

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590 018



A Mini -Project Work on

“Cruise Ship Reservation System”

A Dissertation work submitted in partial fulfillment of the requirement
for the award of the degree

Bachelor of Engineering
In
Information Science & Engineering

Submitted by

Ravi Shankar Roy
Reetik Raj

1AY18IS151
1AY19IS075

Under the guidance of
Prof. Arshiya Mubeen
Assistant Professor



DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING
ACHARYA INSTITUTE OF TECHNOLOGY

(AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI. APPROVED BY AICTE, NEW DELHI,
ACCREDITED BY NAAC, NEW DELHI)

Acharya Dr. Sarvepalli Radhakrishnan Road, Soldevanahalli, Bengaluru-560107

2021-22

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

ACHARYA INSTITUTE OF TECHNOLOGY

(AFFILIATED TO VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI. APPROVED BY AICTE, NEW DELHI,
ACCREDITED BY NAAC, NEW DELHI)

Acharya Dr. Sarvepalli Radhakrishnan Road, Soldevanahalli, Bengaluru-560107



Certificate

This is to certify that the Mini-Project work entitled **"Cruise Ship Reservation System"** is a bonafide work carried out by **Reetik Raj(1AY19IS075)** and **Ravi Shankar Roy(1AY18IS151)** in partial fulfillment for the award of the degree of **Bachelor of Engineering in Information Science and Engineering** of the **Visvesvaraya Technological University, Belagavi** during the year 2021-22. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The Project has been approved as it satisfies the academic requirements in respect of Project work prescribed for the Bachelor of Engineering Degree.

Prof. Arshiya Mubeen
Guide

Dr. Chayapathi A R
HOD

Name of the Examiners

1. _____
2. _____

Signature with date

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of this mini-project would be incomplete without the mention of the people who made it possible through constant guidance and encouragement.

We would take this opportunity to express our heart-felt gratitude to **Sri. B. Premnath Reddy**, Chairman, Acharya Institutes and **Dr. Rajath Hegade**, Principal, and **Prof. C K Marigowda**, Vice Principal, Acharya Institute of Technology for providing the necessary infrastructure to complete this mini-project.

We wish to express our deepest gratitude and thanks to **Dr. Chayapathi A R**, Head of the Department, Information Science and Engineering.

We wish to express sincere thanks to my guide **Prof. Arshiya Mubeen**, Assistant Professor, Department of Information Science and Engineering for helping us throughout and guiding us from time to time.

A warm thanks to all the faculty of Department of Information Science and Engineering, who have helped us with their views and encouraging ideas.

ABSTRACT

The aim of the project is to book and cancel tickets of cruise ships. It provides facility to the passenger to book tickets based on the cruise ship number. The passenger can book multiple seats. The passenger has the option to view all the cruise ships and their timings.

The concept of variable length record, is used to store the record in the file. The program provides the user with a booking and cancel option. It also has a console for the administrator of the system where he can add new ship, delete existing ship and also view the entire list of booking done by the passengers.

The basic need of the software is to save the time of the user with the help of file structure. And also maintain collection of data in the computer systematically so that it's easy to understand.

TABLE OF CONTENTS

Acknowledgement	i
Abstract	ii
Chapter 1: Introduction	1
Chapter 2: Hardware and Software Requirements	4
Chapter 3: System Overview	5
3.1:Problem Definition	5
3.2:Sequential Search	6
Chapter 4: Design	7
4.1:Data Flow Diagram	7
4.2:Class Diagram	8
Chapter 5: Implementation	9
Chapter 6: Testing	10
Chapter 7: Snapshots	12
Conclusion and Future Enhancement	16
Bibliography	17

LIST OF FIGURES AND TABLES

Figures	Page No
Fig 1.1: Data Flow Diagram	11
Fig 1.2: Class Diagram	12
Fig 7.1: Introductory page	17
Fig 7.2: View information page	18
Fig 7.3: Book ticket page	19
Fig 7.4: Cancel ticket page	20
Fig 7.5: Admin password verification page	20
Fig 7.6: Admin page	21
Fig 7.7: Passenger booklist	21
Fig 7.8: Add ship page	22
Fig 7.9: Ship delete page	22
Fig 7.10: Exit page	23
 Tables	
Table 1.1	15
Table 1.1	16

CHAPTER 1

INTRODUCTION

1.1 Introduction to File Structures

File Structures is the Organization of Data in Secondary Storage Device in such a way that minimize the access time and the storage space. A File Structure is a combination of representations for data in files and of operations for accessing the data. A File Structure allows applications to read, write and modify data. It might also support finding the data that matches some search criteria or reading through the data in some particular order.

