DATABASE SPECIFICATIONS

Next-Gen Restaurant Application
Raditya Fahritama
rkf5230@psu.edu

School of Graduate Professional Studies
Information Science Department
IN SC 521 - Introduction to Database Concepts

Fall II, 2021

DOCUMENT CONTROL

Work carried out by:

Name	Email Address	Other
Raditya Fahritama	**rkf5230@psu.edu**	radityafahritama@gmail.com

Revision Sheet

Release No.	Date	Revision Description

DATABASE SPECIFICATIONS

TABLE OF CONTENTS

MILESTONE 1: DATA REQUIREMENTS

System Name or Title

Next-Gen Restaurant Application

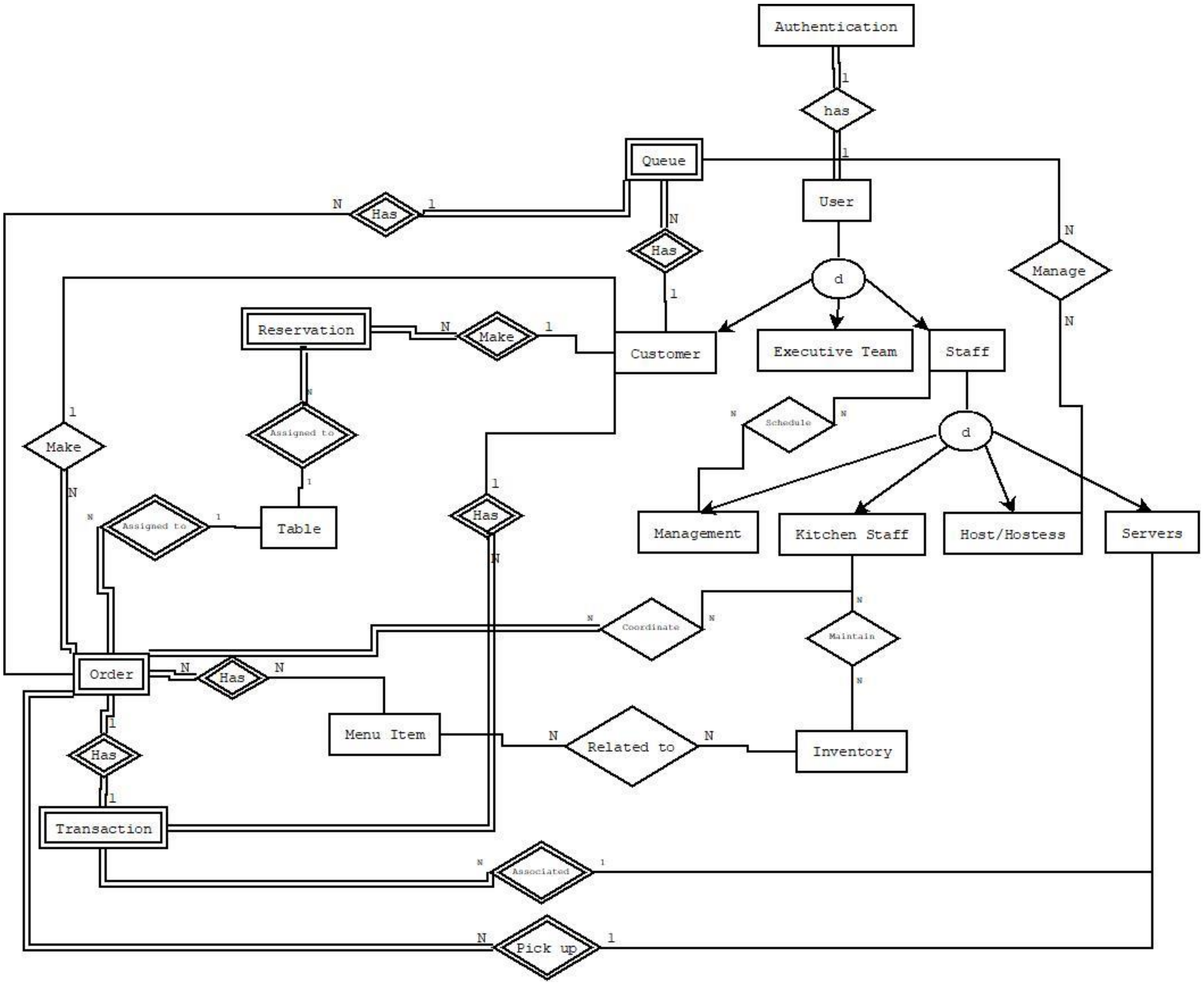
Core requirements

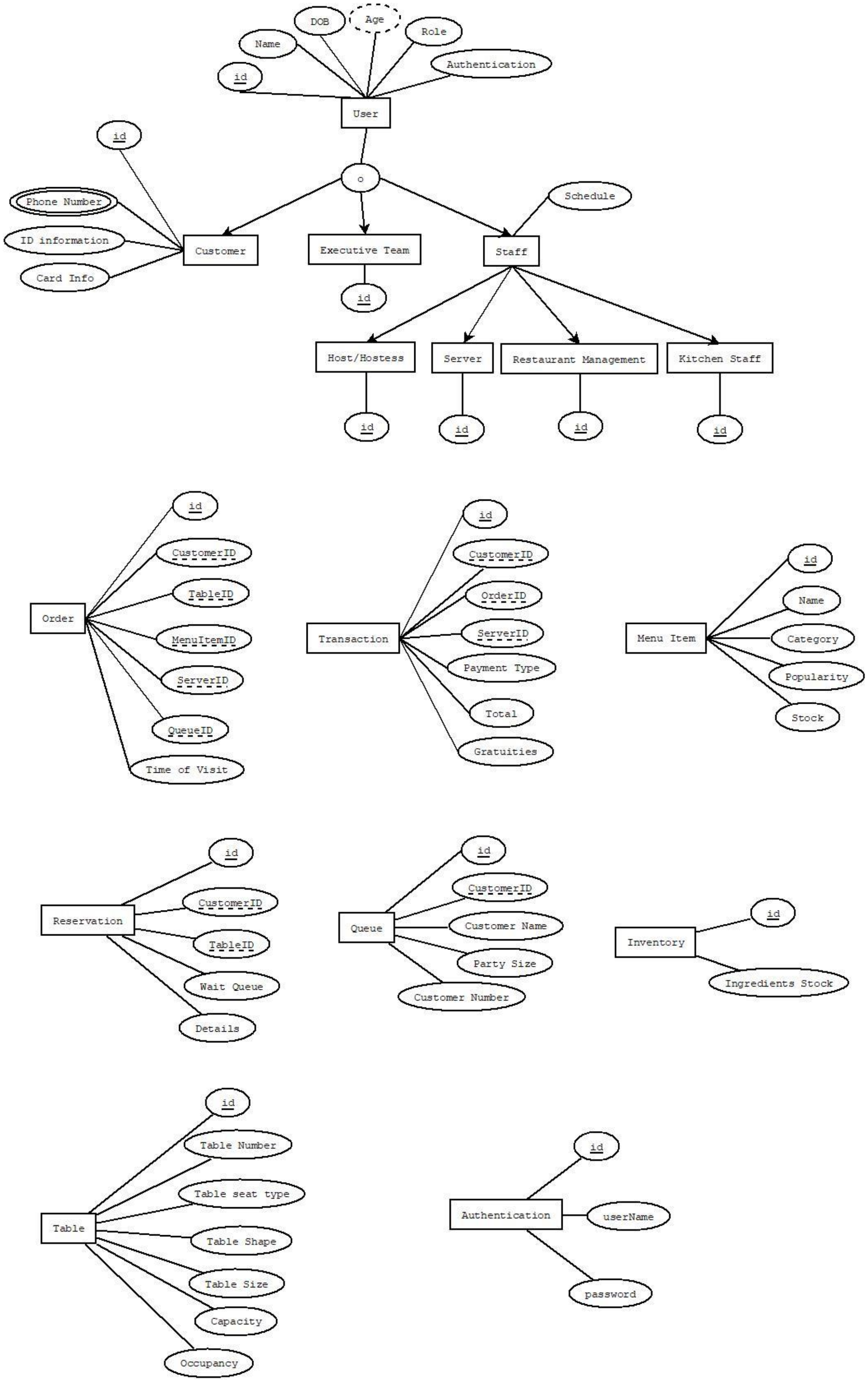
No.	Requirement	Referenced page in SRS	Referenced Section in SRS	Referenced Paragraph in Section
1	The system should store information about the customers	3	1.2	1
2	The system should store information about the order of every customer	3	1.2	2
3	The system should store information about reservation that the customer made	5 12	2.1 3.5.6	3 1
4	The system should store information of the items on the menu.	19 4	8.1.1 1.2	1 2
5	The system should store information about the restaurant's table	39 54	8.1.20 9.5	1 1
6	The system should store information about Restaurant's staffs	5	2.1	3
7	The system should store information of the user of the application.	6	2.3	1
8	The details of the customers: phone number, ID information, credit card info.	5 10 58	2.2 3.5.3 9.9	1 2 1
9	The details of the Order: Customer ID, table assigned ID, Menu Item ID, ServerID, time of customer visit.	5 9	2.2 3.5	1 1
10	The details of the reservation: receipts related to the reservation, reservation details, wait queue, Table Assigned ID, CustomerID	12	3.5.6	1
11	The details of items on the menu: Item name, item Category, item popularity, item stock	19 5	8.1.1 2.1	1 3

