



# JADE PROJECT

Object Orientated Database (IT606)

Benjamin King, Jayekib den\_Dulk  
Southern Institute of Technology

## Contents

Introduction .....	2
Project Team Members and Assigned Tasks.....	3
Entity Relationship Diagrams .....	4
Flight Booking System Database (Ben) .....	4
Oreti Beach 5K Run Database (Jayekib) .....	4
Flight Booking System Database (Ben) .....	5
Instruction Manual.....	5
Starting the Database .....	6
Navigating the Database .....	6
Extra Features not shown in Class .....	7
Future Improvements .....	7
Interface Design .....	8
Wireframes .....	8
Oreti Beach 5K Run Database (Jayekib) .....	9
Conclusion.....	10

## Introduction

In this project, we have created 2 JADE Databases, with both contain 2 Schemas (a Database & a View Schema (User Interface)).

The first Database we have created is a Flight Booking Database which allows the user to log in as different User-Logins allowing them to perform different tasks. Ben has created this Database as well as the accompanying documentation and ERD.

The second Database we have created is an Oreti Beach 5K Run Database which allows the user to log in as different use logins, allowing them to do different tasks such as search and view result information as well as add/edit/delete runner accounts amongst other tasks. Jayekib has created this Database as well as the ERD.

## Project Team Members and Assigned Tasks

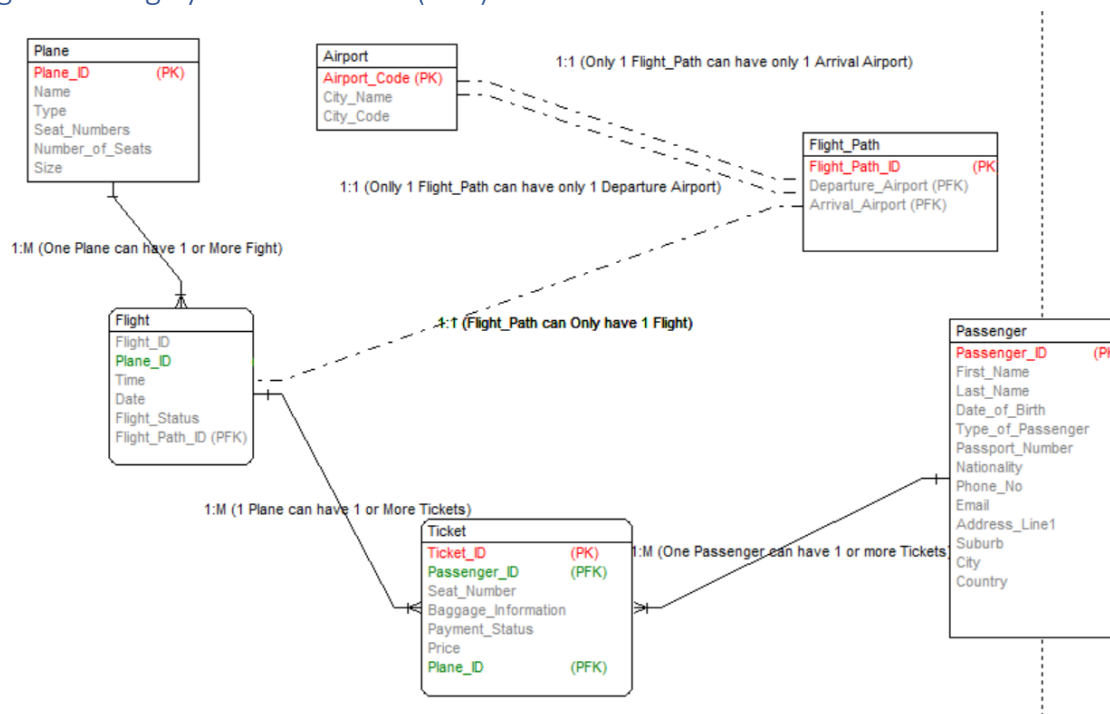
With how this is a Group Project we decided to split up the Databases between the Team Members, with Benjamin King doing the Flight Booking Database while Jayekib den\_Dulk has done the Oreti Beach 5K Database.

