

Bangladesh University of Business and Technology

PROJECT REPORT ON

AIRLINE RESERVATION SYSTEM

Submitted By:

Yasir Rabbani Tanvir

ID: 21224103083

Raihan Sheikh Joy

ID: 21224103184

Sarjid Mia

ID: 21224103143

Pervej Khandakar

ID: 21224103050

Submitted To:

M.M.Fazle Rabbi

Assistant Professor

Dept. of CSE, BUBT

DATE: 14-11-2023

# ABSTRACT

At MA Tours and Travels, we're making booking your trips easier and safer. You might have used systems like Travelport or Galileo before but we noticed they can be a bit tricky and not always up-to-date. That's why we're creating something new! We're using Java and MySQL to build a system that's super easy to use, updates info quickly and makes sure your bookings are safe. Picture a world where planning your trips is simple and stress-free – that's what we're aiming for. Our goal is to make your travel experience smoother and faster. So, come along on this journey with us, and let's make MA Tours and Travels your go-to place for easy and secure travel plans!

# DECLARATION

We, the undersigned, hereby proclaim that the task entitled "**Airline Reservation System** " submitted for the level of Bachelor of Science and Engineering in the personnel Computer Science and Engineering of Bangladesh University of Business and Technology is our unique work. We affirm that it contains no material which has been acknowledged for the honor to the applicants of some other degree or recognition, aside from where due reference is made in the following of the venture. As far as we could possibly know, it contains no materials previously distributed or composed by some other individual, except where due reference is made in this project work.

**-------------------------**

**Yasir Rabbani Tanvir**

**ID: 21224103083**

**Intake: 48**

**Section: 2**

**------------------------**

**Raihan Sheikh Joy**

**ID: 21224103184**

**Intake: 48**

**Section: 2**

**-----------------------**

**Sarjid Mia**

**ID: 21224103143**

**Intake: 48**

**Section:2**

**-----------------------**

**Pervej Khandakar**

**ID: 21224103050**

**Intake: 48**

**Section: 2**

# CERTIFICATION

This project “**Airline Reservation System**” report submitted by Yasir Rabbani Tanvir, Raihan Sheikh Joy, Pervej Khandakar and Sarjid Miastudents of Department of Computer Science and Engineering, Bangladesh University of Business and Technology (BUBT), under the supervision of Mr. M M Fazle Rabbi, Assistant Professor, Department of Computer Science and Engineering has been accepted as satisfactory for the partial requirements for the degree of Bachelor of Science Engineering in Computer Science and Engineering

**------------------------**

**Mr. M M Fazle Rabbi**

**Assistant Professor & Project Supervisor**

**Department of CSE**

# DEDICATION

Dedicated to our parents for all their love and inspiration.

# ACKNOWLEDGEMENTS

We extend our heartfelt gratitude to our supervisor, Assistant Professor,

Mr. M M Fazle Rabbi sir, for his unwavering support, invaluable guidance, and profound insights throughout this project. His expertise and mentorship were indispensable in our success.

We also thank our colleagues and friends for their discussions, suggestions, and feedback, which greatly contributed to the project's improvement. We acknowledge the pioneers in our field whose work inspired us.

Lastly, we express our deepest appreciation to our family for their unwavering support and love, which fueled our determination. We dedicate our achievements to them.

# APPROVAL

This work, titled " **Airline Reservation System**" submitted by Yasir Rabbani Tanvir, Raihan Sheikh Joy, Pervej Khandakar and Sarjid Miastudents of the Department of Computer Science and Engineering at Bangladesh University of Business and Technology (BUBT), under the supervision of Mr. M M Fazle Rabbi, Assistant Professor, Dept. of CSE, has been thoroughly examined and evaluated. It is hereby approved as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science Engineering in Computer Science and Engineering.

**------------------------**

**Mr. M M Fazle Rabbi**

**Assistant Professor & Project Supervisor**

**Department of CSE**

© Copyright by Yasir Rabbani Tanvir, Raihan Sheikh Joy, Pervej Khandakar and Sarjid Mia

All Right Reserved

# ABBREVIATIONS AND NOMENCLATURES

M

MA Tours and Travels

MAT 13

T

The return on investment

ROI 16

Table of Contents

[ABSTRACT ii](#_Toc151212222)

[DECLARATION iii](#_Toc151212223)

[CERTIFICATION iv](#_Toc151212224)

[DEDICATION v](#_Toc151212225)

[ACKNOWLEDGEMENTS vi](#_Toc151212226)

[APPROVAL vii](#_Toc151212227)

[ABBREVIATIONS AND NOMENCLATURES ix](#_Toc151212228)

[Chapter 1 Introduction 13](#_Toc151212229)

[1. Introduction 13](#_Toc151212230)

[2. Existing Model 13](#_Toc151212231)

[3. Problem Statement 13](#_Toc151212232)

[4. Motivation 13](#_Toc151212233)

[5. Objective of the Project 14](#_Toc151212234)

[6. Contribution 14](#_Toc151212235)

[7. Organization of Report 14](#_Toc151212236)

[8. Conclusions 14](#_Toc151212237)

[Chapter 2 Existing System 14](#_Toc151212238)

[9. Introduction 14](#_Toc151212239)

[10. Existing System 14](#_Toc151212240)

[11. Supporting Literature 15](#_Toc151212241)

[12. Used Diagram 15](#_Toc151212242)

[13. Technology Used 16](#_Toc151212243)

[14. Analysis of Existing System 16](#_Toc151212244)

[15. Conclusions 16](#_Toc151212245)

[Chapter 3: Proposed Model 17](#_Toc151212246)

[17. Introduction 17](#_Toc151212247)

[19. Technical Feasibility 17](#_Toc151212248)

[20. Operational Feasibility 17](#_Toc151212249)

[21. Economic Feasibility 17](#_Toc151212250)

[22. Requirement Analysis 17](#_Toc151212251)

[23. Non-Functional Requirement 18](#_Toc151212252)

[24. Functional Requirement 18](#_Toc151212253)

[25. System Design 19](#_Toc151212254)

[26. Agile Method 19](#_Toc151212255)

[27. Data Flow Diagram and Child DFD 21](#_Toc151212256)

[28. ER Diagram 23](#_Toc151212257)

[29. Flowchart 24](#_Toc151212258)

[30. Use Case Diagram 25](#_Toc151212259)

[31. Database Design 26](#_Toc151212260)

[Gantt Chart 27](#_Toc151212261)

[Activity Diagram 29](#_Toc151212262)

[32. Class Diagram 30](#_Toc151212263)

[33. Implementation 31](#_Toc151212264)

[35. Front End 32](#_Toc151212265)

[36. Forms Design 33](#_Toc151212266)

[38. Back End 34](#_Toc151212267)

[39. Database Design (Revisited) 35](#_Toc151212268)

[40. Conclusions 36](#_Toc151212269)

[Chapter 4: Experimental Results 36](#_Toc151212270)

[41. Experimental Results 36](#_Toc151212271)

[42. Introduction 36](#_Toc151212272)

[43. Result Analysis 36](#_Toc151212273)

[44. Applications 37](#_Toc151212274)

[45. Conclusions 37](#_Toc151212275)

[Chapter 5: User Manual 38](#_Toc151212276)

[46. Introduction 38](#_Toc151212277)

[47. System Requirements 38](#_Toc151212278)

[48. Hardware Requirements 38](#_Toc151212279)

[49. Software Requirements 38](#_Toc151212280)

[Web Server: 39](#_Toc151212281)

[50. User Interface 39](#_Toc151212282)

[51. Login 40](#_Toc151212283)

[52. Order 41](#_Toc151212284)

[53. Payment Option 42](#_Toc151212285)

[54. Cancellation 43](#_Toc151212286)

[55. Search 44](#_Toc151212287)

[56. Feedback 44](#_Toc151212288)

[57. Conclusions 45](#_Toc151212289)

[Chapter 6: Conclusion and Next Steps 45](#_Toc151212290)

[58. Conclusion 45](#_Toc151212291)

[59. Next Steps 46](#_Toc151212292)

[62. References 46](#_Toc151212293)

[63. Appendix 46](#_Toc151212294)

# Chapter 1 Introduction

## 1. Introduction

We all use airplanes to go from one place to another. To book our flights, we use computer systems. This project is about creating a better computer system for booking flights. We will use Java and MySQL to do this. The new system will be easier to use, show flight information in real-time and be safer for everyone.

## 2. Existing Model

Right now, the computer systems we use for booking flights like Travelport, Galileo are okay, but they have some problems. They can be a bit hard to use, and sometimes they don't show the most up-to-date flight information. Also, they may not be very safe from hackers. With this project, we want to make these systems better by using Java and MySQL to create one that is easier to use, updates flight information quickly and is very safe.