“**File organization**” refers to the logical relationship among the various records that constitute the file, particular with respect to means of identification access to any specific record. “File structure” refers to the format of label and data blocks and of any logical record control, information. The organization of the given file maybe sequential relative, or indexed.

Co-Sequential Processing

Co-sequential Processing is the coordinated processing of two or more sequential lists to produce single output list. Sometimes the processing results in a merging, or union, sometimes it results in a matching, or intersection and other times the operation is a combination of matching and merging of the items in the lists.

Aim of the File Structure

The need for file structures

- Ideally, we would like to get the information we need once access to the disk
- If it is impossible to get what we need in one access, we want structures that allow us to find the target information with as few accesses as possible.
- We want our file structures to group information so we are likely to get thing We need with only trip to the disk.

1.2 Fundamental Operations on File

Open: A function or system call that makes a file ready for use. It may also bind a logical filename to a physical file. Its arguments include the logical filename, the physical filename and may also include information on how the file is to be accessed.

Close: A function or system call that breaks the link between the logical filename and the corresponding physical filename.

Create: A function or system call that causes a file to be created on secondary storage and may also binds a logical filename to the file's physical filename.

Read: A function or a system call used to obtain input from a file or device. When viewed at the lowest level, it requires three arguments: a source file logical name corresponding to an open file; the destination address for the address and the size or amount of data to be read.

Write: A function or system call used to provide output capabilities. When viewed at the lowest level, it requires three arguments: a destination filename corresponding to an open file; the source address of the bytes that are to be written and the size or amount of data to be written.

Seek: A function or a system call that sets the read/write pointer to a specified position in a file. Languages that provide seeking functions allow programs to access specific.

1.3 Services provided to the user

The functions that the Source Code Storage System provides are as follows:

1. INSERT: This operation is performed when new data needs to be added to the system, for e.g. when developer creates a new source code, the source code entry is inserted in the files. This option has choices:

- a) Source code: This choice allows entering newly created source code into the files.
- b) Sign up: This will create new developer account. This option is selected when a developer is enrolled in the Source Code Storage System.

2. DELETE: This operation clears the existing records in the various files. It is used when for e.g. a member leaves Source Code Storage System or when source code file is to be deleted is disposed of from system.. But care must be taken while performing this operation and permission taken from the author of files because the system could lose any important data.

3. SEARCH: This function is used to search particular data from the System. This function can search for data related in the entities:

a) Source code: To search for a particular source code, to know whether it is currently available in System or not. This can be done by entering name of the file.

b) Login id: This will find out the particular developer who possesses the particular Source Code.

4. STORAGE: This operation is used for storing the source code done by the developers. For this operation to be successful the member must meet some criteria like she developer should have account with unique name and password. All these checks are done by software. If then operation is successful, then the system automatically stores the codes developed.

5. DISPLAY: This is used to display each and every record from the system.

a) Source code: Record of every source code, done by the developer

CHAPTER 2

HARDWARE AND SOFTWARE REQUIREMENTS

2.1 Software requirements

- Documentation Tool: WPS Office
- Development Tool: Visual Studio Code, CodeBlocks

2.2 Hardware requirements

- Windows 8/8.1/10/11
- 4GB RAM
- 64 bit operating system
- 16 GB Hard disk

CHAPTER 3

SYSTEM OVERVIEW

3.1 Problem Definition

Manual system involves a lot of paper work so it becomes time consuming and costly .The chances of errors in booking a ticket are more in the manual system. The procees of keeping of tab of the number of seats booked and which seats are booked by which passenger becomes very tedious and confusing in the manual system.

3.1.1 Solution on problem

Before creating new software we should analyze what is the basic need of the software. Analysis is nothing but a planning of creation of software to get proper output from it. Analysis is detail study of protects that you want to show in your software solving problems. The basic need of the software is to save the time of the user with the help of all useful information. And also to maintain the collection of data in your computer systematically so that it's easy to understand. The proposed system provides lot of facility to the user and the admin to book and view the ship information and the feature to modify the ship information to the later.

