CSI 2132 Project Deliverable 1 Report

Eric Zhou, 300286231

Matin Mobini, 300283854

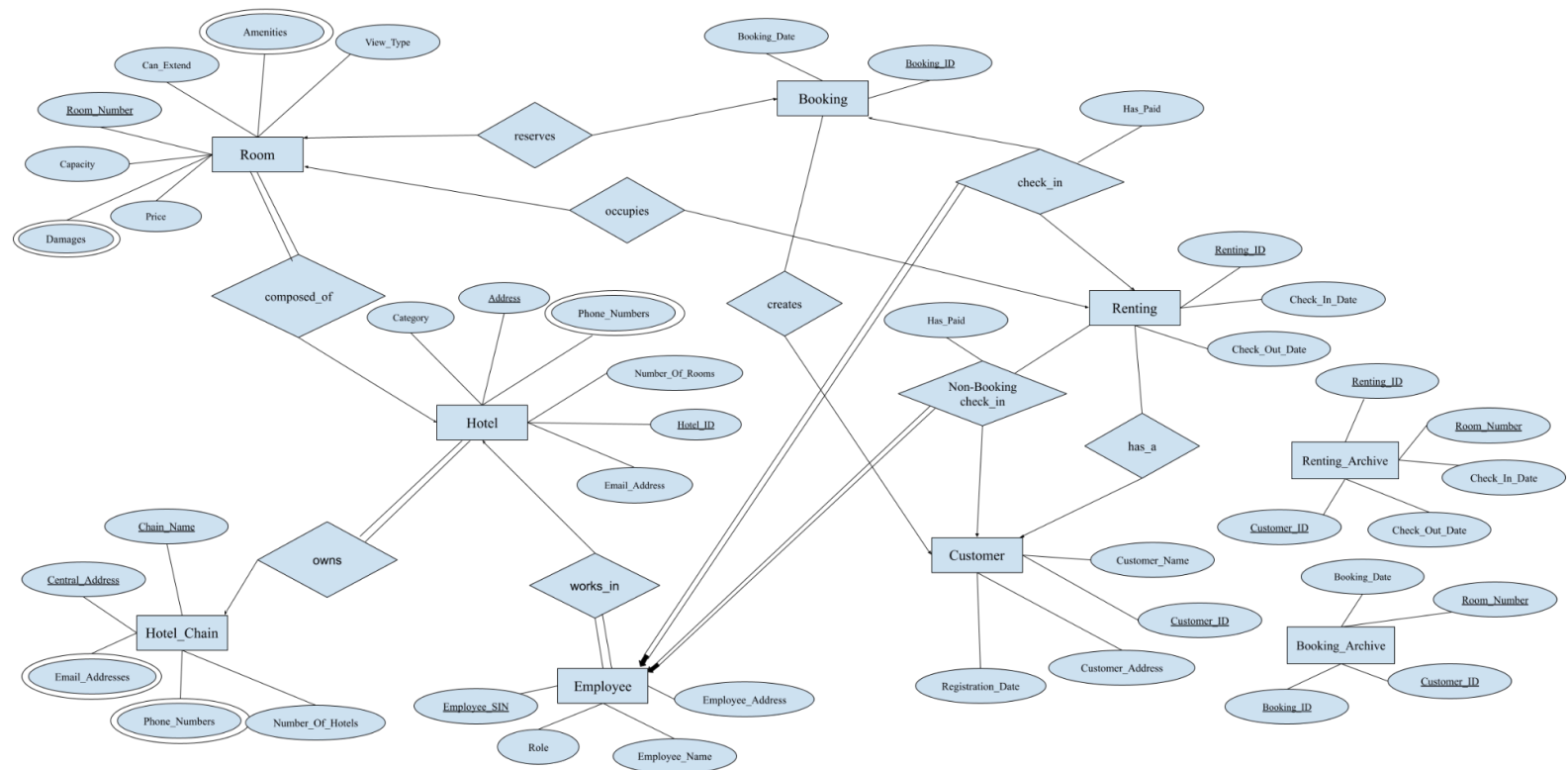Michael Massaad, 300293612

Professor: Verena Kantere
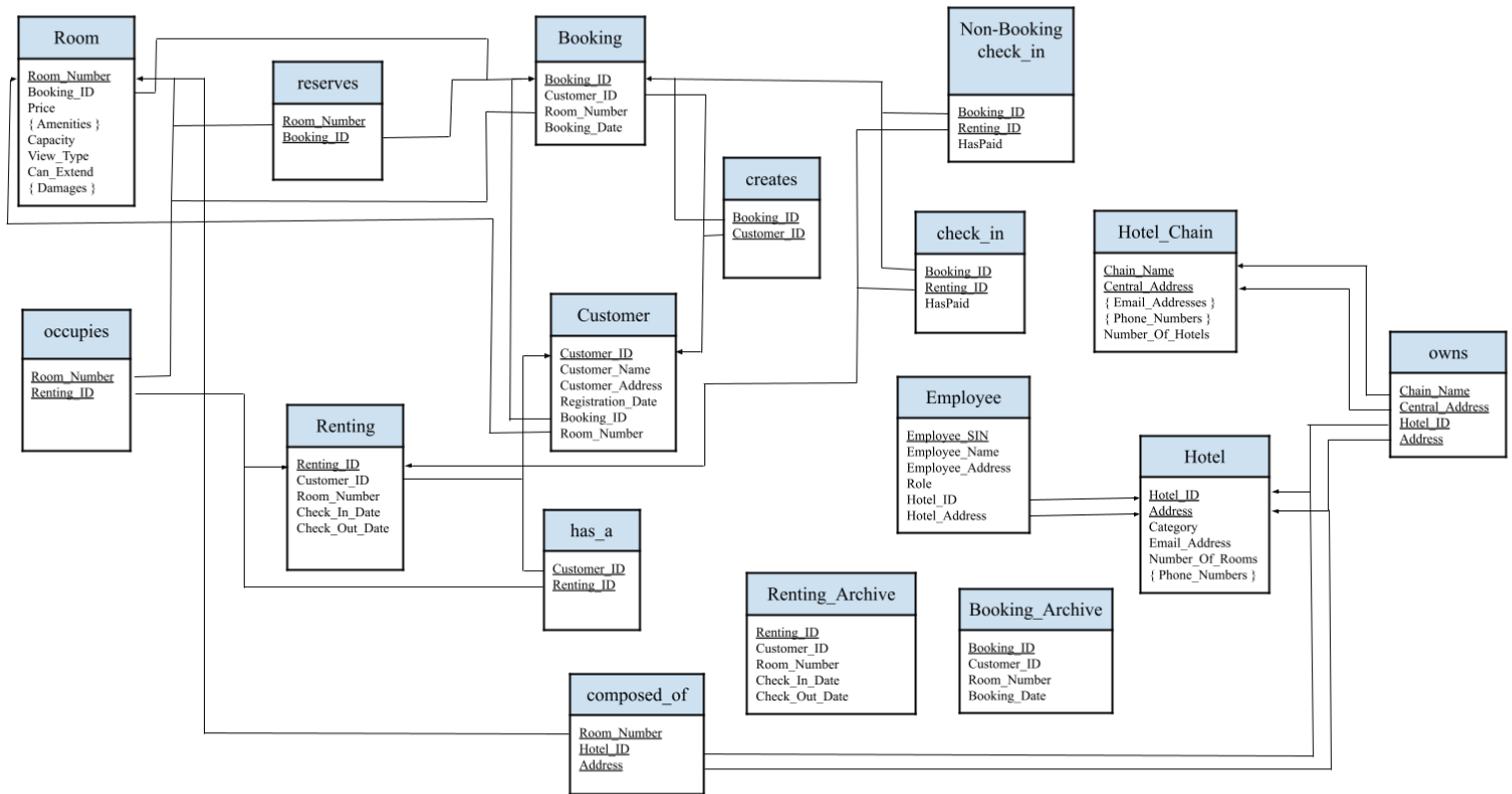**Due Date:** 02/16/2024
Deliverable Number: 1

(1A)The ER diagram:

Amenities

Can_Extend

View_Type

Room_Number

Capacity

Price

Damages

Room

reserves

Booking_Date

Booking_ID

Booking

Has_Paid

check_in

occupies

creates

Renting_ID

Renting

Check_In_Date

Check_Out_Date

composed_of

Category

Address

Phone_Numbers

Number_Of_Rooms

Hotel_ID

Hotel

Email_Address

Has_Paid

Non-Booking
check_in

has_a

Renting_ID

Renting_Archive

Room_Number

Check_In_Date

Check_Out_Date

Chain_Name

Central_Address

Email_Addresses

Hotel_Chain

owns

works_in

Phone_Numbers

Number_Of_Hotels

Employee

Employee_SIN

Role

Employee_Address

Employee_Name

Customer

Customer_Name

Customer_ID

Registration_Date

Customer_Address

Customer_ID

Booking_Date

Booking_Archive

Room_Number

Customer_ID

Booking_ID

(1B)The Relational Database Schema:

Room
Room_Number
Booking_ID
Price
{ Amenities }
Capacity
View_Type
Can_Extend
{ Damages }

reserves
Room_Number
Booking_ID

Booking
Booking_ID
Customer_ID
Room_Number
Booking_Date

Non-Booking check_in
Booking_ID
Renting_ID
HasPaid

creates
Booking_ID
Customer_ID

check_in
Booking_ID
Renting_ID
HasPaid

Hotel_Chain
Chain_Name
Central_Address
{ Email_Addresses }
{ Phone_Numbers }
Number_Of_Hotels

occupies
Room_Number
Renting_ID

Customer
Customer_ID
Customer_Name
Customer_Address
Registration_Date
Booking_ID
Room_Number

owns
Chain_Name
Central_Address
Hotel_ID
Address

Renting
Renting_ID
Customer_ID
Room_Number
Check_In_Date
Check_Out_Date

Employee
Employee_SIN
Employee_Name
Employee_Address
Role
Hotel_ID
Hotel_Address

Hotel
Hotel_ID
Address
Category
Email_Address
Number_Of_Rooms
{ Phone_Numbers }

has_a
Customer_ID
Renting_ID

Renting_Archive
Renting_ID
Customer_ID
Room_Number
Check_In_Date
Check_Out_Date

Booking_Archive
Booking_ID
Customer_ID
Room_Number
Booking_Date

composed_of
Room_Number
Hotel_ID
Address

Entity names (table to help us construct the Required Diagrams):

1) Room	- Room_Number (primary key) - Booking_ID (foreign key) - Price - Amenities - Capacity - View_Type (sea or mountain) - Can_Extend (true or false) - Damages
2) Hotel_Chain	- Chain_Name (primary key) - Central_Address (primary key) - Number_Of_Hotels (integer) - Email_Addresses - Phone_Numbers
3) Hotel	- Hotel_ID (primary key) - Address (primary key) - Chain_Name (foreign key) - Category (1-star to 5-star) - Number_Of_Rooms - Email_Address - Phone_Numbers
4) Customer	- Customer_ID (primary key) - Customer_Name - Customer_Address - Registration_Date - Booking_ID (foreign key) - Room_Number (foreign key)
5) Employee	- Employee_SIN (primary key) - Employee_Name - Employee_Address - Role - Hotel_ID (foreign key) - Hotel_Address (foreign key)
6) Booking (Separate tables for Old and Current Under the Same Entity)	- Booking_ID (primary key) - Customer_ID - Room_Number - Booking_Date
7) Renting (Separate tables for Old and Current Under the Same Entity)	- Renting_ID (primary key) - Customer_ID - Room_Number - Check_In_Date - Check_Out_Date
Check-In **(relational schema)**	- HasPaid - Booking_ID (primary key) - Renting_ID (primary key)

Assumptions (to justify each decision for ER Diagram):

- Assume that to enter the Renting phase from the Booking phase, one must have gone through payment and check-in first (manually through the employee)
- Rooms can only extend one bed
- Each Booking is associated with only one Room
- Each Renting is associated with only one Room
- All Employees must belong to only one Hotel
- Each Hotel must belong to only one Hotel Chain
- Customers can have multiple Renting and Bookings
- A Room can only be rented/booked by one Customer
- Each Employee can only have one role
- Assume that a customer can create a renting through creating an online booking with the creates Relationship or they can create a renting through Non-Booking_Check_in in which no Booking is needed.
- Assume that to check in with or without a booking a customer needs an employee to do so.