## 3. Problem Statement

The issues with the current flight booking systems are quite clear. People often find it hard to use them because the buttons and menus can be confusing. Sometimes, the flight information they show is not the most current, so travelers might not know if a flight is available or not. There's also the issue of safety - sometimes, personal information can be at risk. Another problem is that airlines might sell too many tickets or not enough, which can lead to problems for everyone. We want to fix all these issues and make booking flights much easier and safer.

## 4. Motivation

We're diving into this project because booking flights with systems like Travelport and Galileo is a bit of a hassle. These systems can be confusing, slow and not as safe as we'd like. Our main goal is simple, make booking flights easier, quicker and safer for everyone.

## 5. Objective of the Project

Our big aim is to create a better computer system for booking flights. By using Java and MySQL, we want to make a system that's super easy to use, shows real-time flight info and is really safe. It's not just about fixing problems; we want to make booking flights a breeze for everyone.

## 6. Contribution

It's a big help to make flight booking systems better. With Java and MySQL, we're giving the whole booking process an upgrade. It's not just about tech stuff; it's about making things simpler and more enjoyable. We're on a mission to make booking flights easy for everyone.

## 7. Organization of Report

Think of this report like a well-organized book. It goes from one thing to the next, explaining the problems we found, the cool solutions we came up with, and how everything fits together. You can follow our journey, understanding why we did what we did.

## 8. Conclusions

In the wrap-up, we're summing it all up and talking about what we achieved. It's like putting a bow on a present. We're sharing what we learned, what changed and how our project is making booking flights a better experience. It's not just about fixing; it's about making things awesome for everyone.

# Chapter 2 Existing System

## 9. Introduction

We're starting with the basics. Here, we'll introduce you to the reservation system made for MA Tours and Travels (MAT). We'll explain why it was created and give you a sneak peek into what it can do.

## 10. Existing System

There are many software options available, and Travelport and Galileo are among them. These systems assist travel agencies and businesses in booking flights for people. Travelport serves as a hub connecting agencies, airlines and travelers in finding and booking the right flights effortlessly. Similarly, Galileo functions as a global system linking various components of the travel industry. Both systems offer numerous features, such as displaying real-time flight information and simplifying the process for agents to manage bookings.

## 11. Supporting Literature

We're not just making things up! We've read a bunch of books and articles to understand more about reservation systems, Java, SQL and the techie stuff. We'll share what we learned to make things clearer for you.

The following books provide valuable insights into key areas relevant to our analysis:

1. **"Airline Operations and Scheduling"** by Massoud Bazargan
2. **"Database Management Systems"** by Raghu Ramakrishnan and Johannes Gehrke
3. **"Java: The Complete Reference"** by Herbert Schildt
4. **"Travel Agency Management"** by The Travel Institute

## 12. Used Diagram

In the development of this system, we employed various diagrams to visualize and plan different aspects of the system. Here are the key diagrams used:

* Data Flow Diagram
* ER Diagram
* Flowchart
* Use Case Diagram
* Database Design
* Class Diagram

## 13. Technology Used

This System will be developed using Java as the programming language and SQL as the database management system. The user interface will be designed using Windows Forms.

## 14. Analysis of Existing System

Looking at the current airline reservation systems, Travelport and Galileo, there are reasons to think about making a new one. First, these systems might be too hard for smaller travel agencies to use. If we create a simpler system that still has all the important stuff, it could help these smaller agencies a lot. Also, the current systems can be expensive, especially for smaller businesses. Making a new system that's more affordable but still does what people need could be a good idea. Some businesses might want a system that they can customize easily. If we make a flexible system that fits different needs, it could be a hit. Another thing is that the current systems may not always work well with other tools businesses use. If we create a new system that plays nice with different tools, it could make things smoother for users. Keeping up with new technology and making a system that's up-to-date is important too. If there's a specific group of people who need a simple and mobile-friendly system, making one just for them could be a great idea. Lastly, listening to what users think about the existing systems can give us clues about what's missing or not working well. Using their feedback to make a system that solves these problems would be pretty smart.

## 15. Conclusions

Finally, we're wrapping it up. We'll share what we found out about what's good and what needs fixing in the system. Then, we'll get ready for the next chapters, where we'll suggest cool ideas to make the reservation system even better.

# Chapter 3: Proposed Model

## 17. Introduction

Next up, we introduce our ideas in detail. We lay out the goals and objectives of our model, explaining how it fits with what MA Tours and Travels need

## 19. Technical Feasibility

MA Tours & Travels has access to the necessary technical infrastructure, including computers and servers, to support the implementation of a new reservation system. The required software and development tools are readily available, making the technical setup feasible for the project.

## 20. Operational Feasibility

The staff at MA Tours & Travels should be trained to use the new reservation system effectively. The system's compatibility with the existing business processes should be evaluated to ensure a smooth integration.

## 21. Economic Feasibility

The implementation and maintenance costs of the new reservation system need to be assessed against the potential benefits it will bring to the company. The return on investment (ROI) should be evaluated to ensure the financial viability of the project for MA Tours & Travels.

## 22. Requirement Analysis

In creating our travel software, we thought about what users need and what the system should do. We're thinking about travel agents and regular folks using it. We made searching smarter and added a trick to predict what users might like. Booking a trip should be easy and quick, with some special deals thrown in. We made sure it works well on computers and sends updates in real-time. Keeping everything safe, we locked up user info and added a special login for extra safety. The software should work well almost all the time, and even if something goes wrong, users will get a nice message about it. It should also work on different web browsers like Chrome and Firefox. We're following the rules to keep everything private and letting everyone know how we handle their info. Making it easy to keep everything running, we're building the software so it's easy to update without causing problems. Lastly, we want our software to work well with other systems, so we're making sure it can easily connect with different services and programs.

## 23. Non-Functional Requirement

In our project, we have non-functional requirements that focus on how our travel software should perform, be secure, and provide a good user experience. For performance, we want the system to respond quickly, taking no more than two seconds for users to get what they need. It should also be ready to handle more users during busy times. To keep everything safe, we'll make sure all user information is kept private by encrypting it. Users will also have an extra layer of protection with multi-factor authentication. Making the software easy to use is a big deal for us. The design should be simple and consistent, so everyone can understand and navigate it without any trouble. We also want the system to stay reliable, meaning it should work well most of the time. Even when there's a problem, the system should handle it gracefully, letting users know what's going on in a friendly way. Compatibility is important too. The software should work on different web browsers like Chrome and Firefox and on various devices like computers, tablets, and phones. If more people start using our system, we want it to still work smoothly, so we're planning for that by making it scalable. We also have plans in case something unexpected happens, like server problems or disasters, to get things back on track quickly. It's crucial for us to follow rules and regulations, so our software complies with privacy laws, ensuring users' data is handled safely. To make it easy to keep everything up and running, we're building the software in a way that's easy to update without causing issues. Additionally, we're creating clear instructions for developers and administrators. Lastly, we want our software to work well with other systems, so we're making sure it can easily connect with different services and applications by following accepted standards.

## 24. Functional Requirement

To make our travel software really useful, we made a list of things it should be able to do for users:

* User Sign-Up:

Why it's important: This helps users have a more personalized experience.

* Smart Search and Suggestions:

Why it's important: It makes finding the right flights easy for users.

* Easy Booking:

Why it's important: Users want a hassle-free experience when booking their travel.

* Special Deals:

Why it's important: Users can save money with exclusive offers.

* Computer-Friendly:

Why it's important: Many users prefer using computers for booking their trips.

* Real-Time Updates:

Why it's important: Keeps users informed about their travel plans.

* Safety Measures:

Why it's important: Ensures the security of user information.

* Friendly Error Messages:

Why it's important: Helps users understand and fix issues easily.

* Works on Different Browsers:

Why it's important: Gives users a consistent experience no matter which browser they use.

* Easy Updates:

Why it's important: Allows the software to get better without bothering users.

These are the things we're making sure our travel software can do. It's all about making it user-friendly and practical for everyone who wants to book their travels hassle-free.

## 25. System Design

Now, we're talking about how everything will look and work. We use tools like diagrams to show the flow of information, making sure everyone can understand how the system is going to function.

