

INF272: Homework Assignment 01



- **Themes:** Programmatic problem solving, reasoning and logic.
- **Due date:** 2024-06-03 before 23:59

Notes & Instructions

Capetropolis Tourism [100 Marks]

This assignment will help a student to prepare for the semester test. Read the details and instructions very carefully. Make notes where required.

PURPOSE AND OBJECTIVES.

The purpose and objective of this assignment is:

- **Objective 1:** **Preparing for the Semester Test.**
- **Objective 2:** Evaluating a student's understanding for focussed interventions.
- **Objective 3:** Integrated programming.
- **Objective 4:** Programmatic problem solving.
- **Objective 5:** Programmatic reasoning.
- **Objective 6:** Programmatic logic.

MATERIAL BEING REFERENCED.

The session themes listed below is the exact themes for the semester test. By completing this application, students will have to both study and apply the themes listed. To successfully complete this activity, a student should refer to the notes, details and sample projects that is part of:

- **Session 04:** HTML and CSS within MVC.
- **Session 05:** Responsive web design (RWD), HTML and Bootstrap in MVC.
- **Session 06:** JavaScript and RWD in MVC (Part 1).
- **Session 07:** JavaScript and RWD in MVC (Part 2).
- **Session 08:** Manipulating the MVC HTML Document Object Model (DOM) with jQuery.
- **Session 09:** Processing data with lists in MVC. Moving towards data manipulation and processing.
- **Session 10:** The application of on add-ons and third-party applications in MVC.

READING OR RESEARCH THAT WILL BE NECESSARY.

You will have to conduct independent research and reading on the topics listed below. The topics were discussed and covered in the Monday preparation theory sessions. Students who participated in the sessions will have a broad overview of the topics listed. The topics required is as follows:

- **Topic 1:** Recording your video demonstration and then compressing the demo for upload (self-study).
- **Topic 2:** Any of the aforementioned referenced themes required for the completion of this project.
- **Topic 3:** Making use of localStorage.
- **Topic 4:** The application of Bootstrap for RWD.
- **Topic 5:** Exporting data to a text file (there is an example in a lecture sample project).
- **Topic 6:** The application of JavaScript and jQuery within the context of the DOM.
- **Topic 7:** Extensions, add-in's and prefabricated code libraries.
- **Topic 8:** GUID (globally unique identifier) – this was not covered in any class. Student should read up on this topic.

GENERAL INSTRUCTIONS.

Please ensure that you follow the following general instructions.

- This is an **individual** assignment.
- The assignment is based on assessment objectives and requirements.
- If an objective has been achieved a mark will be allocated.
- By completing this assignment, you will be preparing for the semester test.
- **The themes found in the assignment is the exact themes for the semester test.**

SUBMISSION DUE: 2024-06-03 before 23:59.

- There shall be no extensions to the deadline.
- All work will be marked offline by the Lecturers and the Assistant lecturers.
- Verify the completeness of your upload.
- Incomplete uploads will be considered unsubmitted work.
- If homework submissions are uploaded too late then upload errors WILL happen.
- Do not wait to the last minute to complete the assignment.
- Start working on the assignment as soon as it is posted.
- There are multiple upload opportunities enabled if your upload is incomplete.
- **Email submissions WILL NOT be accepted.**
- **Late submissions WILL NOT be accepted.**
- **NO EXCEPTIONS WILL BE MADE FOR ANYONE.**
- Do not request any form of an exception by email.
- Emails requesting an exception will not receive a reply.
- There shall be no "negotiation" with regards to the mark allocation for missed or incomplete submissions.

SUBMISSION INSTRUCTIONS.

- In this assignment, you will be given high-level requirements and can implement them in a context you see fit.
- **Source Code:** Zip your source code files together and name it **uXXXXXXXX_HW01.zip**, where the XXXXXXXX is your student number, e.g., u12345678_HW01.zip.
- Please make sure to upload the entire project. Refer to **S1L02 Session 02** - Getting started with Visual Studio and GitHub. Ensure that you select, zip and then upload the complete project. This was clearly discussed in the preparation session in the Monday theory class. If only the *.sln is uploaded, then it will be considered and unsubmitted application and a zero will be awarded.
- **Video Demo: DO NOT** Zip your Video Demo. Name the video demo **uXXXXXXXX_HW01.mp4**, where the XXXXXXXX is your student number, e.g., u12345678_HW01.mp4.