We have split up the tasks as seen in the below Table:

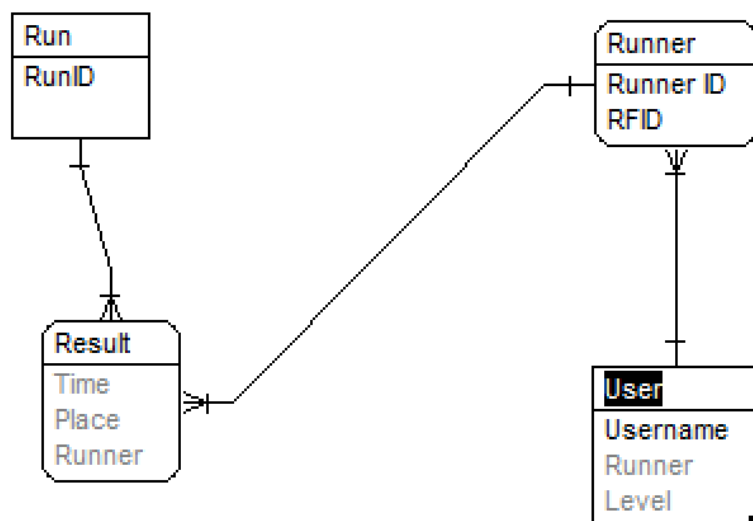
Name	Assigned Tasks
Benjamin King	<b>Flight Booking Database</b> <ul style="list-style-type: none"><li>- Creation of ERD</li><li>- Creation of Database Schema</li><li>- Creation of View Schema</li><li>- Creation of Manual of how to use the Flight Booking Database Application</li><li>- Interface Design</li><li>- Combining of the Documentation as well as the Flight Booking System Documentation</li></ul>
Jayekib den_Dulk	<b>Oreti Beach 5k Run Database</b> <ul style="list-style-type: none"><li>- Creation of ERD</li><li>- Creation of Database Schema</li><li>- Creation of View Schema</li><li>- Creation of Manual of how to use the Flight Booking Database Application</li><li>- Interface Design</li><li>- Documentation for the Oreti Beach 5K Run Database</li></ul>

## Entity Relationship Diagrams

### Flight Booking System Database (Ben)



### Oreti Beach 5K Run Database (Jayekib)



# Flight Booking System Database (Ben)

## Instruction Manual

To Run the Flight Booking System the user needs to DoubleClick the desktop Shortcut or click the Run Application button in Jade.

On first load, the user is required to either enter a username and password or to Click the Guest Access.

Guest Access allows the user to view Flight Information only.

There is also a button which allows the user to generate a new database, such as if this is a database which is being imported into JADE. This is password protected by a secure password –

**JadelsACoolLanguageThatIsAmazing.**

The user can only access anything else in the application if they are an authenticated user with a set access level (Passengers = 2, Managers = 3, Admin = 4).

Once the user has logged in the Main Menu is displayed.

All Object Creations are Recorded in a Log which can be cleared and viewed by an Admin.

From the Main Menu, the user can access different components of the Application.

- View Airports (Passenger & Above)
- Create Airports (Manager & Above)
- View Flights (Everyone)
- Create Flights (Manager & Above)
- View Flight Plans (Manager & Above)
- Create Flight Plans (Manager & Above)
- Create/Edit Passengers (Passenger – Edit Details, Manager & Above – Add Customer)
- Change Password (for User Access = Everyone except Guest)
- View Planes (Manager & Above)
- Create Planes (Manager & Above)
- Create Booking (Passenger only)
- View Ticket (Passenger & Above)
- View Users (Admin)
- View Logs (Admin)
- Create a new Database (Anyone who knows the secret password)

From these the user can access the respective forms showing the information or input boxes needed.

Important Rules:

- A plane must be created first or second
- An Airport must be created second or first
- A Flight Path requires 2 Airports to be created
- A Flight requires a plane and flightpath to be created
- A passenger can be created, where a user is created as well
- A Booking can be created however a flight and passenger must be made. Passenger must be logged in (Managers can't see this due to this – **(due to time constraints we weren't able to allow the manager to emulate the passenger and book flights)**)

- A Booking must be Selected to be able to Manage it (Managers & Above)
- Users can only be created by an Admin unless they are a Passenger where the user and access level are automatically assigned.

### Starting the Database

To start the database double click the desktop shortcut or click the Run Application button in the JADE Client.