The processing time on the data is very fast. It provides required data quickly to the user and also in specified manner to the user. A file system provides a cleaner and more efficient way of storing and maintaing the records.All the information of ship changes is given to the user and also the number of seats available with their timing. Today it is becoming very difficult to maintain record manually. This software system easily does the job of maintaining daily records .

CHAPTER 4

DESIGN

4.1 DATA FLOW DESIGN

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

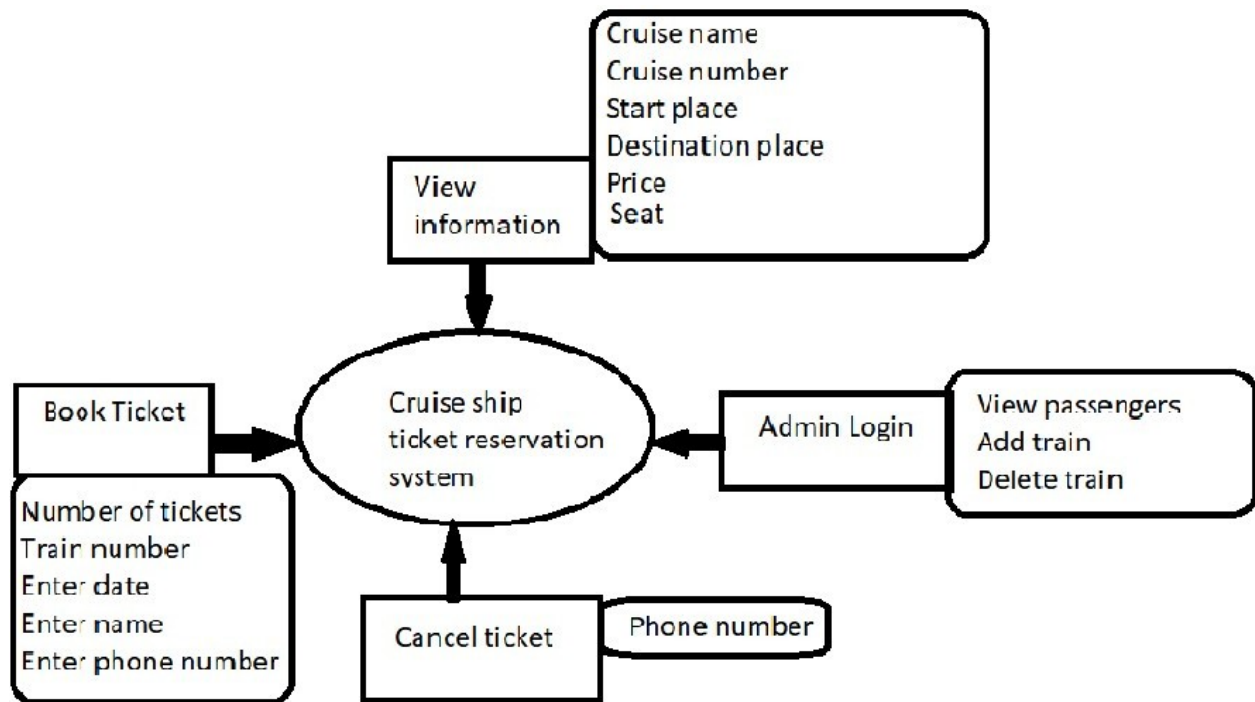


Fig 4.1: Data Flow Design

4.2 CLASS DIAGRAM

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity.

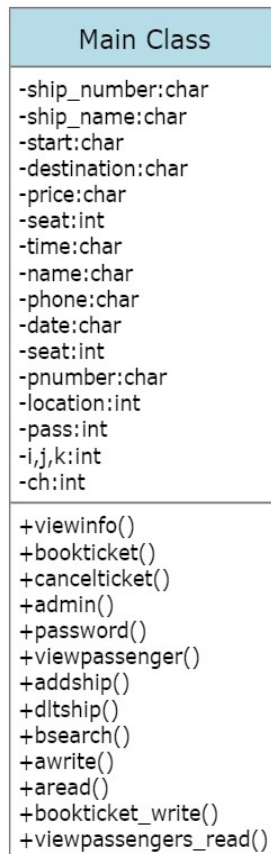


Fig 4.2: Class Diagram

CHAPTER 5

IMPLEMENTATION

Introduction:

We have used C language to implement the system. We have used concept of structure to define the a collection of related variable just like classes in object oriented programming language.