NOTES ON DEMO SOFTWARE.

- Change the compression ratio and Frames Per Second (FPS) of the desktop recording software that you choose so that you reduce the file size of your demo. These details have been noted at the top of the homework submission page.
- Use desktop recording software as suggested on the homework submission page.
- **DO NOT use your phone to record your demo** as it is extremely difficult to follow what is going on in your demo. It creates unnecessarily large files making the video demo upload problematic. ClickUP will reject the upload of a file that is larger than 100Mb. Make sure that your files do not reach the 100Mb file size limit. Desktop recording software streamlines the process and creates smaller files that are easier to upload. Phone recordings create files larger than the upload limit, and, as such, will be rejected by ClickUP.
- You can make the video demo with OBS Studio [<https://obsproject.com/>] or ScreenPal (Formerly Screencast-O-Matic) [<https://screenpal.com/>].
- Please make sure to compress your demo as much as possible. You can use free software called Handbrake [<https://handbrake.fr/>] for this.

VIDEO DEMO INSTRUCTIONS.

- Make sure that everything is running when you start recording the video. The video should not be longer than 10 minutes showing the items in the Checklist. When the 10-minute limit is reached, there shall be no further marking. The maximum limit is 10 minutes and no more. The demonstration may be less than 10 minutes as long as all the required details are presented.
- Start by showing the behaviour and functionality of your program. After showing the behaviour and the general functionality, then you should move onto explaining the application code and the reasoning behind your application code.
- Simply scrolling through the program will be considered not presenting a demo. This will equal to you immediately losing 50% of your mark. You need to explain your reasoning. It is part of what is known as “reflective learning” and helps you solve problems when you encounter them during coding. It is extremely useful for the learning process. Do not neglect your own learning process by skipping out on this important step.
- When showing something in the Checklist (rubric) show us your code and explain why you did it in that way. We need an explanation of why you coded the way you coded. This helps us to understand your approach. Everyone approaches programming slightly differently and we need to contextualize ourselves to your approach. If this is not done, then we would not be able to contextualize ourselves to your approach and you will lose marks. The explanation does not need to be too in-depth.
- Refer to the rubric at the end of this document. Sequence your discussion according to the rubric. Consider the rubric as the script for your demonstration.
- If something did not work in your code, in the demonstration explain to us what you wanted to do and what you wanted to achieve with your approach. Explain the reason why you think the code might not be working. Do not be concerned about being right or wrong with the explanation. It gives us insight into your thinking and your reasoning. We will not mark your code that you show us unless you explain this.
- Both the Video Demo and Homework Source Code should be submitted in the correct upload area. If the Video demo or the Homework Source Code is missing, you will get a zero.
- If files are uploaded to the wrong upload area, we will not go and look for the upload. Uploads should be submitted correctly as indicated online. Incorrect uploads will lead to a zero being allocated.
- If you are caught plagiarising or sharing application code, we will give you 0 and you will be reported for plagiarism or academic dishonesty immediately. This is the first and final warning. No further warnings will be issued.

IMPORTANT FINAL NOTE.

- We download your program source code to make sure that you do not upload empty programs. We follow your demonstration by following the application code. We conduct the evaluation side-by-side, with the demonstration playing on one side and the program being considered based on your demonstration on the other side. The demonstration is used to guide us in evaluating your program (as downloaded) and to assign an appropriate mark based on the code and the discussion in your demonstration.
- We have encountered students demonstrating other students’ applications. That is considered academic dishonesty and there are severe consequences if you are found to have committed academic dishonesty. Please refer to the student code of conduct regarding the consequences of academic dishonesty.
- If you assist another student with regards to their assignment, make sure not to share code but rather explain an approach, a thinking process and help other students find potential problems and how to approach solving these problems.
- Remember. This assignment will help you to prepare for the semester test. If you use shared code then you are not preparing for the test and this will have severe consequences for the final result of the test.
- The assignment is on a higher level of difficulty than the test. The level of difficulty is not the same. That being said, by completing the assignment before the test, and understanding the code being applied as well as the required syntax, then you would have prepared.
- Do not try to complete this program in “one sitting”. That would be considered “cramming” for a test and this never works for a programming test. Please be warned regarding this critical misconception. The only way to study and to prepare for a programming test is to do programming and engage in practical activities and tasks.

