CST3604 FINAL PROJECT 1 – SPRING 2019

Hotel Reservation System – Project Requirements Document

Project Requirement – Design, Development & Implementation Hotel Reservation System

Project Exam Problem Statement

Overview & Objectives

- You are hired as a consultant by **NYC-TECH**, (Represented by Prof. Rodriguez) to design & implement the **Database Tier** for a **Hotel Reservation System** database application named ***EZReservations.com*** for **EZ Hotels Inc**. This application is to automate the managing of reservations guest, hotels and reservations. **EZ Hotels Inc** has hired **NYC-TECH** to develop this application and you job is to do the work, submit the required documentation to Prof. Rodriguez and he will submit these deliverables (fully implemented database tier & required documentation) to the customer **EZ Hotels Inc**.

  - The **Hotel Reservation System** is designed to allow customers, both private and corporate, make reservations.
  - The applications architecture is as follows:

    - **Three-Tiered Web-based Client/Server** – for customers, both consumers and corporate to make hotel reservations online via a browser.
    - **Two-Tiered Windows-Client Client/Server** – hotel employees at the check-in desk and other locations will use this architecture for speed and performance.
    - **Database Tier** – both the Three-tiered Web and Two-tiered Windows client/server **will SHARE the same DATABASE TIER, WHICH IS WHAT YOU WILL CREATE IN THIS PROJECT as Database Analyst, Database Developer & Database Administrator**.

  - The application is to be designed to support dozens of major cities around the world. In addition, provide a great user experience in hotel reservations.

- In this course, this application was divided into several projects (PROJECT 1, PROJECT 2, PROJECT 3 & PROJECT 4) using a Project Management Methodology, thus dividing the project into parts:

  1. In PROJECT 1, you follow the ***Project Management Methodology using the Database Application Development Lifecycle*** (*PLANNING, ANALYSYS, DESIGN, DEVELOPMENT/IMPLMENTATION & OPERATIONS/MAINTENANCE*) ***to create the first version of a full working application*** that includes the following:

  1) ***ER/EER model***, ***Normalized Logical Model***
  2) ***Physical Model (Data Dictionary & Physical Schema Diagram Only)***
  3) ***fully Developed & Implemented Database Tier***.

  2. Please note that as you design & implement the **DBMS** Tier, you are to assume, another group of ***Application*** & ***Web Developers*** will be using a similar methodology to create the following:

  1) ***Front-end Browser/webserver Development for*** **Three-Tiered Web-based Client/Server Application**
  2) ***Custom Windows client executable Development for*** **Two-Tiered Windows-Client Client/Server Application**

Salary & Compensation (Grade) Requirements

- The *client* (Prof. Rodriguez) has the right to **fire you** (***Fail you***) at any time or **reduce the amount of payment** (***Low Grade***) during the development of the system based on the following:

    1) If the system is not design & implemented properly.
    2) Falls behind schedule (submitted late)
    3) You are caught copying other consultants work or giving your work to other consultants! You should be working on your own to implement this system.
    4) **IMPORTANT** – Most important, **NOT** following the program ***REQUIREMENTS***. **Warning! NOT following the requirements can quickly lead to the unemployment line!**

Hotel Reservations System Detailed Business Requirements

Business Requirements

- A Business Analyst was hired by Mr. Rodriguez to compile the list or requirements based on the results of interviews and conversations with the various business stakeholders as part of planning and analysis for the **Hotel Reservation System**.
- Below are the requirements captured by Mr. Rodriguez, which you need to review and analyze to develop the required Conceptual Model Design:

NYC Hotels provides an authentic and contemporary experience for our guests worldwide. With more than 250 locations across six continents. Our hotels are in the US, Canada, Mexico, South America, Caribbean, UK, Japan, Singapore, Australia, and many more locations. Because of our innovative approach to our offerings, amenities & services, we are one of the most recognized names in the industry.

We own a variety of hotels under our brand, such as Hotel Blue, Green Ivey Hotels, The Z Hotels, and many more. Hotels within our brand are of different types such as residential, resorts, casino, convention/conference center, airport hotels, timeshare and suites hotels. Each hotel has a hotel id that uniquely identifies the hotel, hotel name, Manager's name, Manager's contact number and contact email. A hotel can have many locations, but a location can only be for one hotel since one or more hotel brands cannot share the same location.
Our guests or customers can make room reservations at any of our hotel locations. Each of our locations is uniquely identified by a location ID (note that this location id is unique only to the specific hotel brand that owns this location, therefore there may be other of our hotel brands that can possibly have the same location ID), address which is composed of number-street, city, state, zip code, country. Also, telephone number, fax number, email and amenities. Amenities can include any or all of the following: high- quality beds, kitchen facilities, jacuzzi, swimming pool, business center, conference and event facilities, tennis courts, golf course, childcare, gym, restaurants, day spa and entertainment, etc.

NYC Hotels offers its services to primarily two types of guest: corporate guest or consumer/private guest. All our guests are identified by a guest ID, guest's name composed of (first & last), address which is composed of number-street & name, city, state, zip code and country. Other information of guest we need, include date of birth, age, gender, mobile number, email & credit card, which is composed of credit card number, credit card name, merchant name & expiration date. Note that a guest can have many credit cards they can use to pay for a booking a room, in addition, the credit card used by a customer can be co-owned by other individuals such a family member or corporate entity which owns the credit card that may be used by a guest. Additional information of a guest we need is for our corporate guests, which are company name, company id (we store an ID for each company), company primary contact composed of contact name, contact phone number & contact email. When a guest is a consumer, we offer a rewards program called RewardsPlus where they earn points when they stay at any of our hotel facilities. A consumer guest can leverage these points to pay for future reservations, therefore, we need to store their RewardsPlus number and RewardsPlus earned points. In our business, we only have consumer and corporate guests. No other type of guest exists. A guest can only rent a room as a consumer or as a corporation. Each of these booking transactions must be separate guest accounts.

A guest makes a reservation to stay at our hotels. A reservation is a concept that is key to our business. All rentals are based on a reservation, therefore we need to store a unique reservation number to track the reservation and all activities, the date reservation was made, date-in, time-in, date-out, time-out, number of nights and finally number of rooms in reservation.

A reservation is not specific to a room, but to a room category. A room category is a type of room, such as: standard room (double or king beds), suite, studio, penthouse etc. Note that each of these categories have a different price rate. A room category has a room category ID that identifies the category of the room being reserved, and finally Daily category rate.

Some key rules on reserving room(s):

- **Guests make reservations**
- **A reservation is made to a room category not a specific room**
- **A reservation can include multiple room categories for one reservation. For example, a family may want to reserve two rooms under one reservation, a standard king size room for the parent and a second standard double bed for the kids. Therefore, a reservation can include one or more rooms category and a room category can be included in many reservations.**

- **When a reservation includes of one or more room category, we need capture the reservation status code and code description of every room category in the reservation. The status code is the code we assign to the status of the reservation and description is the description of each status code. For example, we use the following code and description to identify a reservation status:**

Reservation Status Code	*Description*
1	**Confirmed**
2	**fulfilled**
3	**Waiting Confirmation**
4	**Cancelled**
5	**Unknown**
Etc..	**Etc..**

- **A reservation is made for a specific hotel location. You cannot have a reservation for more than one hotel. But a hotel location can have many reservations.**
- **A room category can contain many hotel rooms, but a hotel room can only belong to one category.**
- **When a guest rents the room(s) or satisfies the reservation, they check in, stay in hotel (occupies the room/rooms) and check's out. We need to track/store information on the actual stay. This is done as follows:**
  - **When a guest checks in, the guest rents the hotel room based on the reservation. This means a room rental includes the guest, hotel room(s) and the reservation made.**
  - **When the guest rents the room based on a reservation, we need to capture the check-in date, check-in time, check-out date, check-out time.**

Our hotel rooms come in different categories or types: standard, suite, disabled friendly and other. For our standard room, we need to store the room ID, room number, room status code & room status description. The status code is the code we assign to the status or current condition of the room and description describes the status. For example, we use the following code and description to identify a room status:

Room Status Code	*Room Status Description*
1	**Available**
2	**Occupied**
3	**Cleaning**
4	**Inspected**
5	**Out of service**
Etc..	**Etc..**

In addition, for our standard rooms we need to store the bedroom included property which is composed of the bedroom number, number of beds and bed types (e.g. double, king, Full, etc.). Note that room can include more than one bedroom, therefore the bedroom included property must reflect this. Finally, a classification if the room is a smoking or non-smoking room.

For our suites, we need to include the room ID, room number, room status code & status description which are the same code and description as the standard room:

Room Status Code	*Room Status Description*
1	**Available**
2	**Occupied**
3	**Cleaning**
4	**Inspected**
5	**Out of service**
Etc..	**Etc..**

A suite also has bedroom included property composed of the bedroom number, number of beds and bed types (e.g. double, king, Full, etc.). Note that bedroom included property can include more than one bedroom. Also, a classification if the room is a smoking or non-smoking room, and finally suites come with a kitchen, thus we need to store the kitchen style (farmhouse, rustic, modern, traditional, contemporary etc.)

Note that a suite comes in three types, 1) Standard suite, 2) Executive Suite design for business executives and to engage in private meetings, events etc., and 3) Penthouse Suite for our most luxurious apartment style room, with open view and a garden roof top. The standard suite has all the properties of a suite with just a kitchen style, the executive suite has all the

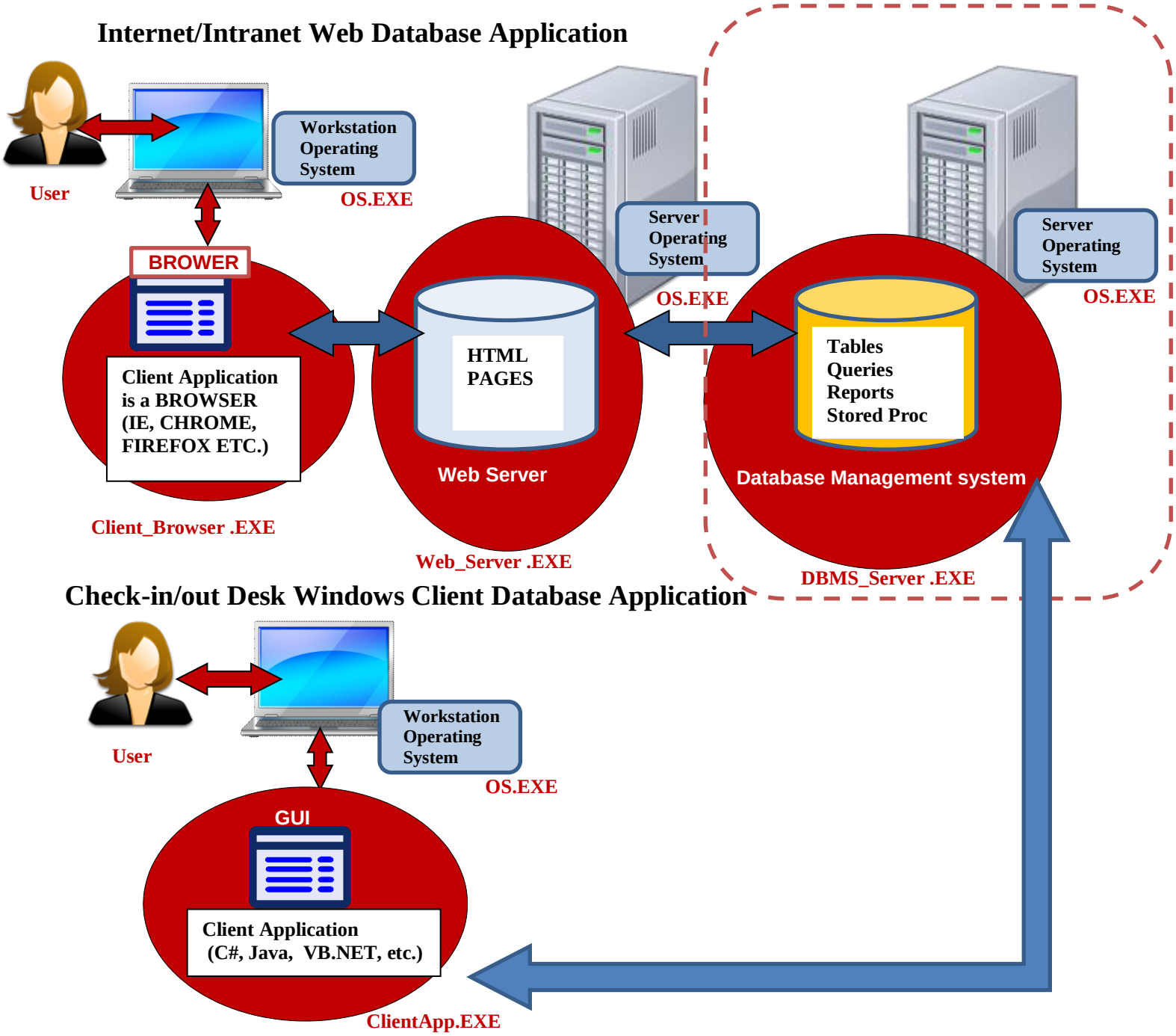
properties of a suite, but also includes a conference room type (e.g., boardroom, classroom, u-shape, etc.). Our penthouse suite also has all the properties of a suite, with the addition to an indication if penthouse has elevator or not, finally, if the penthouse has a personal gymnasium or not.