Functions and the members involved:

We have declared two structure named bookticket and adddata.

Members in bookticket structure includes ship number, name, phone, date and allocated seat.

Members in adddata structure includes serial number, ship number, shipname, start, destination, Price, seat, time.

Methods are viewinfo(), bookticket(), cancelticket(), admin(), password(), viewpassenger(), addship(), dltship(), awrite(), aread(), bookticket_write(), viewpassengers_read().

CHAPTER 6

TESTING

The integral part of any system's development life cycle is testing without which the system developed is sure to fail and result in loss of economic and manpower investments besides user's dissatisfaction and downfall of reputation. System testing is the stage of implementation, which aims at ensuring that the system works accurately and efficiently before actual operation commences. No program or system design is perfect, communication between the user and the designer is not always complete or clear. All this can result in errors.

Another reason for system testing is its utility as a user oriented vehicle before implementation. The application system is worthless if does not meet user needs, thus the system should be tested to see whether it meets the user requirements.

Testing here is conducted in bottom up approach as follows:

- Module testing: Here testing is done at each module level. Each case has been thoroughly tested to discover pitfalls.
- System testing: Here testing is done after all the modules have been integrated.

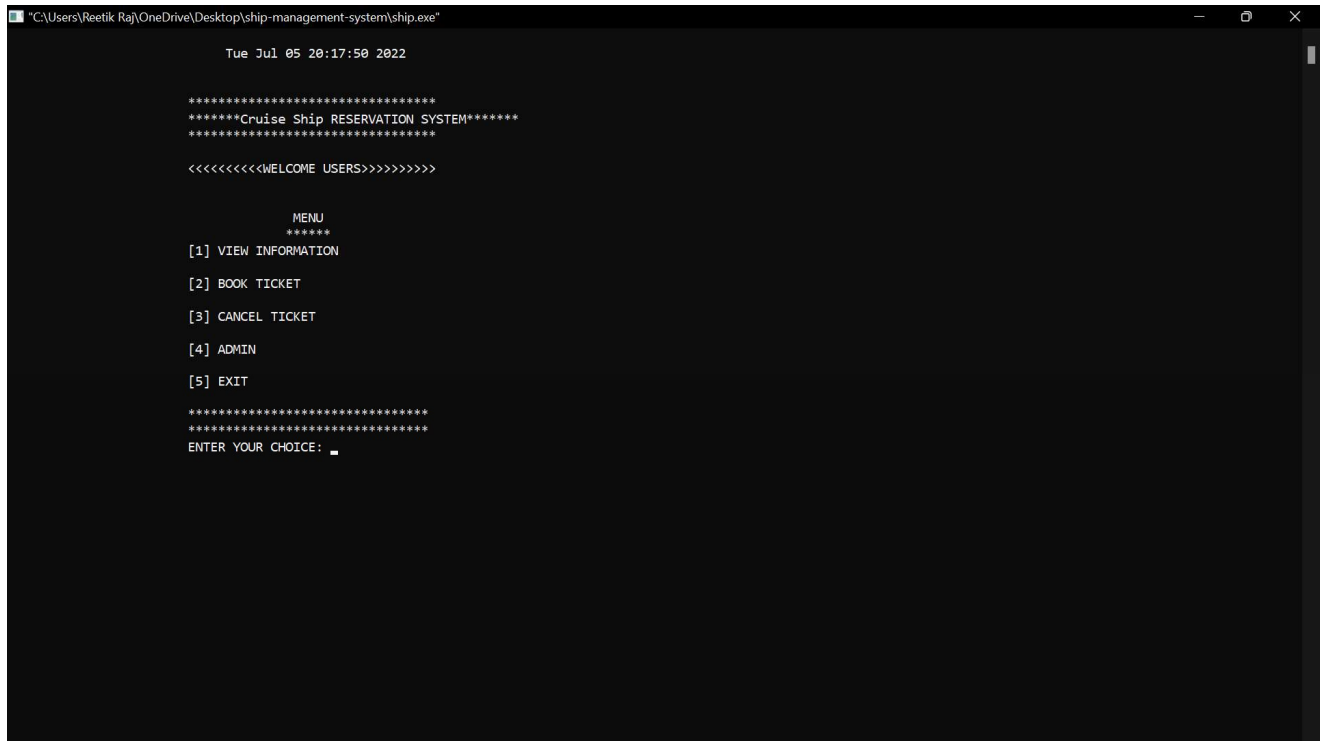
Test cases

Sl.No	Module	Test Case Description	Input	Expected Output	Actual Output	Status
1	Welcome Page	Verify whether Menu page displays.	Run the program.	Displays Menu page	Displays Menu Page	Pass
2	Menu Display	Displays the menu list.	Enter your choice.	Goes to the part of code you entered for to execute.	Goes to the part of code you entered for to execute.	Pass
3	Book ticket	Takes data to insert into the file	Enter choice 1 to book	Takes the input from the user and stores into file.	Accepts the input form the user and stores into file	Pass
4	Cancel ticket	Delete a booked ticket	Accept a valid phone number.	Ticket deletion successfull	Ticket deletion successfull	Pass
5	Admin sign in	Verify admin password	Enter a valid password	Login successfull	Login successfull	Pass
6	Admin page menu	Verify whether Menu page displays.	Sign in through valid password	Menu for Admin is displayed	Menu for Admin is displayed	Pass
7	View passenger booklist	Display the list of all the booking done by the passenger.	Enter 1 from option .	Passenger booklist is displayed.	Passenger booklist is displayed.	Pass
8	Add ship	Add a new ship to the booking console.	Enter all the fields prompted in the console .	Ship added successfully is displayed.	Ship added successfully is displayed.	Pass