12	Item will have label whether they contain alcohol or not	8	3.1	1
13	Order Payment use three forms of payment: cash, credit card, gift card	9	3.5.1	9
14	The detail of restaurant’s table: table number, table seat type, capacity, table shape, table shape size, Occupancy	39 54	8.1.20 9.5	1 1
15	Table seat is divided to bar and casual dining	39	8.1.20	1
16	The detail about staff: staff schedule	5	2.1	3
17	The detail of User: user age, user role, authentication, user DOB, user Name	6	2.3	1
18	The user roles: Executive team, Restaurant Management, Servers, Host/Hostess, Kitchen Staff, Customer	6	2.3	2
19	The Database should store information about kitchen inventory	7	2.3	2
20	The detail of inventory: Ingredients.	7	2.3	2
21	The database should store information about transaction receipts	5	2.2	4
22	The detail of transaction: Order ID, Payment type. Total. Gratuities, ServerID	5 9	2.2 3.5	4 1
23.	The database should store information about Wait queue	10	3.5.3	1
24	The detail of Wait Queue: Customer Name, Party Size, Customer Number	10	3.5.3	2

MILESTONE 2: CONCEPTUAL DESIGN

Diagram



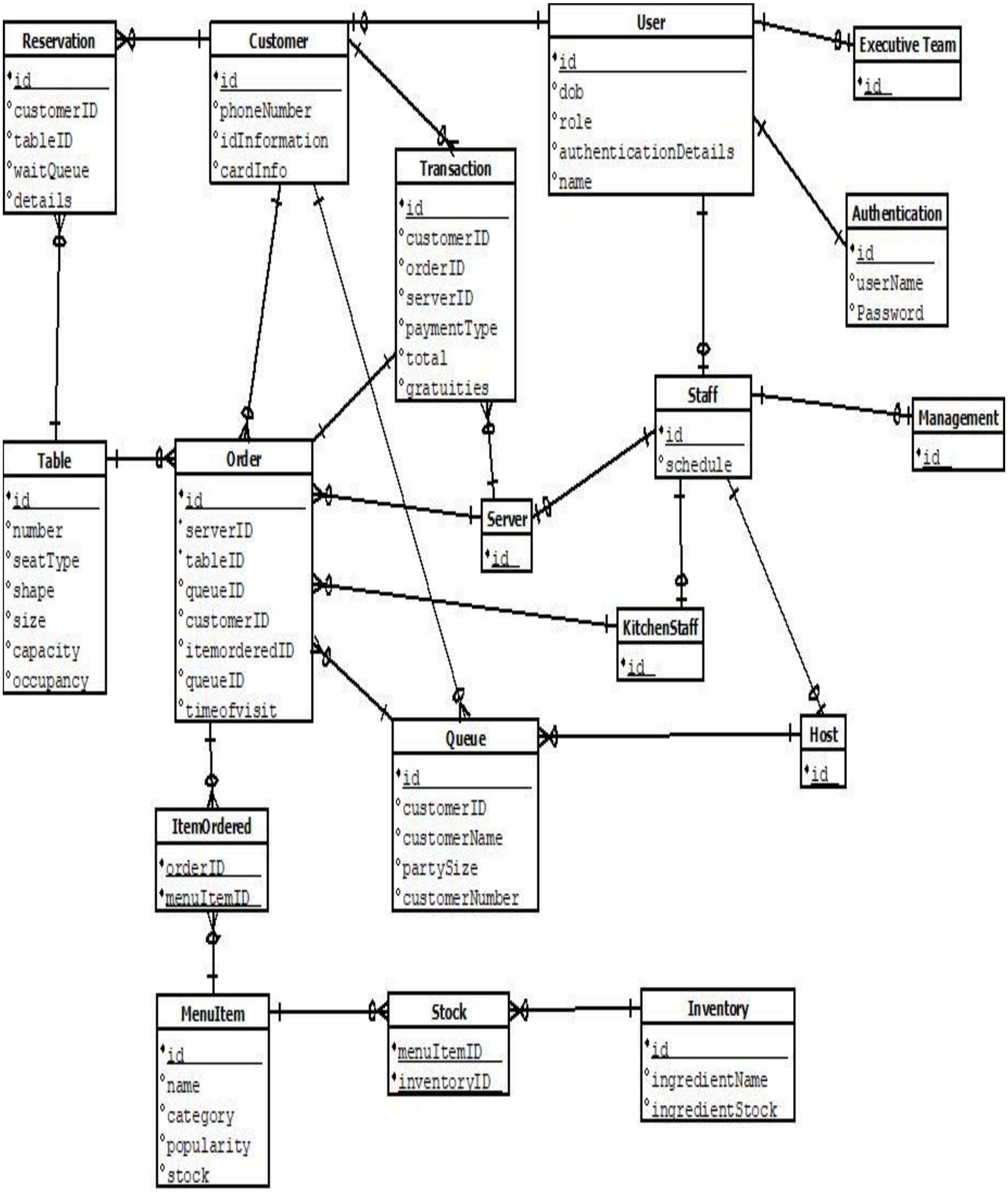

DOB
Age
Role
Name
Authentication
id
User
id
o
Schedule
Phone Number
ID information
Customer
Executive Team
Staff
Card Info
id
Host/Hostess
Server
Restaurant Management
Kitchen Staff
id
id
id
id
id
CustomerID
TableID
Order
MenuItemID
ServerID
QueueID
Time of Visit
id
CustomerID
OrderID
Transaction
ServerID
Payment Type
Total
Gratuities
id
Name
Menu Item
Category
Popularity
Stock
id
Reservation
CustomerID
TableID
Wait Queue
Details
id
CustomerID
Queue
Customer Name
Party Size
Customer Number
id