The final type of room we have is our disabled friendly room, which have a room ID, room number, room status code & status description with the same room code and description as the other types. In addition, for our disabled friendly rooms, we need to store the bedroom included property composed of the bedroom number, number of beds and bed types (e.g. double, king, Full, etc.), indication if the room is a smoking or non-smoking room and finally indication if the room has a wheeled shower chair.

Hotel Reservations System Detailed Technical Requirements

Technical Objectives

- The system is a combination of 1) **Three-Tiered Web Database Application** with a web-based browser front-end for the users that want to make reservations via the internet or hotel employees that require a browser-based application. In addition, 2) **Two-Tiered Windows Database Application**, with a custom client front-end to for the application used by check-in/check-out desks and other users that require fast performance. Note that both these architecture (Web & Custom client) share the same DBMS back-end data store. Below is the full architecture we are targeting:

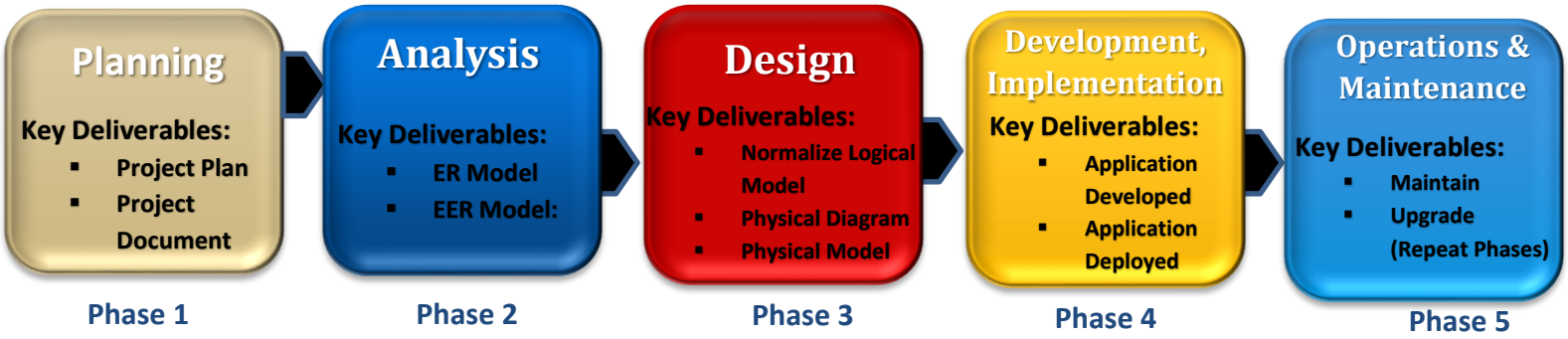


- This project will focus only on the **DBMS or backend design and implementation** using a DBMS, NOT on the front-end or client application hosted in the Browser/Web Server or Windows Client (another group will work on this in parallel).

Hotel Reservations System Project Management Requirements

Project Management & Implementation Objectives

- We will continue to implement our project management ***database lifecycle methodology*** for this final project by ***COMPLETING*** the **Design Phase**, and ***INPLEMENTING*** the **Development/Implementation Phase & Operations Phase** for the **Hotel Reservations System**.
- Below is a listing of all the 5 PHASES & their main deliverables:



Database Application Development Lifecycle Methodology Details

- As stated, the goal of this project is to finalize the **Design Phase** and execute the last two phased of our methodology, **Development/Implementation & Operations**.
- Below tables lists the details of each of these phases and their expected deliverables.

PLANNING PHASE – The first phases is the planning phase. Short description, activities & deliverables are as follows:

Phases	Description
Phase 1: Planning & Discovery 	▪ ***Purpose:*** - *Develop the plan, understand the business requirements and discover existing information systems, etc.* - *Interview, discover, etc.* ▪ ***Database Professional Role:*** - *Database/Systems Analyst* - *Business Analyst* ▪ ***Deliverables:*** - ***Primary Deliverables:*** *1) Project Plan – using project plan tool such as Microsoft Project, etc.* *2) **Project Document** – which includes the methodology or project life-cycle & Documentation* ▪ ***Timeline/duration:*** - ***Week 1***

Main deliverables in this phase for this PROJECT!

❑ For this phase, you are responsible for the following:

- **PLANNING PHASE**

1. **Primary Database Deliverable – Project Document**

   - You are responsible **to create & submit the PROJECT DOCUMENT with your submission**. This document will be the ***Project Planning, Design and Implementation Document*** in this project

2. **Timeline/duration** – Must be done within the first week of the project:

   - ***Done in Week 1 (See Requirement #1 in this document for details)***

ANALYSIS PHASE – Short description, activities & deliverables are as follows:

Phases	Description
Phase 2: Analysis 	▪ ***Purpose:*** 1. *Analyze/derive detailed user requirements for data & develop data model to represent requirements.* 2. *Identify the Business Rules, use correct Naming convention guidelines etc.* 3. *Create a detailed **ER Conceptual Data Model** **(Entity-Relational Model - ER Diagram)** based on requirements.* 4. *Create a detailed **EER Conceptual Data Model** **(Enhanced Entity-Relational Model - EER Diagram)** based on requirements.* ▪ ***Database Professional Role:*** - *Database/Systems Analyst* ▪ ***Deliverables:*** 1. ***Primary Database Deliverable – Detailed Conceptual Data Model:*** ○ *E-R Diagram* ○ *Enhanced E-R Diagram* 2. ***Updated Project Document*** **Main deliverables in this phase for this PROJECT!** ▪ ***Timeline/duration:*** - ***Done in Week 1 (See Requirement #1 in this document for details)***

- For this phase, you are responsible for the following:
  - **ANALYSIS PHASE**

1. **Primary Database Deliverable – Detailed Conceptual Data Model (EER Model)**
   - **The *Conceptual EER Model* will be provided to you**.
   - The assumption is you've already implemented this model.
   - For those students who still lack the skills to develop ***Conceptual ER & EER Models***, I recommend you go through this exercise yourself. I will provide reference in this document within each section to location of examples and theory.

2. **Updated Project Document**
   - Note that although the ***Conceptual ER & EER Model*** is provided to you, it must be included in your final submission which is the overall ***Project Planning, Design and Implementation Document***. Therefore, you need to update the ***Project Planning, Design and Implementation Document***
   - Instructions & details on what to submit will be provided in later sections of this document.

3. **Timeline/duration** – Must be done within the first week of the project:

- ***Done in Week 1 (See Requirement #2 & #3 in this document for details)***

DESIGN PHASE – Short description, activities & deliverables are as follows:

- ***Purpose:***
  1. *Develop a detailed design of database Information System based on all specifications and requirements.*
  2. *Create a **Logical Data Model (Logical Schema)***
  3. *Normalize the **Logical Data Model***
  4. *Create a **Physical Data Model (Physical Schema)** **– composed of three parts:***
     1) **Data Dictionary**
     2) **Physical Model Schema Diagram**
     3) **Technical Specifications for performance, efficiency, data integrity, security, disaster recover, backup etc. (CST3604 topic)**
- ***Database Professional Roles that own phase:***
  - ***Database Analyst** – Logical Data Model*
  - ***Database Administrator & Analyst** – Physical Data Model*
  - ***Database Developer or other roles involved***
- ***Activities include:***
  1. *Create detailed **Logical Data Model (Logical Schema)** as follows:*
     - ***TRANSFORM** – **Convert ER/EER Diagram** to **Logical Model Diagram***
     - ***NORMALIZATION** – Process of breaking down tables with abnormalities to produce smaller, well-structure tables to reduce redundancy and inconsistencies.*
  2. *Create detailed **Physical Data Model (Physical Schema)** as follows:*
     - ***DATA DICTIONARY** – Create a Dictionary Table(s) mapping of all Tables, Attributes, Data Types, properties etc. The idea is to capture all METADATA in one place.*
     - ***PHYSICAL SCHEMA DIAGRAM** – Convert Normalized Logical Model to the Physical Schema Diagram based on **DBMS** to be used.*
     - ***PHYSICAL TECHNICAL SPECIFICATIONS** – Technical specifications for performance, storage & hardware requirements, security, backup & disaster recovery, compliance, etc.*
- ***Project Management Activities:***
  - *PM Activities – Create Design Document, Update Plan etc.*
- ***Deliverables:***
  1. ***Primary Database Deliverable – Normalized Logical Data Model (Logical Schema)***
  2. ***Primary Database Deliverable – Data Dictionary***
  3. ***Primary Database Deliverable – Physical Model Schema Diagram***
  4. ***Primary Database Deliverable – Physical Model Technical Specifications***
  5. *Other project related documents such as UPDATE of Project Plan & document, etc.*
- ***Timeline/duration:***
  - ***Done in Week 1 – Normalized Logical Model (See Requirement #4 in this document for details)***
  - ***Week 1 & Week 2 – Data Dictionary & Physical Model Schema Diagram Model (See Requirement #5 & #6 in this document for details)***

Main deliverables in this phase for this PROJECT!

- The design phase is composed of four deliverables: 1) ***Normalized Logical Model*** & 2) ***Physical Data Model composed of Data Dictionary, Physical Model Diagram or Schema & Physical Model Technical Specifications***.
- For this phase, you are responsible for the following:

  - **DESIGN PHASE – NORMALIZED LOGICAL MODEL**:

    - **Normalized Logical Model** – **This will also be provided to you as well. Again, the assumption is you've already implemented this model**.
    - For those students who still lack the skills to develop ***Logical Model***, I recommend you go through this exercise yourself. Again, I will provide reference in this document within each section to location of examples and theory.
    - Note that although this model is provided to you, it must be included in your final submission which is the overall ***Project Planning, Design and Implementation Document***.

  - **DESIGN PHASE – PHYSICAL DATA MODEL** – The *Physical Model* is divided into three parts. Below are the parts and what you are responsible for:

    1. **Physical Model Schema Diagram** – **YOU ARE RESPONSIBLE TO IMPLEMENT THIS DELIVERABLE**:

       a) **Dictionary Table** – Create a **Data Dictionary table** that contains information on all the relational tables, their attributes, selected datatypes, properties, etc., targeting ***Oracle Express 11g DBMS***. Use MS Word or MS Excel to capture the information. The Dictionary Table must be included in your final submission or ***Project Planning, Design and Implementation Document***.

       b) **Physical Model Schema Diagram** – Using the **Data Dictionary table** & the **Normalized Logical Model**, CREATE the **Physical Model Schema Diagram** using a drawing tool targeting ***Oracle Express 11g DBMS***. The Physical Model Schema Diagram must be included in your final submission or ***Project Planning, Design and Implementation Document***

    2. **Physical Model Technical Specifications** – Technical Specifications for performance, efficiency, data integrity, security, disaster recover, backup etc. **YOU ARE NOT RESPONSIBLE TO IMPLEMENT THIS PHYSICAL MODEL ACTIVITY. THIS WILL BE IMPLEMENTED IN FUTURE PROJECT EXAMS DURING THIS CST3604 COURSE**.

