Computing Unit 3: Coursework: Discussion Chapter

Oliver Looney 2089

Contents:

Overview

For my A-Level project I shall build a database driven order management, billing and delivery system for a locally owned family business called Pedro's Pizzas.

Currently Pedro's Pizzas does not have a computer system in use and so they use a paper-based system for everything. This causes difficulties when taking orders as some people's poor handwriting can slow things down or cause a mistake. Also, as all of Pedro's Pizzas' data is recorded on paper it is easily lost, which causes major problems, or data must be copied out many times so that everyone who needs it can have access, this causes data to become redundant which can lead to inconsistences across Pedro's Pizza's flat file paper-based system. An example of this is a customer's address being copied from out to give to the driver, data inconsistence here could cause the driver to deliver the food to the wrong address.

Also, as data is unable to be accessed quickly it makes it harder for the managers to evaluate and track the workers, e.g.: how often they are late, wrong orders, or how fast they work. This can cause problems when the employees want a raise for example. Another problem is that with the current system the managers have no way of seeing what dishes are most popular. As they do not currently permanently store customer's details, they cannot target certain customers with ads or discounts, which other restaurants in the local area can do.

Customers can **order** multiple items from a set **menu**, which is created and edited by the **managers**, or the **customer** can create their own pizza. This will require multiple validation algorithms as some items will not be able to be put together on a custom pizza. This will also require a more complex algorithm to calculate the cost as it will be based on input by the **customer**. The **customer** can order over the phone(data is then entered into the system by a member of the **counter-staff**), through a kiosk(data is entered into the system by the **customer** onsite) or online through this database driven system. If the **customer** orders online or on the phone, the system will automatically e-mail the customer a receipt but if the customer orders in the restaurant, their receipt will be printed out as well as receiving an e-mail.

For deliveries a separate **order** is printed out for the driver with other useful information about the **customer**, e.g.: address, phone number. All of the deliveries will have an added bill that will be at a set rate per mile from the customer to the restaurant plus a flat fee. This will be calculated before the driver leaves, using google maps API. The **drivers** would also need to see all of their delivery information online through this system. I will also design an algorithm that displays the customers' addresses on a map for the **drivers**.

The **managers** will need to be able to see the total sales and transactions for that day/week/month. Along with this should be profit and the system should also display what else money was spent on, e.g: how much of their revenue was spent on supplies, wages. This should be graphed by the system. As the system is not going to calculate wages the **managers** will manually input it.

The pizzeria would like to implement a discount scheme ere loyal **customers** are rewarded with vouchers/discount. **Customers** spending over a certain amount will receive discount vouchers via email. Different **staff** must have different views of the system, including counter-staff, drivers and managers.

Every receipt will have a barcode that when scanned will bring up information on the **customer** who ordered it, and another barcode that will bring up information on the **order**.

The **staff** will have the orders that they created tracked so that they can receive a bonus based on how well they do.

Identification of Stakeholders:

- Customers
- Staff
    - Managers [2]
    - Drivers [8]
    - Counter Staff [12]
    - Cooks [5]

Stakeholders' Requirements:

Management:

- View, edit, add and delete all customer & staff details and the menu, ingredients and suppliers' details
- View staff hours
- Change staff hours
- Sales details at the end of the day/week/month
- Ability to change the cost of items on the menu
- View high level data processing on the customer
- Data should be able to leave the system through csv & pdf files
- View pdf files of orders, order tickets and information on the customer
- Have admin access, i.e be able to do everything that all of the staff can do as well
- Create coupons for discounts
- View graphed output of:
    - How efficient staff are
    - Most loyal customers
    - Most popular items on the menu
    - Be able to see if there is a trend of what certain customers are buying, e.g.: age v type of pizza

Driver:

- Be able to view their own details
- View customers' addresses when they have to deliver to that customer
- View their deliveries' food order, so they can make sure that they have everything
- Get a printed note for each order
- View order receipts as a pdf file

Server:

- Be able to view their own details
- Be able to see what table a customer has booked
- Be able to input the customer's food order

- Be able to see their weekly pay
- Be able to see their booking information
- Be able to view the customer's receipt as a pdf file
- Be able to view all orders and order items on the system