## 26. Agile Method

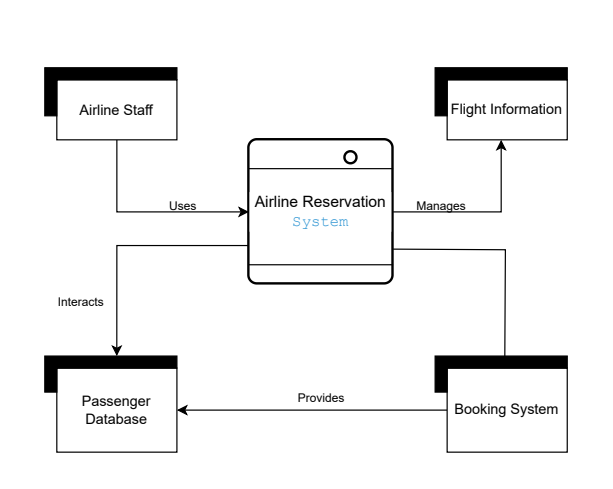
We use Agile for this project because it offers several advantages that align well with the needs of MA Tours & Travels:

* **Customer-Centric Approach:** Agile focuses on customer collaboration and continuous feedback. This ensures that the developed system meets the specific needs and expectations of MA Tours & Travels, leading to a more satisfying end product.
* **Adaptability to Changes:** In the travel industry, requirements can change frequently. Agile allows us to be flexible and adapt to evolving needs, ensuring that the system remains up-to-date and responsive throughout its development.
* **Incremental Development:** Agile iterative approach allows us to deliver working software in short phases. This enables MA Tours & Travels to start utilizing the system sooner and provide feedback for continuous improvement, enhancing the system's overall effectiveness.
* **Continuous Testing and Quality Assurance:** Agile places a strong emphasis on continuous testing throughout the development process. This ensures that the airline reservation system will be robust, secure, and of high quality, providing a reliable platform for MA Tours & Travels and their customers.
* **Collaboration and Communication:** Agile promotes close collaboration between the development team and stakeholders. Regular meetings and effective communication channels keep everyone informed about the project's progress, fostering a strong sense of teamwork and shared responsibility.
* **Faster Time-to-Market:** Agile incremental delivery approach enables us to swiftly deliver core features, accelerating the system's time-to-market. This allows MA Tours & Travels to start benefiting from the system sooner and gain a competitive advantage in the travel industry.
* **Risk Mitigation:** Agile facilitates early identification and mitigation of potential risks. By proactively addressing challenges, we can minimize the likelihood of encountering major obstacles during the development process, ensuring a smooth and successful implementation.

By leveraging the Agile methodology, we aim to deliver a successful and user-friendly airline reservation system that meets MA Tours & Travels' unique requirements while enhancing their operational efficiency and customer experience.

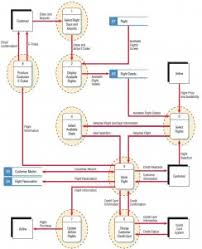
## 27. Data Flow Diagram and Child DFD

This is like a map showing how information moves around in the system. It helps us see the big picture of how things connect.

****

**Fig. 1.1 (Data Flow Diagram)**

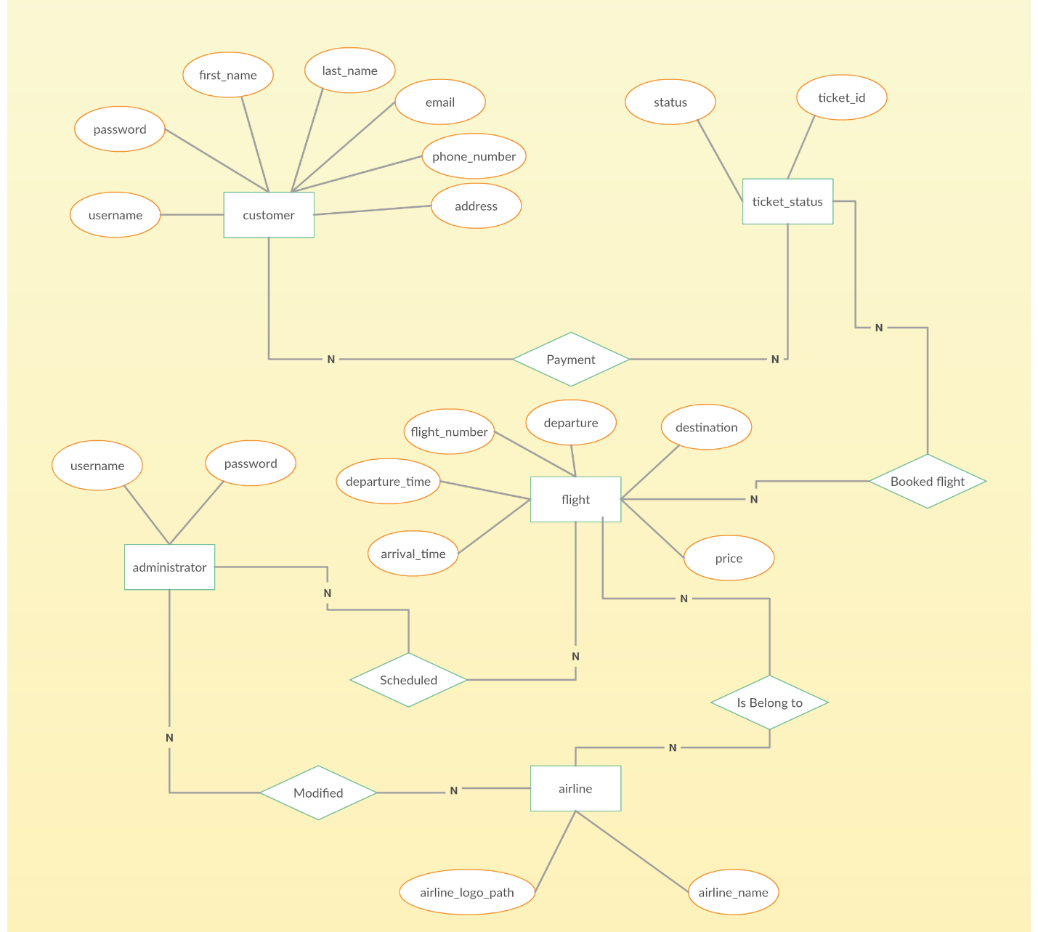
## 



**Fig. 1.2 (Child DFD)**

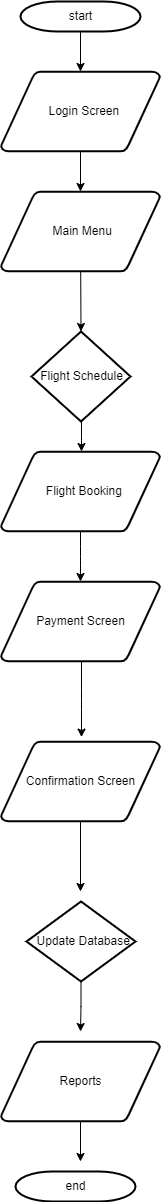
## 

## 28. ER Diagram

****

**Fig. 2 (ER Diagram)**

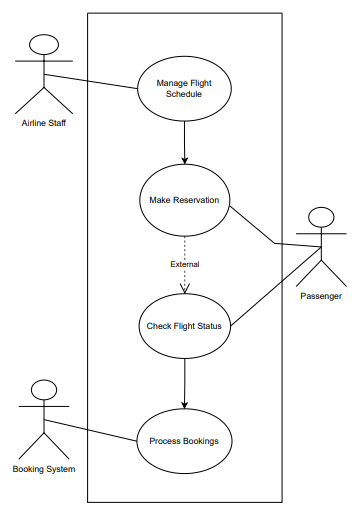
## 29. Flowchart

****

**Chart no. 1 (Flowchart)**

## 

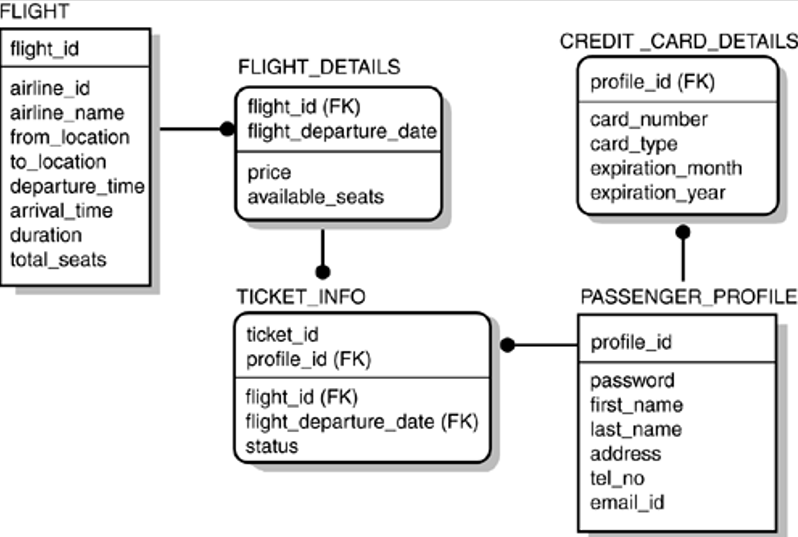
## 30. Use Case Diagram

****

**Fig. 3 (USECASE Diagram)**

This diagram shows how different people or things interact with our system. It helps us see the different scenarios and make sure our system covers all the bases.

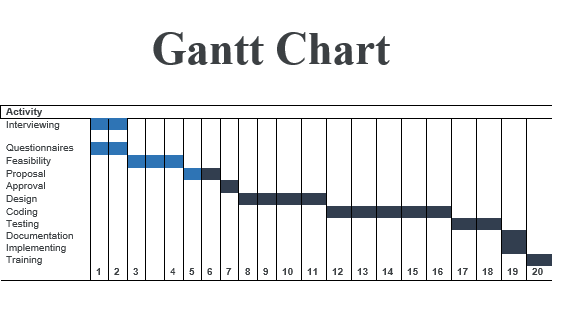
## 31. Database Design



Now, we're getting into how we organize and store information. This is crucial for making sure data is managed well and can be easily accessed when needed.

