REQUIREMENTS DOCUMENTS

Group 5: Rutvik Marakana, Zachary Okwuosa, Liam Hardage, Adam Chung

* **Requirements Elicitation**
* Functional Requirements

**Q.1.** Is the customer given the flexibility to update the personal information?

**Ans 1:** Yes, the customer can update the personal information anytime. The customer can also update the payment information by adding or deleting cards.

**Q.2.** What access is given to the customer?

**Ans 2:** The customer can access all the menu items and choose what to order. The customer can further access the personal information.

**Q.3.** How would the customer know what items he/she has ordered and what are yet left to order?

**Ans 3:** The customer will be able to check cart items, cart total and process the payment.

**Q.4.** How would the employees of the shop know the orders to be prepared?

**Ans 4:** The cashier computer will have a list of all the orders. As soon as the order is prepared, it will be deleted from the list.

**Q.5.** How will the employees know the details of the customer?

**Ans 5:** The cashier computer will be able to generate the details of the online orders. It will include the customer name, customer address and the mode of payment.

**Q.6.** How does the store check the number of customers who have an account with them?

**Ans 6:** The manager of the store has access to the customer database. He/she can pull up all the details of the customer like the personal information, number of orders till date, number of coupons used till date and many more.

**Q.7.** Do you change the menu items regularly or not? If so, who decides the new items?

**Ans 7:** Yes, our store does change items regularly to give customers more items to select from. The store manager decides what items to keep and what to remove based on monthly sales.

* Non-functional requirements

**Q.8.** What steps have been taken about customer information security?

**Ans 8:** Security of customer information is one of the most important non-functional requirements. So, we have restricted access of customer database. Only the manager can view the customer database.

**Q.9.** What if the system fails to operate suddenly? Have you considered any steps to deal with this situation?

**Ans 9:** Yes, we have planned to back-up our data so that the company don’t face any problems in case of failures. As soon as the problem is resolved, the data can be restored easily.

**Q.10.** Did you consider usability of the software while deciding the functional requirements of the software?

**Ans 10:** Yes, we did take into consideration the usability of website. The website is very easy to access. The steps to order are very easy to follow.

**Q.11.** How many post-release people will the website need?

**Ans 11:**  According to us, 15 post-release people would be enough. But we have kept this number to be flexible. We can hire some more people depending on the number of problems faced after release.

* + Design Constraint

**Q.12.** What programming language have you decided to use?

**Ans 12:** We all are familiar with Java, so we have decided to use Java for coding needed in the software development.

**Q.13.** Will the website be accessible from all type of devices?

**Ans 13:** No, our website will not be accessible from all devices. The website will be accessible only on computer having at least intel core I-3 processor.

**Q. 14.** How many steps will it take for the customer to create an account on the website?

**Ans 14:** It will take 5 steps to create an account on our website.

* + Process Constraint

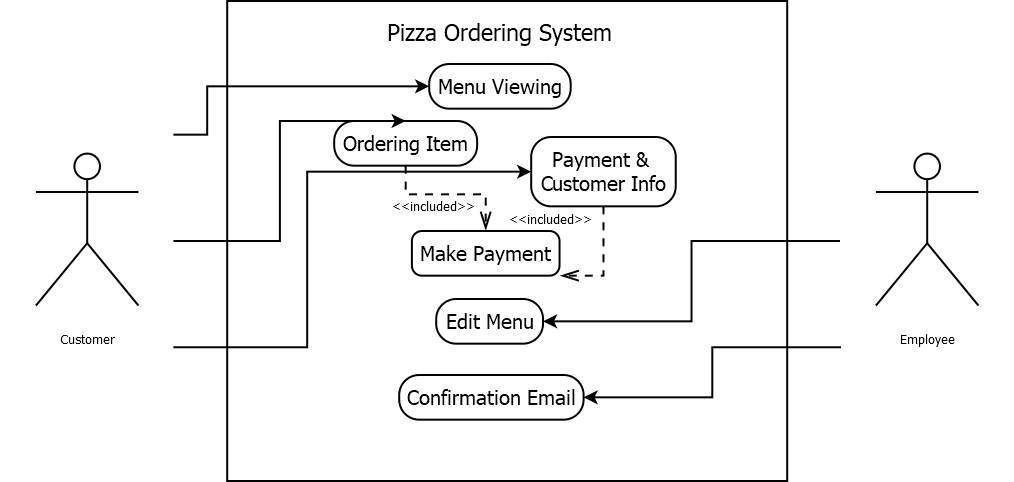
**Q.15.** How many processes will the software handle in 1 minute?

