Part One

Hotel System

1) Design the database model to solve the hotel system, it can be No SQL or SQL relation.

2) Briefly explain the functionality of your model, highlight important situations such as: reservation control, data security, fault tolerance, replication, etc.

3) Think about scenarios in case the system collapses and how to create a temporal solution, for example: database corruption, an error not considered, customers with the same first and last name, etc.

4) Implement your model in the selected database manager, specify the reason why you decided to use this database. In this point, you must convert your visual model (diagram) to a physical model (create collections in MongoDb or create tables in SQL).

5) Create at least 5 endpoints to display/update/delete some data of your model, validate the data entry and clearly indicate the error message in case of occurrence.

Upload all the files you have generated to the LMS system

Part Two

Guess the number

Create a react app that guess a number. The game starts when the player presses the Start button, the application will generate a random number between 1 to 30, called x. There will be a text box (input type) where the user will write a number, called y. If x = y then displays an announcement that says, "Congratulations you win", otherwise specify if the number given is greater or less than the number y. The player will have 3 attempts to win, if user fails the number x will be displayed.

Example:

Random Number = 17

Write your number: 5

No, you must write a number greater than 5

Write your number: 25

No, you must write a number less than 25

Write your number: 19

No, you must write a number less than 19

You fail: The number was: 17

Random Number = 24

Write your number: 20

No, you must write a number greater than 20

Write your number: 26

Congratulations you win

Upload your project (without modules folder) to the LMS system