❖ **IMPORTANT!!!!**

- **Follow the guidelines and examples from the Lecture 5A in the CST3504 course to create your DICTIONARY TABLE & PHYSICAL SCHEMA DIAGRAM**
- **Everything you need to succeed in this phase of the project are in this lecture notes.**

- ***Timeline/duration:***

- ***Done in Week 1 – Normalized Logical Model (See Requirement #4 in this document for details)***
- ***Week 1 & Week 2 – Data Dictionary & Physical Model Schema Diagram Model (See Requirement #5 & #6 in this document for details)***

DEVELOPMENT & IMPLEMENTATION PHASE – Short description, activities & deliverables are as follows:

Phases	Description
Phase 4: ***Development and Implementation*** 	▪ ***Purpose:*** - *Develop & physically implement the design created in design phase.* - *Develop & implement* ***Physical Model*** ***1) Physical Schema Diagram*** *2)* ***Technical Specifications for performance, efficiency, data integrity, security, disaster recover, backup etc. (CST3604 topic)*** ▪ ***Database Professional Role:*** - *Database Developer* - *Database Administrator* ▪ ***Activities include:*** 1. *Install & configure Database Management System (DBMS)* 2. *Write, test & execute all program/scripts that access, created or modify the database (use SQL, Programming Language or special database processing language)* 3. *Write, test & execute all program/scripts for required reports, User-views or displays, graphs etc. (use SQL, Programming Language or special database processing language)* 4. *Unit & Integration Testing* 5. *Load/Import data into DBMS* 6. *Finalize documentation* ▪ ***Deliverables:*** 1. ***Fully Tested & Implemented Database Information System Based on Requirements*** 2. *Final Project Document & diagrams completed* ▪ ***Timeline/duration:*** - ***Weeks 2, 3 & 4 (See Requirement #7, #8 & #9 in this document for details)***

Main deliverables in this phase for this PROJECT!

- For this phase, you are responsible for the following:
  - **DEVELOPMENT & IMPLEMENTATION PHASE** – You are responsible to implement this phase as follows:
    - **Develop the Physical Model Schema in Oracle Express 11 g** – Using **Oracle SQL Developer** implement the Physical Schema Diagram in ***Oracle Express 11g***.
    - Included snap-shots images of your implementation and schema script file in your ***Project Planning, Design and Implementation Document***.

❖ **IMPORTANT!!!!**

- **Follow the guidelines and examples from the Lecture 6A, 6B & 6C in the CST3504 course to complete this phase**
- **Everything you need to succeed in this phase of the project are in these lecture notes.**

OPERATIONS & MAINTENANCE PHASE – Short description, activities & deliverables are as follows:

Phases	Description
Phase 4: Operations & Maintenance  **Main deliverables in this phase for this PROJECT!**	▪ ***Purpose:*** - *Maintain, operate & backup the Information System* - *Repeat entire Methodology as changes are required* ▪ ***Database Professional Role:*** - *Database Analyst – Re-design for new business requirements & errors in database design* - *Database Developer – Re-design & implement for new business requirements & errors in database design* - *Database Administrator – Maintain, operate, backup & improve performance* ▪ ***Deliverables:*** 1. ***Keep the lights on*** 2. ***Upgrade as needed (Repeat Methodology)*** 3. *Updated Operations & Maintenance Document* ▪ ***Timeline/duration:*** - ***Weeks 2, 3 & 4 (See Requirement #7, #8 & #9 in this document for details)***

- For this phase, you are responsible for the following:
  - **OPERATIONS & MAINTENANCE PHASE**:
    - **Maintenance section – Test & Verify the DBMS implementation or Database is working as requirements** – Using **Oracle SQL Developer** you will test and validate that this database design is working as per requirements. Included snap-shots images of your verification in your ***Project Planning, Design and Implementation Document***.
    - **Upgrade section** of the methodology will be executed during the CST3604 course, via new features and updates.

TOTAL PROJECT TIMELINES – 4 Weeks

Hotel Reservation System – Project Datailed Requirements

Requirement #1 (4 POINTS) – (Phase 1 – Planning) – Create Design/Implementation Document

- Create a Microsoft Word document with the following requirements:

❖ **Requirement #1 – Create MS Word Design/Implementation Project Document:**

1. Create a Microsoft Word document to store all your plan, analysis & design phases information of **this PROJECT EXAM! THIS WILL BE YOUR PROJECT DOCUMENT THAT YOU WILL SUBMIT TO YOUR CLIENT (PROF RODRIGUEZ)**
2. This document is to contains the ***goals and requirements*** for the ***EZReservations.com*** **Hotel Reservations System** with explanation of what you are going to do, how you are going to do it, etc.
3. **THIS DOCUMENT IS INTENDED TO BE SUBMITTED TO THE CUSTOMER EZ Hotels Inc therefore MUST be created professionally & targeted to the customer, NOT NYC-TECH or anyone else.**
4. **CREATE A SECTIONS IN THE DOCUMENT FOR EACH OF THE REQUIREMENTS LISTED IN THIS REQUIREMENTS DOCUMENT - DO NOT INCLUDE IN YOUR PROJECT DOCUMENT THE INSTRUCTIONS OR REQUIREMENTS STATEMENTS FROM THIS DOCUMENT, such as Requiremet Number or INSTRUCTIONS ON WHAT TO DO. YOUR PROJECT DOCUMENT IS A DELIVERABLE FOR THE CUSTOMER THAT CONTAINS THE PLAN, ANALYSIS, DESIGN & IMPLEMENTATION OF THE SOLUTION, NOT INSTRUCTIONS FROM THIS REQUIREMENTS DOCUMENT! Therefore this requirement #1 and any other instructions statement from this document SHOULD NOT BE IN YOUR PROJECT DOCUMENT!**
5. Below are the sections **you will need to CREATE HEADERS** this project document:

    1) **Executive Summary**
    2) **Problem objectives**
    3) **Targeted Technical Architecture**
    4) **Targeted Methodology & plan (including this Planning phase or steps above)**
    5) **A section for Planning phase Business requirements – *requirement #2* of this document**
    6) **A section for Analysis phase ER/EER Conceptual Model – *requirement #3* of this document**
    7) **A section for Design Phase Normalized Logical Model – *requirement #4* of this document**
    8) **A section for Design Phase Data Dictionary – *requirement #5* of this document**
    9) **A section for Design Phase Physical Model Schema Diagram – *requirement #6* of this document**
    10) **A section for Development & Implementation Phase, implementation of the database tier design via scripts – *requirement #7* of this document**
    11) **A section for Development & Implementation Phase, generation of the Physical Schema Diagram via Oracle SQL Developer – *requirement #8* of this document**
    12) **A section for Development & Implementation Phase, testing of your implementation of the database tier by executing some queries on a selected group of tables – *requirement #9* of this document**
    13) **Finally, a conclusion section**

6. **Note that every section of the document should be clearly labeled and professionally created**. Don't just paste information without explanation of each section. Have an introduction section explaining the objectives or requirements of the project etc.
7. For each section, you are going to describe, keep your explanations short. I am not asking for an essay or report, but a well-documented information, that is easy to read and makes sense to the reader.
8. Your goal is to make it easy for the reader to understand from a high-level what you have done.
9. **The information from the next sets of requirements in this document are to be entered in this document as per instructions in step 5.**
10. **THIS DOCUMENT IS YOUR DELIVERABLE and you will be PAID/GRADED ON HOW THIS DOCUMENT IS FORMATTED, AND THE ABILITY OF THE READER TO EASILY UNDERSTAND WHAT YOU ARE DOING AND HOW.**

Requirement #2 (2 POINTS) – (Phase 1 & 2 – Planning & Analysis) – List the Business Requirements from PROJECT 1 in the Implementation Document

- In this requirement #2, you need to:

❖ Requirement #2a – COPY/PASTE the BUSINESS REQUIREMENTS from PROJECT 1 to the Design/Implementation Project Document of Requirement #1

1. Go to the business requirements you listed in **PROJECT 1** & **COPY/PASTE** to the **Design/Implementation Project Document**, of **requirement #1**.
2. Now your **Design/Implementation Project Document** will contain the business requirements.

❖ Requirement #2(b) – REQUIREMENT FORMAT & PRESENTATION

1. As stated in **requirement #1**, remember to ***CREATE A HEADER FOR THIS REQUIREMENT*** in your **Design/Implementation Project Document**
2. As stated in **requirement #1**, remember to ***CREATE A PARAGRAPH DESCRIBING THIS SECTION AND WHAT YOU ARE DELIVERING***. **Only a SHORT PARAGRAPH IS REQUIRED**.
3. **REMEMBER THAT YOU ARE BEING GRADED ON THE FORMAT AND QUALITY OF YOUR DOCUMENT AND THE ABILITY OF THE READER (PROJECT SPONSOR/PROFESSOR) TO UNDERSTAND WHAT YOU ARE DOING AND HOW**.

Requirement #3 (2 POINTS) – (Phase 2 – Analysis, EER Model DIAGRAM) – List the FINAL resultant ER/ER Conceptual Database Model from PROJECT 1 based on the Requirements of Project 1

EER Conceptual Model

- **DIAGRAM #1 –** In PROJECT 1, you implemented the EER Model for the **HOTEL RESERVATION SYSTEM**, this diagram needs to be included in your **Design/Implementation Project Document** as stated below.

❖ **Requirement #3a – (DIAGRAM #1) COPY/PASTE THE FINAL EER MODEL DIAGRAM from PROJECT 1 OR USE THE ONE PROVIDED BELOW to your Design/Implementation document of Requirement #1**

- **NOTE THAT YOU CAN USE THE DIAGREM I AM PROVIDING BELOW FOR THIS FINAL PROJECT, OR YOU CAN USE YOUR EER MODEL OF PROJECT 1, IF YOU FIX IT OR YOU WERE CORRECT IN YOUR DESIGN**

1. In next two pages, my rendition of the **EER Conceptual Model** is being listed for you.
2. **COPY/PASTE**, the **EER CONCEPTUAL MODEL** listed below or your version from PROJECT 1, into your **Design/Implementation Project Document** of **requirement #1**.

❖ **Requirement #3b – REQUIREMENT FORMAT & PRESENTATION**

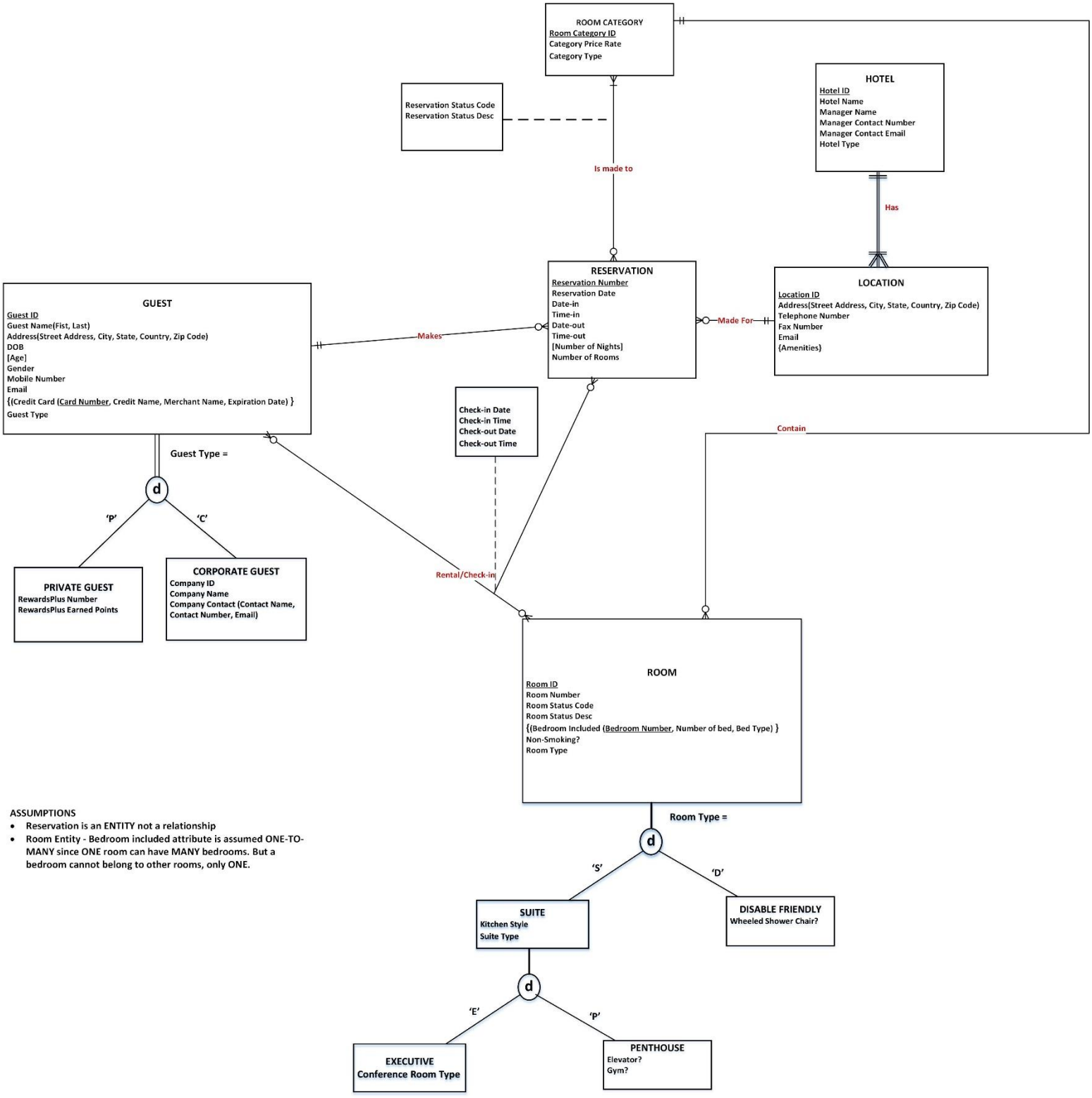
1. As stated in **requirement #1**, remember to ***CREATE A HEADER FOR THIS REQUIREMENT*** in your **Design/Implementation Project Document**
2. As stated in **requirement #1**, remember to ***CREATE A PARAGRAPH DESCRIBING THIS SECTION AND WHAT YOU ARE DELIVERING***. **Only a SHORT PARAGRAPH IS REQUIRED**.

