

**PHASE 1: SUBMISSION CHECKLIST/SIGNOFF SHEET****GROUP #: 20****GROUP NAME: Bobby Drop Tables**

## Deliverables:

- ✓ Requirements description
- ✓ ER diagram with min/max specifications
- ✓ Constraints not in ER schema
- ✓ Summary of processing needs  
(Categorization into expected forms, reports, and queries)

## Assessment:

- ✓ Group Status Report
- ✓ Returned Phase 1 Intermediate
- ✓ Confidential Peer Evaluation (submitted separately)

We have each reviewed the contents of this deliverable.

	<i>Printed Name</i>	<i>ASU Email</i>	<i>Signature</i>
Phase Leader	Ryan	code@asu.edu	_____
Phase Recorder	Pat	paoconno@asu.edu	_____
Phase Checker	Connor	clmaddox@asu.edu	_____
Technical Advisor 1	Lisa	lrashidi@asu.edu	_____
Technical Advisor 2	Todd	twenker@asu.edu	_____

## **CSE 412/598**

### **Database Management**

### **Microsoft Access Project**

### **Restaurant Database (TACO TABLES)**

### **Phase 1 Deliverable**

**Due Date:** September 24  
**Group No:** 20  
**Group Name:** Bobby Drop Table  
**Members:** Ryan Schachte  
Todd Wenker  
Connor Maddox  
Pat O'Connor  
Lisa Ranjbar

**Table of Contents**

<b>1. Introduction</b>	
<b>2. Requirements and Description of the Enterprise</b>	<b>4</b>
<b>3. Conceptual Design ER Diagram</b>	<b>5</b>
<b>4. ER Diagram Uncaptured Constraints</b>	<b>6</b>
<b>5. Processing Needs</b>	<b>7</b>
5.1 Forms	7
5.2 Queries	11
5.3 Reports	13
<b>GROUP STATUS REPORT</b>	<b>14</b>

## **1. Introduction**

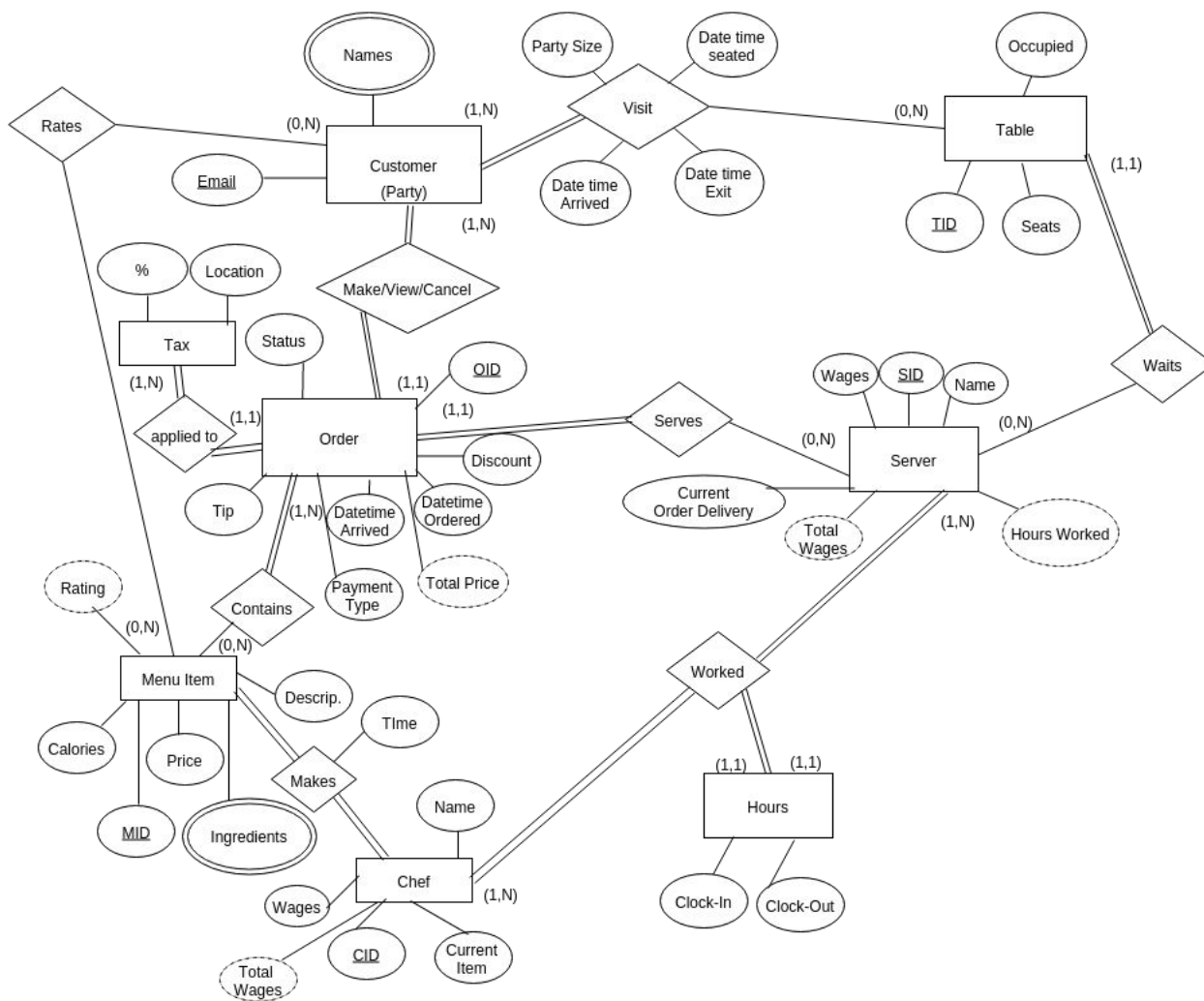
The objective of this software is to allow for a comfortable and friendly experience for customers ordering at our restaurant.

