Title

Room Management Module within the Resort Complex Management System.

Background

- ATZ Resort Complex is a hospitality facility offering accommodation, dining, and wellness services. The complex requires a software system to manage its service operations. You are assigned to develop a module responsible for managing rooms and guest stays, featuring 10 functions as detailed in the technical specification table below.

- Students are required to analyze and design the program using an object-oriented programming (OOP) approach. Features such as abstraction, polymorphism, encapsulation, and inheritance must be applied during the development process.

Program Specifications

Build a management system for Room Management Module with the following functionalities:

1. Import Room Data from Text File

2. Display Available Room List

3. Enter Guest Information

4. Update Guest Stay Information

5. Search Guest by National ID

6. Delete Guest Reservation Before Arrival

7. List Vacant Rooms

8. Monthly Revenue Report

9. Revenue Report by Room Type

10. Save Guest Information

Others- Quit.

Features:

Software Requirements Specification (**SRS**) Module:
Room and Guest Management System – ATZ Resort Management Software

This system contains the following functions:

- **Function 1: Import Room Data from Text File - 50 LOC**

    Description:
    The current module is provided with a list of service rooms exported by another module; therefore, this function is responsible for reading data from a plain text file named *"Active_Room_List.txt"*.

    Input Requirements:
    - File path to the *"Active_Room_List.txt"* file.

- **File format**: .txt, each line represents one room.
- **Line format**: RoomID; RoomName; RoomType; DailyRate; Capacity; FurnitureDescription.
- **Example**: `R101; Sunset Suite; Deluxe; 150.00; 4; Double bed, Sofa, TV, Balcony`

Validation Rules:

1. File must exist
2. All six fields must be provided per line.
3. RoomID must be unique.
4. DailyRate must be a positive decimal number.
5. Capacity must be a positive integer

Operation Workflow:

1. Check file existence.
2. File is read line-by-line.
3. Each line is validated and parsed.
4. Valid entries are saved; invalid ones are logged

Sample Output

10 rooms successfully loaded.
2 entries failed.

▪ Function 2: Display Available Room List – 50 LOC

Description:
Displays all currently available room data in tabular format. If data is not available, appropriate notification should be given.

Input Requirements:

o No external input required.

Validation Rules:

o Room list must not be empty.

Operation Workflow:

1. User selects "**Display Available Room List** ".
2. System retrieves all room records
   o If the data exists:
     ▪ Data is formatted and presented to the user
   o If the data does not exist:
     ▪ Display the message: **" Room list is currently empty, not loaded yet."**
3. Data is formatted and presented to the user

Sample Output

```
Active Room List

RoomID | RoomName        | Type      | Rate   | Capacity | Furniture
-------+----------------+----------+-------+----------+----------------------
R101   | Sunset Suite    | Deluxe    | 150.0 |        4 | Double bed, Sofa, TV
R102   | Garden View     | Standard  | 90.0  |        2 | Twin beds, Desk
......
```

▪ Function 3: Enter Guest Information – 50 LOC

Description:

Records a new guest's information for room rental.

Input Requirements

- o National ID number
- o Full name
- o Birthdate
- o Gender
- o Phone number
- o Desired room ID
- o Number of rental days
- o Start date
- o Name of co-tenant

Validation Rules

- o National ID number: *Includes 12 digits and must be unique*
- o Full name: *between 2 and 25 characters long and must start with a letter*
- o Birthdate: *This value represents a date of birth*
- o Gender: *Represents only one of two values indicating male or female gender*
- o Phone number: *A 10-digit number matching the formats of mobile carriers in Vietnam.*
- o Desired room ID: *Up to 5 characters, starting with a letter followed by digits*
- o Number of rental days: *Must be a positive integer*
- o Start date: *Must be a future date*
- o Name of co-tenant *(optional)*

Operation Workflow:

1. User selects "Enter Guest Information "
2. System prompts for input fields:
3. Inputs are validated**.**

4. Guest is assigned to room

5. Return to Main Menu:

   o   After displaying the results or the message, prompt the user to return to the main menu.

Sample Output:

Guest registered successfully for room R102

Rental from 2025-05-08 for 3 days

▪ **Function 4: Update Guest Stay Information – 50 LOC**

Description:

Modifies existing guest rental details.

