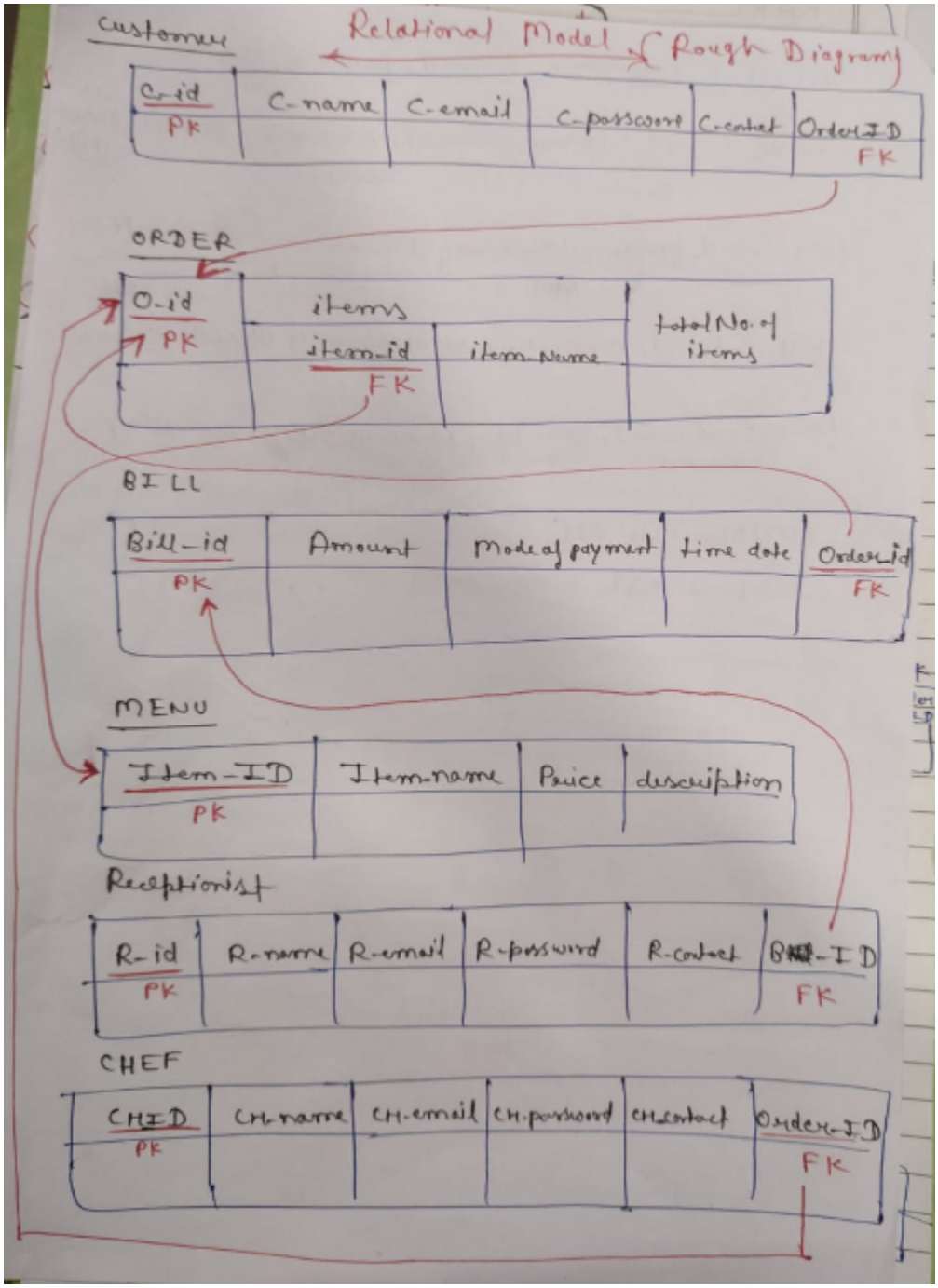
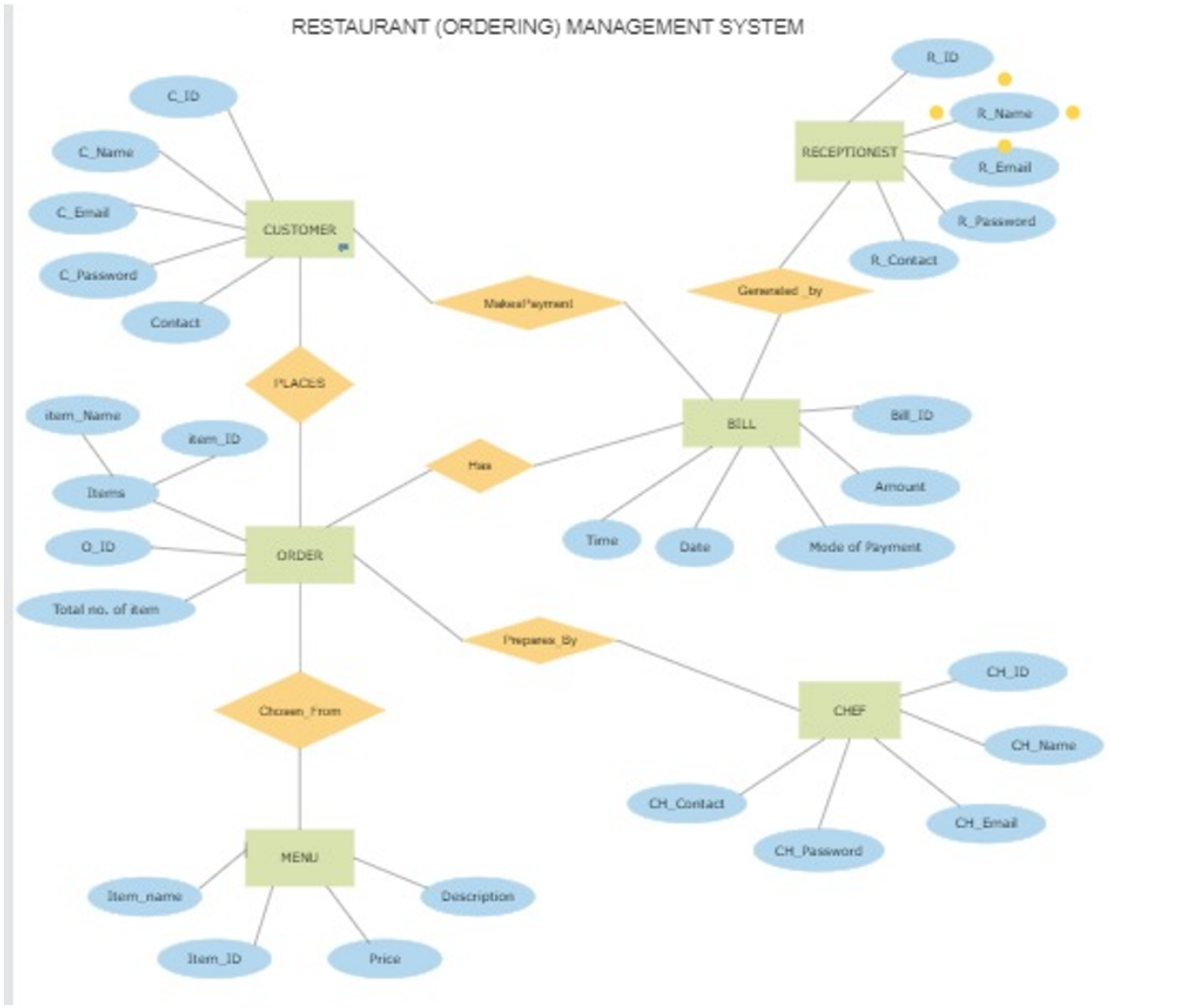
Final report on the Restaurant Management System