INF272: Homework Assignment 01 Description



Homework Assignment 01 – Capetropolis Tourism Out of 100 Marks

DUE: 2024-06-03 before 23:59

1. HIGH-LEVEL CASE DESCRIPTION

The island of Capetropolis is a highly sought-after tourist destination. All year-round tourists from different parts of the world visit Capetropolis to enjoy the beauty, attractions and pleasures available on the island. Capemint Holdings is a tourism management company that has a chain of five hotels located in Capetropolis. The management of Capemint Holdings have decided to create/develop an online hotel booking portal that can serve the five hotels. The main objective is to create a one-stop shop that will allow tourist looking for accommodation to make a booking in any of the five hotels based on their preferences (want they want). The booking portal must also allow tourists to order meals depending on their meal preferences, irrespective of the specific hotel where they are residents. The management of Capemint Holdings must be able to track sales performance through a dashboard that enables bookings at individual hotels and total sales and total income for Capemint Holdings from the five hotels to be tracked per time. The online booking portal must allow meal providers (caterers) to register to provide services. Tourists/users must be able to search the list of registered meal providers and place an order to have a meal delivered to their hotel accommodation.

Your task as a developer is to create this online booking portal for Capemint Holdings. The portal must conform to specific requirements outlined in the remaining part of this document.

Once the online portal is implemented, Capemint Holdings will experience a significant boost in tourism in Capetropolis. This, in turn, will lead to improved revenue from all our hotels and their activities, marking a new era of success for our company.

2. STANDARD REQUIREMENTS.

The following is a list of mandatory requirements. Mandatory requirements are not optional. Not complying with the mandatory requirements will lead to a loss of marks.

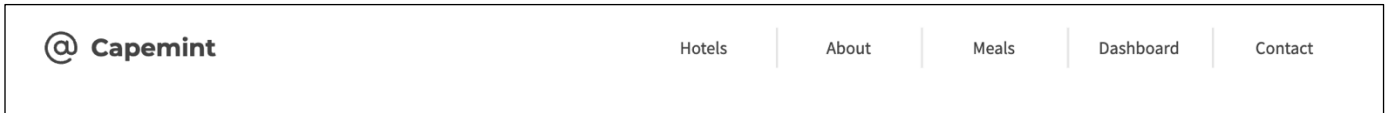
- You must follow the guidance of the wireframes provided to complete the assignment.
- Your assignment does not have to look exactly like the wireframes.
- The wireframes serve mainly as guidance of what we expect from you.
- You are allowed to choose any approach to complete the assignment. You must however stay within the technology parameters specified. Exceeding these parameters will cost you marks.
- Marks will be allocated for the creativity and neatness of your assignment. Not displaying any original thinking in the layout and design of the pages will cost you marks.
- Do not add any functionalities that are not specified.
- The dynamic data in your program must be persistent. You will most likely need to make use of various lists in localStorage and a folder for any images.

3. LIST OF SYSTEM REQUIREMENTS.

3.1. Navigation Bar and main page.

The specifications for the navigation bar are as follows:

- Each screen must contain a navigation bar.
- Five links should be present on the navigation bar: Home, Ride History, and Manage.
- Refer to the following wireframes for an example of what the navigation bar could look like.



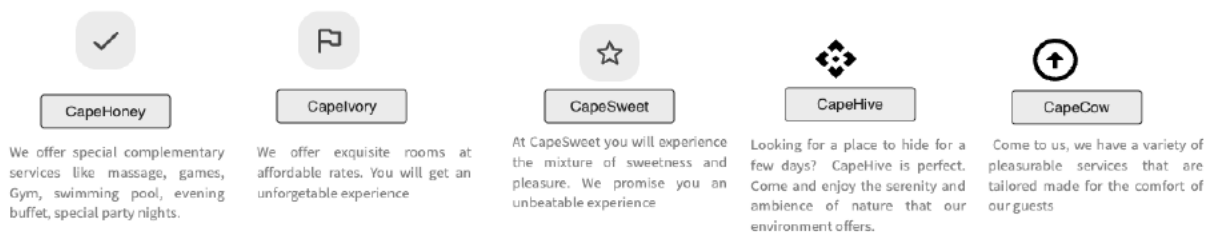
- **Hotels** should take the user to the hotel booking page C
- **About** should take the user to a page that contains an informative narrative (storyline) about Capemint holdings (it is your task to formulate this)
- **Meals** should take the user to a page that allows a meal provider to register. A tourist should also be able to order a meal from here.
- The **Dashboard** should display an informative dashboard (summary report) for administrative purposes.

