

Programming Methodology

Group Name: “AIRROR”

Members: 1) Chotiwat Silarak (#6731312821) 2) Yanaphat Chatphokhinkun (#6731313421)

What is our project about?

We are making application; specifically, “Airplane Booking” Application. Our application provides basic functions such as booking, purchases list and sorting features. In addition, it contains user sign in and sign up function.

Implementation Details

Please note that : + (public), - (private), # (protected), underlined (static), *italic* (abstract), All_CAPS (final)

1. Package application

1.1 Class MyApplication

Fields

- <u>Scene scene</u>	Scene of the Application
- <u>Stage stage</u>	Stage of the Application
- <u>ArrayList<Parent> listOfRoots</u>	List of all roots/parents node in the scene graph

Methods

+ <u>void main(String[] args)</u>	Main Function
+ void start(Stage stage)	Start the Application
+ <u>void loadData()</u>	Loading the data such as flights, purchases, airline name and etc. from Text File (Via Utils.IOReaderWriter.java)
+ <u>void keepData()</u>	Putting the data such as flights, purchases, username and password, and etc. to Text File (Via Utils.IOReaderWriter.java)
+ <u>void setStageStyle(Stage stage)</u>	Setting the stage's style and add Event Handler
+ Getter & Setter	-

2. Package Exceptions

2.1 Class IncorrectAndPartialFillFormException

Methods

+ IncorrectAndPartialFillFormException()	Initiate incorrect and partial filling of the form Exception
--	--

2.2 Class IncorrectFillFormException

Methods

+ IncorrectFillFormException()	Initiate incorrect filling of the form Exception
--------------------------------	--

2.1 Class PartialFillFormException

Methods

+ PartialFillFormException()	Initiate partial filling of the form Exception
------------------------------	--

3. Package interfaces

3.1 Interface Searchable

Methods

void addSearchEachPane(Object obj)	Used by any classes which can search or filter the data.
------------------------------------	--

4. Package logics

4.1 FlightData

Fields

- String departAbbr	Abbreviation of a departure (such as BKK, CNX)
- String destinyAbbr	Abbreviation of a destination (such as BKK, CNX)
- String airlineName	Name of an airline
- String departAirportName	Name of departure's airport
- String destinyAirportName	Name of destination's airport
- ImageView airlineImage	Image of an airline's logo
- String departTime	Time of departure (such as 19:14)
- String arrivalTime	Time of arrival (such as 19:14)
- double price	Price of a flight
- LocalDate departDate	Date of departure

Methods

+ FlightData(String departAbbr, String departAirportName, String destinyAbbr, String destinyAirportName, String airlineName, ImageView airlineImage, String departTime, String arrivalTime, double price, LocalDate departDate)	Initialize all fields
+ String toString()	Return departAbbr + "," + departAirportName + "," + destinyAbbr + "," + destinyAirportName + "," + airlineName + "," + departTime + "," + arrivalTime + "," + price + "," + departDate ("dd/MM/yyyy")
+ Getter & Setter	-

4.2 PurchaseData

Fields

- String classes	Class (Such as Economy Class)
- int adultField	The Number of adults
- int childrenField	The Number of children
- int toddlerField	The Number of toddlers
- FlightData flightData	The purchased flight

+ PurchaseData(FlightData flightData, String classes, int adultField, int childrenField, int toddlerField)	Initialize all fields
+ String toString()	Return flightData.toString() + “,” + classes + “,” + adultField + “,” + childrenField + “,” + toddlerField
+ Getter & Setter	-

4.3 RequestData

Fields

- String departField	A departure (such as “CNX – Chiang Mai International Airport”)
- String destinyField	A destination (such as “BKK – Don Mueng International Airport”)
- LocalDate departDate	Date of departure
- LocalDate destinyDate	Date of destination
- int adultField	The number of adults
- int childrenField	The number of children
- int toddlerField	The number of toddlers
- boolean isReturn	True if it is two-way flights. Otherwise, it is false.
- String classes	Class (Such as Economy Class)

Methods

+ RequestData(String departField, String destinyField, LocalDate departDate, LocalDate destinyDate, int adultField, int childrenField, int toddlerField, Boolean isReturn, String classes)	Initialize all fields
+ RequestData(RequestData requestData)	Copy constructor
+ Getter & Setter	-

4.4 UserData (No Implementation, just for the future user features)

5. Package pages

5.1 Abstract Class Page

Fields

# Canvas canvas	A canvas of the instance
-----------------	--------------------------

Methods

+ void setHeader(GraphicsContext gc)	Set header of the page
+ void setStyle()	Style the page
+ void setTopLeftAnchor(Node node, double left, double top)	AnchorPane.setTop(node, top) AnchorPane.setLeft(node, left)
+ Getter	

5.2 Class LoginPage extends Page

Fields

- <u>Map<String, String> passwords</u>	A map which has a username as a key and a password as a value
+ <u>String loginUsername</u>	A Username String

Methods

+ LoginPage()	Initialize Login page
+ <i>void setHeader(GraphicsContext gc)</i>	No implementation needed
+ <i>void setStyle()</i>	Set style of the login page
+ <u>Getter & Setter</u>	-

5.3 Class MainPage extends Page

Fields

+ <u>MainPage mainPageInstance</u>	An instance of MainPage
+ <u>ScrollPane departSearchBar</u>	A searchbar for departure
+ <u>ScrollPane destinySearchBar</u>	A searchbar for destination
+ <u>TextField departField</u>	An input TextField for depature
+ <u>TextField destinyField</u>	An input TextField for destination
+ <u>DatePicker departDatePicker</u>	A DatePicker for picking depart date
+ <u>DatePicker destinyDatePicker</u>	A DatePicker for picking returning date