Customer:

- Be able to view their own details, so that the company can adhere to new data protection laws e.g.: GDPR
- Be able to edit their own details
- Be able to create their own account
- Be able to view the menu
- Be able to add a booking or a delivery
- Be able to see all of their previous orders
- Be able to see what table they have associated to their booking
- Be able to see what server they have for their booking
- Be able to view their order receipt as a pdf file
- Be able to receive confirmation of their order through email or SMS
- Be able to receive an email confirming their order, with a pdf file of their receipt attached
- Should get recommendations on other products based on what they are ordering
- Use coupons for discounts when ordering

Cook:

- Be able to view their own details
- View the ingredients table
- Be able to view order details in a pdf file
- Be able to view order ticket details in a pdf file
- Be able to view all upcoming & live orders and order items on the system
- View the food orders table to see what they must prepare

Limitations:

- The system will not be able to process payments. All payments must be done with cash or through a card machine in the restaurant
- The system will not be able to automatically order ingredients when they are running out
- Will not exchange/port data to accounting software used by the shop.
- It will not be able to take user input simultaneously without extensive use of third party libraries and hosting servers
- It will not have an in built messaging service for the employees to communicate
- It will not be able to handle the payment of the staff
- Will not integrate with driver GPS. This will have to be completed manually.

Pedro's Pizzas

Discussion of Proposed Solution

Background

- I shall build a database driven web application for a locally owned family business called Pedro's Pizzas.
- Pedro's Pizzas serves a variety of Italian cuisine and they offer sit-in, delivery or collection.
- Pedro's Pizza has five delivery drivers on the weekend and three during the weeknights. They have a weekend manager and a weekday manager. They also have ten cooks and ten servers who work different shifts across the week.

Problem

- Currently Pedro's Pizzas does not have a computer system in use and so they use a ***paper-based*** system for everything
- This causes ***data inconsistences and redundant data*** and leads to many mistakes which are harmful to the business
- Data can be ***slow*** to access
- All bookings and deliveries/collections must be done over the phone, ***wastes time and employees*** which can be harmful to the business.

.

Identification of Stakeholders

- Customers

- Staff
  - Managers [2]
  - Drivers [8]
  - Cooks [10]
  - Servers [10]

The Scope

- CUSTOMERS can ORDER multiple PIZZAS, that are selected form a set menu or created by the customer. These orders are processed over the phone or by drop ins to the shop. ORDERS can be collect or take away. CUSTOMERS must be INVOICED for the transaction. If this is a delivery, a delivery docket must be created to include address and delivery instructions. Deliveries are billed at a set rate and are limited to a 6 mile radius. All ORDERS & SALES must be individually billed and include a printed output. MANAGERS must be able to access all orders and the end of a day to see total sales. The pizzeria would like to implement a discount scheme ere loyal customers are rewarded with vouchers/discount. Customers spending over a certain amount will receive discount vouchers via email. Different staff must have different views of the system, including floor staff, drivers and managers.

Limitations

- The system will not be able to process payments. All payments must be done with cash or through a card machine in the restaurant
- The system will not be able to automatically order ingredients when they are running out
- Will not exchange/port data to accounting software used by the shop.
- It will not be able to take user input simultaneously without extensive use of third party libraries and hosting servers
- It will not have an in built messaging service for the employees to communicate
- It will not be able to handle the payment of the staff
- Will not integrate with driver GPS. This will have to be completed manually.

Stakeholder Requirements

Management

- View, edit, add and delete all customer & staff details and the menu, ingredients and suppliers' details
- View staff hours
- Change staff hours
- Sales details at the end of the day/week/month
- Ability to change the cost of items on the menu
- View high level data processing on the customer
- Data should be able to leave the system through csv & pdf files
- View pdf files of orders, order tickets and information on the customer
- Have admin access, i.e be able to do everything that all of the staff can do as well
- Create coupons for discounts
- View graphed output of:
- How efficient staff are
- Most loyal customers
- Most popular items on the menu
- Be able to see if there is a trend of what certain customers are buying, e.g.: age v type of pizza