Inventory
Ingredients Stock
id
Table Number
Table seat type
Table
Table Shape
Table Size
Capacity
Occupancy
id
Authentication
userName
password

Assumptions and Constraints

-Age is derived from DOB.
-Customer’s phone number is multivalued.
-Order contains ID from Customer, Table, Menu Item, Server.
-Transaction contains ID from Order, Server.
-Reservation contains ID from Customer, Table.
-Queue contains ID from Order, Customer

MILESTONE 3: LOGICAL DESIGN

Entity Relationship Diagram



Entity name: User

Attributes:

userID, name, DOB, role, authenticationDetails,

Functional dependencies:

userID -> name, DOB, role, authenticationDetails

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	userID		Name, DOB, role, authenticationDetails

Attribute closures (if any):

userID+ = userID, name, DOB, role, authenticationDetails

(userID) is a super key

Unique keys: the key for this table is/are

userID

Entity name: Table

Attributes:

tableID, number, seatType, shape, size, capacity, occupancy

Functional dependencies:

tableID -> number, seatType, shape, size, capacity, occupancy

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	tableID		number, seatType, shape, size, capacity, occupancy

Attribute closures (if any):

tableID + = number, seatType, shape, size, capacity, occupancy

(tableID) is a super key

Unique keys: the key for this table is/are

tableID

Entity name: Staff

Attributes:

staffID, schedule

Functional dependencies:

staffID -> schedule

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	staffID		schedule

Attribute closures (if any):
staffID+ = staffID, schedule
(staffID) is a super key