RESTAURANT (ORDERING) MANAGEMENT SYSTEM
R_ID
R_Name
R_Email
R_Password
R_Contact
RECEPTIONIST
C_ID
C_Name
C_Email
C_Password
Contact
CUSTOMER
Makes Payment
Generated_by
PLACES
BILL
Bill_ID
Amount
Mode of Payment
Date
Time
item_Name
item_ID
Items
O_ID
Total no. of item
ORDER
Has
Prepares_By
Chosen_From
CH_ID
CH_Name
CH_Email
CH_Password
CH_Contact
CHEF
MENU
Item_name
Item_ID
Price
Description

```
84 •    SELECT * FROM CHEF;
85
```

Result Grid | Filter Rows: | Edit:

CH_ID	CH_NAME	CH_EMAIL	CH_PASSWORD	CH_CONTACT
NULL	NULL	NULL	NULL	NULL

```
86 •    select * from menu;
```

Result Grid | Filter Rows: | Edit:

ITEM_ID	ITEM_NAME	PRICE	RATING
NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

R_ID	R_NAME	R_EMAIL	R_PASSWORD	R_CONTACT	B_ID
1001	NAMAN	NAMAN@gmail.com	NG#1001	978466600	3001
1002	NAMAN	NAMAN@gmail.com	NG#1001	978466600	3001
1003	NAMAN	NAMAN@gmail.com	NG#1001	978466600	3003
1004	AAFIYA	AA@gmail.com	AA#1004	800369696	3004
NULL	NULL	NULL	NULL	NULL	NULL

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

CH_ID	CH_NAME	CH_EMAIL	CH_PASSWORD	CH_CONTACT	ORDERID
5001	RAMNIK	RC@gmail.com	RC@1014	906969805	201
5002	RAMNIK	RC@gmail.com	RC@1014	906969805	202
5003	RAMNIK	RC@gmail.com	RC@1014	906969805	203
5004	AMAN	AC@gmail.com	AC@1014	701729337	204
NULL	NULL	NULL	NULL	NULL	NULL

CREATE DATABASE restaurant;

```
SET FOREIGN_KEY_CHECKS=0;
USE restaurant;

CREATE TABLE CUSTOMER
(C_ID VARCHAR (15) NOT NULL,
C_NAME VARCHAR(15) NOT NULL,
C_EMAIL VARCHAR (15) NOT NULL,
C_PASSWORD CHAR (9) NOT NULL,
C_CONTACT  INT (15) NOT NULL ,
ORDER_ID  VARCHAR (10) NOT NULL,

PRIMARY KEY (C_ID),
FOREIGN KEY (ORDER_ID) REFERENCES ORDERFOOD(O_ID));



CREATE TABLE ORDERFOOD
(O_ID VARCHAR (15) NOT NULL,
ITEM_ID VARCHAR (10) NOT NULL,
ITEMNAME VARCHAR (15) NOT NULL,
TOTAL_ITEMS INT (10) NOT NULL,

PRIMARY KEY (O_ID),
FOREIGN KEY (ITEM_ID) REFERENCES MENU(Item_ID));

CREATE TABLE BILL
(BILL_ID VARCHAR (15) NOT NULL,
AMOUNT DECIMAL (10, 2) NOT NULL,
