

**COMP132: Advanced Programming
Programming Project Report**

KU Travel Agency

<SOFIIA KHOKHAR, 86015>

<Fall 2024>



Part 1

General Demo Information:

- **Customers (username, password):** [lunaV, vizik13] , [alina,1344] , [maia_hg, 123456]
,[gilbert_b, wdydooyale]
Admin (username, password): [admin, Kacka]
- **Existing reservations:**
- **Existing Travel Packages made by Admin:**
[Luxury Amsterdam Tour, Four Seasons, Standard room , Black Taxi , \$3782.7798,
2025-01-06, 2025-01-08,60]
[Istanbul Deluxe, Atlantis hotel ,Deluxe, Black Taxi, \$3265.44, 2025-05-02, 2025-05-08,40]
[Helsinki Cold, InterContinental hotel ,Suite, Black Taxi, \$4463.4204, 2025-01-22,
2025-01-27,60]

Application usage information

Start : The main.java method of the application should be run. The first page to appear would be the login page.

- Sign up/Login Guide
 - The login page has two fields : Username and password. Both should be filled if the user is registered.
 - If the user is not registered, ‘register’ button should be clicked. It redirects the user to the Registration page.
 - The user is not allowed to leave the fields empty, enter a username longer than ten characters, and enter un matching passwords.
 - If the user does not exist, or the username exists but the password doesn’t match, the screen displays a message.
 - If correct credentials are entered, the system decides if the user is a customer or an admin and redirects him to the next page accordingly.
- User’s Guide

- The user is directed to the main menu. It has four options: Creating Custom Travel Package, Selecting an Existing package, booking individual entities, and user booking history.
- Note: The booking history page could not be created because of time constraints.
- Note: Booking Existing Packages works but it does not save the booked package to the users statistics.
- Admin's Guide
- Admin is directed to the main menu with three options: Creating a new travel package, managing existing travel packages, and viewing user statistics.
- Note: Managing existing travel packages can delete a travel package but not modify it.
- Creating a New Reservation Guide
- A customer can create a new reservation in three ways:
- Create a custom travel package.
- Choose from an existing package.
- Book an individual entity.

Selecting any of these options redirects the user to the page. Each page has multiple combo boxes. The user must select from the combo boxes in order to proceed forward. Subsequent combo boxes are enabled based on previous user input.

At the end, the user receives a confirmation message whether they would like to proceed, the user selected 'yes' to make a reservation.

- Creating a New Travel Package

Creating a travel package involves there steps:

- Flight Selection
- Hotel Selection
- Taxi Selection
- All steps are completed in order and at the end, the travel package is saved.
- User Statistics & Agency Reports

User statistics involve the number of reservations made, total amount spent, number of individual entities booked, individual spendings on each entity.

Users are ranked based on decreasing order of amount spent.

- How to check log information

Unfortunately, I could not complete the log information requirement because of time constraints.

Part 2

Project Design Description:

1. Class Relations

The project consists of multiple packages, each responsible for specific functionalities:

- GUI Package: Handles user interface components and interactions.
- Handler Package: Manages data processing and resource handling.
- Package Components Package: Defines core entities (e.g., flights, hotels) and their managers.
- Travel Package Package: Focuses on creating and managing travel packages.
- User Package: Manages users, including customers and admins, and their data.

Classes in each package interact with others, e.g., UserManager interacts with FileHandler to manage user data.

2. Inheritances, Type Hierarchies, Interfaces, Abstract Classes

- Abstract Classes:
- Manager: Abstract class extended by HotelManager, FlightManager, and TaxiManager.
- Interfaces:
- Comparator: Implemented by UserMoneySpentComparator.
- Inheritance:
- User is the superclass for Admin and Customer.

3. GUI Components

- The GUI Package contains components for different functionalities:
- Admin-related GUIs: Admin dashboard, package management, user management.

- Customer-related GUIs: Package browsing, auto-selection, flight/taxi/hotel selection.
- Login and Registration GUIs: For user authentication.
- Design Highlights:
- Each GUI is built using Java Swing (via Window Builder).
- Navigation is implemented through events triggered by button clicks.
- Scrollable panels for displaying multiple travel packages.

4. `.txt File Processing Details

- File Formats:
- User data, reservations, and travel packages are stored in separate text files.
- Each file uses a custom format (e.g., comma-separated values) or JSON-like structures.
- File Handling:
- FileHandler is responsible for reading and writing data to text files.
- Data is loaded into memory (e.g., maps or lists) during initialization and saved upon termination.

5. Finding Availability for Reservations

- Hotel Availability:
- Iterate through dates between check-in and check-out to find available rooms.
- Flight Availability:
- Check seat availability for the selected flight.
- Taxi Availability:
- Verify if a taxi is free during the requested time.

The agency has implementations of methods to save txt files. There exists a separate text file for each entity. They record the number of reserved entities for a specific date.

6. Calculating the Profits of the Agency

- I was unable to implement this part of the project.

7. Implementation of Save/Load

- Save:
 - Triggered on program termination or at regular intervals.
 - Writes the updated data (users, reservations, profits) to the respective .txt files.
- Load:
 - Occurs at program startup to initialize the system with the latest data.
 - Reads from text files into memory structures (e.g., ArrayList, HashMap).

8. Additional Design Details

- Custom Comparators:
 - UserMoneySpentComparator is used for sorting users based on total spending.
- Singleton Pattern:
 - Used for UserManager to ensure a single instance manages user data throughout the application.
- Travel Package Builder:
 - Follows the Builder pattern to simplify the creation of complex travel packages.
- Error Handling:
 - Includes exception handling for file I/O and invalid user inputs.

Note: I have added the index.html file to the KUtravel package.

References

- Java Full Course for Free. YouTube, uploaded by Bro Code, youtu.be/xk4_1vDrzzo.
- "Java Swing Tutorial 2 : Creating First Project and InvokeLater() Method." YouTube, uploaded by Bijoyan Das, www.youtu.be/XttvrNIIInoU.
- "Java Flushing Output with the Flush Method - Send Buffer to Output Stream - APPFICIAL." YouTube, uploaded by Appficial, youtu.be/MjK3dZTc0Lg.
- "Java GUI - How to Code a GUI in Java | Java Swing How To." YouTube, uploaded by Choobtorials, [U5nfmc4](https://youtu.be/U5nfmc4).
youtube.com/playlist?list=PLWfJeLqH0KUUicazPYhTdnZydkC10h6wB&si=2GqCa70Mu
- "Java Tutorial #89 - Java File Class to Create and Delete Files in Directory." YouTube, uploaded by Programming For Beginners, www.youtu.be/Vq1gcNiA4yE.
- "Java Scanner HasNext() Vs. HasNextLine()." Baeldung.com, www.baeldung.com/java-scanner-hasnext-vs-hasnextline.
- StackOverflow
- "Shuffle or Randomize a List in Java." Geeksforgeeks.Org, www.geeksforgeeks.org/shuffle-or-randomize-a-list-in-java/.