**Ans 15:** Our software can do 10 processes in one minute.

**Q.16.** Does the website have capability to track the order?

**Ans 16:** The website is unable to track the order because we don’t have enough technical staff and resources available to create the tracking mechanism. Further, there is not enough budget to develop the tracking mechanism.

A) **Use-Case Diagram (Use flow of events document)**



Flow of Events

X Flow of Events for the <name> Use Case

X 1. Preconditions

X 2. Main Flow

X 3. Sub-flows

X 4. Alternative Flows

1.0 Flow of Events for the Pizza Ordering System Use Case

1.1 Preconditions

The application must be operational before the Use Case can start. The payment system must log into the interface and be viewing the menu.

1.2 Main Flow

The user can see a wide arrange of items. By selecting an item(s), it is added it to their cart. After selecting a main item, an option menu will display for the size, crust type, toppings, etc. When the user is done ordering, they can make their way to their cart to checkout. At the cart page they are asked if the user is an Existing User or New User. Existing users are greeted with a login page followed by a prompted to select a method of order (take-out or delivery) and payment page. All previous information saved on their profile will be automatically filled in (Address, phone, credit card). For new users, they are prompted to sign up or continue as a guest. This is also followed by the same pages as the existing user but need to fill out the delivery information.

After the order has been confirmed, an email is sent out to the user that indicates the customer’s order information.

1.3 Sub-flows

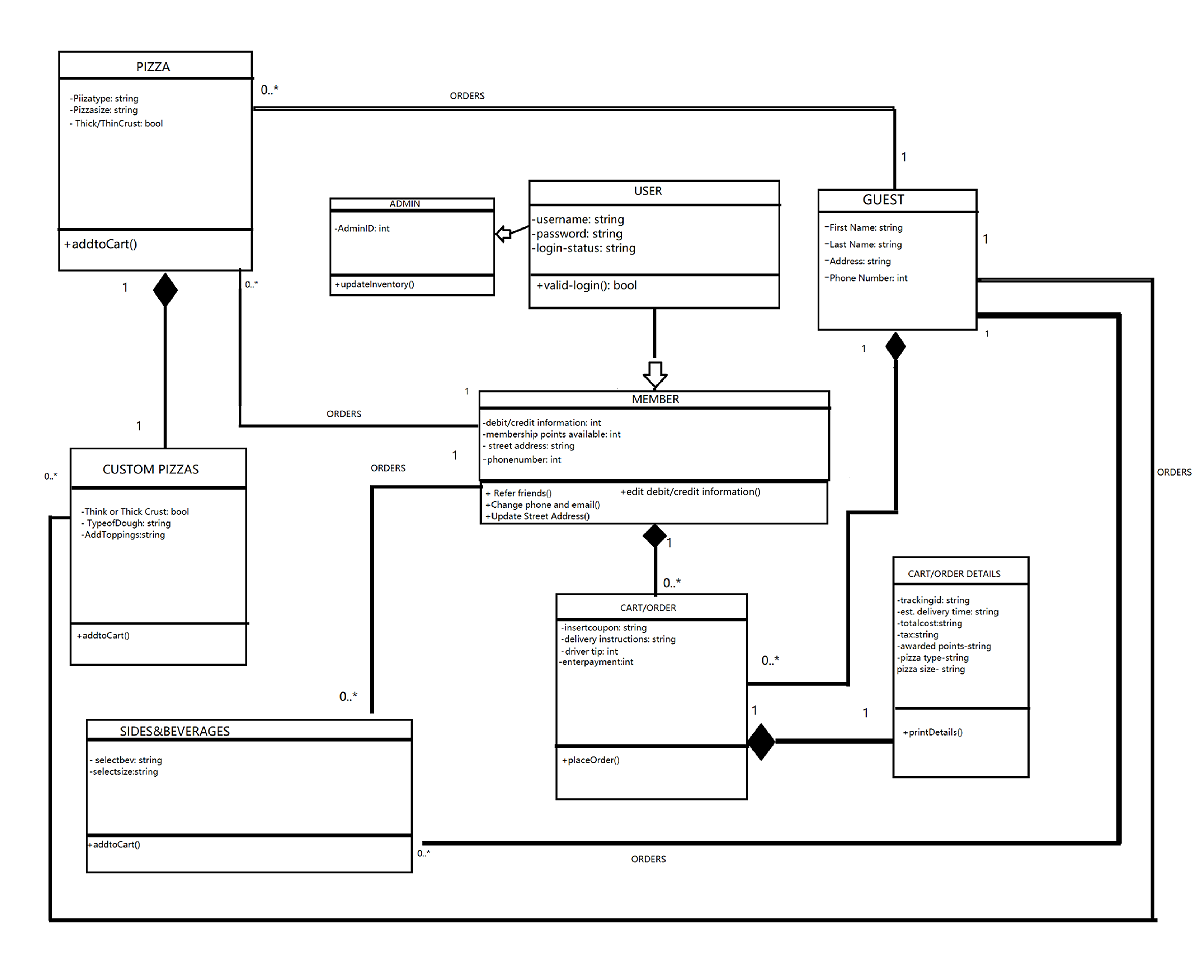
S-1: User-created item: This system allows the customization of a pizza from scratch starting from displaying multiple options such as size, crust type, toppings, and cheese type.