If this is the first time you are running the database (with no instances) you need to click the Create New Database and enter the security password – JadelisACoolLanguageThatIsAmazing . Once done instances of the dictionaries will be created, as well as a user of username – admin – and password of – admin – which can be used to sign in once restarting the application.

Using the Credentials (or credentials you already have if the database has already been set up) you can then login to the database.

If all you want to do is View the Flights you can view them via the **Guest Login** button.

### Navigating the Database

After having logged into the Database you will be greeted with the Main Menu. From here you can access several features as discussed earlier depending on your access level.

Only managers or Admins have access to the following features:

Action	Permission Required:	Necessary Notes:
Create Airport	Manager or Admin	Can be created anytime but ensure that the Airport ID is Unique
Create Flight	Manager or Admin	A Flight Path must be already created, as well as a Plane.
Create a Plane	Manager or Admin	A plane can be created at any time. Does not need to have any unique values as has its own Plane ID.
Create Flight Path	Manager or Admin	Must have 2 Airports already Created before creating
Create User	Admin	Can be created anytime however they will not have a passenger instance created (for Managers/Admins etc).
Create Passenger	Manager or Admin	A passenger has automatically associated a User. Ensure that the username is unique (e.g. first letter of First Name and then the last name such as bKing).
View Users	Admin	Users must be created to view this
View Log File	Admin	Logs must exist to view this however a Log is created at Login. Logs can be cleared by

### Extra Features not shown in Class

We have included a number of features which were not shown in class. These are:

- Pictures on Several Forms which have the size set to the size of the Picture Box
- Maximize Button and the Frame disabled set so they don't maximize/adjust.
- A Print Dialogue that appears on the first launch of booking a ticket
- We have changed the visibility of the Select Buttons depending on the previous action which created the form.
- Functions that perform depending on the MsgBox\_OK\_Cancel Action
- Get the Current Date/Time and assign them to the variables and attributes in the Log Files.
- Counting of Instances of Object to see whether the database has been created or not.

### Future Improvements

If we had more time to complete the project the following would be added/improved in the project:

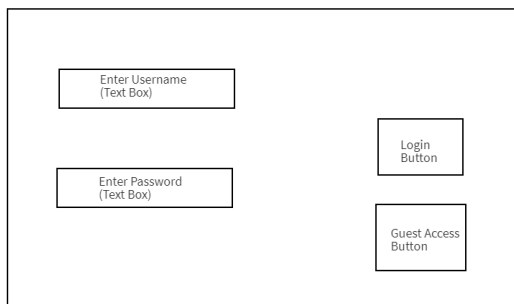
- The ability for Managers and Admins to emulate customers – allowing them to Book Tickets under their name
- An Actual Dedicated Payment Form which will allow the passenger to choose payment type and details (e.g. enter Credit Card Details, Bank Transfer Details)
- Validation checking that the User has entered all the required details on the form and showing a message box if not.
- Not allowing a Flight Path to have the same Arrival and Departure Airport
- Not allowing the user to have the same username (currently throws an exception) and instead show a message box informing the user they need to choose a username
- Ability to Edit objects (such as Planes, Flight Path, Bookings etc)



## Interface Design

### Wireframes

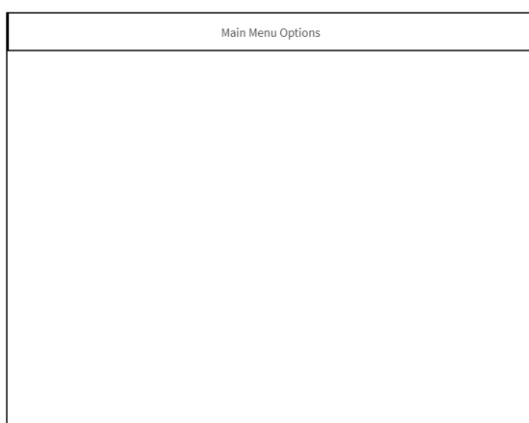
#### Login



A wireframe for a login form. It consists of a large rectangular container. Inside, on the left side, there are two text boxes stacked vertically. The top one is labeled 'Enter Username (Text Box)' and the bottom one is labeled 'Enter Password (Text Box)'. On the right side, there are two buttons stacked vertically. The top button is labeled 'Login Button' and the bottom button is labeled 'Guest Access Button'.