(1C)Constraints:

Primary Keys Constraints

List of the primary keys:
> **PK → Entity name**
- Renting_ID → Renting
- Booking_ID → Booking
- Employee_SIN → Employee
- Customer_ID → Customer
- (Hotel_ID, Address) → Hotel
- (Chain_Name, Central_Address) → Hotel_Chain
- Room_Number → Room

Referential Integrity Constraints
List of the Foreign Keys:
> **FK in Entity →PK in Entity**
- Room_Number in Renting → Room_Number in Room
- Room_Number in Booking → Room_Number in Room
- Room_Number in Customer → Room_Number in Room
- Room_Number in reserves → Room_Number in Room
- Room_Number in occupies → Room_Number in Room
- Room_Number in composed_of → Room_Number in Room
- Customer_ID in Renting → Customer_ID in Customer
- Customer_ID in Booking → Customer_ID in Customer
- Customer_ID in has_a → Customer_ID in Customer
- Customer_ID in creates → Customer_ID in Customer
- Customer_ID in Non-Booking_Check_in → Customer_ID in Customer
- Booking_ID in reserves → Booking_ID in Booking
- Booking_ID in Room → Booking_ID in Booking
- Booking_ID in Customer → Booking_ID in Booking
- Booking_ID in creates → Booking_ID in Booking
- Booking_ID in check_in → Booking_ID in Booking
- Renting_ID In occupies → Renting_ID in Renting
- Renting_ID in has_a → Renting_ID in Renting
- Renting_ID in check_in → Renting_ID in Renting
- Renting_ID in Non-Booking_check_in → Renting_ID in Renting
- Chain_Name in Hotel → Chain_Name in Hotel_Chain
- Chain_Name in owns → Chain_Name in Hotel_Chain
- Hotel_ID in composed_of → Hotel_ID in Hotel
- Hotel_ID in owns → Hotel_ID in Hotel
- Hotel_ID in works_in → Hotel_ID in Hotel
- Hotel_ID in Employee → Hotel_ID in Hotel
- Central_Address in owns → Central_Address in Hotel_Chain
- Address in composed_of → Address in Hotel
- Address in owns → Address in Hotel
- Address in works_in → Address in Hotel
- Address in Employee → Address in Hotel
- Employee_ID in works_in → Employee_ID in Employee
- Employee_ID in Check_in → Employee_ID in Employee
- Employee_ID in Non-Booking_Check_in → Employee_ID in Employee

Domain Constraints And Attributes Constraints

Attributes	Entity	Domain Constraints	Attribute Constraints
Room_Number	Room	Integer	$0 <= x <= 999$
Price	Room	Integer	$x >= 0$
Amenities	Room	Varchar	length >= 0
Capacity	Room	Integer	length > 0
View_Type	Room	Varchar	$x \in \{$Sea, Mountain$\}$
Can_Extend	Room	Boolean	True, False
Damages	Room	Varchar	length >= 0
Chain_Name	Hotel_Chain	Varchar	length > 0
Central_Address	Hotel_Chain	Varchar	length > 0
Number_Of_Hotels	Hotel_Chain	Integer	$x >= 0$
Email_Addresses	Hotel_Chain	Varchar	length > 0
Phone_Numbers	Hotel_Chain	Integer	$x >= 0$
Hotel_ID	Hotel	Integer	$x >= 0$
Address	Hotel	Varchar	length > 0
Category	Hotel	Integer	$1 <= x <= 5$
Number_Of_Rooms	Hotel	Integer	$0 <= x <= 999$
Email_Address	Hotel	Varchar	length > 0
Phone_Numbers	Hotel	Integer	$x >= 0$
Registration_Date	Customer	Varchar	length > 0
Customer_Address	Customer	Varchar	length > 0
Customer_Name	Customer	Varchar	length > 0
Customer_Name	Customer	Varchar	length > 0
Customer_ID	Customer	Integer	$x >= 0$
Employee_SIN	Employee	Integer	$x >= 0$
Employee_Name	Employee	Varchar	length > 0

Employee_Address	Employee	Varchar	length > 0
Role	Employee	Varchar	length > 0
Booking_ID	Booking	Integer	x>=0
Booking_Date	Booking	Varchar	length > 0
Renting_ID	Renting	Integer	x>=0
Check_In_Date	Renting	Varchar	length > 0
Check_Out_Date	Renting	Varchar	length > 0
HasPaid	check_in	Boolean	True, False
HasPaid	Non-Booking_check_in	Boolean	True, False

User-defined Constraints

- Rooms can only be extended by one bed; this affects the Capacity variable in the Room Entity, for which we will have to address during the implementation phase through a method.
- Each Booking is associated with only one Room
- Each Renting is associated with only one Room
- All Employees must belong to only one Hotel; this affects the Employee Entity.
- Each Hotel must belong to only one Hotel Chain; This affects the Hotel Entity.
- Customers can have multiple Renting and Bookings
- A Room can only be rented/booked by one Customer
- Each Employee can only have one role; This affects the Role variable in the Employee Entity.