S-2: Change user information: When being prompted to confirm to be an existing user, they can change/update their address, phone number, email, etc. After updating the user can save the information to and use the updated information for future orders.

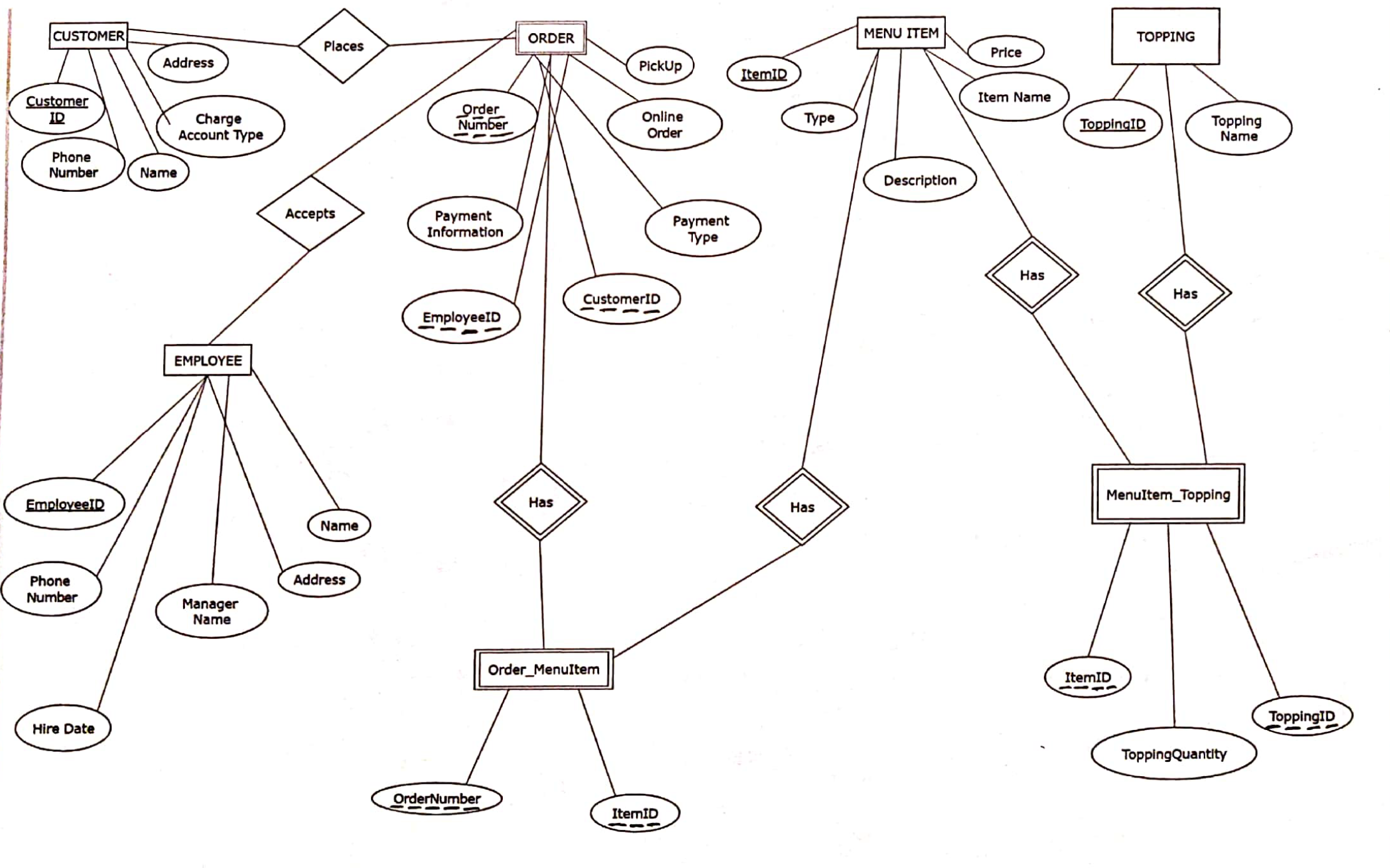
1.4 Alternative Flows

E-1: An invalid Username and password are entered. The user will be prompted to reenter their username and password again or create an account.

E-2: An invalid address is entered. The address will be checked if the formatting to match up a set input. Users will be warned and asked to reenter or transfer the information to a take-out order.

B) **Class-Diagrams (Class Documentation)**

C) **Entity relationship diagram (ERD)**



In the ERD, the strong entities are CUSTOMER, EMPLOYEE, MENU ITEM and TOPPING. The weak entities are ORDER, Order\_MenuItem and MenuItem\_Topping. There is one-to-many relationship between CUSTOMER and ORDER, EMPLOYEE and ORDER. There was many-to-many relationship between ORDER and MENU ITEM & MENU ITEM and TOPPING. To remove the many-to-many relationship, a new entity called Order\_MenuItem is created between ORDER and MENU ITEM. Further, a new entity called MenuItem\_Topping is created between MENU ITEM and TOPPING.

D) **Decision table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | If a user logs in | | | |  |
|  | See Menu | Ordering | Info/Pay | Edit Menu | Conf/Email |
| Customer | True | True | True | False | False |
| Employee | True | False | False | True | True |

|  |  |  |  |
| --- | --- | --- | --- |
| Topping price increase | | | |
|  | Meats | Veg | Cheese |
| 1 | 0.35 | 0.50 | 0.25 |
| 2 | 0.70 | 1.00 | 0.50 |
| In addition to base products | | | |
| Pizza w/1 topping | 10.34 | 10.49 | 10.24 |
| Pizza w/2 topping | 10.69 | 10.99 | 10.49 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Potential rewards | | | | |
| Membership\/ | Coupon A | Coupon B | Coupon C | Coupon D |
| >1 year | O | X | X | X |
| 1-2 years | O | O | X | X |
| 3-4 years | O | O | O | X |
| 5+ years | O | O | O | O |