DATABASE REQUIREMENTS SPECIFICATION

PREPARED FOR

Project Phase - I

DATA AND APPLICATIONS

PREPARED BY

DAKSH RAWAT - 2018101087

VASU SINGHAL - 2018101074

PRIYANSHU MADAAN - 2018101097

1. INTRODUCTION

Here, we have considered a restaurant as our mini-world. The restaurant has several customers coming in everyday and several employees working in different departments.

1.1 PURPOSE

The purpose of this document is to build an online system to manage the orders received and the working employees of the restaurant.

1.2 INTENDED AUDIENCE

This project is a prototype for a restaurant management system. It is useful for the administrative team of the restaurant. Only the restaurant's administrative team will have access to the management system.

1.3 INTENDED USE

The users can use the application to manage the orders received by the restaurant. They will be able to query the database to fetch the required information about any particular order. They can also make database transactions regarding employees.

2. DATABASE REQUIREMENTS

2.1 ENTITIES AND ATTRIBUTES

- **EMPLOYEES**
    - Employee Id
    - First Name
    - Last Name
    - Name (Composite)
    - Contact Number
    - Salary (can be used to calculate average_salary)
    - Category
- **EMPLOYEE DEPENDENTS** (Weak Entity)
    - Employee Id
    - First Name
    - Last Name
    - Name (Composite)
- **MENU**
    - Food Id

    - Food Item
    - Price
- **CUSTOMER**
    - Customer Id
    - First Name
    - Last Name
    - Name (Composite)
- **ORDER**
    - Order Id
    - Customer Id
    - Order Date
    - Quantity
    - Food Id (Multi-Value)
    - Status (Served or Preparing)
- **ORDER ITEM** (Weak Entity)
    - Order Id
    - Food Id
    - Quantity
    - Order Date
    - Unit Price
    - Chef Id
    - Status (Item prepared or not)
- **CHEF**
    - Employee Id
    - Order Id
- **PAYMENT**
    - Payment Id
    - Customer Id
    - Order Id
    - Amount
    - Payment Type
    - Payment Date
    - Status (Paid or Pending)
- **TABLE**
    - Table Id
    - Customer Id
    - Order Id
- **WAITER**
    - Table Id
    - Customer Id
    - Order Id
    - Employee Id

2.2 CLASSES

- **EMPLOYEE**
    - CHEF
    - WAITER

2.3 RELATIONSHIPS

- **<IsServing>**: Exists between waiter & customer and waiter & table. (TERNARY).
- **<IsPreparing>**: Exists between chef and order item.
- **<HasPaid>**: Exists between customer and payment.
- <**HasOrdered>**: Exists between customer and order.
- **<PaymentFor>:** Exists between order and payment.

3. FUNCTIONAL REQUIREMENTS

3.1 ADD/DELETE MENU ITEM

- For adding/removing an item to/from the menu.

3.2 CHANGE PRICE

- Change the price of an existing item in the menu.

3.3 ADD ORDER

- When a customer gives his order that order is added to the ORDER table with status ‘Preparing’.
- Add food items corresponding to the food ids given in an order to the ORDER ITEM table with status Preparing.
- Add a new entry to the PAYMENT table with corresponding order id, customer id, status pending.

3.4 MODIFY ORDER ITEM STATUS

- Change the status of on order item to ‘Prepared’ when it is prepared.
- Change the status of an order to ‘Served’ when all corresponding order items are prepared

3.5 ADD CUSTOMER

- Add the details of a new customer i.e first time visit.

3.6 ADD/REMOVE EMPLOYEE AND EMPLOYEE DEPENDENT DETAILS

- Add/Remove an employee and to/from the EMPLOYEE and EMPLOYEE DEPENDENT tables.

3.7 MODIFY PAYMENT

- Change the payment status to ‘Paid’ when the payment is received for the corresponding order.

3.8 MOST ORDERED ITEM

- Get the most ordered food item between two given dates.

3.9 NUMBER OF ORDERS

- Get the number of orders received between two given dates.