of the same value as previously entered the price is then changed.

* Add Item:

Actors: Admin, Staff Site, Database

Inputs: Admin Login, Food description, Food name, Price

Normal Operation: The admin will login to the staff site and navigate to the add new item screen. This will then ask them for the new product name, description and price. They will then press the add new item button and a record will be added to the database with an id that is auto generated.

Exception: Item already exists – The admin will be prompted by the system with a message saying that the item already exists, and they should either enter another or leave the page.

* Remove item:

Actors: Admin, Staff Site, Database

Inputs: Admin Login, Item ID

Normal Operation: The Admin will logon to the staff site and navigate to the Remove item page. They will be prompted to enter the id of the item that they would like to remove. They will then press the remove button which will prompt them with a message asking if they are sure. The item will select that they are sure and the item with the entered id will be removed from the database.

Exception: Item Does not exist – The Admin is prompted with a message stating that the item does not exist on the system and they are asked to either re-enter the id or to leave the page.

They are not sure – If the Admin is not sure then they are asked to either enter a different id or to leave the page.

# System Design

Many problems found with the existing system, such as to place the order customer needs to be present, check the menu choose the items required, then place the order and proceed with the payment, which demands more manual work and customers time, but when the customer wants to order over the phone, customer is unable to see the physical copy of the menu, this also lacks the verification that the order was placed for the appropriate items and customers’ address.

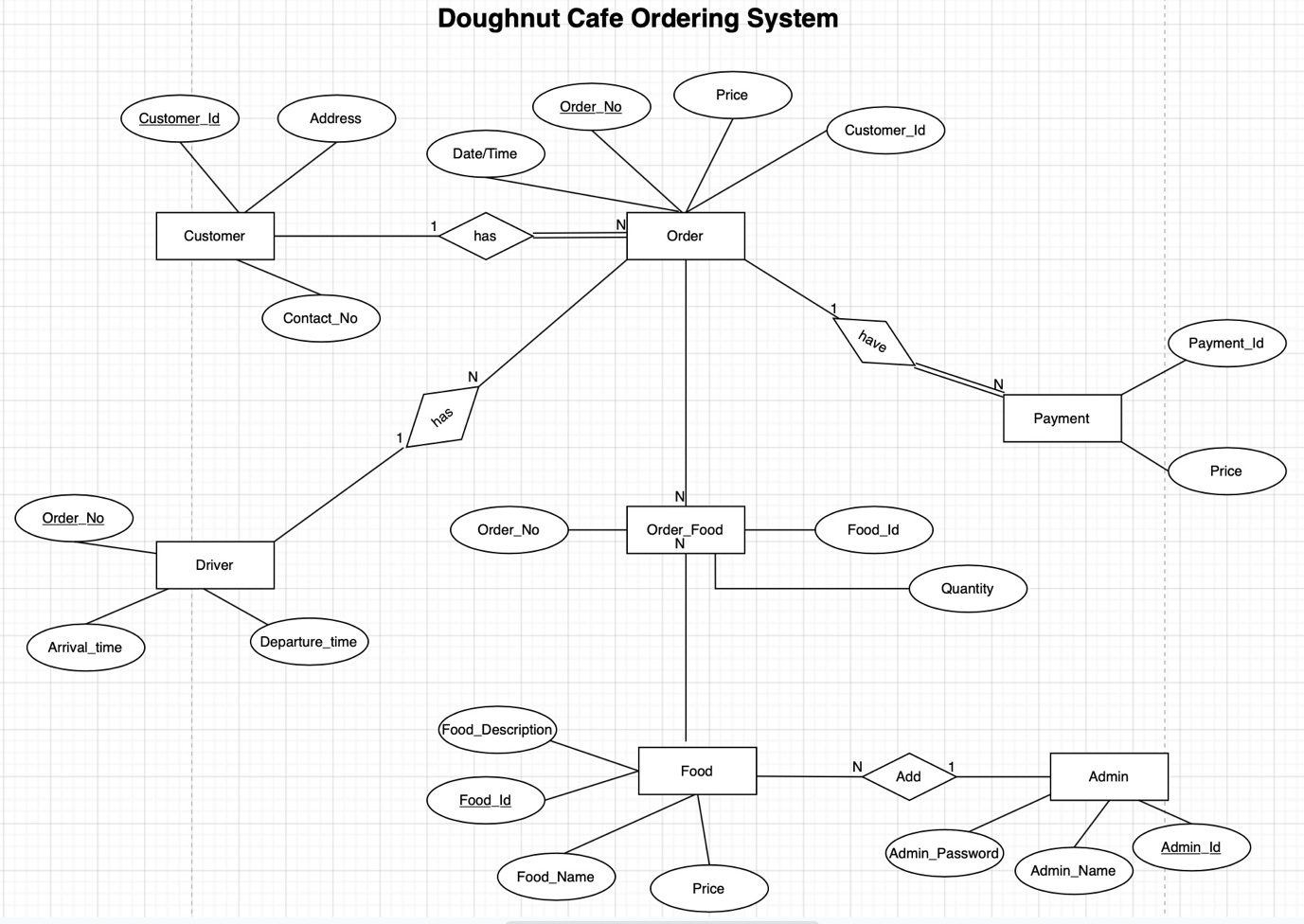
The new system begins with the customer entering their details (ID and password). Whenever that has been confirmed, the customer can put in a request determining the food and his amount required. Presently we get a window that shows the order number, customer ID, food name, cost and amount. When the customer finishes their request, they are diverted to the payment window where the absolute cost is shown, and the customer then can proceed with the payment and afterward the customer gets a message of affirmation of request, and if you are an administrator, you can choose the typical login alternative and enter the administrator accreditations (ID and password). When you enter the administrator entrance, you get the choice of adding food, erasing food or refreshing food. Once the selected operation is carried out, the final product, for example the additional food or the refreshed food list is shown and in the event that you have erased a food, that specific food vanishes from the principal menu.

