Capgemini

HOTEL RESERVATION/MANAGEMENT SOFTWARE

Low-Level Design Version Draft v0.2

Document Control:

Project Revision History			
Date	**Version**	**Author**	**Brief Description of Changes**
15-01-2023	2	Group 5	User authentication added in low level design

Team members

Employee id	Name
46282418	Arveti Chandana Gayathri
46282420	Sian Gijo
46282421	Manisha L
46290138	Kappa Latha
46290139	Karishma Shaik
46290156	Anusha Nallam

Table of Contents

I) Low-Level Design

Low-Level Design

1. Introduction

The aim of this document is to gather, analyze and give an in-depth insight into the complete Hotel Reservation/Management software by defining the problem statement in detail.The intended audience include developers,project managers, hotel managers etc. The detailed low-level design of the Hotel Reservation System is provided in this document.

1.1 Purpose

The purpose of this document is to describe the low-level design flow of the Hotel Management/ Reservation software.

1.2 Intended Audience and Reading Suggestions

The document is primarily intended for development team and project team members.

1.3 References

The references are:

1. System Requirements Specification Document

2. Detailed System Design

2.1 Design Descriptions:

This project design mainly focuses on implementing the Hotel Management/ Reservation system. Users will be able to choose services given by the hotel. Reservation of rooms of a particular type can be made by requesting the server that checks its availability and initiates the booking.

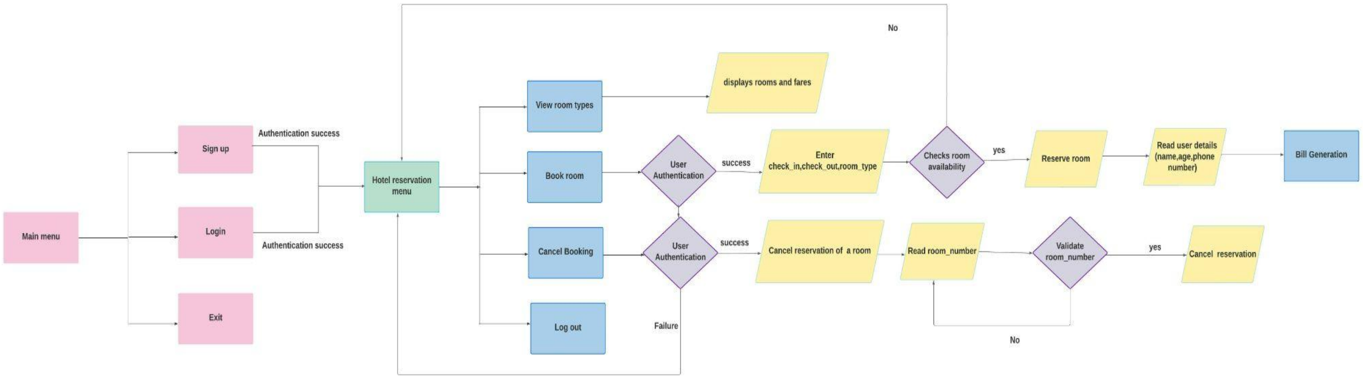
User Authentication menu:

The user will be displayed a menu to login or sign up. If the user is new to the system , he/she should sign up by setting up the username and password. The user authentication details will be stored in a file. If the user is an existing user, he/she should login by entering the username and password which will be authenticated and the next main menu will be displayed.

Main menu:

- The main menu has options to view rooms, book rooms, cancel rooms and quit the system.
- Depending on the choice entered by the user, corresponding functions are performed
- For choice 1, the server will display the type of rooms, price and services provided by the hotel to the user.
- For choice 2, the client will provide the room type, check in and check out date to book a room, The server receives this information and checks if the room of that particular type is available, If available the server takes confirmation from the user and books the room. The user will also be able to add-on the services provided by the hotel. The server then generates the bill and displays it to the user. The server will also store the user details in a file which maintains the booking history.
- For choice 3 , the user will be able to cancel the booked room, The client sends the room number and username to the server, the server will deallocate the room for that user and send a message to the client.
- For choice 4 , the client quits from the system.