## **2. Requirements and Description of the Enterprise**

The Restaurant Database will need to store information about employees, customers, menu items, orders, and tables pertaining to the restaurant. This data will organized as follows:

- Chef: Chefs will have names, IDs, and wages. We will also see the Hours for each chef as well as what menu items they are making at a specific time.
- Server: Servers will have names, IDs, wages, and will be assigned tables to serve. We will also store what orders they have served and track the Hours that they worked. They also have an attribute to keep track of what item they are in the process of serving.
- Menu Item: Menu items will have descriptions, names, ingredients, calories, price, and rating. The rating will be derived from overall customer ratings.
- Table: Tables will be given IDs. We will also track the number of seats and whether or not it is occupied.
- Customer/Party: This is a group of customers visiting the restaurant. Parties will be tracked by a party leader's email. We will also store that party leader's name. Parties will make orders where they select menu items from the menu and will be able to rate the menu items ordered. Once the customer makes an order, they will have the ability to see the status of an order or cancel the order. We will store each party's visit as well as relevant information to that visit.
- Order: Orders will contain menu items. They will also have IDs, status, as well as attributes pertaining to the final price. The time the order is placed as well as when it is completed will be stored. An order also has a type of payment(cash/credit) applied to it.
- Tax: We will also track the tax percentage of the restaurant's location. This will enable the database to be used in multiple states for franchise restaurants.

### 3. Conceptual Design ER Diagram



#### 4. *ER Diagram Uncaptured Constraints*

The following is a list of constraints that are not captured by the ER Diagram:

- $0 < \text{Table.Seats} \leq 12$
- $4 \leq \text{Server.Wages} \leq 5.50$
- $0 < \text{Visit.PartySize}$
- $\text{Visit.PartySize} \leq \text{Table.Seats}$
- $\text{Visit.DateTimeArrived} \geq 10:00$  (the time the restaurant opens)
- $\text{Visit.DateTimeExit} \leq 21:00$  (the time the restaurant closes)
- $\text{Visit.DateTimeArrived} < \text{Visit.DateTimeExit}$
- $\text{Visit.DateTimeArrived} < \text{Visit.DateTimeSeated} < \text{Visit.DateTimeExit}$
- $\text{Hours.ClockIn} \geq 08:00$
- $\text{Hours.ClockOut} \leq 24:00$
- $8.50 \leq \text{Chef.Wages} \leq 10$
- $10:00 < \text{Makes.Time} < 20:30$  (the kitchen is open from 10:00 to 20:30)
- $\text{Menuitem.Price} > 0$
- $0 \leq \text{Menuitem.Rating} \leq 5$  (the rating scale for menu items must be between 0 and 5)
- $\text{Order.Tip} \geq 0$
- $0 \leq \text{Order.Discount} \leq 100$  (in terms of percentages, discounts can range from 0 to 100, with 100 being a complimentary Order)
- $10:00 < \text{Order.DateTimeOrdered} < 20:00$  ( the kitchen stops taking orders at 20:00)
- $10:00 < \text{Order.DateTimeArrived} < 21:00$
- $\text{Order.DateTimeOrdered} < \text{Order.DateTimeArrived}$

## 5. *Processing Needs*

The processing needs of the COMPANY enterprise are characterized with respect to operations on customers, orders, menu items, chefs, and servers.