**Fig. 4 (Database Diagram)**

## Gantt Chart



**Total 20 weeks**

Chart no. 2 (Gantt Chart)

**Description:** We're working on an airline reservation system project with 10 steps. These steps are:

**Understanding Requirements (2 weeks):** We'll talk to the airline to understand what they need for their booking system.

**Designing User Interface (2 weeks):** We'll create the screens and forms that people will use to book flights, making it easy to use.

**Checking Feasibility (2 weeks):** We'll make sure the system we're planning is possible by looking at what's needed and how it can work.

**Creating System Blueprint (2 weeks):** We'll make a detailed plan for how the system will work based on what's possible and what's needed.

**Gaining Approval (1 week):** The airline will agree to our plan so we can move forward with confidence.

**Building the System (4 weeks):** We'll start making the actual software for the system, piece by piece.

**Testing the Software (2 weeks):** We'll test everything to be sure it works well and fix anything that's not right.

**Documenting Guidelines (1 week):** We'll write down clear instructions for how to use the system.

**System Implementation (1 week):** We'll put the system on the airline's computers and make sure it's set up correctly.

**Training Staff (1 week):** We'll teach the airline's employees how to use the new system so they can do their jobs well.

## Activity Diagram

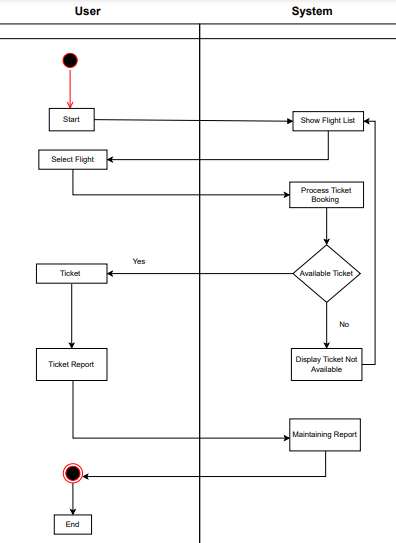
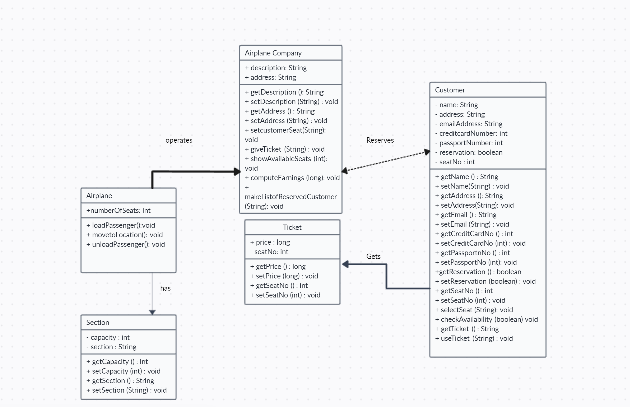


Fig. 5 (Activity Diagram)

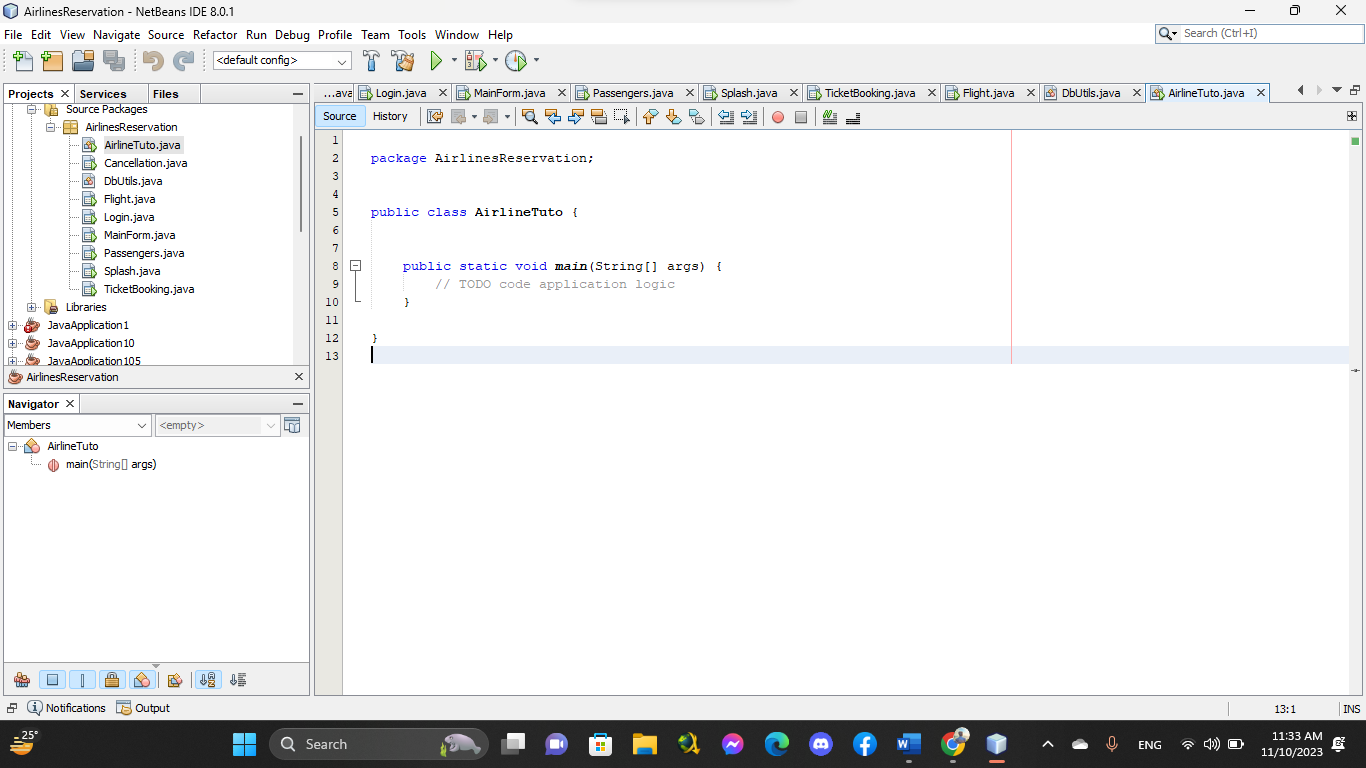
## 32. Class Diagram



**Fig. 6 (Class Diagram)**

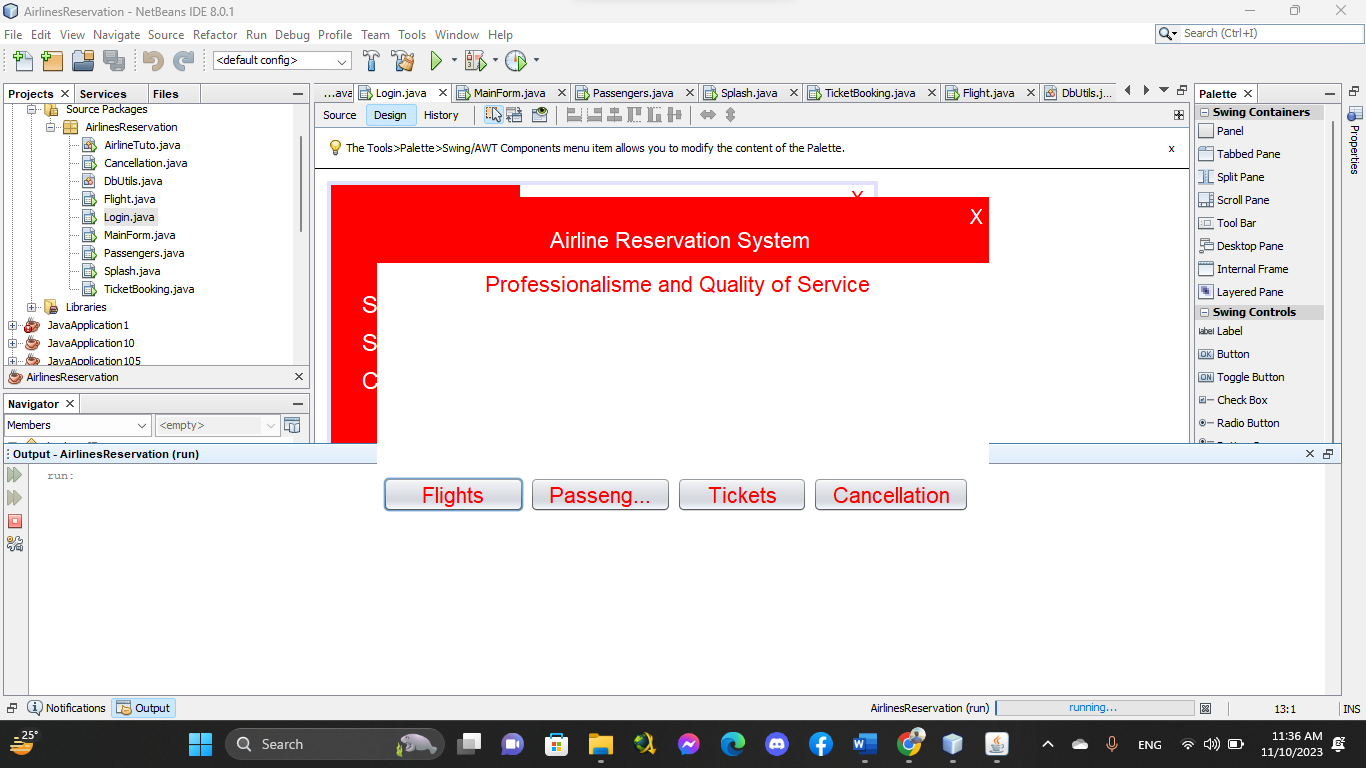
We use this diagram to show how different parts of the system interact. It's like a blueprint for how everything is structured in the system