## Flow Chart:

Diagram, engineering drawing

Description automatically generated

## ER Diagram:



This system will help people to easily order food, it can ensure that people won’t waste their vital time, instead use in other work productively. We can ensure that this system proves to be more cost effective, user friendly, and is much more reliable than other systems. This system will also help to reduce labor cost as there aren’t any limitations as such in the system.

# User Acceptance Criteria

# Team Working Strategy

## Choosing a Team Leader:

To approach this design document, our team first needed to choose a team leader, this is because in a team, we need one person (sometimes more) to make the calls and decide what each member does during the project, to coach each member and make sure the quality of the work the team is producing goes above the standards.

As a group we decided to have Henry as the team leader, this is because of his personality being more of a leader type of person compared to everyone else in the group.

## Assigning Roles:

We then looked at what we had to cover in the design document which were 4 main sections, the sections being:

* Firstly, write up a formal requirements analysis.
* Secondly, the design of a system that meets the requirements.
* Thirdly, user acceptance.
* For the last section, write up the team working strategy we used to approach the design document.

Because there are 4 sections and we have 4 people in the team, we all got to say what section we wanted to work on and fortunately, we all chose different sections, meaning there were no disagreements in the group.

## Communication Between Members:

For communication we used the Microsoft Teams application, we chose to use this application because it’s also used by professionals. The Teams application allows people to create group chats, audio and video calls which allows for great communication, in the chat you also have a “Files” tab and a “Notes” tab which helps with file organisation.

To ensure good communication we decided to have all members in a scheduled group call while we work so if we have anything to say to each other, we unmute our microphone and speak to the group, that way we get quick responses in the group.

We would also like to do face to face meetings however at the moment that is not possible because of the ongoing pandemic and lockdown so we all have to work remotely.

## Working Together:

While working together, apart from the Microsoft Teams application, were also using GitHub which is also a piece of software used by professionals. GitHub allows us to all work on the same file/repository, this software allows us to push documents and updates into the cloud and the other members can pull it to their computers, this way we all work on the latest files. This software is greatly used by software developers to send code and work on code as a team as well. Every time we can update the repository, we do by pushing and pulling the work.