### 5.1 Forms

Form Name	Description
	<b>ORDER</b>
Make Order	An Order will be made by a Party after they are seated at their table. The Party will be shown Menu Items and will be able to select those items to build an Order. When the Party is ready they can submit the Order to be made.
Edit Order	After an Order is made, edits to that Order can be made. Editing will include adding or removing Menu Items from the Order, or adding special cooking instructions to certain Menu Items
Cancel Order	After an Order is made, the Order can also be cancelled. The Party will then be prompted to make a new Order or they can leave.
Set Discount	After an Order is made, Parties can enter coupon codes in order to apply a discount to the Order.

Form Name	Description
	<b>CUSTOMER/PARTY</b>
Add Customer	Before making an Order, a Party will be asked to input their email address. This will be used to identify them. The names of the people in the Party will also be recorded.
Rate Menu Item	After an Order has been received, the Party will be able to rate the Menu Items that they received. This will be out of 5 stars.

Pay Meal	After an Order has been received, the Party will be responsible to pay for the Order. This can be done with cash, credit, or debit card. If a card is used, they can simply swipe it on the side of the monitor. If Cash is being used, the server will need to assist in the transaction at the front.
----------	---

Form Name	Description
	TABLE
Add Table	Adding a Table to the restaurant can be done in order to keep the number consistent with how many actual tables there are.
Remove Table	Tables can also be removed from the restaurant in order to stay consistent.
Update Table Status	This will be used to indicate if a Table is occupied or not. This will help servers track which Tables are open or need to be cleaned.

Form Name	Description
	SERVER
Add Server	Servers will be hired by the managers and will need to added to the application. The manager will add all the relevent information about the Server to the application. The Server will be asked to confirm that the information is correct.
Remove Server	When a Server is terminated, they will also be removed from the application. This can be done at the discretion of the managers. Terminated Servers will be kept in the database for reference, but will not be assigned any hours.
Update Server Name	If a Server changes their legal name, they should be able to change it in the application as well. This will need the manager's approval.
Assign Table	When a Server is working, they will be assigned a Table that they are working. This will track what Servers are interacting with what Tables during their shift.

Form Name	Description
-----------	-------------



	HOURS
Clock In	When a Server/Chef arrives at the restaurant, they will need to clock in to the application to start getting paid for that shift.
Clock Out	When a Server/Chef is finished with their shift, they will need to clock out of the application to finish being paid for that shift.

Form Name	Description
	CHEF
Add Chef	Chefs will be hired by the managers and will need to added to the application. The manager will add all the relevant information about the Chef to the application. The Chef will be asked to confirm that the information is correct.
Remove Chef	When a Chef is terminated, they will also be removed from the application. This can be done at the discretion of the managers. Terminated Chefs will be kept in the database for reference, but will not be assigned any hours.
Queue Menu Item	After an Order is Placed, Menu Items will enter the queue. This queue will tell a Chef what Menu Item to make next. Multiple Chefs can work on separate Menu Items from an Order, but Menu Items within that Order should be guaranteed to be completed in a timely matter together.
Complete Menu Item	After a Chef completes a Menu Item, the Chef will be responsible for marking it as complete so that the status of the Order can update and the Server can server the food to the Table it belongs to.

Form Name	Description
	MENU ITEM
Add Menu Item	Managers will be able to add Menu Items to the to the application. They will be responsible for adding all of the necessary information about the Menu Item they are adding and will be shown an error if any of it is incomplete.

Remove Menu Item	Managers will be able to remove Menu Items from the application. When a Menu Item is removed from the application it will still exist for reference in the database.
Update Menu Item	Managers will be able to edit any attributes relating to the Menu Item including ingredients, price, description, and calories.

## 5.2 Queries

Query Name	Description
	<b>CUSTOMER/PARTY</b>
View All Customer Visits	For each Party that Visits a Table, retrieve Party Size, Datetime Arrived/Seated/Exit.
Last Visit	For a Party who has Visited before, retrieve information on last Visit and Order.

