IV. Analysis

4.1 Identify Boundary,Controller,Entity, and Tool classes

We separated the source codes of the project into 4 parts: Boundary,Controller,Entity, and Tool classes.

4.1.1 Boundary Class

The boundary class shoulders the responsibility for presenting our Graphic User Interface , listening to their actions, and interacting with them. The boundary class is at the top of our software architecture and only interacts with the controller classes, making our software loose coupling. In our system, the boundary classes include:

```
Boundary
      AdditionalServiceView.java
      CheckinView.java
      ChooseFlightView.java
      ChooseMealView.java
      ChooseSeatView.java
      ConfirmationView.java
      CreditcardView.java
      IDNoCheckinView.java
      WelcomeView.java
```

4.1.2 Controller Class

We use the controller class to link the Boundary class to the Entity class, which processes data requests sent by the Boundary class and linking to the corresponding JSON-data file based on the content of the request. After getting the data from the JSON-data file in the project, it parses the return packet and returns the specific data to the Boundary class. In our system, the controller classes include:

```
Controller
      AdditionalServiceController.java
      CheckinController.java
      ChooseFlightController.java
      ChooseMealController.java
      ChooseSeatController.java
      ConfirmationController.java
      Controller.java
      CreditCardController.java
      IDNoCheckinController.java
      WelcomeController.java
```

4.1.3 Entity Class

The entity class in our project manages the data administration. It consists of the basic data structure of the passengers data and all the get and set methods to modify and search. In our system, entity class includes:

```
Entity
      BookingInformation.java
```

The get and set methods include:

```
    getFirstName()
        setFirstName(String)
        getLastName()
        setLastName(String)
        getFlightNumber()
        setFlightNumber(String)
        getBoardingTime()
        setBoardingTime(String)
        getBoardingGate()
        setBoardingGate(String)
        getSeat()
        setSeat(String)
        getPrimaryFood()
        setPrimaryFood(String)
        getCredFirst()
        setCredFirst(String)
        getCredSecond()
        setCredSecond(String)
        getCredNumber()
        setCredNumber(String)
        getSecurCode()
        setSecurCode(String)
        setExtraServiceFee(int)
        getExtraServiceFee()
        setExtraService(String)
        getStartWhere()
        getDestWhere()
        getDuringTime()
        getBookingNo()
        getExtraService()
        setStartWhere(String)
        setDestWhere(String)
        setDuringTime(String)
        setBookingNo(int)
        getIdNo()
        setIdNo(String)
```

```
getFilePath()
setFilePath(String)
getSeatHelpNumber()
setSeatHelpNumber(int)
```

4.1.4 Tool class

The Tool class handles the file reading and file writing for the JSON files which store the data as a database. `FileReaderWriter` class handles the Java file reading and writing, while `JsonTool` handles the conversion of data into JSON to store. The Tool classes include:

```
Tool
    FileReaderWriter.java
    FileReaderWriterTest.java
    JsonTool.java
```

4.2 Conceptual Class Diagram

Having analyzed the relationships and associations between classes, we drew the conceptual dia-gram. (Click here to view the whole Conceptual Class Diagram)

4.3 Reusability

To make our code re-usable, we store our data in an appropriate data structure and design our code following the design process. For example, we create JSON file `checkinData`to store all the check in data,`flightData` file to store all the plane information, and `passengerData` file to store the passengers data. Thus, the code can easily adapt to changes, the system can add new categories and new membership is up a notch by just modifying the JSON file.
Besides, the methodology we used to store data into JSON file is just a new way to store the data aside from sql-database technique.