Input Requirements

   o   National ID number

   o   Updated fields (e.g., rental days, co-tenant name, phone number)

Validation Rules

   o   Guest must exist in the system.

   o   Updated data must follow original format validations.

Operation Workflow:

1. User selects "Update Guest Stay Information":

2. Retrieves and displays current info

3. Search guest by national ID :

   o   **If the "natioanal ID" is existed**:

     ▪   Retrieves and displays current info.

     ▪   User submits changes

     ▪   System validates and saves updates if valid

     ▪   Display success message

   o   **If it does not exist**:

     ▪   Display the message: **"No guest found with the requested ID !"**.

4. Return to Main Menu:

   o   After displaying the results or the message, return to the main menu.

Sample Output:

   Case 1: the "National ID**" is existed**

   Guest information updated for ID: 123456789

   Case 2: the "National ID**" does not exist**

   No guest found with the requested ID !.

- **Function 5: Search Guest by National ID – 50 LOC**

Description:

Retrieves guest rental data using National ID ...

Input Requirements:

- o National ID number

Validation Rules:

- o Must be a valid format.

Operation Workflow:

1. User selects "**Search Guest by National ID**"
2. System prompts for ID input and validates
3. **Search by ID in the guest list:**
   - o **If the ID already exists**:
     - System locates and displays guest info
   - o **If does not exist**:
     - Display message "**No guest found with the requested ID !**".
4. **Return to Main Menu:**
   - o After displaying the message, prompt the user to return to the main menu or continue with place another order.

Sample Output:

Case 1: National ID exists in the guest list

```
----------------------------------------------------------------
Guest information [National ID: 031066501254]
----------------------------------------------------------------
 Full name      : Thuy, Hoang Thi Thu
 Phone number   : 0902331127
 Birth day      : 01/04/1999
 Gender         : Female
----------------------------------------------------------------
 Rental room    : R105
 Check in       : 29/04/2025
 Rental days    : 3
 Check out      : 02/05/2025
----------------------------------------------------------------
Room inforamtion:
 + ID       : R105
 + Room     : Garden View
 + Type     : VIP
 + Daly rate: 270$
 + Capacity : 2
 + Funiture : Double bed, dressing table, TV, sea view
```

Case 2: National ID does not exist in the guest list

```
----------------------------------------------------------------
No guest found with the requested ID '031066501254'
```

- **Function 6: Delete Guest Reservation Before Arrival – 50 LOC**

Description:
Deletes guest booking if stay has not yet started.

Input Requirements:

- o National ID number.

Validation Rules:

- National ID is valid.
- Booking must exist.
- Rental must not have started

Operation Workflow:

1. User selects "Cancel Booking"
2. Prompt the user to enter the **National ID**.
3. Check if the National ID exists in the guest list:
   - o If the ID exists and the start date is in the future:
     a. Ask the user to confirm the room booking cancellation.
     b. Cancel the guest's registration and save the change to the database.
   - o If the ID is valid but the start date is in the past or today:
     - Display the message: **" The room booking for this guest cannot be cancelled "**
   - o If the ID does not exist:
     - Display the message: **"Booking details for ID … could not be found."**
4. Ask the user whether to continue with another delete or return to the main menu

Sample Output:

Case 1: If the ID exists but the start date is on or before April 26, 2025
```
----------------------------------------------------------
Guest information [National ID: 031066501254]
----------------------------------------------------------
 Full name      : Thuy, Hoang Thi Thu
 Phone number   : 0902331127
 Birth day      : 01/04/1999
 Gender         : Female
----------------------------------------------------------
 Rental room    : R105
 Check in       : 29/04/2025
 Rental days    : 3
 Check out      :
----------------------------------------------------------
Room inforamtion:
 + ID           : R105
 + Room         : Garden View
 + Type         : VIP
```

```
                + Daly rate: 270$
                + Capacity : 2
                + Funiture : Double bed, dressing table, TV, sea view
                -------------------------------------------------------------
                Do you really want to cancel this guest's room booking ? [Y/N]:Y
                ... System message ...
                "The booking associated with ID 12345 has been successfully canceled."
```

Case 2: National ID does not exist in the guest list

```
                -------------------------------------------------------------
                Booking details for ID '031066501254' could not be found
```

Case 3: The ID exists but the start date is on or before April 29, 2025

```
                -------------------------------------------------------------
                The room booking for this guest cannot be cancelled !
```

▪ Function 7: List Vacant Rooms – 50 LOC

Description:

Displays rooms that are currently not rented.