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING!
HERE IS WHERE:
Theory & Examples on this TOPIC can be found:

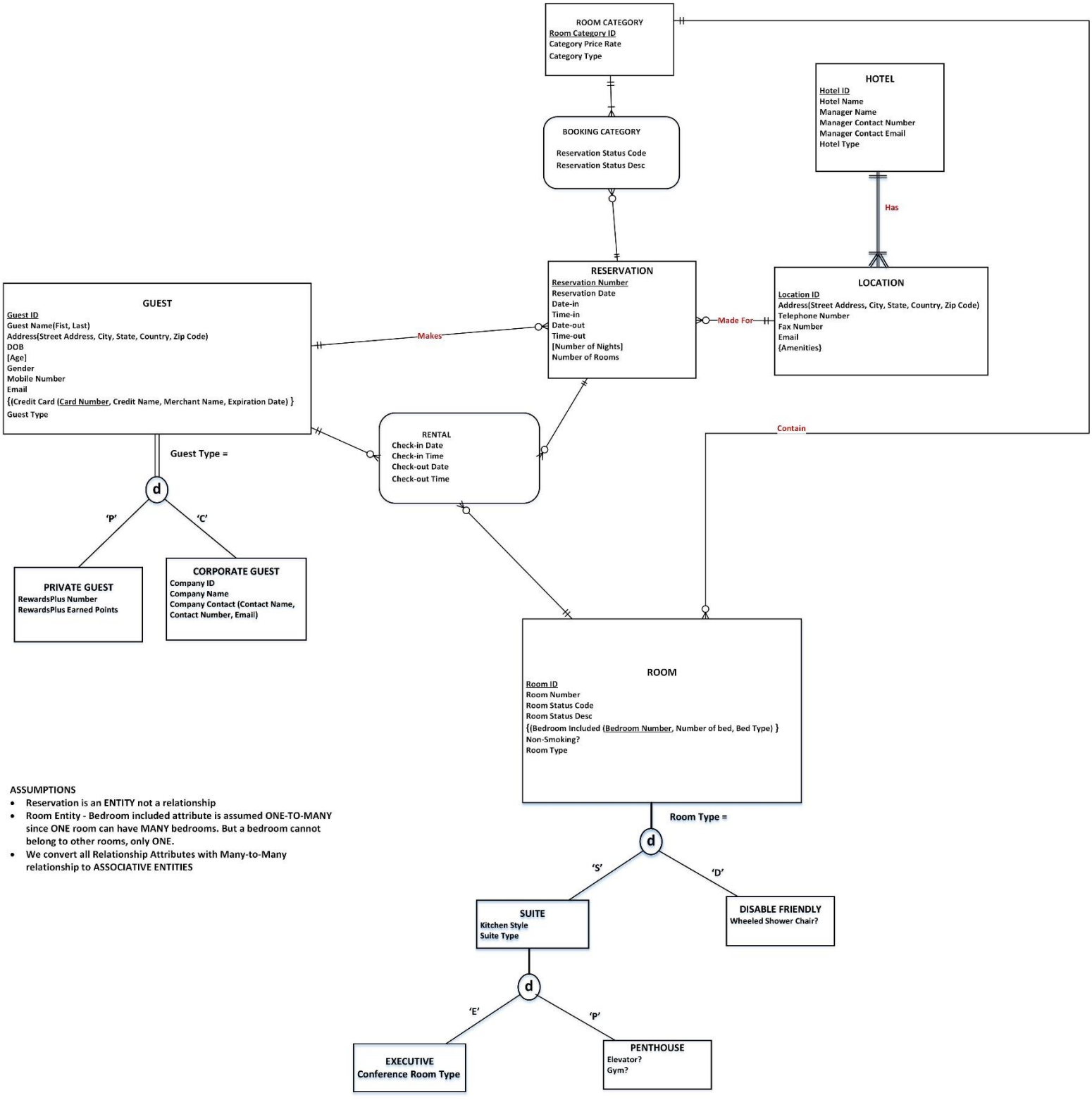
- **My Lectures:**
  - **Lecture 2A, B, & C - CST3504 Data Modeling**
  - **ER modeling HWS & examples in lecture 2C**
  - **Lecture 3A & B - CST3504 Enhanced ER Model**
  - **EER Modeling HWs & examples in Lecture 3B**
- **Book: Modern Database Management (Chapter 2 & 3)**

- Below is my rendition of the EER Model of the Business Requirements of Project 1, with standard crow foot notation and including Associative entities. Use the below version if you like on this project or your version if properly designed and modified based on version below:

EER Conceptual Model (no associative entity conversion)



EER Conceptual Model with Associative Entities conversion



ASSUMPTIONS

- Reservation is an ENTITY not a relationship
- Room Entity - Bedroom included attribute is assumed ONE-TO-MANY since ONE room can have MANY bedrooms. But a bedroom cannot belong to other rooms, only ONE.
- We convert all Relationship Attributes with Many-to-Many relationship to ASSOCIATIVE ENTITIES

❖ **PLEASE NOTE THAT AT THE END OF THIS DOCUMENT, THERE IS AN APPENDIX SECTION WHERE I ALSO LIST OTHER VERSIONS OF THIES EER CONCEPTUAL MODEL WHICH COULD HAVE BEEN POSSIBLE RESULTS**

Requirement #4 (2 POINTS) – (Phase 3 – Design Phase Normalized Logical Model Diagram) – Place the Normalized Logical Model from Project 1 to this section

- In this section, place the final **NORMALIZED LOGICAL MODEL FROM PROJECT 1 to your Design/Implementation Project Document**

  - **THIS DIAGRAM IS PROVIDED TO YOU FOR YOU TO USE OR USE YOUR VERSION IF IT IS DESIGN CORRECTLY OR MODIFIED BASED ON WHAT IS SHOWN BELOW.**

❖ **Requirement #4a – (DIAGRAM #2) COPY/PASTE the final NORMALIZED LOGICAL MODEL FROM PROJECT 1 OR USE VERSION BELOW into the Design/Implementation Document of Requirement #1:**

1. In next page, my version of the **Normalized Logical Model** is being listed for you to use or use your version from project 1 if is correct.
2. **COPY/PASTE**, the **Normalized Logical Model** listed below or your version from PROJECT 1, to the **Design/Implementation Project Document** of **Requirement #1**.

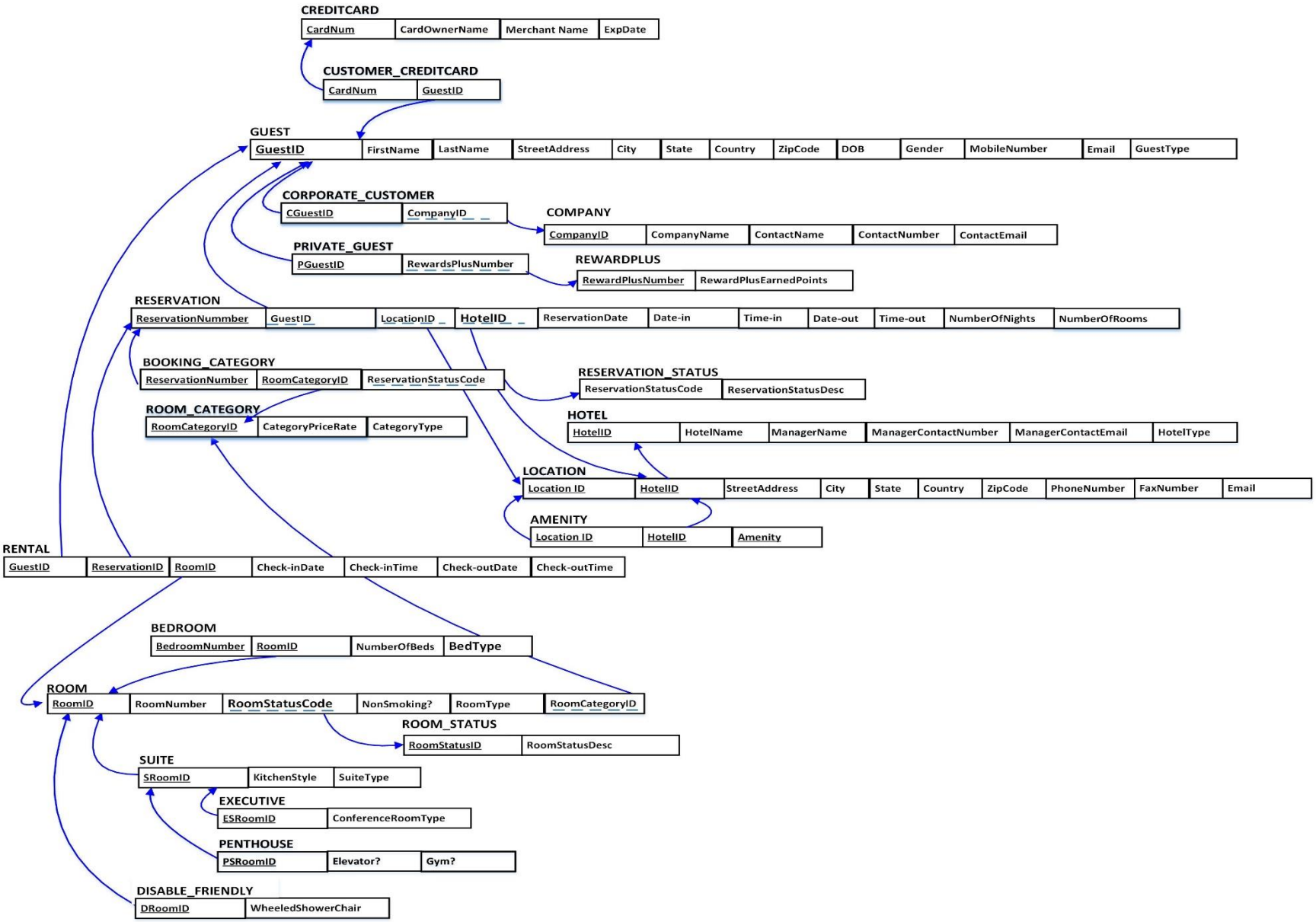
❖ **Requirement #4b – REQUIREMENT FORMAT & PRESENTATION**

1. As stated in **requirement #1**, remember to ***CREATE A HEADER FOR THIS REQUIREMENT*** in your **Design/Implementation Project Document**
2. As stated in **requirement #1**, remember to ***CREATE A PARAGRAPH DESCRIBING THIS SECTION AND WHAT YOU ARE DELIVERING***. **Only a SHORT PARAGRAPH IS REQUIRED**.