Sl.No	Module	Test Case Description	Input	Expected Output	Actual Output	Status
9	Delete ship	Delete an existing ahip from the record.	Enter the correcct ship number.	Ship deleted successfully is displayed.	Ship deleted successfully is displayed.	Pass
10	Quit	Exits from the program	Select the choice to quit	Quits from the program	Quits from the program	Pass

CHAPTER 7

SNAPSHOTS

A screenshot of a Windows command prompt window titled "C:\Users\Reetik Raj\OneDrive\Desktop\ship-management-system\ship.exe". The window displays the introductory page of the Cruise Ship Reservation System. The text is as follows:

```
Tue Jul 05 20:17:50 2022

*****Cruise Ship RESERVATION SYSTEM*****

<<<<<<<<<WELCOME USERS>>>>>>>>>>>>

      MENU
      *****
[1] VIEW INFORMATION
[2] BOOK TICKET
[3] CANCEL TICKET
[4] ADMIN
[5] EXIT

*****
*****
ENTER YOUR CHOICE: _
```

Fig 7.1: Introductory page

This is the welcome page that displays the menu. The user chooses from the menu provided what he wants to execute.

```
"C:\Users\Reetik Raj\OneDrive\Desktop\ship-management-system\ship.exe"

*****
*****Cruise Ship RESERVATION SYSTEM*****
*****

SI  ship number  ship name  start place  destination place  price  seat  time
89  560107      waves     BLR          MUB             4000   337   2:00
65  114456      Sunset    PDY          KER             6700   7000   8:00
*****

press '1' for main menu & press any key for exit: _
```

Fig 7.2: View information page

This page displays the list of ships available for booking along with the start place, destination place, price, seat and time.

```
C:\Users\Reetik Raj\OneDrive\Desktop\ship-management-system\ship.exe

=====
*****Cruise Ship RESERVATION SYSTEM*****
=====

how many ticket do you want to buy: 1

Enter ship number: 560107

enter book 1 no ticket:
enter date: 30:06:2022

enter your name: Reetik

enter your phone number: 1234567890

seat number : 337

enter '1' for main menu & press any key to exit: _
```

Fig 7.3: Book ticket page

This page is for booking the ticket. It first asks the number of seats to book. After which it confirms the ship number entered by the user. Once, confirmed its asksthe the user to enter the date of travel , name, phone number, seat number. Once entered the ticket is booked.

```
C:\Users\Reetik Raj\OneDrive\Desktop\ship-management-system\ship.exe
Tue Jul 05 20:22:10 2022

*****Cruise Ship RESERVATION SYSTEM*****
*****

<<<<<<<<WELCOME USERS>>>>>>>>>>

      MENU
      *****
[1] VIEW INFORMATION
[2] BOOK TICKET
[3] CANCEL TICKET
[4] ADMIN
[5] EXIT

*****
*****
ENTER YOUR CHOICE: 3

enter phone number: 1234567890

<<<<<<<<<<<ticket cancelled successfully>>>>>>>>>>>>>>>>
```

Fig 7.4: Cancal ticket page

The ticket is cancelled based on the phone number of the client.