Input Requirements

- None

Validation Rules

- System checks current rental status of all rooms.

Operation Workflow:

1. User selects "**List Vacant Rooms**":

2. System queries and filters unoccupied rooms

   - If the data exists: Data is formatted and presented to the user

   - If the data does not exist: Display the message:

     "All rooms are currently rented out — no availability at the moment !."

3. **Return to Main Menu**:

   - After saving the data, return to the main menu.

Sample Output

Case 1: If the list contains any available rooms

Available Room List

```
RoomID | RoomName       | Type      | Rate  | Capacity | Furniture
-------+----------------+-----------+-------+----------+---------------------
R101   | Sunset Suite   | Deluxe    | 150.0 |        4 | Double bed, Sofa, TV
R102   | Garden View    | Standard  |  90.0 |        2 | Twin beds, Desk
......
```

Case 2: If there are no available rooms in the list

```
All rooms have currently been rented out; no rooms are available
```

- **Function 8: Monthly Revenue Report – 50 LOC**

Description:
Summarizes income by month from all room rentals.

Input Requirements
  - Target month (MM/YYYY)

Validation Rules
  - Month must be valid and within system's data range.

Operation Workflow:

1. User selects "**Monthly Revenue Report**"
2. Inputs target month:
   - Verify that the month and year provided for the statistics are valid.
3. **Check the list of room guests and display data:**
   - If the list contains entries:
     - The system calculates total revenue and displays the results with room attributes:
       - Room ID
       - Room name
       - RoomType
       - DailyRate
       - Amount
   - If the list does not contain entries:
     - Display the message: **"There is no data on guests who have rented rooms."**
4. **Return to Main Menu:**
   - After handling, return to the main menu.

Sample Output:

Case 1: List Containing Data

Monthly Revenue Report - 'MM/YYYY'

```
----------------------------------------------------------------
RoomID  | Room Name       | Room type | DailyRate |  Amount
----------------------------------------------------------------
R101    | Sunset Suite    | Deluxe    | 150.00    |    450.00
R102    | Ocean Breeze    | Standard  |  90.00    |     90.00
R117    | Overflow Room   | Suite     | 220.00    |   3300.00
...
----------------------------------------------------------------
```

Case 2: List does not contain entries

```
There is no data on guests who have rented rooms
```

- **Function 9: Revenue Report by Room Type – 50 LOC**

Description:
Shows income grouped by room type.

Input Requirements

- Target room type

Validation Rules

- If month is provided, must be valid.

Operation Workflow

1. User selects "**Revenue report by Room Type**".
2. Validate the roomType value
3. If the value is valid

    - The system aggregates data and displays results with room attributes:

        - RoomType
        - Amount

4. If the value is not valid, display a message "Invalid room type !." And let the user re-enter
5. Aggregates data and displays results.

Sample Output

```
Revenue Report by Room Type
---------------------------
  Room type  | Amount
--------------------------
  Deluxe     |    $450,050
  Standard   |     $90,000
  Suite      |  $1,200,000
  Superior   |    $650,500
```

- ## Function 10: Save Guest Information – 50 LOC

Description:
This function will allow saving guest list data in the system, into the files **guestInfo.dat**.

Operation Workflow:

1. **Data Collection:**

    - Gather all current data from the program, including guest details
      (National ID, full name, phone, …).

2. **Serialization**:

    - Convert the data into an object format suitable for file storage as a binary object file.

3. **Save to File**:

    - Write the serialized data to a file. The file should be named appropriately
      (e.g., **guestInfo.dat**).

4. **Confirmation Message**:

    - Display a confirmation message once the data is successfully saved.

5. **Return to Main Menu**:

- o   After saving the data, return to the main menu.

Sample Confirmation Message:

- Guest information has been successfully saved to **"guestInfo.dat".**

The above specifications provide basic information. You are required to conduct a detailed requirements analysis and build the application based on the real-world requirements.

The lecturer will explain the full set of requirements only once during the initial slot of the assignment.