3.2. Main Page

The main page should emulate Figure 1. You can use any image/logo of your choice to depict the five hotels (CapeHoney, Capelvory, CapeSweet, CapeHive, CapeCow). Also, add a short description of what you imagine Capetropolis could be like and images of it.

The specifications for the Home screen are as follows.

- The main page should emulate Figure 1. You can use any image/logo of your choice to depict the five hotels (CapeHoney, Capelvory, CapeSweet, CapeHive, CapeCow). Also add a short description of what you imagine Capetropolis could like, and images of it. The specifications for the Home screen are as follows:
- This screen will serve as the home page for the application.
- The get started button must unlock (activate) the command button for all hotels which makes booking possible
- Refer to the wireframe in **Figure 1** for some inspiration as to the layout and design of your main page.
- The user will need to select the hotel they want to make a booking for on this page.
- Five clickable options (as displayed in **Figure 1**) should be available.



- When a user clicks on one of the hotels, they should be taken to the Booking Form page, where they must make a booking for the specific hotel selected.

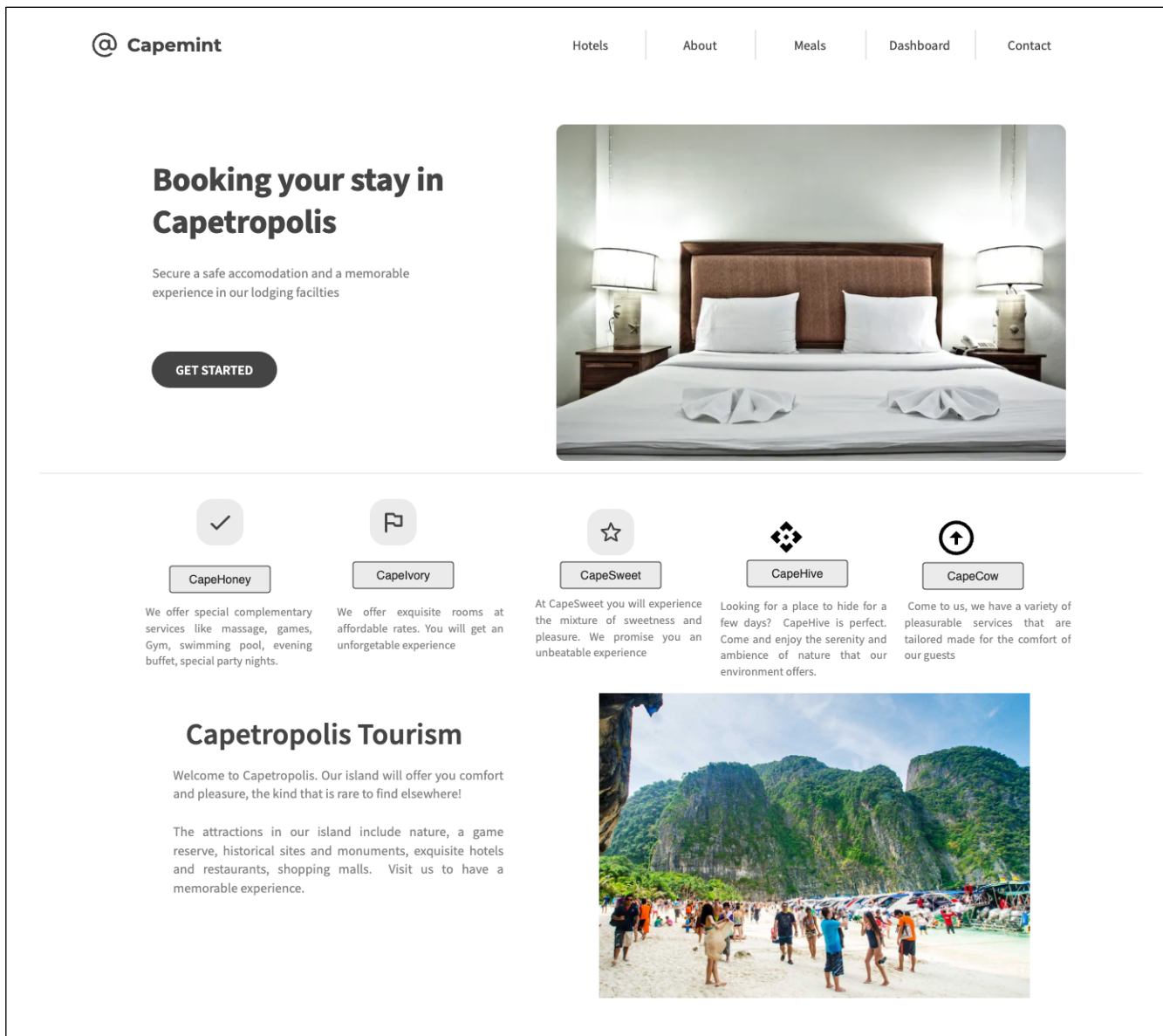


Figure 1. Main page

3.3. Hotel Booking Form

The specifications for the Booking Form screen are as follows:

- The purpose of this screen is to enable the user to book a room at a hotel of choice.
- The title of the specific hotel selected must be displayed on the screen.
- The booking should contain all the input fields present on the wireframe in **Figure 2**.
- The “**Id type**” dropdown should contain the following: { “South African id”, “Passport” }
- The “**Meals req.**” dropdown should contain eight meal types from which the user will select one:
 - { “Regular”, “Vegan”, “Halal”, “Kosher”, “Indian”, “Gluten-free”, “Keto”, “Paleo” }
- **Table 1**. shows the type of meals available in the different hotels.
- The “**Room type**” dropdown must only show the types of rooms that are available in a specific hotel (see **Table 2** for a list of room types in specific hotels).
- The “**price**” and “**features**” must be automatically displayed once the user selects a room type from the “**Room type**” dropdown.

- **NOTE:** An event should be tracked to determine when the user has selected a room type so that the price and features of that room for the particular hotel are displayed. Each hotel should have a booking form that is displayed when the user has selected the hotel.
- When the **Confirm Booking** button is clicked, all the details should be saved as a booking to localStorage, and the user should be taken to the Booking Confirmed screen.
 - **NOTE:** When the booking is created you must save a GUID (globally unique identifier) as the BookingID, and the date and time the booking was made as well. That each booking must have GUID, and be timestamped.
- Refer to the wireframe in **Figure 2** for some inspiration as to the layout and design.

@ CapeCow Logo

Booking form

First name Date of Birth

Surname Address

Initials Telephone

Email address Meals req.

Id type Room Type

Id number Price

Features

Figure 2. An example of booking form for a specific hotel

Table 1. Available meal types in different hotels

Hotel	Regular	Vegan	Halal	Kosher	Indian	Gluten-free	Keto	Paleo
CapeHoney	Yes	Yes	No	No	No	Yes	No	No
Capelvory	Yes	Yes	Yes	Yes	No	Yes	Yes	No
CapeSweet	Yes	Yes	Yes	Yes	No	No	No	No
CapeHive	Yes	Yes	No	No	No	Yes	No	Yes
CapeCow	Yes	Yes	Yes	Yes	Yes	No	No	No

Table 2. Room Types

Hotel	Room type	Price per night (ZAR)	Features
CapeHoney	Standard	R3,000	Breakfast excluded, single bed, TV, internet
	Deluxe	R5,000	Breakfast included, double (2) single beds, TV, internet
	Exquisite	R7,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator
	Executive	R10,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator; laundry, room service, tourism tour of Capetropolis

Capelvory	Single	2,500	Breakfast excluded; single bed, TV, internet
	Double	5,000	Breakfast included, double (2) single beds, TV, internet
	Executive	8,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator; laundry, buffet voucher
CapeSweet	Tasty	3,500	Breakfast included; two single beds, TV, internet
	Comfort	5,000	Breakfast included, two single beds, TV, internet, laundry, free parking
	Executive	9,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator; laundry, free parking, pet allowed, separate dining area
	Diamond	12,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator; laundry, free parking, pet allowed, room service, separate dining area, buffet
CapeHive	Standard	5,000	Breakfast included, single bed, TV, internet, laundry, free parking
	Double	7,000	Breakfast included, double (2) single beds, TV, internet
	Executive	10,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator; laundry, room service, tourism tour of Capetropolis
CapeCow	Budget	2,500	Breakfast excluded; single bed, TV, internet
	Spotlight	4,000	Breakfast included; two single beds, TV, internet
	Deluxe	6,000	Breakfast included, single bed, TV, internet, laundry, free parking, tourism tour of Capetropolis.
	Gold	8,000	Breakfast included, single bed, TV, internet, complimentary drinks in refrigerator, laundry, free parking, tourism tour of Capetropolis.
	Diamond	12,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator; laundry, room service, tourism tour of Capetropolis
	Platinum	15,000	Breakfast included, queen-sized double beds, TV, internet, complimentary drinks in refrigerator; laundry, room service, tourism tour of Capetropolis

3.4 Booking Confirmed

The specifications for the Booking Confirmed screen are as follows:

- As soon as the Booking is made, the user should be taken to the booking confirmation screen.
- All the details related to the Booking should be shown. You can design this booking confirmation screen as you want.
- You must show the **BookingID**, the **date**, and the **time** when the Booking was made.
- The name and logo/image of the hotel in which the Booking was made should be shown on this screen.