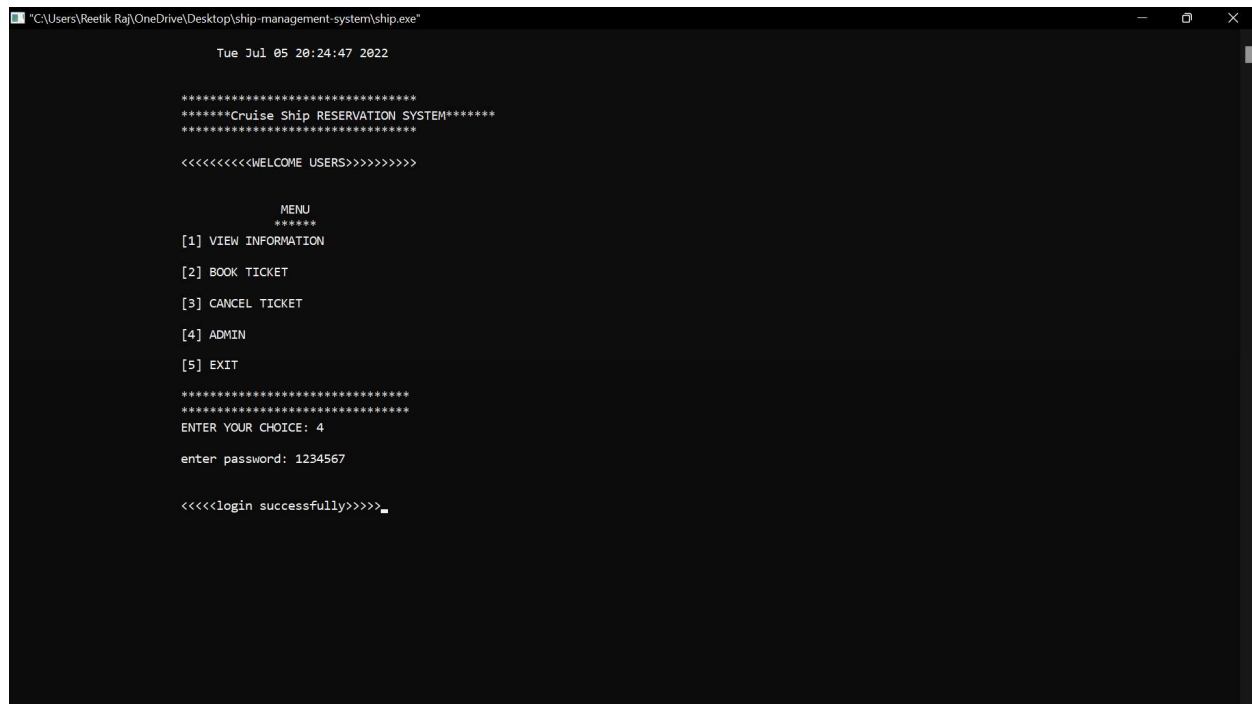


Fig 7.5: Admin password verification page

This page confirms the password entered by the administrator. On entering the correct password ,the admin is successfully logged in.