`2.2 Low Level Design



2.3 Modules

2.3.1 client .c

main() :

Name	**main()**			
Input	Parameter Name :	NA	Initial value:	NA
Output	Return value type:	int	-	-
Description	The main function will perform socket creation and connection to the server. Display user authentication menu and hotel reservation system menu to the user			
Pseudo Code	1. Display the main menu 2. Takes user's option 3. After successful user authentication it will move to display_hotel_menu function			

2.3.2 **display_hotel_menu()** :

Name	**display_hotel_menu()**			
Input	Parameter Name:	NA	Initial Value:	NA
Output	Return value type :	int	-	-
Description	This function is called user authentication is successful and it displays the options to view room type, book room, cancel room and log out			
Pseudo Code	1. Display the hotel reservation menu 2. Takes user’s choice 3. Navigates to respective functions based on the user’s choice.			

2.3.3 **display_room_details()** :

Name	**display_room_details()**			
Input	Parameter Name:	NA	Initial Value:	NA
Output	Return value type :	NA	-	-
Description	This program is used to view room types such as single ,double, deluxe, suite along with the fare per each day.			
Pseudo Code	1.Displays the hotel sapphire. 2.User can view the room types as per their choice. 3.Enters into respective functions to proceed further in reserving a room.			

2.3.4 book_rooms()

Name	**book_rooms()**			
Input	Parameter Name	NA	Initial value:NA	-
Output	Return value type	int	-2 if the client closes	-1 if the booking information is invalid
Description	Reads the check in, check out date and room type that the user wishes to book and checks the availability of that room type from the server and confirms booking after user authentication.			
Pseudo Code	1. Prompt user to enter username and password for authentication. 2. Read the check in and check out date and required room type. 3. Requests the server to confirm the availability of the required room type 4. If available, move to generate_ticket() function.			

2.3.5 generate_ticket()

Name	**generate_ticket()**			
Input	Parameter Name	msg_t* recv msg_t recv1	Initial value:	*recv :
Output	Return value type	void	NA	-
Description	Prompts the user to enter the user’s information and reads the room details of the user from the structure . Calculates the fare of the room and displays the bill to the user.			
Pseudo Code	1.Prompts the user to enter the user's information. 2. Stores the room information into valiables 3. Navigate to calculate_fare function which calculates fare of the room booked according to the number of days of stay 4. Displays the bill to the client.			

2.3.5 cancel_room()

Name	**cancel_room()**			
Input	Parameter Name	NA	Initial value:NA	-
Output	Return value type	NA		-
Description	Reads the user’s name and password to authenticate and after authentication is successful reads the room_no of the room that is to be cancelled , the information is sent to the server through the socket			
Pseudo Code	1. Prompts the user to user name and password 2. Authenticates the user 3. If successful reads the user’s room_no that is to be deleted 4. If unsuccessful returns -2 5. Process the information read to the server through socket.			

2.5 Structure used

- **Message Data structure was created in a common header file for all communications between client and server.**

```
typedef struct
{
  msg_type_e  msg_type;
  int         msg_int;
  char        msg_data[MAX_DATA_LEN];
  char        msg_add_data[MAX_DATA_LEN];
  char        msg_add1_data[MAX_DATA_LEN];
  char        msg_add2_data[MAX_DATA_LEN];

} msg_t;

typedef struct
{
  void        (*msg_action_function) (int, msg_t *);

} msg_action_t;
```

2.3 Modules

2.3.1 server.c

main()

Name	main			
Input	Parameter Name	NA	Initial value	NA
Output	Return value type	int	-	-
Description	The main function will perform socket creation and the socket listens until a client gets connected to the server .After successful connection , a thread is created for the client which is connected.			
Pseudo Code	1.After a client is connected,a thread is created for the client 2.That thread will go to the process_client_messages()			

2.3.2 handle_view_room()

Name	handle_view_room			
Input	Parameter Name	int sockfd msg _t *msg	Initial value	NA
Output	Return value type	Msg		
Description	This function open the room details file and sends the rooms details and fares from server to client			
Pseudo Code	1. Here the server open the room details file and sends the room details server to the client 2. After completion of it will pass some message			

2.3.3 handle_book_room ()

Name	handle _book_room()			
Input	Parameter Name	int sockf msg_t *msg	Initial value:	-
Output	Return value type	int	-	-
Description	This function books the rooms according to the user's choice and checks the availability of a particular room type and displays the total occupied and available rooms and allocates the room to the user.			
Pseudo Code	1.Reads the check_in,check_out time from the user, the server asks the user to select the room type. 2.Server checks the availability of the room using the flag. 3.Displays the total occupied and available rooms. 4.If available goes to the handle_services().			