## 33. Implementation



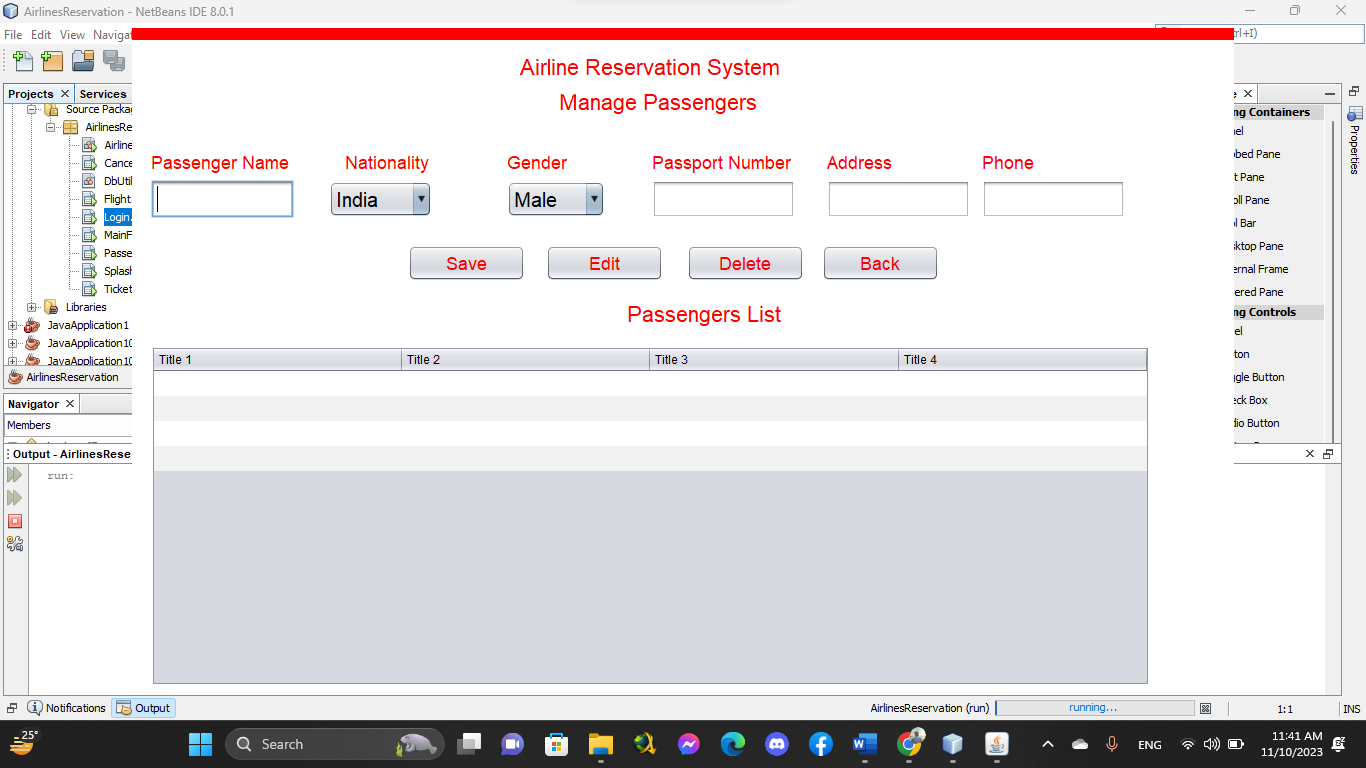
This is where we start making our ideas a reality. We talk about the actual steps involved in building the system, from writing code to making sure everything works together.

## 35. Front End

****

We're focusing on what users see and interact with. We design the user interface, making sure it's easy to use and looks good.

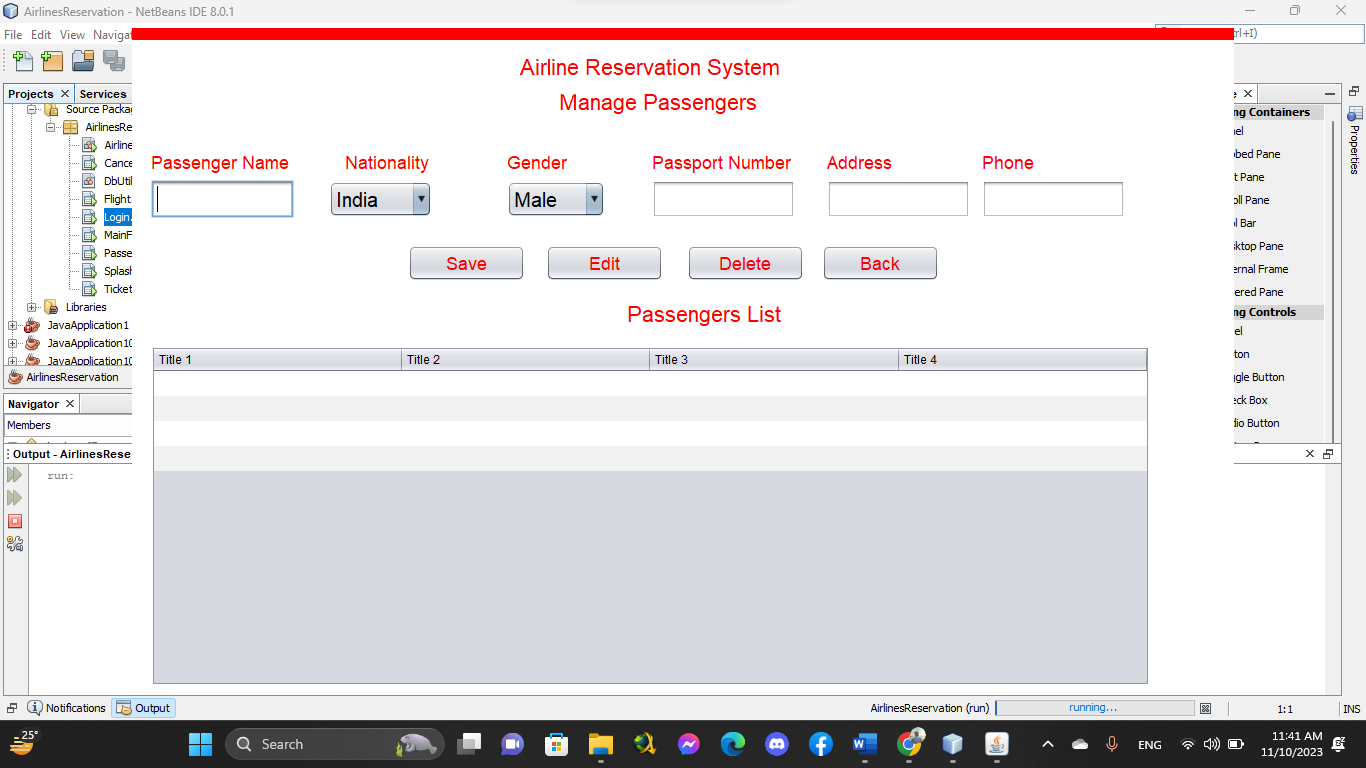
## 36. Forms Design



This section is all about designing the input forms. We want to make sure users can easily input information without any confusion.