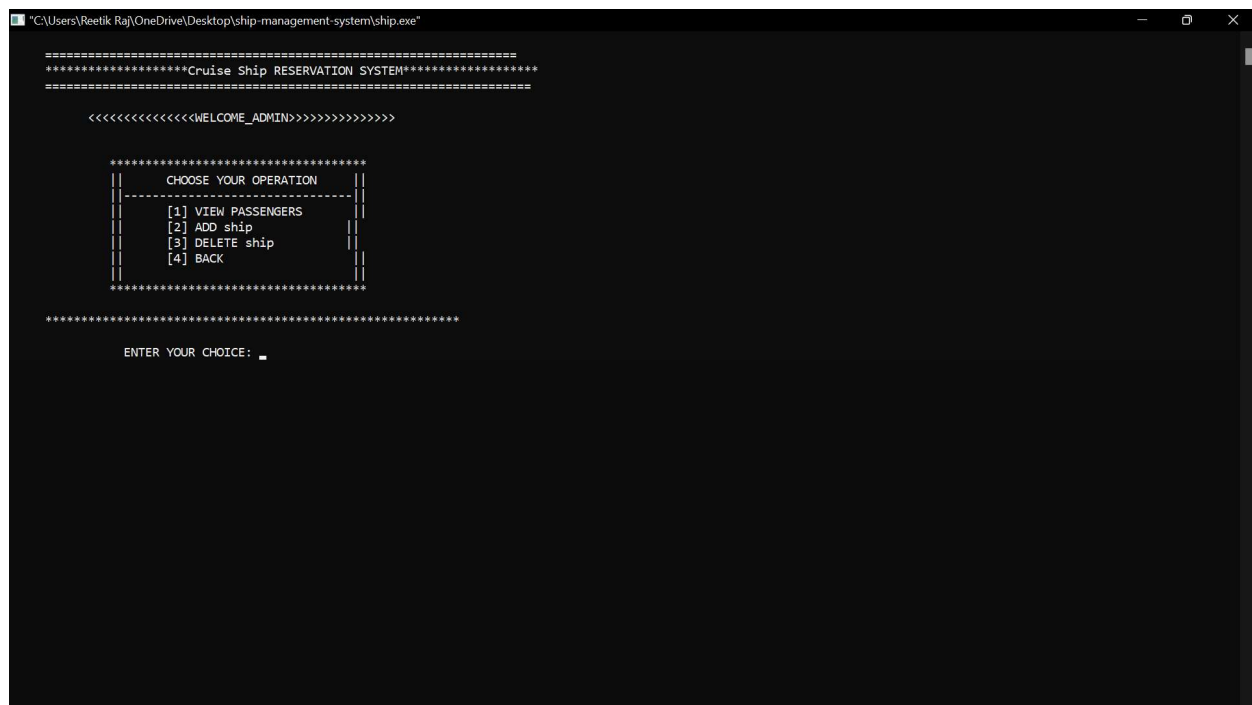


Fig 7.6: Admin page

The admin page displays the various functionality available to the admin.


```
C:\Users\Reetik Raj\OneDrive\Desktop\ship-management-system\ship.exe
```

```
=====
*****Cruise Ship RESERVATION SYSTEM*****
=====

<<<<<<<<<<<<<WELCOME_ADMIN>>>>>>>>>>>>>>>>

      *****
      ||          CHOOSE YOUR OPERATION          || | |
      ||-----||                               ||
      || [1] VIEW PASSENGERS                     ||
      || [2] ADD ship                            ||
      || [3] DELETE ship                        ||
      || [4] BACK                              ||
      ||-----||                               ||
      ||                                          ||
      *****

*****

ENTER YOUR CHOICE: 3

enter ship number: 560107

<<<<<<<<<<<<<ship deleted successfully>>>>>>>>>>>>>>>>
```

Fig 7.9: Ship delete page

We can delete the ship by entering its ship number.

```
C:\Users\Reetik Raj\OneDrive\Desktop\ship-management-system\ship.exe

=====
*****Cruise Ship RESERVATION SYSTEM*****
=====

BROUGHT TO YOU BY

***REETIK AND RAVI***
```

Fig 7.10: Exit page

This is the exit page.

CONCLUSIONS AND FUTURE ENHANCEMENT

Conclusions

The concept of variable length record, is used to store the record in the file. The program provides the user with a booking and cancel option. The admin is allowed to access various features through a password verification. The admin is provided with option to add new ship, delete existing ship and an access to all the booking done through a View Passenger option.

Limitation

As the number of records grows comparison processes also grows hence maintaining the large files becomes very difficult as we cannot read.

Future Enhancements

Given the File Structure design that we have built, we foresee a lot of areas that need enhancements.

1. First of all searching of records based on primary key indexes is the major void that needs to be filled.
2. Faster retrieval is one goal that the designers constantly try to achieve. One way of accomplishing the goal by using index structures itself is by creating multiple levels of these structures. The improvement will definitely show up as the database increases in size.

BIBILOGRAPHY

Book references

1. File Structures An Object-Oriented Approach with C by Michael J.Folk, Bill Zoellick, Greg Riccardi
2. Software Engineering-by Ian Sommerville
3. Let Us C by Yashavant .P.Kanetkar
4. The C Programming Language by Kernighan and Ritchie

Web references

1. www.stanford.edu
2. www.awl.com
3. www.codeproject.com
4. www.sourceforge.com
5. www.w3schools.com