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING!
HERE IS WHERE:
Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 4 - CST3504 Logical DB Design (see conversion rules)**
- **Book: Modern Database Management (Chapter 4)**

- Below is my rendition of the **Normalized Logical Model** based on EER diagram of Project 1. Use this version or your modified version:



- **PLEASE NOTE THAT AT THE END OF THIS DOCUMENT, THERE IS AN APPENDIX SECTION WHERE I ALSO LIST THE NON-NORMALIZED LOGICAL MODEL THAT WAS THE FOUNDATION OF THE NORMALIZED MODEL ABOVE.**
- **I AM LISTING IT FOR TEACHING PURPOSE SO YOU CAN SEE THE TRANSITION TO NORMALIZATION.**

Requirement #5 (20 POINTS) – (DESIGN PHASE – DATA DICTIONARY) – Create a Data Dictionary that contains information about attributes and selected datatypes, properties, etc. targeting Oracle DBMS

- Create a **Data Dictionary** by **SELECTING THE BEST DATA TYPES FOR ALL ATTRIBUTES OF THE** Logical Normalized Models created in **Requirement #4** as follows:

❖ **Requirement #5a – CREATE A DATA DICTIONARY TARGETING ORACLE 11g DATA TYPES:**

- Requirements are as follows:

1. Using **Microsoft Excel** or **Microsoft Word TABLE** create a **Data Dictionary** and populate with required Metadata targeting the **ORACLE DBMS 11g.**
2. The data dictionary includes tables using the format of your choice. FEEL FREE TO USE MY FORMAT in Lecture 5A as shown below.

- Here is my example using **MS Word**:

CUSTOMER						
Attribute Name	**Data Type**	**Oracle Data Type**	**Required?**	**Length/Size /Format**	**Constraints**	**Description/ purpose**
Customer_ID	Number	`NUMBER(X)`	Y	4	Primary key	Unique identifier for a customer instance
Name	Variable Character	`VARCHAR2(X)`	Y	50	NOT NULL	Customer full name
BDate	Date	`DATE`		N/A	NOT NULL	Customer's Date of Birth
Address	Variable Character	`VARCHAR2(X)`	Y	75	NOT NULL	Customer full address
Phone	Variable Character	`VARCHAR2(X)`	Y	20	NOT NULL	Phone number
Gender	Variable Character	`CHAR(X)`	Y	1	NOT NULL	Customer's gender (M/F)
Email	Variable Character	`VARCHAR2(X)`	Y	50	NULL	Customer's Email (optional)

- Here is one example using MS Excel:

	A	B	C	D	E	F	G	H	I	J
1	Attribute Name	Table / Concept	Required	Type	Source	DB Field Name	Field Length	Default Values	Repeat?	Notes
2	Job Title	Job Posting	Yes	Text	X Job Board	JobTitle	40	n/a	No	Up to 40 characters
3	Job Description	Job Posting	Yes	Text	X Job Board	JobDesc	4000	n/a	No	Up to 255 characters
4	Hiring Organization	Job Posting	Yes	Look-Up	X Job Board	HiringOrg	40	n/a	No	Look-up Active Employers
5	Salary	Job Posting	Yes	Numeric	X Job Board	Salary	6	n/a	No	No decimal points allowed.
6	Required Qualification	Job Posting	No	List	X Job Board	ReqQual	n/a	n/a	Yes	See Required Qualification List tab
7	First Name	Hiring Manager	Yes	Text	X Job Board	Fname	40	n/a	No	
8	Last Name	Hiring Manager	Yes	Text	X Job Board	Lname	40	n/a	No	
9	Manager Job Title	Hiring Manager	No	Text	X Job Board	ManJobTitle	40	n/a	No	
								Email		

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:

Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 5A - CST3504 Physical Design Intro**
- **Book: Modern Database Management (Chapter 5)**

3. **YOU WIL BE GRADED on your choice of DATA TYPE** targeting the **ORACLE 11g DBMS**. Open the list of **Oracle 11g PL/SQL data types** (Also located in **Lecture 5A in APPENDIX SECTION**) & follow the ***design guidelines covered in class lecture 5A*** discussed in class to select your **data types**.
4. In some cases, where is NOT easily clear, you may want to provide a short comment on why a data type was selected. Not trying to give you more work, but where it makes sense and you feel I WON'T be able to clearly see why you chose something, provide a short comment.

❖ Requirement #5b – COLUMNS REQUIRING TIME DATA TYPE USE MILITARY TIME USING ORACLE 11g NUMBER(4) DATA TYPE:

1. In this project, for any database table column that requires **TIME VALUES** using a **STANDARD FORMAT** such as **09:00 AM, 11:00 PM etc**., use **MILITARY TIME FORMAT** instead for your data type. Example **0900** to represent **09:00 AM** and **1700** to represent **05:00 PM**.
2. I am giving you this option since **TIME STAMP DATA TYPE** which is what you should be using, may be a bit tricky to use in Oracle and I don't have a lecture that will explain this for your convenience. To save you time, we will use 4 numbers to represent time. **PLEASE NOTE THAT IN THE REAL WORLD YOU WILL NOT BE DOING THIS UNLESS THERE IS A PERFORMANCE BENEFIT**. But for this course, let's keep it simple. We have enough complexity already.
3. Below is a chart of **Military Time** to **standard time** conversions for your convenience:

Military Time Chart

12 Hour am-pm Clock	24 Hour Military Time
12:00 am Midnight	0000
1:00 AM	0100
2:00 AM	0200
3:00 AM	0300
4:00 AM	0400
5:00 AM	0500
6:00 AM	0600
7:00 AM	0700
8:00 AM	0800
9:00 AM	0900
10:00 AM	1000
11:00 AM	1100
12:00 PM	1200
1:00 PM	1300
2:00 PM	1400
3:00 PM	1500
4:00 PM	1600
5:00 PM	1700
6:00 PM	1800
7:00 PM	1900
8:00 PM	2000
9:00 PM	2100
10:00 PM	2200
11:00 PM	2300
12:00 Midnight	2400

❖ Requirement #5c – WHICH EVER METHOD YOU USE (WORD OR EXCEL) PLACE THE DATA DICTIONARY IN YOUR DESIGN/IMPLEMENTATION PROJECT DOCUMENT

1. You should be creating your data dictionary in MS Word in your **project document** or if created in MS Excel, or other tool, place your **DATA DICTIONARY** in your **Project Document**.
2. Your **DATA DICTIONARY** has to be listed in your **Project Document** in the section indicated in **Requirement #1**.

❖ Requirement #5d – REQUIREMENT FORMAT & PRESENTATION

1. As stated in **requirement #1**, remember to ***CREATE A HEADER FOR THIS REQUIREMENT*** in your **Design/Implementation Project Document**
2. As stated in **requirement #1**, remember to ***CREATE A PARAGRAPH DESCRIBING THIS SECTION AND WHAT YOU ARE DELIVERING***. **Only a SHORT PARAGRAPH IS REQUIRED**.

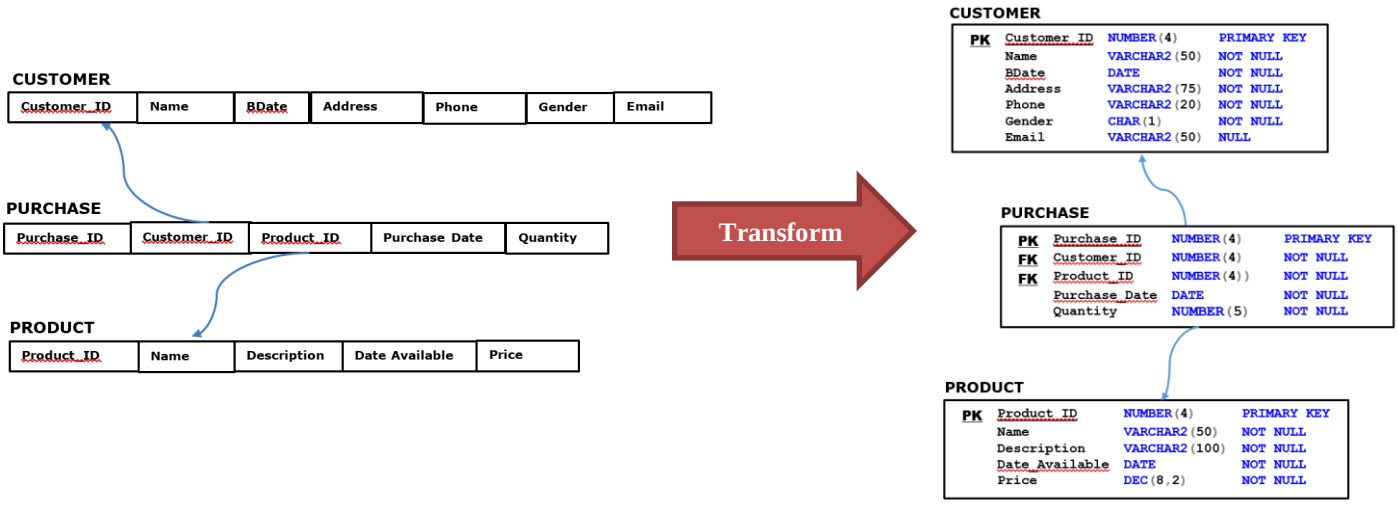
Requirement #6 (20 POINTS) – (DIAGRAM #3) – Create the Physical Data Model Diagram based on your Normalized Logical Diagram of Requirement #5 targeting ORACLE 11g DBMS

- Create the Physical Model/Schema diagram by CONVERTING the **NORMALIZED** logical model of **Requirement #4 & the DATA DICTIONARY of Requirements #5 to a Physical Diagram**:

❖ Requirement #6a – (DIAGRAM #3) CREATE PHYSICAL MODEL using a DRAWING TOOL:

1. Using a combination of the **NORMALIZED LOGICAL** Model listed in **Requirement #4**, and the **DATA DICTIONARY** of **Requirement #5** and a **MODELING/DRAWING TOOL such as Visio, Oracle Data Modeler, or drawing program, *CREATE*** the **PHYSICAL MODEL SCHEMA DIAGRAM**.
2. Below is an example of the ***high-level steps*** to create the physical model diagram from the logical model based on ***Oracle 11g***. we will use this example to give you an idea of what is required.
3. **We assume a Logical Model containing a CUSTOMER, PURCHASE & PRODUCT tables and their respective RELATIONSHIPS**. Then we convert these Logical Model, leveraging the **Data Dictionary** to populate the data types.
4. **IMPORTANT** – note that the **final Physical Schema Model Diagram is IDENTICAL to the NORMALIZED LOGICAL MODEL EXCEPT EACH TABLE IS NOT LISTED AS ONE ROW AND COLUMNS BUT A SINGEL BOX OR ENTITY POPULATED WITH THE DATA TYPE METADATA FROM THE DATA DICTIONARY. MOST IMPORTANT, THE RELATIONSHIPS OF THE NORAMLIZED LOGICAL MODEL MUST BE KEPT INTACT AND WHIOUT CHANGE**.

Step 1- We start with this Normalized **Step 2 – We transform to this Physical Schema**



- Your goal is to CREATE a physical model using the Normalized Logical Model and the Data Dictionary of **Requirements 4 & 5**.