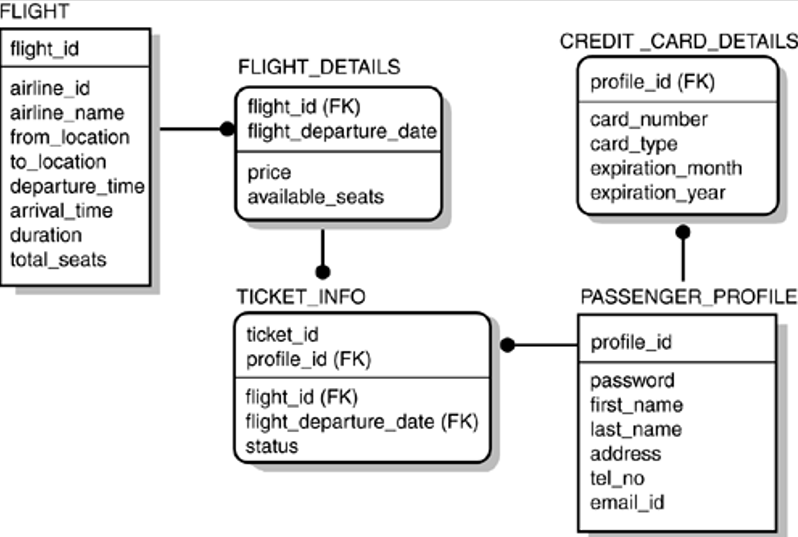
## 

## 38. Back End



This is the **backend part** that makes everything run smoothly.

## 39. Database Design (Revisited)



We revisit the database design during implementation to ensure it supports efficient data management. This is crucial for the overall performance of the system.

## 

## 40. Conclusions

Finally, we wrap it up by summarizing what we've achieved. We reflect on the journey from planning to making things happen, emphasizing how our changes can transform the reservation system for MA Tours and Travels.

# Chapter 4: Experimental Results

## 41. Experimental Results

It is like the part where we see how well our new airline reservation system worked for MA Tours and Travels. We're not just talking about it; we're showing the actual results from trying it out. First, we introduce what we wanted to learn in our experiments. Then, we dive into the numbers and what we observed. It's like looking at the real-world performance of our system and figuring out how it did. After that, we think about how the things we learned can actually be useful in real life that's the applications part.

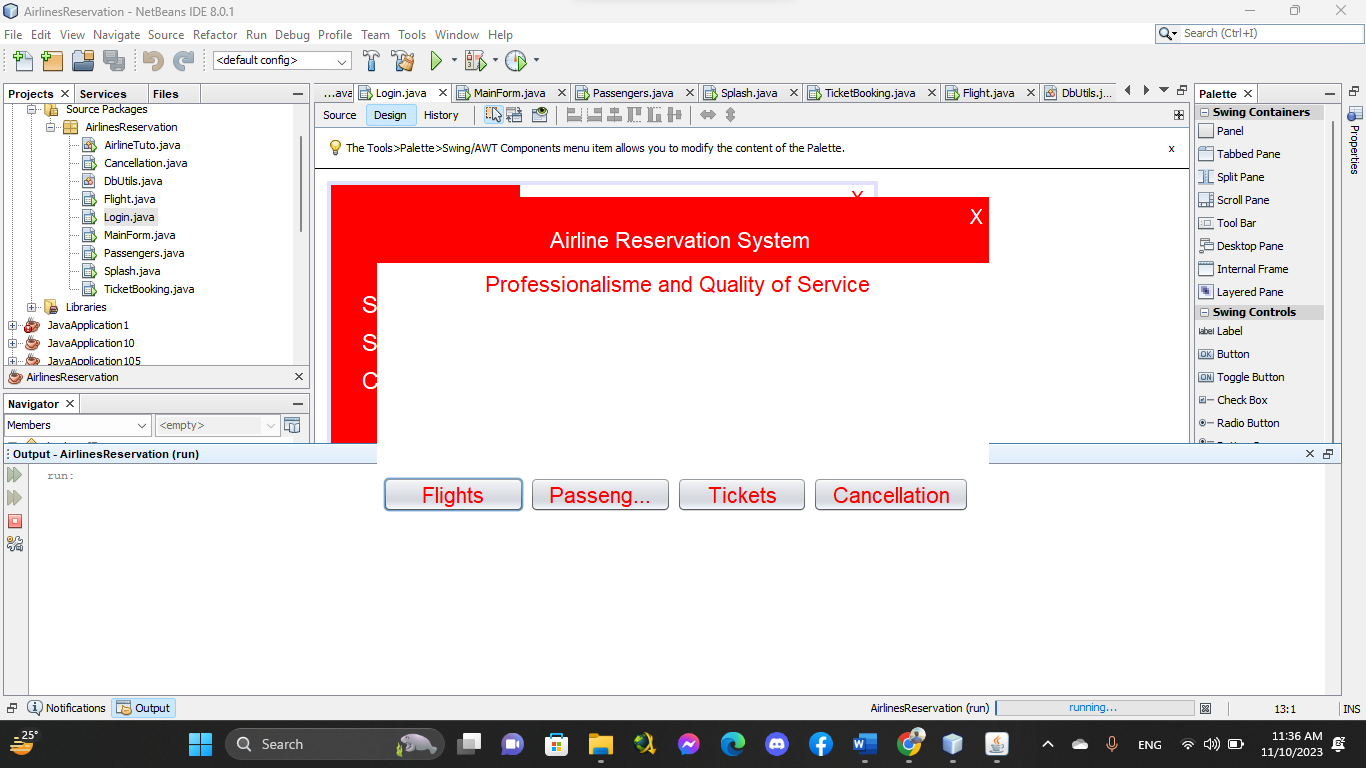
## 42. Introduction

This section sets the scene for understanding the reasons behind our experiments.

## 43. Result Analysis

Now, we dig into the numbers and observations from the experiments. We analyze the data to understand the performance of our system. This section is about breaking down the details and figuring out the story that the results are telling us.

## 44. Applications

****

After looking at the results, we explore how the insights we gained can be applied in real-life situations. It's like thinking about the practical uses and benefits that can come from what we observed during the experiments.

## 45. Conclusions

In this chapter we wrap up everything we found out from the experiments. It's the part where we look at the big picture and think about what it all means for our airline reservation system. This chapter is where our ideas meet reality, and we see how they actually work in practice.

# Chapter 5: User Manual

## 46. Introduction

We start with a quick hello and an overview to let users know what they'll find in the user manual. It's like a friendly invitation to explore and learn.

## 47. System Requirements

The system requirements outline what's needed to run our travel software smoothly. Here's a breakdown:

Operating System:

Requirement: The software is compatible with Windows 10, macOS, and Linux.

Web Browser:

Requirement: The software supports the latest versions of popular browsers, including Chrome, Firefox, and Safari.

Internet Connection:

Requirement: A stable internet connection with a minimum speed of 5 Mbps.

Screen Resolution:

Requirement: Minimum screen resolution of 1366x768 pixels.

## 48. Hardware Requirements

Requirement**:** Dual-core processor (e.g., Intel Core i3 or equivalent).

4 GB RAM, Minimum 20 GB free storage space.

Graphics Card:

Requirement: Integrated graphics or dedicated graphics card.

## 49. Software Requirements

These specifications detail the necessary software components:

## Web Server:

Requirement: Apache or Nginx server.

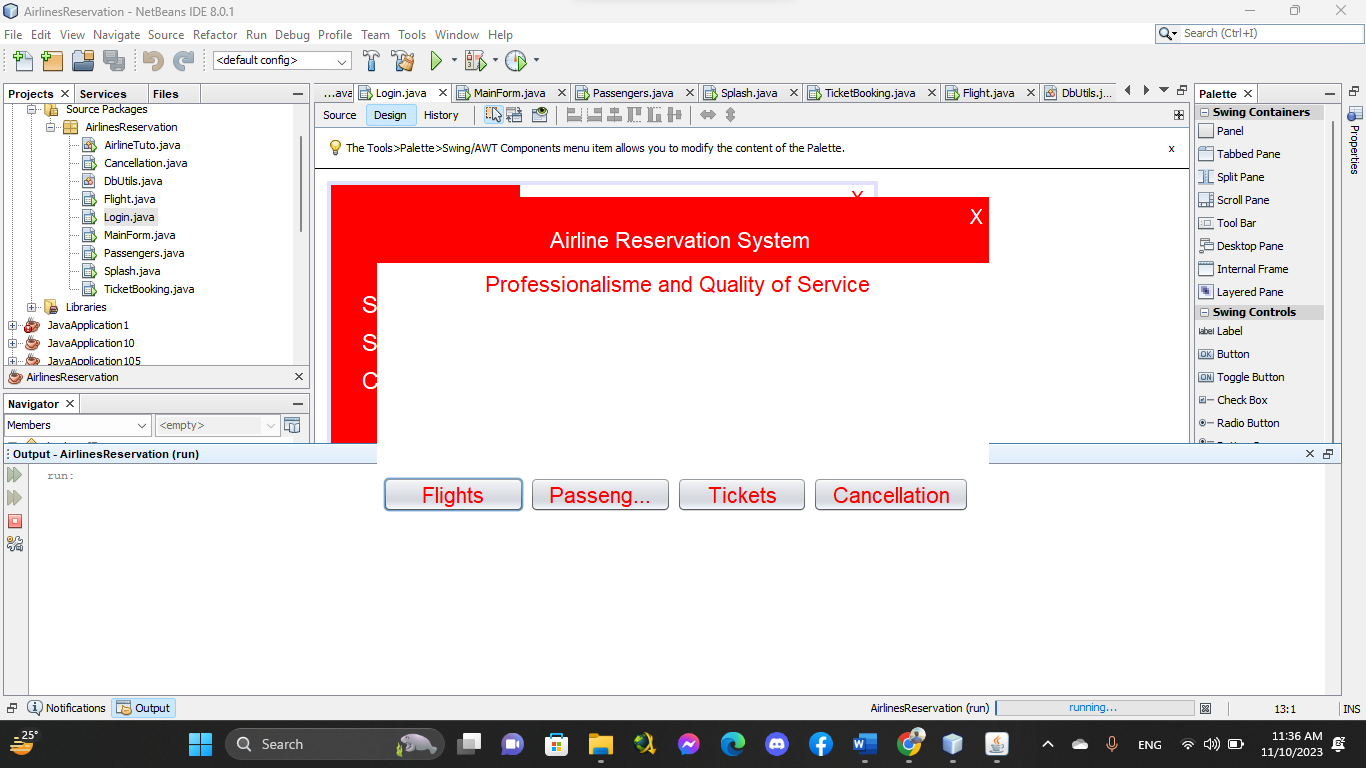
Database Management System:

Requirement: MySQL or PostgreSQL.

Programming Language:

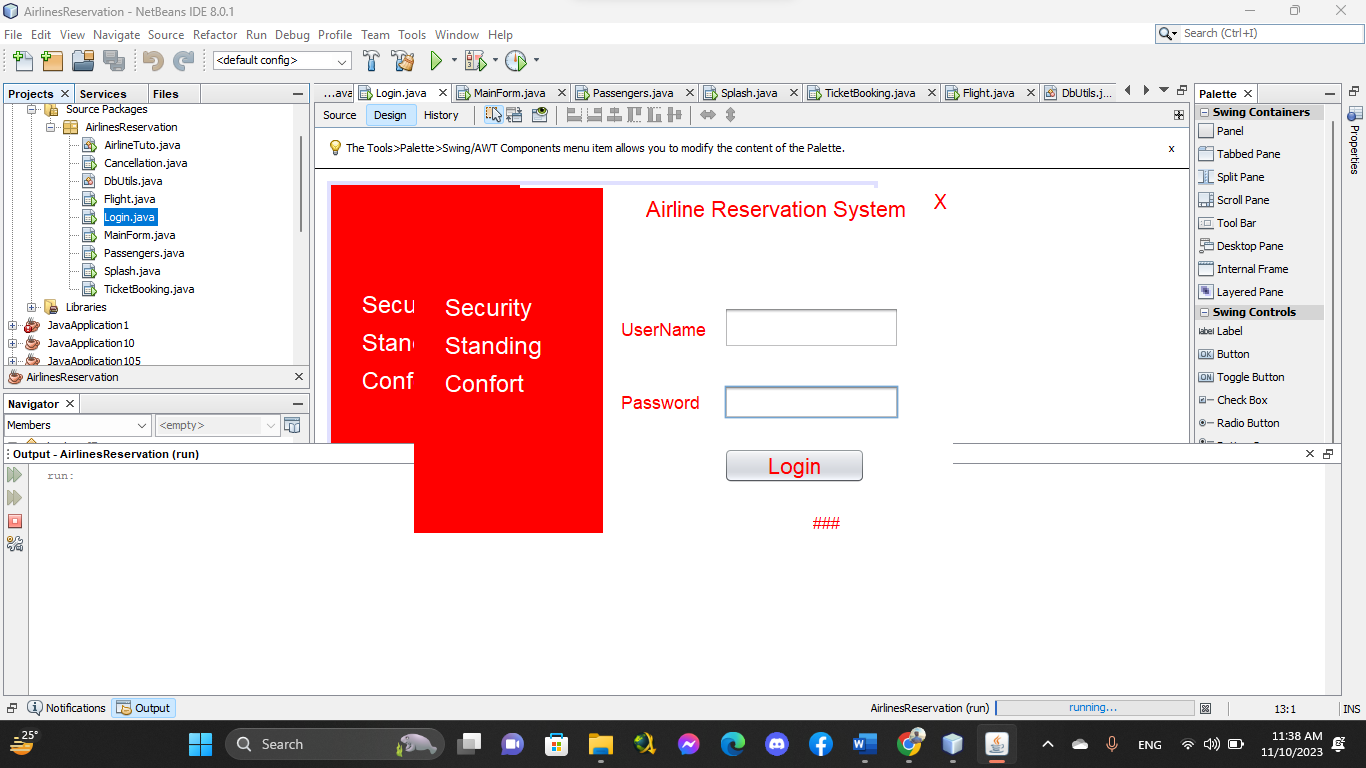
Requirement: Java (JDK 8 or later).

## 50. User Interface



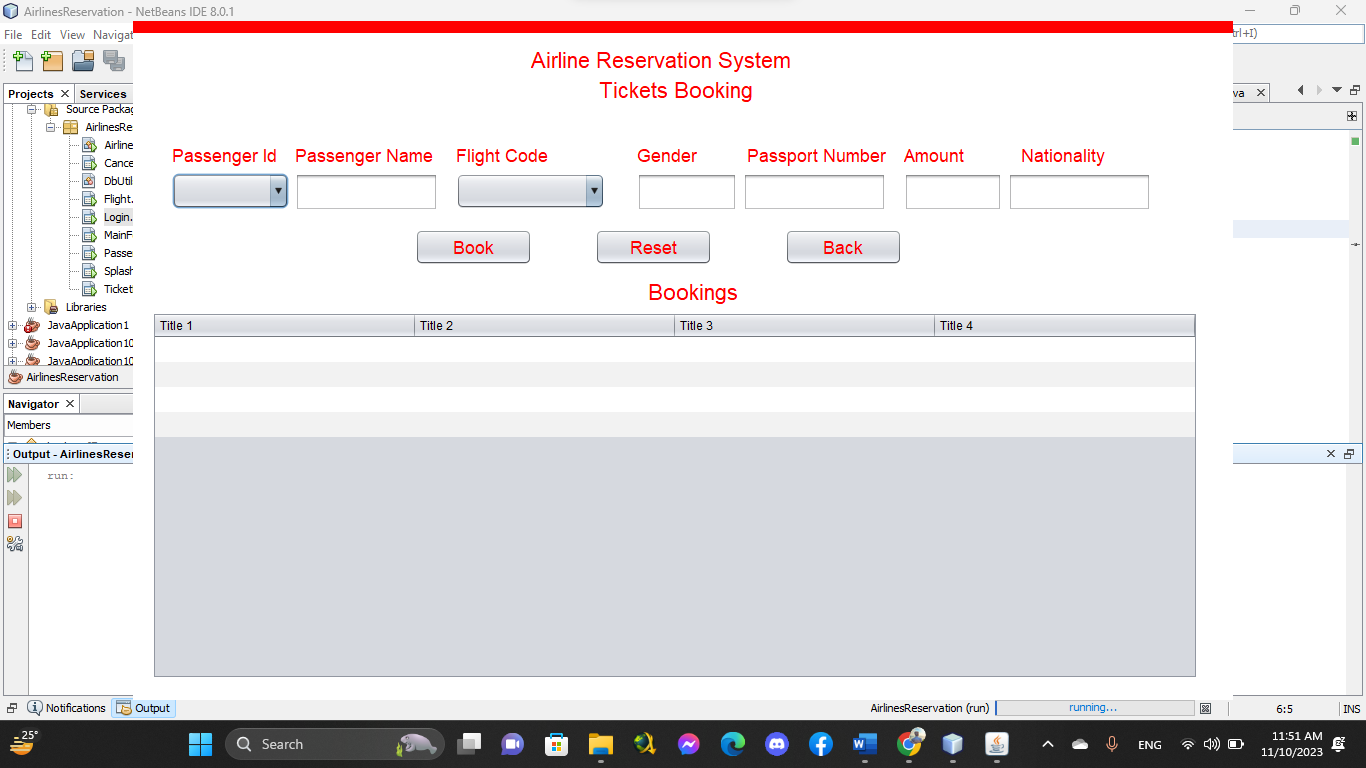
This is the **user interface** of our system.

## 51. Login



This is the **login interface** of our system.