Unique keys: the key for this table is/are
staffID

Entity name: Inventory

Attributes:

inventoryID, ingredientsName, ingredientsStock

Functional dependencies:

inventoryID -> ingredientsName, ingredientsStock

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	inventoryID		ingredientsName, ingredientsStock

Attribute closures (if any):
inventoryID+ = inventoryID, ingredientsName, ingredientsStock
(inventoryID) is a super key

Unique keys: the key for this table is/are
inventoryID

Entity name: MenuItem

Attributes:

menuItemID, name, category, popularity, stock

Functional dependencies:

menuItemID -> name, category, popularity, stock

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	menuItemID		name, category, popularity, stock

Attribute closures (if any):
menuItemID+ = menuItemID, name, category, popularity, stock
(menuItemID) is a super key

Unique keys: the key for this table is/are
menuItemID

Entity name: Staff

Attributes:

staffID, schedule

Functional dependencies:

staffID -> schedule

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	staffID		schedule

Attribute closures (if any):
staffID+ = staffID, schedule
(staffID) is a super key

Unique keys: the key for this table is/are
staffID

Entity name: Order

Attributes:

orderID, customerID, tableID, menuItemID, serverID, queueID, timeOfVisit

Functional dependencies:

orderID -> customerID, tableID, menuItemID, serverID, queueID, timeOfVisit

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	orderID		customerID, tableID, menuItemID, serverID, queueID, timeOfVisit

Attribute closures (if any):
orderID+ = orderID, customerID, tableID, menuItemID, serverID, queueID, timeOfVisit
(orderID) is a super key

Unique keys: the key for this table is/are
orderID

Entity name: Queue

Attributes:

queueID, customerID, customerName, partySize, customerNumber

Functional dependencies:

queueID -> customerID, customerName, partySize, customerNumber

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	queueID		customerID, customerName, partySize, customerNumber

Attribute closures (if any):
queueID+ = queueID, customerID, customerName, partySize, customerNumber
(queueID) is a super key

Unique keys: the key for this table is/are
queueID

Entity name: Transaction

Attributes:

transactionID, orderID, serverID, paymentType, total, gratuities

Functional dependencies:

transactionID -> orderID, serverID, paymentType, total, gratuities

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	transactionID		orderID, serverID, paymentType, total, gratuities

Attribute closures (if any):
transactionID+ = transactionID, orderID, serverID, paymentType, total, gratuities
(transactionID) is a super key

Unique keys: the key for this table is/are
transactionID

Entity name: Reservation

Attributes:

reservationID,

Functional dependencies:

reservationID -> customerID, tableID, waitQueue, details

Attributes not in FD	Attributes on the left	Attributes on both sides	Attributes on the right side
	reservationID		customerID, tableID, waitQueue, details

Attribute closures (if any):
reservationID+ = reservationID, customerID, tableID, waitQueue, details
(reservationID) is a super key

Unique keys: the key for this table is/are
reservationID

Assumptions and Constraints

-Order contains ID from Customer, Table, Menu Item, Server.
-Transaction contains ID from Order, Server.
-Reservation contains ID from Customer, Table.
-Queue contains ID from Order, Customer

MILESTONE 4: NORMALIZATION AND

MILESTONE 5: PHYSICAL DESIGN

Assumptions and Constraints

Naming Conventions

Discuss the naming standards and conventions that you have used for table creation.

Tables

Name of the table	***Employee***			
Description	an employee is a person that works for our university. There are three types of employees: Faculty, staff, and contract employees.			
Attribute	**Description**	**Type**	**Examples of values**	**Notes**
…				
Functional Dependencies and Keys				
Functional dependencies				
Candidate keys	**id**			
Normalization				
1NF	**Yes**	Reason…		
2NF	**Yes**	Reason...		
3NF	**Yes**	Reason...		
BCNF	**Yes**	Reason...		
Physical Design				
Primary Key				
Foreign Keys	**-**			
SQL Code				
Count of records in the table	**Note**: Please make sure you add 2 records in each table.			

…

MILESTONE 6: SQL QUERIES AND

Note: Please make sure you add/have 25 records in each table, on average.

Query 1	
English version	
Source for the query need in the SRS document	
SQL sentence	
Example of returned rows (cropped screen caption)	

…