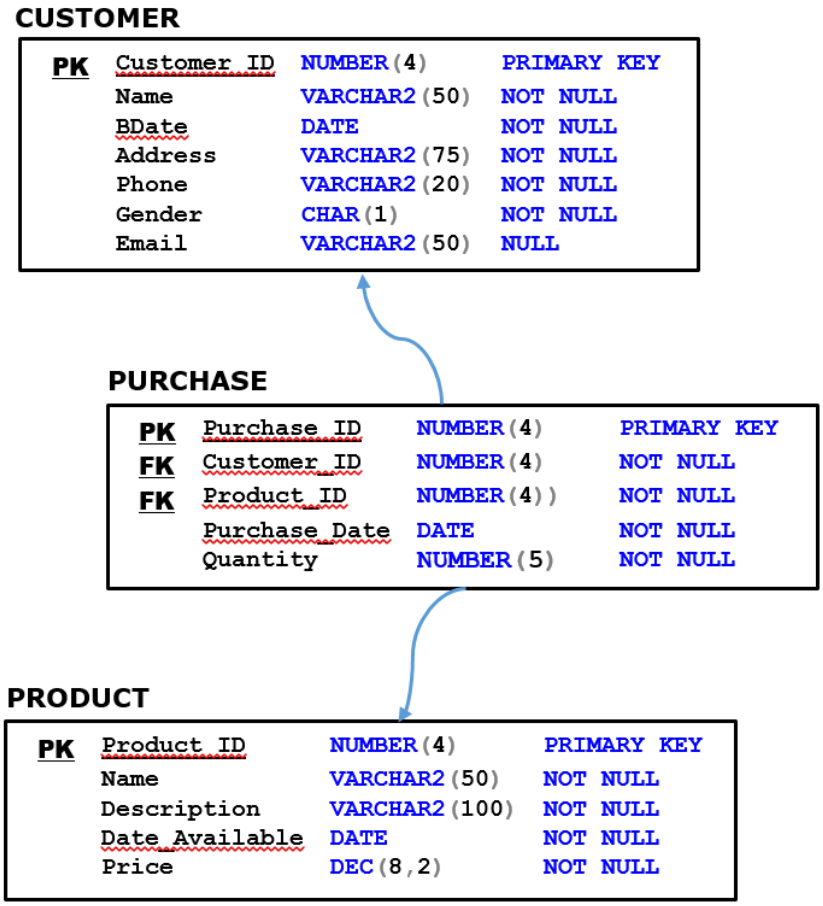
TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:

Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Dictionary & Physical Diagram – Lecture 5A - CST3504**
- **Book: Modern Database Management (Chapter 5)**

❖ Requirement #6b – COPY/PASTE YOUR FINAL PHYSICAL DIAGRAM to the Design/Implementation document of Requirement #1

1. **AFTER YOU CREATE YOUR PHYSICAL MODEL DIAGRAM USING A DRAWING TOOL, COPY/PASTE** you **Physical Model Diagram** to your **Design/Implementation Project Document**.
2. This will be your 3rd diagram. Make sure you label you section etc., as shown in the example below:



❖ Requirement #6c – REQUIREMENT FORMAT & PRESENTATION

1. As stated in **requirement #1**, remember to ***CREATE A HEADER FOR THIS REQUIREMENT*** in your **Design/Implementation Project Document**
2. As stated in **requirement #1**, remember to ***CREATE A PARAGRAPH DESCRIBING THIS SECTION AND WHAT YOU ARE DELIVERING***. **Only a SHORT PARAGRAPH IS REQUIRED**.

Requirement #7 (20 POINTS) – (Phase 4 – DEVELOPMENT/IMPLEMENTATION PHASE- SCRIPT FILE #1) – Implement the Physical Model using Oracle 11g & Oracle SQL Developer by CREATING & EXECUTING a Script File that contains ALL the Data Definition Language (DDL) Statements to create the Schema

- Implement your schema design (Physical Model) in Oracle 11g as follows (See L:

❖ Requirement #7a – (DB SCRIPT FILE #1)) CREATE A SCRIPT FILE TO HOST CREATE TABLE STATEMENTS TO IMPLEMENT YOUR SCHEMA:

1. Using **Oracle SQL Developer**, ***CREATE*** and ***NAME*** a **SCRIPT FILE** targeted to store all the DDL **CREATE TABLE** statements to be used to implement the physical design. **This will be your FIRST SCRIPT FILE** (**CREATE TABLE SCRIPT FILE #1)**
2. **SEE LECTURE 6A for installation of Oracle & SQL Developer and setting up your development environment**.
3. **SEE LECTURE 6B to learn how to create a Script File using Oracle SQL Developer**.
4. Make sure you **NAME** & **SAVE** your **Script File**.

❖ Requirement #7b – CREATE the PHYSICAL SCHEMA inside the SCRIPT FILE USING DDL CRATE TABLE STATEMENTS & EXECUTE TO IMPLEMENT EACH CREATE TABLE STATEMENT:

1. Using **Oracle SQL Developer**, ***IMPLEMENT*** the entire **PHYSICAL MODEL SCHEMA** by creating all ***tables*** and ***relationships*** (Foreign-key/Primary-key) using **DDL CREATE TABLE** statements ***INSIDE*** **THIS SCRIPT FILE**.
2. ***EXECUTE*** each of the **CREATE TABLE statements** in the appropriate order to create your **physical schema**.
3. Make sure you **NAME** & **SAVE** your **Script File**.

❖ Requirement #7c – COPY/PASTE the entire SCRIPT CODE (DDL STATEMENTS) ONTO YOUR PROJECT DESIGN/IMPLEMENTATION DOCUMENT:

1. **COPY/PASTE** all the **CONTENT** or **CREATE TABLE STATEMENTS** from the **CREATE TABLE script file** onto this section of your **Design/Implementation Project Document**.
2. Make sure you label and properly describe this section.

❖ Requirement #7d – REQUIREMENT FORMAT & PRESENTATION

3. As stated in **requirement #1**, remember to ***CREATE A HEADER FOR THIS REQUIREMENT*** in your **Design/Implementation Project Document**
4. As stated in **requirement #1**, remember to ***CREATE A PARAGRAPH DESCRIBING THIS SECTION AND WHAT YOU ARE DELIVERING***. **Only a SHORT PARAGRAPH IS REQUIRED**.

❖ Requirement #7e – SAVE & SUBMIT THE PHYSICAL CREATE TABLE SCRIPT FILE #1 WITH YOUR PROJECT DESIGN/IMPLEMENTATION DOCUMENT:

1. **SAVE** your final **CREATE TABLE SCRIPT FILE #1.**
2. **SUBMIT** your final **CREATE TABLE SCRIPT FILE #1**, along with your **Design/Implementation Project Document**. **SEE deliverable section of this document for details on what to submit!!!!**

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:

Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 6B - CST3504 DB Dev with Oracle PART 2**
- **Book: Modern Database Management (Chapter 6)**

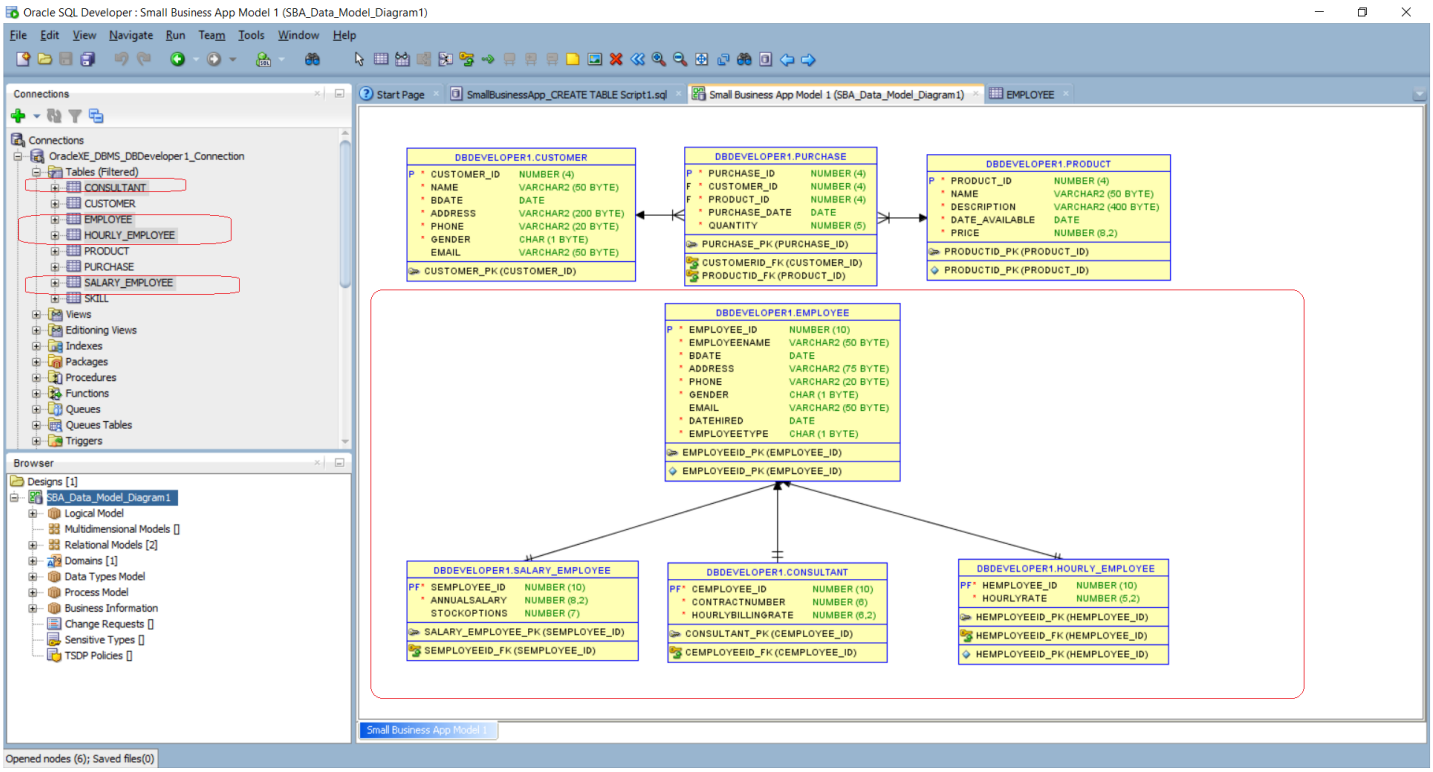
Requirement #8 (5 POINTS) – (Phase 4 – DEVELOPMENT/IMPLEMENTATION PHASE: GENARATED SCHEMA DIAGRAM #4) – Generate the Table Schema Diagram using Oracle SQL Developer of your Physical Model Implemented in Requirement #7 to Validate & Compare to your Physical Model of Requirement #6

- **Generate** a diagram of your implemented Physical Model as follows:

❖ Requirement #8(a) – (DB DIAGRAM #4 & DB FILE #2) GENERATE THE DATABASE SCHEMA TABLE DIAGRAM:

1. Using **Oracle SQL Developer**, ***GENERATE*** a table diagram of your schema created in **requirement #7**.
2. **SEE LECTURE 6B for instruction on generating & formatting this diagram using SQL Developer**.
3. This **DIAGRAM #4** should contain enough details and information on the entities or tables symbols to match your **PHYSICAL MODEL DIAGRAM (Diagram #3)** of **Requirement #6**.
4. **Save & Name** this **DIAGRAM #4** to file. Name accordingly.

- Below is an example of this type of diagram:



❖ Requirement #8b – COPY/PASTE the TABLE SCHEMA DIAGRAM ONTO YOUR PROJECT DESIGN/IMPLEMENTATION PROJECT DOCUMENT:

1. **COPY/PASTE** all the **SCHEMA DIAGRAM #4** from SQL developer onto a section of your **Design/Implementation Project Document**.
2. Make sure you label and properly describe this section.

❖ Requirement #8c – REQUIREMENT FORMAT & PRESENTATION

1. As stated in **requirement #1**, remember to ***CREATE A HEADER FOR THIS REQUIREMENT*** in your **Design/Implementation Project Document**
2. As stated in **requirement #1**, remember to ***CREATE A PARAGRAPH DESCRIBING THIS SECTION AND WHAT YOU ARE DELIVERING***. **Only a SHORT PARAGRAPH IS REQUIRED**.

❖ Requirement #8d – SAVE & SUBMIT THE ACTUAL TABLE SCHEMA DIAGRAM FILE #2 WITH YOUR PROJECT DESIGN/IMPLEMENTATION DOCUMENT:

1. **SAVE** your final **TABLE SCHEMA DIAGRAM FILE #2**.
2. **SUBMIT** your final **TABLE SCHEMA DIAGRAM FILE #2**, along with your Design/Implementation word document. **SEE deliverable section of this document for details on what to submit!!!!**

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:

Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 6B - CST3504 DB Dev with Oracle PART 2**
- **Book: Modern Database Management (Chapter 6)**

Requirement #9 (20 POINTS) – (Phase 4 – DEVELOPMENT/IMPLEMENTATION PHASE: Testing & Validation via Queries) – Create a Script to Test your database/Tables Using Oracle SQL Developer to Create & Execute Data Manipulation Language (DML) SQL Statements (SELECT, INSERT, UPDATE & DELETE)

- Test your tables as follows:

❖ **Requirement #9a – (DB SCRIPT FILE #3) CREATE A DML SQL TEST QUERIES SCRIPT**
❖ **READ CAREFULLY AND IMPLEMENT AS YOU ARE REQUIRED:**

1. Using **Oracle SQL Developer**, **CREATE A SCRIPT FILE** to ***STORE*** **TEST QUERIES** against your schema using **DML SQL STATEMENTS**. This will be your **3RD FILE BUT SECOND SCRIPT FILE** (**QUERY TEST Script File #3**)
2. **Save** & **Name** the script to file accordingly.