MODEOF_PAYMENT VARCHAR (15) NOT NULL,
TIMEOF_PAYMENT TIME (6) NOT NULL,
DATEOF_PAYMENT DATE NOT NULL,
Order_ID varchar(15) NOT NULL,

PRIMARY KEY (BILL_ID),
FOREIGN KEY (Order_ID) REFERENCES ORDERFOOD(O_ID));

CREATE TABLE MENU
(ITEM_ID VARCHAR (10) NOT NULL,
ITEM_NAME VARCHAR (15) NOT NULL,
PRICE INT (10) NOT NULL,
RATING VARCHAR (15) NOT NULL,

PRIMARY KEY (ITEM_ID));
```

```
CREATE TABLE RECEPTIONIST
(R_ID VARCHAR (15) NOT NULL,
R_NAME VARCHAR(15) NOT NULL,
R_EMAIL VARCHAR (15) NOT NULL,
R_PASSWORD CHAR (9) NOT NULL,
R_CONTACT  INT (15) NOT NULL ,
B_ID  VARCHAR (10) NOT NULL,

PRIMARY KEY (R_ID),
FOREIGN KEY (B_ID) REFERENCES BILL(BILL_ID));

DROP TABLE RECEPTIONIST;

CREATE TABLE CHEF
(CH_ID VARCHAR (15) NOT NULL,
CH_NAME VARCHAR(15) NOT NULL,
CH_EMAIL VARCHAR (15) NOT NULL,
CH_PASSWORD CHAR (9) NOT NULL,
CH_CONTACT  INT (15) NOT NULL ,
ORDERID  VARCHAR (10) NOT NULL,

PRIMARY KEY (CH_ID),
FOREIGN KEY (ORDERID) REFERENCES ORDERFOOD(O_ID));

DROP TABLE CHEF;

-- INSERT COMMAND



SELECT * FROM CUSTOMER ;
SELECT * FROM ORDERFOOD ;
SELECT * FROM BILL ;
SELECT * FROM MENU ;
SELECT * FROM RECEPTIONIST ;
SELECT * FROM CHEF ;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

ITEM_ID	ITEM_NAME	PRICE	RATING
afghchp14	afghani chaap	250	4.0
alugbhi11	aloo gobhi	100	4.1
bdroll06	breadroll	20	4.0
bttrchk18	butter chicken	400	4.1
bug05	burger	35	3.8
chana12	chana daal	130	3.8
chcopstry5	choco pastry	260	4.9
chkbir15	chicken biryani	300	4.1
chle10	chhole	90	3.9
chocshke24	choco shake	250	4.9
coke20	coke	60	5.0
icecrm23	ice cream	280	5.0
lmnade22	lemonade	90	4.3
mixvg13	mix vegetable	200	4.8
mojito21	mojito	80	4.8
mtrpnr08	matar paneer	90	4.2
mutbir16	mutton biryani	375	3.9
naan02	naan	15	4.6
poori03	poori	12	4.7
rice04	rice	35	4.4
roti01	roti	10	4.7
rsgulla26	rasgulla	30	4.2
shipnr09	shahi paneer	150	4.3
tawachk17	tawa chicken	400	4.0
vegwrp07	vegwrap	35	4.2
water19	mineral water	25	5.0
NULL	NULL	NULL	NULL