- Text Boxes Allowing the User to enter information (Login Credentials)
- Buttons which launch the respectful commands/forms

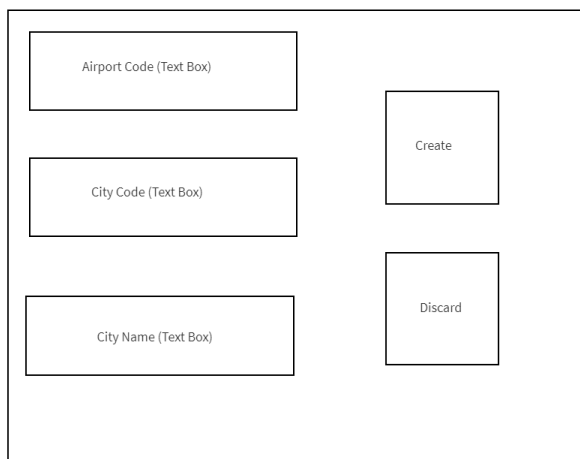
#### Main Menu



A wireframe for a main menu. It consists of a large rectangular container. At the top, there is a horizontal bar labeled 'Main Menu Options'. Below this bar, the rest of the container is empty, representing the area for menu items.

- Main Menu at the top Screen of the Window easily allowing the user to access different forms

#### Create Airport/Flight/Flight Path/User etc



A wireframe for a form to create new entities. It consists of a large rectangular container. On the left side, there are three text boxes stacked vertically. The top one is labeled 'Airport Code (Text Box)', the middle one is labeled 'City Code (Text Box)', and the bottom one is labeled 'City Name (Text Box)'. On the right side, there are two buttons stacked vertically. The top button is labeled 'Create' and the bottom button is labeled 'Discard'.

- Text Boxes allowing the user to Enter information Button
- Buttons to launch forms (e.g. Select Flight Path in the Flights Creation Form) as well as buttons to run Commands etc

### View Airports/Flights/FlightPaths/Users etc

<input type="checkbox"/>	Country	Companies
<input checked="" type="checkbox"/>	USA	Apple Inc , Microsoft
<input type="checkbox"/>	Sweden	IKEA Furnitures , Spotify
<input checked="" type="checkbox"/>	Finland	Nokia Communications

Select

- Tables which allow the user to see information (Objects) that have been created
- Buttons allowing the user to select specific instances of Objects

### Oreti Beach 5K Run Database (Jayekib)

## Conclusion

In Conclusion, we have created a Flight Booking System and Oreti 5K Beach Run Database that allows the user to perform a range of tasks and actions. Throughout the development of the Flight Booking System we have incorporated several self-researched features, as well as noting in this documentation several features that we were unable to incorporate into the database (despite wanting too) due to time constraints.

NAMES: Ben
☒ This assignment is completely my own or my team's original work.

STUDENT SIGNATURE:

*Benjamin King*

### Marking Schedule

Section	Components	Marks
Database	Classes (Flight, Passengers, Tickets, ...)	5
	Properties (attributes/references)	5
	Relationships	5
	Encapsulation (full)	5
	Methods	5
	Collections	5
		5
Report and files	Comments/Layout	10
	Complete reports and files	5
	Extra feature (using functions not covered in class)	50
Interface		Total
	Main page	5
	Login function (level of access)	5
	Menus function	5
	Search function	5
	Printing ticket	5
		5
	Backup (user log file)	10
	Form layout/formatting	5
	Extra content (using controls not covered in class)	5
Presentation	Live demonstration of a random task	
	Good handling of Q&A	
Total Points		100

NAMES: Jayekib

☒ This assignment is completely my own or my team's original work.

STUDENT SIGNATURE:

*Jayekib den Dulk*

### Marking Schedule

Section	Components	Marks
Database	Classes (Flight, Passengers, Tickets, ...)	5
	Properties (attributes/references)	5
	Relationships	5
	Encapsulation (full)	5
	Methods	5
	Collections	5
		5
Report and files	Comments/Layout	10
	Complete reports and files	5
	Extra feature (using functions not covered in class)	50
Interface		Total
	Main page	5
	Login function (level of access)	5
	Menus function	5
	Search function	5
	Printing ticket	5
	Backup (user log file)	5
	Form layout/formatting	10
	Extra content (using controls not covered in class)	5
		5
Presentation	Live demonstration of a random task	
	Good handling of Q&A	
		Total Points
		100