Query Name	Description
	<b>MENU ITEM</b>
Most Popular	For Menu Items that have been rated, retrieve a list of most popular items with a default limit or a limit that can be specified.
Get Price	For any Menu Item, retrieve the price of the Menu Item based on the <u>MID</u>
Get Ingredients	For any Menu Item, retrieve a list of all ingredients.

Query Name	Description
	<b>ORDER</b>
Retrieve Status	For any Order, retrieve the status of the Order.

Query Name	Description
	<b>TABLE</b>
Retrieve Seat Count	For any Table, retrieve the number of seats at that Table.
Retrieve Occupied Status	For any Table, retrieve whether or not the Table is occupied.
List By Seat Count	For unoccupied Tables, retrieve a list of those Tables with their number of seats.

Query Name	Description
	<b>SERVER</b>
Retrieve Wages	For any Server, retrieve their current wages.
Retrieve Current Tables	For any clocked-in Server, retrieve a list of Tables that they are currently serving.
Retrieve Hours Worked	For any Server, retrieve the number of hours they have worked within a specified datetime range.

Query Name	Description
	<b>CHEF</b>
Retrieve Wages	For any Chef, retrieve their current wages.
Retrieve Items Cooked	For any Chef, retrieve the Menu Items they have cooked within a datetime range.
Retrieve Hours Worked	For any Chef, retrieve the number of hours they have worked within a specified datetime range.
Retrieve Total Items Cooked Count	For any Chef, retrieve a count of how many Menu Items they have cooked.
Retrieve Currently Queued Items	For any clocked-in Chef, retrieve a list of their queue Menu Items that they will be cooking next.

### 5.3 Reports

Report Name	Description
	CUSTOMER/PARTY
Most frequent customers	Finds the customers that frequent the restaurant the most based on the amount of past visits.

Report Name	Description
	CHEF
Hardest Working Chef	Finds hardest working chef based on total count of items cooked
Most Cooked Items	Lists, in descending, order the most cooked items based on cook count for each associated item.

Report Name	Description
	MENU ITEM
Total amount sold between time span	List the number of each menu item sold for the given time span.
Total profit made between time span (gross income)	Calculate the profit made by adding up all the paid orders for the given time span.

**GROUP STATUS REPORT****GROUP #:** 20**GROUP NAME:** Bobby DROP TABLE**PHASE #:** 1

We have each reviewed the contents of the following group status report:

	<i>Printed Name</i>	<i>Signature</i>
Phase Leader	Ryan	_____
Phase Recorder	Pat	_____
Phase Checker	Connor	_____
Technical Advisor 1	Lisa	_____
Technical Advisor 2	Todd	_____

Dates & attendance at group meetings in this phase:

Friday, Aug 28 3:00 - 5:00 PM All group member present

Monday, Sept 7 3:00 - 5:00 PM All group member present

Thursday, Sept 10 5:00 - 6:30 PM All group members present

Tuesday, Sept 22 9 - 9:45 PM(Google Hangouts) All group member present

Overview of progress on project as of September 22:

The ER diagram has been updated and all forms, queries, and reports have been created and are almost completed.

The final deliverable has been started and each item is being added as it is completed

**CONTRIBUTIONS OF GROUP MEMBERS**

( · indicates completed contribution o indicates contribution to be completed )

Leader:

- Attended first meeting to create rough draft of ER Diagram
- Initialized Github repository
- Translated rough draft of ER Diagram to digital format
- Attended third meeting to revise ER Diagram and requirements
- Made first draft of forms, queries, and reports

Recorder:

- Attended first meeting to create rough draft of ER Diagram
- Assisted in Requirements document final draft
- Updated requirements on Github repo

- Attended third meeting to revise ER Diagram and requirements
- Wrote Descriptions for all forms and queries

Phase Checker:

- Attended first meeting to create rough draft of ER Diagram
- Updated ER Diagram with constraints
- Final analysis of Phase 1 Intermediate Document
- Attended third meeting to revise ER Diagram and requirements
- Made first draft of forms, queries, and reports
- Final analysis of Phase 1 Final Document

Technical Advisor 1:

- Attended first meeting to create rough draft of ER Diagram
- Assisted in Requirements document final draft
- Attended third meeting to revise ER Diagram and requirements
- Made first draft of forms, queries, and reports

Technical Advisor 2:

- Attended first meeting to create rough draft of ER Diagram
- Assisted in Requirements document final draft
- Attended third meeting to revise ER Diagram and requirements
- Made first draft of forms, queries, and reports