```
--                  Inserting data in Receptionist Table

INSERT INTO RECEPTIONIST VALUES('1001','NAMAN','NAMAN@gmail.com','NG#1001','978466600','3001');
INSERT INTO RECEPTIONIST VALUES('1002','NAMAN','NAMAN@gmail.com','NG#1001','978466600','3001');
INSERT INTO RECEPTIONIST VALUES('1003','NAMAN','NAMAN@gmail.com','NG#1001','978466600','3003');
INSERT INTO RECEPTIONIST VALUES('1004','AAFIYA','AA@gmail.com','AA#1004','800369696','3004');
INSERT INTO RECEPTIONIST VALUES('1005','AARTI','AR@gmail.com','AA#1005','800369696','3005');
INSERT INTO RECEPTIONIST VALUES('1006','RAMAN','RA@gmail.com','RA#1006','600369697','3006');
```

```
129 •    select* from chef;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

CH_ID	CH_NAME	CH_EMAIL	CH_PASSWORD	CH_CONTACT	ORDERID
5001	RAMNIK	RC@gmail.com	RC@1014	906969805	201
5003	RAMNIK	RC@gmail.com	RC@1014	906969805	203
5004	AMAN	AC@gmail.com	AC@1014	701729337	204
5005	AMAYRA	AM@gmail.com	AM@1014	701729338	205
5006	RIYA	RI@gmail.com	RI@1014	801729638	206
5007	PRIYA	PR@gmail.com	PR@1014	701729458	207
NULL	NULL	NULL	NULL	NULL	NULL

```
--                  Inserting data in Chef Table

INSERT INTO CHEF VALUES('5001','RAMNIK','RC@gmail.com','RC@1014','906969805','201');
INSERT INTO CHEF VALUES('5002','RAMNIK','RC@gmail.com','RC@1014','906969805','202');
INSERT INTO CHEF VALUES('5003','RAMNIK','RC@gmail.com','RC@1014','906969805','203');
INSERT INTO CHEF VALUES('5004','AMAN','AC@gmail.com','AC@1014','701729337','204');
INSERT INTO CHEF VALUES('5005','AMAYRA','AM@gmail.com','AM@1014','701729338','205');
INSERT INTO CHEF VALUES('5006','RIYA','RI@gmail.com','RI@1014','801729638','206');
INSERT INTO CHEF VALUES('5007','PRIYA','PR@gmail.com','PR@1014','701729458','207');
```

```
131 •    SELECT* FROM RECEPTIONIST;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

R_ID	R_NAME	R_EMAIL	R_PASSWORD	R_CONTACT	B_ID
1001	NAMAN	NAMAN@gmail.com	NG#1001	978466600	3001
1002	NAMAN	NAMAN@gmail.com	NG#1001	978466600	3001
1003	NAMAN	NAMAN@gmail.com	NG#1001	978466600	3003
1004	AAFIYA	AA@gmail.com	AA#1004	800369696	3004
1005	AARTI	AR@gmail.com	AA#1005	800369696	3005
1006	RAMAN	RA@gmail.com	RA#1006	600369697	3006
NULL	NULL	NULL	NULL	NULL	NULL

```
133
134      --      Data manipulation queries  (DML)    --
135
136 •    ALTER TABLE CHEF ADD COLUMN STARS VARCHAR(15);
137
138 •    SELECT * FROM CHEF;
139
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap C

CH_ID	CH_NAME	CH_EMAIL	CH_PASSWORD	CH_CONTACT	ORDERID	STARS
5001	RAMNIK	RC@gmail.com	RC@1014	906969805	201	NULL
5003	RAMNIK	RC@gmail.com	RC@1014	906969805	203	NULL
5004	AMAN	AC@gmail.com	AC@1014	701729337	204	NULL
5005	AMAYRA	AM@gmail.com	AM@1014	701729338	205	NULL
5006	RIYA	RI@gmail.com	RI@1014	801729638	206	NULL
5007	PRIYA	PR@gmail.com	PR@1014	701729458	207	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Sample Report:

Restaurant order and Bill Management System
INTRODUCTION:
Database Management is the core of modern data, as handling and managing data is the key in making exponential progress in today's world. Here, the undersigned students of 2nd year:

1. Afshan Khan (211210006)
2. Arpita (211210015)
3. Kajal Verma (211210030)

Have completed a project on the Database of the Restaurant order and Billing system which has all the information related to orders placed by the customer , customer details, chef and other employees and Bill(amount) details.

Case Study: Database design for Restaurant order and Bill Management.
Aim: Burp is a restaurant order and billing system that aims to help restaurant and customers with Order,Booking and Billing.When a customer visits a restaurant for a meal, they choose what they want to eat from a menu that is provided.s

The case study's goal is to create a database that the restaurant can use to manage and preserve its records of customers, receptionists, chefs, orders,bookings and bills. It includes a unique feature providing the facility to either book a table in a particular time slot (then they will be provided with a particular table number) or to book an order i.e. a customer can book his /her order already and can pick the same in his given time.

Description:

a) In a restaurant there are many sections like receptionist, customer, bill, order food section, chef, and menu.

b) The customer entity has the ID provided by the restaurant, its name, contact information, and other facts. The customer then orders items from the menu with item IDs, item names, and quantities.

c) The employees(Receptionists and chefs) are given their IDs and passwords. The receptionist generates a bill and Each bill has a unique Bill ID associated with that particular order.

d) The chef prepares the order which has a unique Order ID, which is then delivered to the customer when he arrives in his booked time slot. The Receptionist then generates the bill for the customer, which includes the order's specifics and the total amount that must be paid.

Functionalities:

• Database stores the information details of customer like name, ID, contact details and the order they place which includes time and date of order placed, order ID and food items present in their order, which is then analyzed by the restaurant to prepare the order and generate the bill according to the respective items with the important credentials of other staffs including Receptionist and chef.

• Restaurant order and billing Management System is an online application, from which users can easily generate a bill and manage other details of employees.

• Administrator can edit, add, delete and update the records of his/her customer and can keep a record of most ordered dishes and bills.