Driver

- Be able to view their own details
- View customers' addresses when they have to deliver to that customer
- View their deliveries' food order, so they can make sure that they have everything
- Get a printed note for each order
- View order receipts as a pdf file

Cook

- Be able to view their own details
- View the ingredients table
- Be able to view order details in a pdf file
- Be able to view order ticket details in a pdf file
- Be able to view all upcoming & live orders and order items on the system
- View the food orders table to see what they must prepare

Server

- Be able to view their own details
- Be able to see what table a customer has booked
- Be able to input the customer's food order
- Be able to see their weekly pay
- Be able to see their booking information
- Be able to view the customer's receipt as a pdf file
- Be able to view all orders and order items on the system

Customer

- Be able to view their own details, so that the company can adhere to new data protection laws e.g.: GDPR
- Be able to edit their own details
- Be able to create their own account
- Be able to view the menu
- Be able to add a booking or a delivery
- Be able to see all of their previous orders
- Be able to see what table they have associated to their booking
- Be able to see what server they have for their booking
- Be able to view their order receipt as a pdf file
- Be able to receive confirmation of their order through email or SMS
- Be able to receive an email confirming their order, with a pdf file of their receipt attached
- Should get recommendations on other products based on what they are ordering
- Use coupons for discounts when ordering

Elements to be Considered	Feedback	Accept or Reject	Justification	Action to be taken
Management requirements	The manager should get daily, weekly, monthly and yearly financial reports to see how the total sales for that day/ week/ month/ year	Accept	It would allow the manager to clearly see how much profit that they have made in a period of time.	Calculates all financial transactions over the past day/ week/ month/ year and outputs it as a hard copy or pdf to the manager.
Management requirements	The manager should be able to see the most loyal customers and give them discounts	Accept	The manager would need this for an easy targeted discount campaign	Implement a feature that allows the manager to see who has ordered the most/spent the most
Driver requirements	They should receive a printed copy of the delivery information	Accept	They should have this in case they drive to an area with low mobile coverage	Give the driver the option to parint
Driver requirements	They should get the customers addresses on top of a map *	Accept	It will save the driver from copying all of the customers addresses out into another map app and will allow	Give the driver the option to put all of the addresses that they are going to deliver to on a map.

			them to easily plan their route	
Cook requirements	Allow them to view the menu entity	Accept	They need to see what ingredients are in each dish	Give them a larger view access
Cook requirements	Allow them to see the average amount of each dish that is ordered on a day **	Accept	So that when they arrive they can see how busy they are going to be and how much preparation they need to do	Give the cook an option to view how much of each dish is ordered on a given day
Server requirements	The server needs to be able to print a bill for the customers	Accept	So the customer gets a receipt	Give the server an option to print a bill for the booking
Customer requirements	Select favourite items to order	Accept	It would be a helpful feature for the customer and relatively simple to implement	Allow the user to select a specific previous order and link that order with the customer's ID in a new favourite order table
Customer requirements	When ordering they should be able to edit their order until they lock it in	Accept	Allows the customer to change their mind as they order without cancelling the whole order	Not save the order till the customer confirms it
Staff requirements	Allow staff to edit their own details	Reject	The manager needs accurate contact information of the staff, so the manager must be the confirm it	----------
Manager & Customer requirements	The system should automatically email customers on special occasions, e.g.: birthdays With offers	Accept	It would benefit the customer and the business	If the current date is 1/2 days before the customers birthday, the customer will get an email with a discount for their birthday
Manager requirements	The manager should be able to hide items on the menu from customer's so	Accept	This would make managing the products easier. It would also allow the	Each product will have a Boolean flag which the manager will be able to toggle.

	that they can be removed from the menu but no t deleted from the database		manager to view data on old items, instead of loasing that data. This also increase the data integrity of the database as less orders will be affected by the removal of items from the customer's option	A SQL select statement will use the WHERE clause to only show the customer items the manager has set the availability to true.

*This will require me to make use of the google maps api to do so, which will greatly expand the scope of this project.

**This shall require high level data processing, which will be further planned in the design chapter using flowcharts and pseudocode.