3.4. Meals management

3.4.1. Meals Agent Registration

Meals management must support two main functionalities, which are registering as a meal agent (Caterer) and ordering a meal. The specifications for the meals management are as follows:

- From the main page, the user should be able to navigate to a screen that allows a meal agent to register as a service provider to tourists lodged at any of the five hotels under Capemint Holdings.
- The meal agent registration page should support **CRUD** operations i.e. **Create** (add), **Retrieve** (find), **Update** and **Delete**.
- Immediately, a meal agent has been registered by clicking the **"Add"** button. The record should be displayed in the **table** below the **search** button.
- A click on the **Find** button should retrieve the details (a record) of a meal agent that has been registered when the user has provided the first name /surname.
- A click on the **Edit** button should change the colour of a retrieved button to "red" and allow changes to be made to an existing record.
- A click on the **Update** button should enable recent changes to a record to be saved and reflected in the table.
- A click on the **Delete** button should remove a record from the table.

- An entry into the **Search** button should filter the content of the table, ensuring that only the record that matches the search is shown in the table.
- Refer to the wireframe in **Figure 3** for some inspiration as to the layout and design.

@ Capemint

Meal Agents - Caterers

First name

Surname

Email address

Telephone

Meal Services

☒ Regular
 ☒ Kosher

☒ Vegan
 ☒ Keto

☒ Indian
 ☒ Paleo

☒ Halal
 ☒ Gluten-free

Add

Find

Edit

Update

Delete

Name + Surname	Email	Telephone	Meal Services

Figure 3. Meals agent registration form

3.4.2 Meal order form

The specifications for the meal order booking form are as follows:

- The purpose of this screen is to enable the user to order a meal from a meal agent.
- When the form is loaded (displayed). A list of registered meal agents should be displayed in the table (see Figure 4).
- The information of the user ordering the food must be captured on the form. (see Figure 4).
- The meal type dropdown should contain the eight different meal types: { "Regular", "Vegan", "Halal", "Kosher", "Indian", "Gluten-free", "Keto", "Paleo"}.
- The meal agent must be one of those registered and listed in the table.
- The description of the meal being requested (meal details) should be provided.
- The hotel residence dropdown should display a list of the five hotels under Capemint Holdings.
- A click on the **Order Meal** button should ensure that the booking is made (stored). When an order is made, you must save a GUID (globally unique identifier) as the **OrderID** and the date and time the booking was made.
- Refer to the wireframe in **Figure 4** for some inspiration as to the layout and design.
- After a meal has been ordered, the user should be able to see the meal order by clicking the **View order** button as a confirmation.
- You can design this **View order** screen as you want. However, you must show the **OrderID** and the **date** and **time** the order was made.

@ Capemint

Order a Meal

Date: 24-04-2024:

First name

Placeholder

Meal details

Describe your here here

Surname

Placeholder

Meal type

Select

Hotel residence

Select

Meal agent

Placeholder

Order meal

View oder

Meal providers

Name + Surname	Email	Telephone	Meal Services

Figure 4. Meal order form

3.6. Administration dashboard reporting

The administrative dashboard should have the following:

- Show the total number of bookings (addition of all bookings in the five hotels of Capemint Holdings).
- Show the total number of guests (i.e., the sum of guests lodged in the five Capemint Holdings hotels).
- Show the total income (addition of sales of the five hotels of Capemint Holdings).
- The sale performance table should show the number of bookings and total sales for each of Capemint Holdings's five hotels.
- The customers' record table should show details of customers (including booking date & time, phone number, hotel, room type, number of persons in the room, and amount paid) that have lodged in each of the five hotels of Capemint Holdings. (**Hint:** these data are expected to be retrieved from local storage).
- You must be able to export the content of customers table to a text file on an Export button click. Make sure the demo data makes use of different names and variations of names and hotels.
- Refer to the wireframe in **Figure 5** for some inspiration as to the layout and design.