Design Theory for Implementing this Requirement

- After creating and saving your file, the objectives are that you will Add **DML SQL STATEMENTS** to tests your tables. I will now explain the details requirements on HOW you will go about this>

Step 1 – Analyze & Select Table Group (s)

- These test require that you analyze your schema and select table groups as maximize your test. The idea is as follows:

  - You are not testing ALL tables, but selecting a set or group of tables that will allow you to efficiently and accurately test the following:

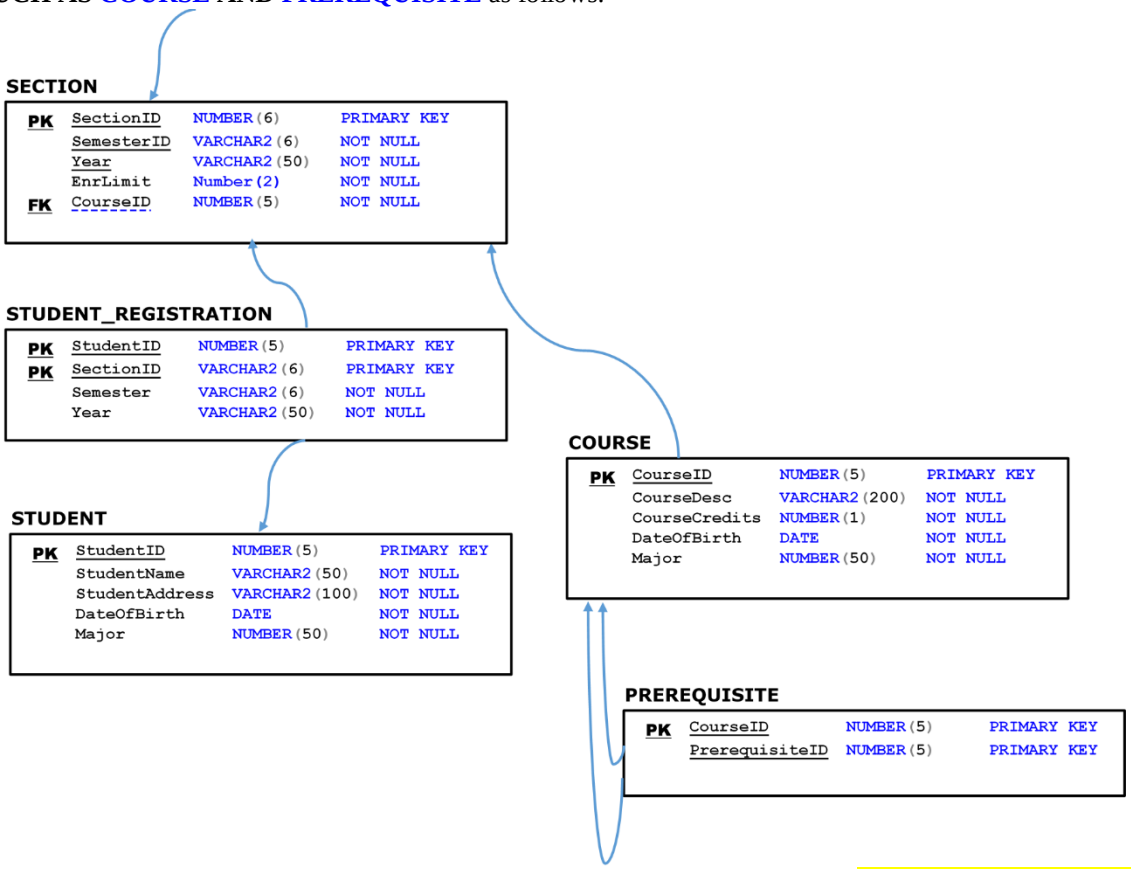
    - **SELECT** SQL Statements
    - **INSERT** SQL statements
    - **UPDATE** SQL Statements
    - **DELETE** SQL Statements

- You are going to be asked to create **DML SQL STATEMENTS** to tests your database schema, here is the first step in your design of your testing procedure:

  - When you create your **DML SQL STATEMENTS**, keep in mind the following:

    - **YOU DO NOT HAVE TO TEST EVERY TABLE ON ENTIRE SCHEMA!!! ONLY A SELECTED FEW. THEREFORE, YOU NEED DECIDE ON A TABLE GROUP**. But here is the basic idea of the requirements:

      1. Select a **TABLE GROUP** or business scenario that will required a combination of minimum 3 tables such as a **BINARY or TERNARY ASSOCIATIVE TABLE RELATIONSHIP (3 or more tables)**.
      2. You need to think of a business scenario based on your schema or database and put some thoughts on which **TABLE GROUP** to select.
      3. Here is an example of a chosen **TABLE GROUP FOR A SCHOOL MANAGEMENT SYSTEM APPLICATION**, containing a **BINARY ASSOCIATIVE TABLE RELATIONSHIPS BETWEEN STUDENT, STUDENT_REGISTRATION AND SECTION & OTHER DEPENDENT RELATIONSIPS SUCH AS COURSE AND PREREQUISITE** as follows:



      - **STUDENT**, **STUDENT_REGISTRATION** & **SECTION** tables make up a **BINARY ASSOCIATIVE TABLE RELATIONSHIP**. Note, some queries may require additional tables for example **SECTION** is dependent on **COURSE** and some of its information come from the **COURSE** table may be needed, which in turn may include **PREREQUISITE** if needed, depending on the query.

    - In my **STUDENT**, **STUDENT_REGISTRATION** & **SECTION** tables example, you would perform tests queries using **SELECT**, **UPDATE INSERT** & **DELETE** SQL statements as indicated in the next section.
    - In addition to **COURSE** & **PREREQUISITE** tables which was selected to be part of this group as well.
    - **THERE IS NO LIMITATION AS TO THE NUMBER OF TABLES YOU CAN SELECT FOR A TABLE GROUP**
    - ***Note that this will require that you populate these 3 or all 5 tables with data prior to executing your test statements***.

- Next, I will list the remaining detailed requirements to complete this **Requirement #9**.

❖ **Requirement #9b (5 POINTS) – Create INSERT statements to POPULATE your chosen TABLES from your TABLE GROUP:**

- In your **TEST SCRIPT FILE**, create **DML SQL INSERT STATEMENTS** to **POPULATE** selected table as follows:

1) **INSERT STATEMENTS (#1 - #5)** – Create **INSERT** Statements that ***inserts all fields/columns*** to a minimum of **FIVE RECORDS** into **THE FIRST TABLE OF YOUR CHOSEN TABLE GROUP. NOTE, THIS TABLE SHOULD NOT AN THE ASSOCIATIVE ENTITY!**
2) ***Execute*** each **SQL INSERT Statement**.
3) **IMPORTANT! –** After inserting all your records, VERIFY the data is in the TABLE. Take a **SCREEN-SHOT** of the **results of verification** & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**.
4) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & how you got the results & proof**

5) **INSERT STATEMENTS (#6 - #10)** – Create **INSERT** Statements that ***inserts all fields/columns*** to a minimum of **FIVE RECORDS** into **THE SECOND TABLE OF YOUR CHOSEN TABLES TABLE GROUP. NOTE, THIS TABLE SHOULD NOT BE AN ASSOCIATIVE ENTITY!**
6) ***Execute*** each **SQL INSERT Statement**.
7) **IMPORTANT! –** After inserting all your records, VERIFY the data is in the TABLE. Take a **SCREEN-SHOT** of the **results of verification** & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**.
8) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & how you got the results**

9) **INSERT STATEMENTS (#11 - #15)** – Create **INSERT** Statements that ***inserts all fields/columns*** to a minimum of **FIVE RECORDS** into **THE ASSOCIATIVE ENTITY TABLE OF YOUR CHOSEN TABLES TABLE GROUP.**
10) **IMPORTANT! –** For the **ASSOCIATIVE ENTITY, the DATA MUST MATCH/ALIGN WITH THE OTHER TWO TABLES** for it to make business sense. In other words, **FOREIGN-KEYS MUST MATCH PRIMARY KEYS of parent tables**.
11) ***Execute*** each **SQL INSERT Statement**.
12) **IMPORTANT! –** After inserting all your records, VERIFY the data is in the TABLE. Take a **SCREEN-SHOT** of the **results of verification** & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**.
13) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & how you got the results & proof**

14) **REPEAT** the above steps for any additional tables that are part of your **TABLE GROUP.**

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:

Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 6C - CST3504 DB Dev with Oracle PART 3**
- **Book: Modern Database Management (Chapter 6)**

❖ **Requirement #9c (5 POINTS)** – Create **SELECT** statements to **RETURN RECORD(S)** your chosen **TABLE** from your **TABLE GROUP**:

- In your **TEST SCRIPT FILE**, create **DML SQL SELECT STATEMENTS** to **TEST** the schema as follows:

1) **SELECT STATEMENT #1** – Create **SELECT** Statements that ***queries*** **ONE of the CHOSEN TABLE WITH DATA, RETURNING ALL COLUMNS OF ONE RECORD ONLY BASED ON PRIMARY KEY.**
2) ***Execute*** **SQL SELECT Statement.**
3) **IMPORTANT! –** After executing, take a **SCREEN-SHOT** of the results & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**.
4) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & proof**

5) **SELECT STATEMENT #2** – Create **SELECT** Statements that ***queries*** **ONE of the CHOSEN TABLE WITH DATA, RETURNING ALL COLUMNS OF MULTIPLE RECORDS BASED ON SOME CRITERIA.**
6) ***Execute*** **SQL SELECT Statement.**
7) **IMPORTANT! –** After executing, take a **SCREEN-SHOT** of the results & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**.
8) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & proof**

9) **SELECT STATEMENT #3** – Create **SELECT** Statements that ***queries*** **ALL THREE of the CHOSEN TABLE WITH DATA, RETURNING SOME COLUMNS OF ONE RECORD ON A CRITERIA FOR SOME BUSINESS SCENARIO.** For example, using my STUDENT, STUDENT_REGISTRATION & SECTION tables, we could create a business query for ***what section & course is Joe Smith, whose ID = 1111, taking this semester***.
10) ***Execute*** **SQL SELECT Statement.**
11) **IMPORTANT! –** After executing, take a **SCREEN-SHOT** of the results & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**.
12) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & proof**

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:

Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 6C - CST3504 DB Dev with Oracle PART 3**
- **Book: Modern Database Management (Chapter 6)**

❖ **Requirement #9d (5 POINTS) –** Create **UPDATE** statements to **UPDATE RECORD(S)** your chosen **TABLE** from your **TABLE GROUP**:

- In your **TEST SCRIPT FILE**, create **DML SQL UPDATE STATEMENTS** to **TEST** the schema as follows:

1) **UPDATE STATEMENT #1** – Create **UPDATE** Statements that ***UPDATES*** **ONE of the CHOSEN TABLE WITH DATA, UPDATING ALL COLUMNS OF ONE RECORD ONLY BASED ON PRIMARY KEY.**
2) ***Execute*** **SQL UPDATE Statement**.
3) **IMPORTANT! –** After updating the record, VERIFY the row in the TABLE was updated. Take a **SCREEN-SHOT** of the **results of verification** & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**
4) **Design/Implementation Document**.
5) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & proof**

6) **UPDATE STATEMENT #2** – Create **UPDATE** Statements that ***UPDATES*** **THE ASSOCIATIVE ENTITY of the WITH DATA, UPDATING A RECORD INDICATING AN ASSOCIATION BETWEEN THE TWO PARENT TABLES.**
7) ***Execute*** **SQL UPDATE Statement**.
8) **IMPORTANT! –** After updating the record, VERIFY the row in the TABLE was updated. Take a **SCREEN-SHOT** of the **results of verification** & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**
9) **Design/Implementation Document**.
10) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
    a) **Label or describe the query(s) you are about to execute**
    b) **List the QUERY(S) from your script**
    c) **Screen-shot of results & proof**

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:

Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 6C - CST3504 DB Dev with Oracle PART 3**

- **Book: Modern Database Management (Chapter 6)**

❖ **Requirement #9e (5 POINTS) –** Create **DELETE** statements to **DELETE RECORD(S)** from you chosen **TABLE** from your **TABLE GROUP**:

- In your **TEST SCRIPT FILE**, create **DML SQL DELETE STATEMENTS** to **TEST** the schema as follows:

1) **DELETE STATEMENT #1** – Create **DELETE** Statements that ***DELETES*** **ONE row ONE RECORD BASED ON PRIMARY KEY FROM ONE of the CHOSEN TABLE WITH DATA(NOT ASSOCIATIVE TABLE). IF YOU ARE NOT PERMITTED, TAKE THE NECESSARY STEPS REQUIRED THAT WILL ALLOW YOU TO DO SO.**
2) ***Execute*** **SQL DELETE Statement**.
3) **IMPORTANT! –** After deleting the record, VERIFY the row in the TABLE was deleted. Take a **SCREEN-SHOT** of the **results of verification** & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**
4) **Design/Implementation Document**.
5) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
   a) **Label or describe the query(s) you are about to execute**
   b) **List the QUERY(S) from your script**
   c) **Screen-shot of results & proof**

6) **DELETE STATEMENT #2** – Create **DELETE** Statements that ***DELETES*** **ONE RECORD BASED ON PRIMARY KEY from ASSOCIATIVE TABLE of the CHOSEN TABLE GROUP.**
7) ***Execute*** **SQL DELETE Statement**.
8) **IMPORTANT! –** After deleting the record, VERIFY the row in the TABLE was deleted. Take a **SCREEN-SHOT** of the **results of verification** & ***PASTE*** the **SCREEN-SHOT** in your **Design/Implementation Document**
9) **Design/Implementation Document**.
10) **IMPORTANT! – LABEL EACH SECTION PROFESSIONALLY IN THE DOCUMENT as follows:**
    a) **Label or describe the query(s) you are about to execute**
    b) **List the QUERY(S) from your script**
    c) **Screen-shot of results & proof**

TO UNDERSTAND HOW TO SOLVE THIS REQUIREMENT, GO FISHING! HERE IS WHERE:
Theory & Examples on this TOPIC can be found:

- **My Lectures:**
  - **Lecture 6C - CST3504 DB Dev with Oracle PART 3**
- **Book: Modern Database Management (Chapter 6)**

Project Exam Expected Deliverables

Project Deliverables

Submit the following based on requirements All package with WinZip:

1. **DOCUMENT/FILE #1 – (From Requirement #1)** – **PDF version** of the **Project Design & Implementation Project Document** which includes the information:

   a) ***ALL REQUIREMENTS EXECUTED*** (steps **1 through 9** executed).

   b) ***CONVERT*** this Word document to a **PDF file.** **SUBMITT THE PDF ONLY!**

2. **FILE #2** – Also submit the actual **CREATE TABLE SCRIPT FILE #1** ***saved*** from **Oracle SQL Developer**.

3. **FILE #3** – Also submit the actual **DATABASE SCHEMA TABLE DIAGRAM FILE** ***saved*** from **Oracle SQL Developer**.

4. **FILE #4** – Also submit the actual **DML SQL TEST QUERIES SCRIPT FILE #2** ***saved*** from **Oracle SQL Developer**.

Final Deliverable Summary

- **PART 1 –** You are to submit the following 4 documents with the details listed in requirements above:

  1) **PDF Project Design/Implementation Project Document with all requirements information, description, screen shots etc., File #1 (PDF ONLY!)**
  2) **Create Table Script File #2**
  3) **Schema Table Diagram File #3**
  4) **Test SQL Queries Script File #4**

- **PART 2 –** Put all these documents/files in a folder and **WINZIP PACKAGE** (in windows simply: **right-click folder|send to|compress (zip)** to compress folder. **DO NOT USE WINRAR!!!! WINZIP ONLY!**
- **PART 3 –** Send **WINZIP PACKAGE** via email the following email address:

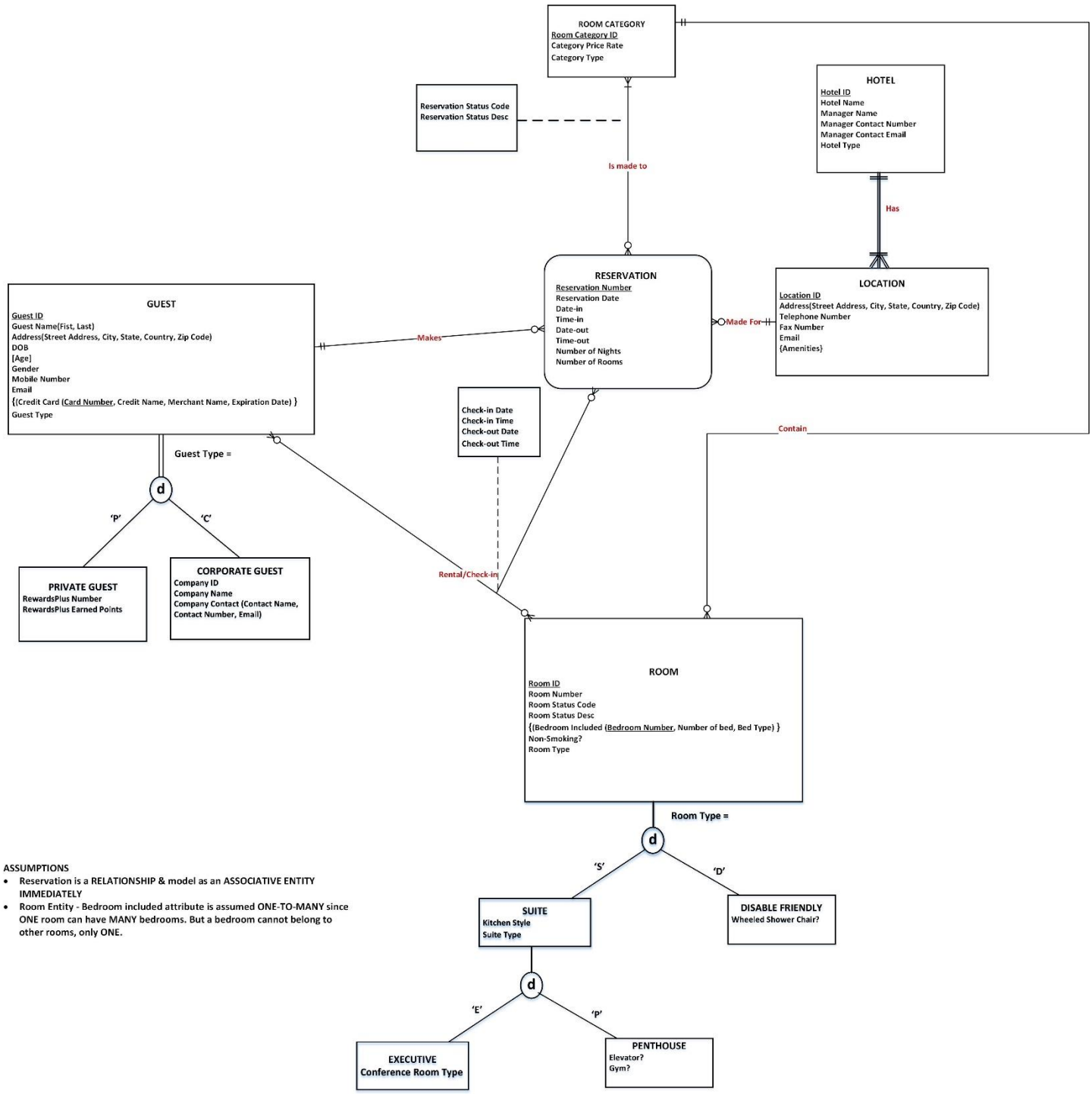
  1. Email a **WINZIP PACKAGE** to **arod1212@outlook.com** (**DO NOT cc arod@microsoft.com**)
  2. Email subject line should have the following syntax: **CST3604-YOUR FULL NAME-PROJECT 1**

Project Due Date

- Due date **MONDAY MARCH 4TH (I suggest you continue to target FEBRUARY 25TH, 2019)**
  - **DON'T WAIT TO START THIS PROJECT ON THE WEEKEND THAT IS DUE! START IMMEDIATELY! IT WILL REQUIRED A LOT OF WORK!!!**

Appendix A

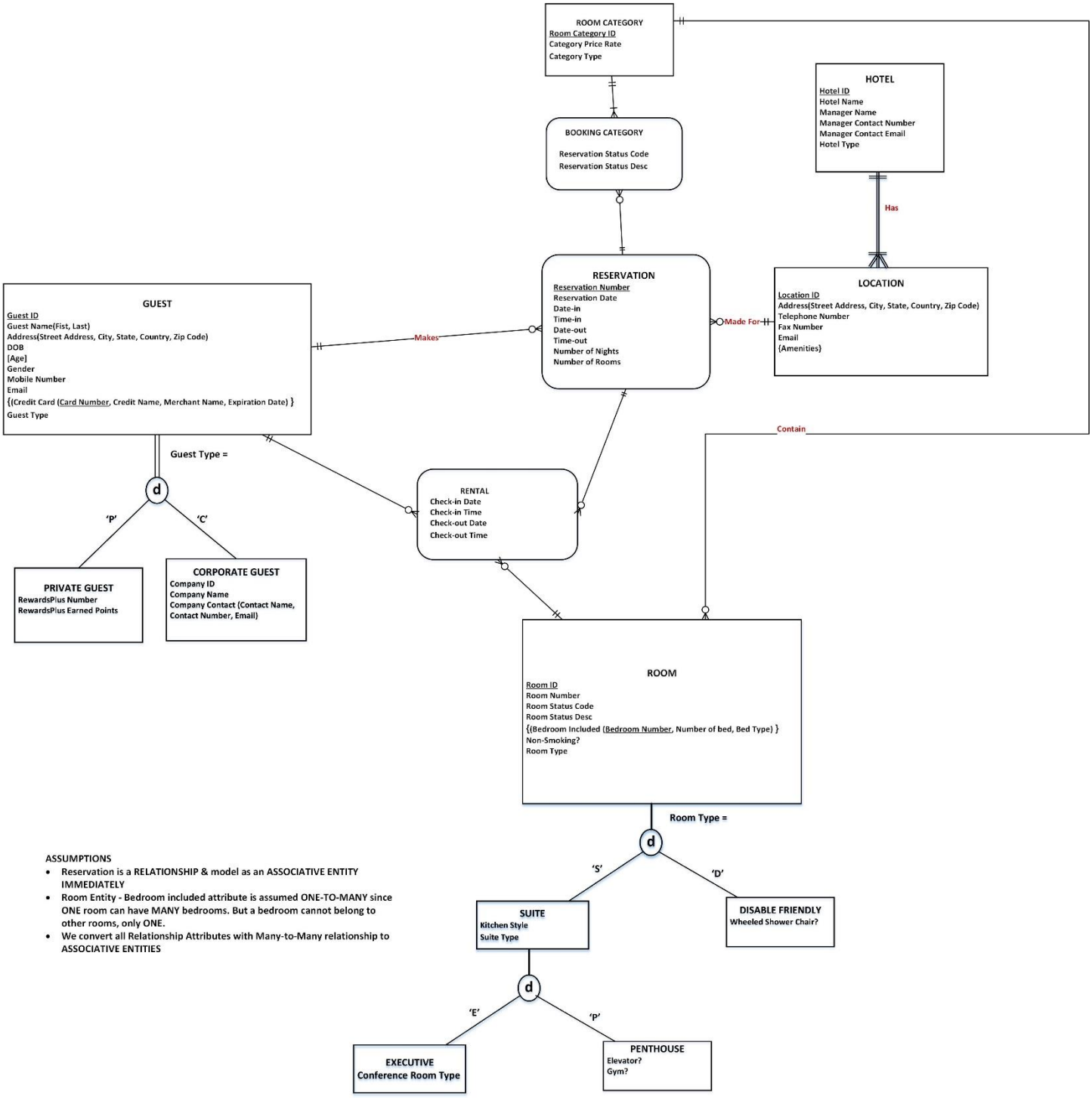
Other Possible EER Model Versions – Reservation modeled as an Associative Entity from the start



ASSUMPTIONS

- Reservation is a RELATIONSHIP & model as an ASSOCIATIVE ENTITY IMMEDIATELY
- Room Entity - Bedroom included attribute is assumed ONE-TO-MANY since ONE room can have MANY bedrooms. But a bedroom cannot belong to other rooms, only ONE.

Other Possible EER Model Versions – Reservation modeled as an Associative Entity from the start & entire diagram converted to Associative Entities



ASSUMPTIONS

- Reservation is a RELATIONSHIP & model as an ASSOCIATIVE ENTITY IMMEDIATELY
- Room Entity - Bedroom included attribute is assumed ONE-TO-MANY since ONE room can have MANY bedrooms. But a bedroom cannot belong to other rooms, only ONE.
- We convert all Relationship Attributes with Many-to-Many relationship to ASSOCIATIVE ENTITIES

Appendix B

Logical Model Derived from EER Model After Transformation (Non-Normalized)