## 52. Order

****

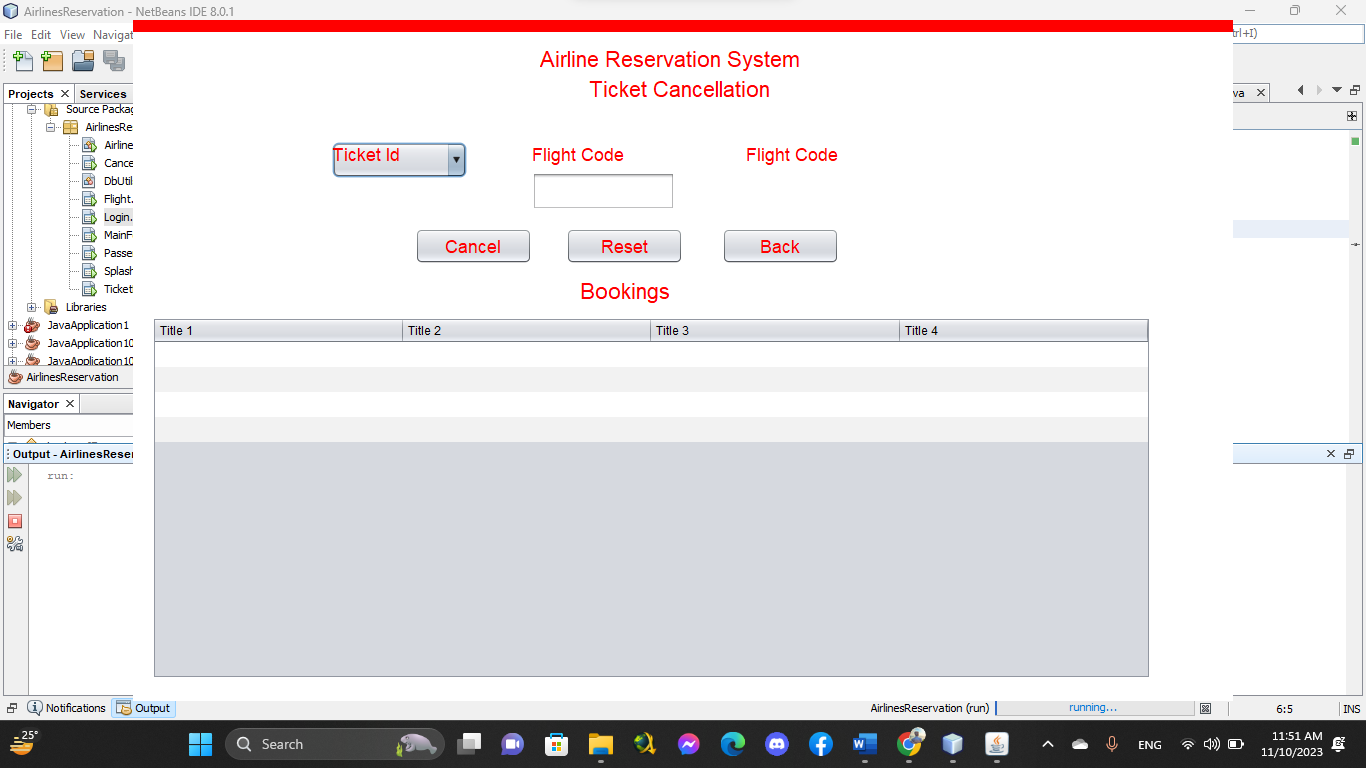
By this **interface** we can order the tickets.

## 53. Payment Option



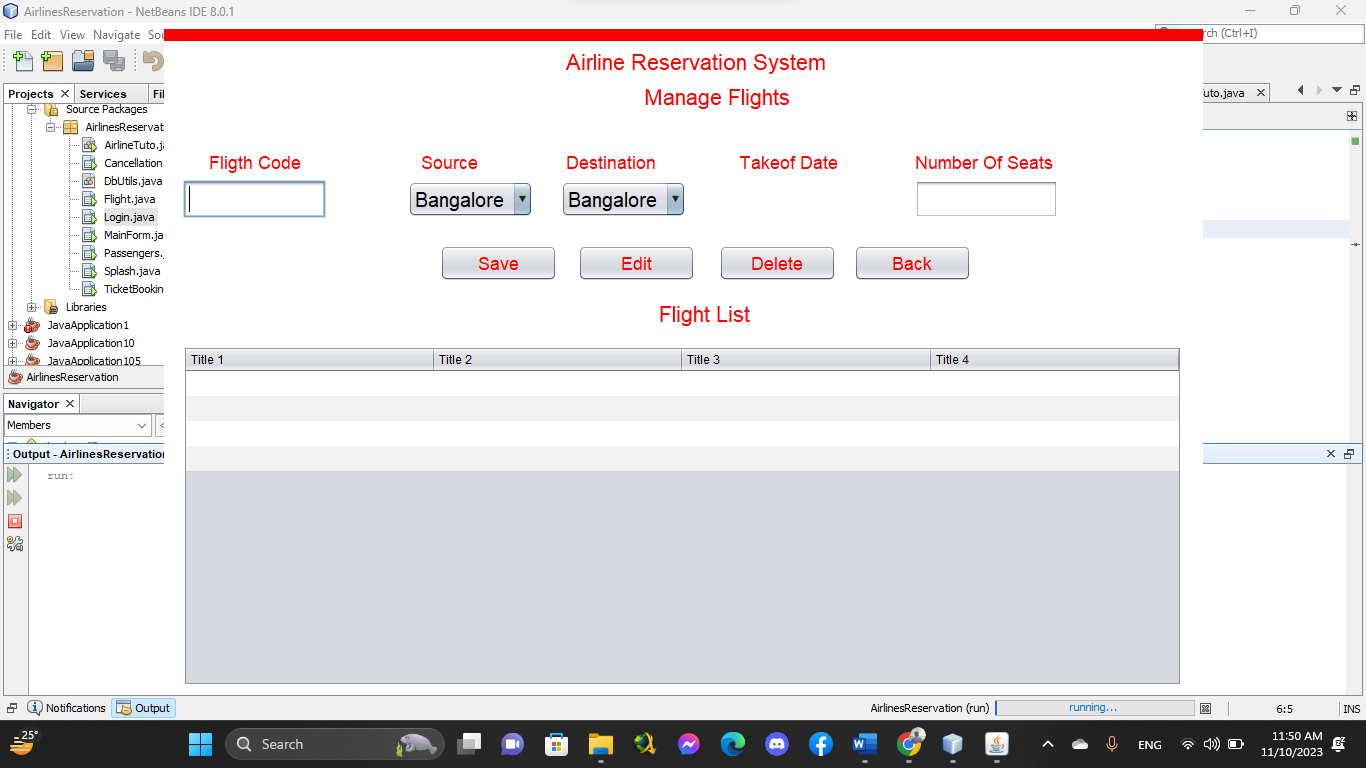
## 

## 54. Cancellation

****

By this **interface** we can cancel the tickets.

## 55. Search

****

By this **interface** we can search the flights.

## 56. Feedback

Users rated their overall experience with the new system as positive with an average rating of 4.5 out of 5. The majority found the system easy to navigate and use, praising its user-friendly interface. Users appreciated the improved real-time display of flight information.

## 57. Conclusions

In the end, we think user manual is like a friendly companion, helping users understand and use the airline reservation system without any hassle.

# Chapter 6: Conclusion and Next Steps

## 58. Conclusion

Finally, we're thrilled to introduce a better way to book flights for MA Tours and Travels. With the help of Java and MySQL, it's now not just easier but also safer and more enjoyable for everyone. Along the way, we faced challenges, turning each one into an opportunity to make things better. Our main aim was to change how people experience booking flights, making it hassle-free. We want to express our thanks to everyone who joined us on this journey partners, users and supporters. This project isn't just about technology; it's about making travel plans simple and secure.

In closing, we see a future where MA Tours and Travels is known for stress-free and safe travel adventures. The success of our project isn't just in the lines of code. It's in the positive changes for travelers. Here's to a future where planning your journey is as pleasant as the journey itself where every traveler feels at ease and excited about their next adventure. Cheers to a simpler, safer and more enjoyable way of booking flights.

## 59. Next Steps

Looking forward, our next steps involve testing the new flight booking system at MA Tours and Travels to make sure it works well, is easy to use and keeps everyone's information safe. We'll ask a group of users to try it out and give us feedback so we can fix any issues and make it even better. At the same time, we want to work closely with travel agencies, airlines and tech partners to improve how everything connects. This collaboration will help us make booking flights even smoother for both users and travel providers. In the next few months, we'll focus on letting more people know about the upgraded system. We'll use ads, team up with travel influencers and use social media to make sure more people hear about it and give it a try.

We're committed to making booking flights at MA Tours and Travels as easy and enjoyable as possible.

## 62. References

In building our travel software, we leaned on various resources to gather valuable insights and knowledge. Here are some key references that played a significant role in guiding our project:

* **"Java: The Complete Reference"** by Herbert Schildt
* **"Database Design for Mere Mortals"** by Michael J. Hernandez
* **"Learning SQL"** by Alan Beaulieu
* **Online Documentation and Developer Communities** (e.g., Stack Overflow)
* **Industry Reports and Blogs on Travel Software Trends**

## 63. Appendix

[Activity Diagram 29](#_Toc150817757)

[Gantt Chart 27](#_Toc150817756)

[Feedback 44](#_Toc150817783)