Total bookings



32

Total Income



R22,500.00

Total Guests








50

April 2024



Current Date: 22-04-2024

Sales Performance

Hotel	Number of booking	Total Sales (R)	Logo
Capecow	5	25000	
Capesweet	4	20000	
Capehoney	12	35000	
Capeivory	8	23000	
Capehive	6	15000	

Export Table

Customers



Name	Booking Date	Phone Number	Hotel	Room Type	Number of guests in the room	Amount
Jerry Mattedi	19 April, 2024 : 10:10 AM	27-713-220-521	Capecow	Deluxe	2	R5,000
Elianora Vasilov	18 April, 2024 : 3:12 PM	171-534-330-1262	Capesweet	Executive	1	R20,000
Alvis Daen	19 April, 2024 : 2:15 PM	233-661-440-5110	Capehoney	Regular	2	R6,000
Lissa Nkosimo	17 Apr, 2024 : 1:15 PM	27-761-210-304	Capeivory	Standard	1	R8,000

Figure 5. Administrative Dashboard

INF272: Homework Assignment 01 Rubric



Homework Assignment 01 – Capetropolis Tourism Out of 100 Marks

DUE: 2024-06-03 before 23:59

REQUIREMENT [Marking]	DYSFUNCTION	PARTIAL	MAX
<ul style="list-style-type: none"> Demonstration should be in the sequence of the rubric. Demo time refers to the time allocated for each requirement being demonstrated. Plan your time allocation carefully. When the demo reaches 10 minutes, marking will stop. 			
1. Requirement 1 – Functionality [Demo time: 0.5 minutes].	0	5	10
1.1. Start by showing the functionality of the program.	0	2.5	5
1.2. Run the program and then go through one complete action.	0	2.5	5
2. Requirement 2 – Navigation and Main Page Functionality [Demo time allocation = 1 minute].	0	6	12
2.1. Navigation bar is present and active as a functionality.	0	2	4
2.2. The GET STARTED button activates (enables) the hotel selection functionality.	0	2	4
2.3. Overall design and layout of screen.	0	2	4
3. Requirement 3 – Select a Hotel Functionality [Demo time: 1 minute].	0	2.5	5
3.1. five clickable options' functionality.	0	1.5	3
3.2. Overall design and layout of screen.	0	1	2
4. Requirement 4 – Booking Form Screen Functionality [Demo time: 2 minutes].	0	7.5	15
4.1. Title and logo/image of hotel are displayed.	0	1	2
4.2. All the Input fields functionality are working.	0	1	2
4.3. The dropdown options displayed for Id Type, Meal Req. and Room type are correct. The dropdown functionality are working.	0	1.5	3
4.4. The price and features of a room are displayed when a room type has been selected from the dropdown.	0	2	4
4.5. The Confirm booking button functionality is working	0	1.5	2
4.6. Overall design and layout of screen.	0	1	2
5. Requirement 5 – Hotel Booking Confirmed Functionality [Demo time: 1.5 minutes].	0	5	10
5.1. Booking confirmation details correct.	0	2	4
5.2. The booking details including the BookingID, time and date, and the logo/image of the hotel are correct.	0	2	4
5.3. Overall design and layout of screen.	0	1	2
6. Requirement 6 – Meal Agents Registration Functionality [Demo time: 1 minute].	0	6	12
6.1. CRUD of Meal Agent	0	3	6
6.2. List of meal agents and their details are correct	0	1	2
6.3. Search functionality.	0	1	2
6.4. Overall design and layout of screen.	0	1	2
7. Requirement 7 – Meal Order Form Functionality [Demo time: 1 minute].	0	6	12
7.1 CRUD of Meal Order	0	3	6
7.2 List of meal orders in the table	0	1	2
7.3 View order functionality	0	1	2
7.4 Overall design and layout of screen.	0	1	2
8. Requirement 8 – Administrative Dashboard Functionality [Demo time: 3 minutes].	0	12	24
8.1. Total bookings, guests, income correctly retrieved.	0	3	6
8.2. Sale performance functionality.	0	3	6

8.3. List of customers.	0	3	6
8.4. Export functionality.	0	1	2
8.5. Overall design and layout of screen.	0	2	4
TOTAL MARK ALLOCATION [Maximum Demo time available: 10 minutes].	0	50	100