2.3.4 handle_cancel_room()

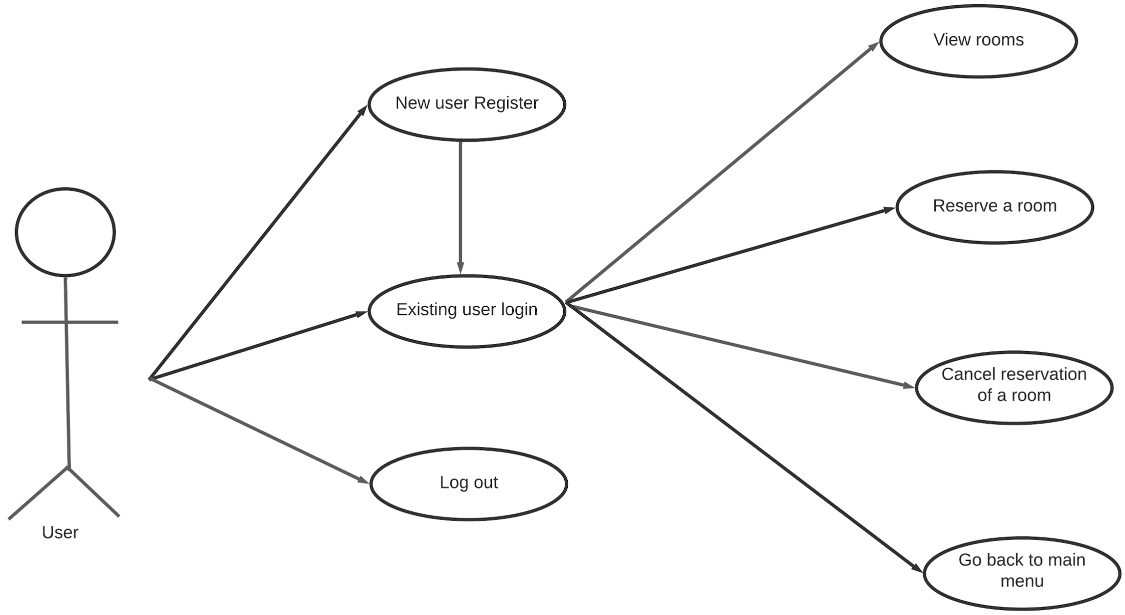
Name	handle_cancel_room()			
Input	Socket fd Message structure msg	int struct msg_t	Initial value:NA	-
Output	Return value type	void		-
Description	Handles the cancel booking request message sent by the client. Checks the room number provided by the user and confirms that it is booked, then it is made available. If not already booked by the user, sends a response error message to the client that cancellation request is invalid.			
Pseudo Code	1. Handle the request message sent by client 2. Check the room number provided by the user. 3. If the room number provided matches with the room number and corresponding username stored in the structure, the room is made available again and hence canceled. 4. If not validated, the server sends an error response to the client saying cancellation request is invalid.			

2.5 Structure used

User's information and the room types present in the hotel structure is created in server header file to store the data.

```
typedef struct user_info
{
  char uname[32];
  char pwd[8];
  struct user_info *next;

} user_info_t;

typedef struct single_room
{
  int room_no;
  bool available;
  char check_in[10];
  char check_out[10];
  char uname[32];

} single_t[4];

typedef struct double_room
{
  int room_no;
  bool available;
  char check_in[10];
  char check_out[10];
  char uname[32];

} double_t[4];

typedef struct deluxe_room
{
  int room_no;
  bool available;
  char check_in[10];
  char check_out[10];
  char uname[32];
} deluxe_t[4];

typedef struct suite_room
{
  int room_no;
  bool available;
  char check_in[10];
  char check_out[10];
  char uname[32];

} suite_t[4];
```

2.4 Use case Diagram


User
New user Register
Existing user login
Log out
View rooms
Reserve a room
Cancel reservation of a room
Go back to main menu