**Report Template: Object-Oriented Programming and Design**

# Student Name:

Kira Zholtikova

# Group:

*ОПД 2*

# Task:

# *Design and develop a console-based flight booking system. The system should read input configuration with flight information from a text file and allow users to interact with the system in different ways.*

# System Model:

A close-up of a paper

Description automatically generated

# GitHub Link:

[Click here](https://github.com/kzholtikova/flight-booking-system)

# Solution Description:

The system ties all classes together. For simplifies flight searches airplanes stored in map first my flightNo 1st char by alphabet (same for users), then by date in the format of time point. Used binary search for easy access to the ticket by id.

# Testing:

- **Steps:**

*Checked commands and argument validation.*

**- Screenshots:**

Help/Exit

# *A screenshot of a computer program Description automatically generated*

Check (valid)A screen shot of a computer

Description automatically generated

Check (invalid)

A screen shot of a computer

Description automatically generated

Book (valid)

A black screen with white text

Description automatically generated

Book (invalid)

A screen shot of a computer screen

Description automatically generated

Return (valid)

A black screen with white text

Description automatically generated

View by ID

A screen shot of a computer program

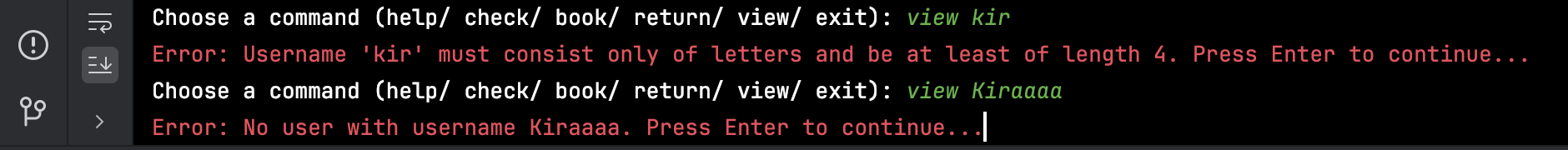
Description automatically generated

View by Username (valid)

A screen shot of a computer

Description automatically generated

View by Usename (invalid)



View by Flight (valid)

A screen shot of a computer

Description automatically generated

# Conclusions:

- **What was implemented:**

*Classes System, InpurReader, User, Ticket, Airplane, Seat.*

- **Differences from the proposed model:**

*The core structure corresponds to the plan, though some fields are either removed or added.*

*The most significant change is that Seat now doesn’t hold user pointer as it caused circular dependency. Instead, ticketId field is introduced.*

**- Additional Notes:**

*Faced circular dependency while trying to optimize the commands. First I didn’t comprehend what the compiler tried to tell me. When I ‘d identified the problem there was a bit of a challenge what relation to break to not affect efficiency. I found using weak and shared pointers my best bet.*