+ <u>TextField adultField</u>	An input TextField for the number of adults
+ <u>TextField childrenField</u>	An input TextField for the number of children
+ <u>TextField toddlerField</u>	An input TextField for the number of toddler
+ <u>CheckBox checkbox</u>	A CheckBox for checking whether the flight will be two-way or not
+ <u>ChoiceBox<String> choiceBox</u>	A ChoiceBox for selecting class (Such as Economy Class)

Methods

- MainPage()	Initialize home page
+ void <i>setHeader(GraphicsContext gc)</i>	Create header section using BorderPane
+ void <i>setStyle()</i>	Set width of MainPage to UIComponent.USER_MAX_SCREEN_WIDTH
+ void <i>setMiddle()</i>	Create and style content section (A booking system)
+ <u>void reset()</u>	Reset the content section (A booking system)
+ <u>MainPage getInstance(Boolean bool)</u>	Create MainPage's instance Bool specifies whether or not we are going to get new mainPageInstance (True) or not (False)

5.4 Class FlightAvailablePage extends Page

Fields

+ <u>ArrayList<FlightData> flightsList</u>	All available flights
+ ArrayList<String> airlinesCondition	Conditions for filtering
+ <u>Map<String, String> townName</u>	A map which has an abbreviated city name as key and full name as value (Such as (“BKK”, “Bangkok”))
- RequestData requestData	An RequestData for finding flights

Methods

+ FlightAvailablePage(RequestData requestData)	Initialize available flights page
+ <i>void setStyle()</i>	No implementation needed
+ void setHeader(GraphicsContext gc)	Create header section of flight available page
+ void setMiddle(GraphicsContext gc, RequestData requestData)	Create content section of flight available page
+ void <u>setSortingEventListener(SearchFlightAvailablePane searchPane, ImageView imageView, Comparator comparator)</u>	Set event handler to ImageView, which when it is clicked, the available flights will sort determined by comparator
+ void getByConditions(SearchFlightAvailablePane searchPane, RequestData requestData)	Get the flights determined by the condition strings (Airline names)
+ void <u>setAirlineEventListener(SearchFlightAvailablePane</u>	Add event handler to checkbox for filtering by the

searchPane, RequestData requestData, CheckBox checkbox, String condition)	condition strings (Airline names)
+ Getter & Setter / <u>Getter & Setter</u>	-

5.5 Class PurchasePage extends Page

Fields

+ <u>ArrayList<PurchaseData> pendingList</u>	An ArrayList which contains the pending flights which are not confirmed yet.
+ <u>ArrayList<PurchaseData> purchasesList</u>	An ArrayList which contains all of PurchaseData

Methods

+ PurchasePage()	Initialize purchase page
+ <i>void setHeader(GraphicsContext gc)</i>	Create header section of purchase page
+ <i>void setMiddle(GraphicsContext gc)</i>	Create content section of purchase page (Purchases List)
+ <i>void setStyle()</i>	No implementation needed
+ <u>Getter & Setter</u>	-

5.6 Class AboutUsPage extends Page

Fields

+ <u>AboutUsPage</u> aboutUsPageInstance	Instance of AboutUsPage
--	-------------------------

Methods

- AboutUs()	Initialize about us page
+ void setHeader(GraphicsContext gc)	Create header section of about us page
+ void setMiddle(GraphicsContext gc)	Create content section of about us page
+ void setStyle()	No implementation needed
+ <u>Getter</u>	-

6. Package panes

6.1 Class SearchAirportPane extends VBox implements Searchable

Fields

+ <u>ArrayList<String></u> airports	List of all airports
---	----------------------

Methods

+ SearchAirportPane(String condition)	Initialize SearchAirportPane with the given string condition
+ void addSearchEachPane(Object obj)	Add SearchAirportEachPane to SearchAirportPane
+ <u>Getter</u>	-

6.2 Class SearchFlightAvailablePane extends VBox implements Searchable

Fields

+ ArrayList<FlightData> flightsSelected	Contain flights which satisfies the requestData
---	---

+ SearchFlightAvailablePane(RequestData requestData)	Initialize SearchFlightAvailablePane with the given RequestData
+ void addSearchEachPane(Object obj)	Add SearchFlightAvailableEachPane to SearchFlightAvailablePane
+ Getter	-

6.3 Class SearchPurchasePane extends VBox implements Searchable

Methods

+ SearchPurchasePane()	Initialize SearchPurchasePane (Purchase List)
+ void addSearchEachPane(Object obj)	Add SearchPurchaseEachPane to SearchPurchasePane

6.4 Class SearchAirportEachPane extends BorderPane

Fields

- String name	A name of an airport
---------------	----------------------

Methods

+ SearchAirportEachPane(String name, double prefWidth)	Initialize SearchAirportEachPane with the given string name and double prefWidth
+ Getter	-

6.5 Class SearchFlightAvailableEachPane extends AnchorPane

Fields

- FlightData flightData	FlightData of a flight
-------------------------	------------------------

+ SearchFlightAvailableEachPane(FlightData flightData, RequestData requestData, double prefWidth, double prefHeight)	Initialize SearchFlightAvailableEachPane with the given FlightData RequestData and double prefWidth and prefHeight
+ <u>setTopLeftAnchor(Node node, double left, double top)</u>	AnchorPane.setTopAnchor(node, top); AnchorPane.setLeftAnchor(node, left);
+ Getter	-

6.6 Class SearchPurchaseEachPane extends AnchorPane

Fields

- PurchaseData purchaseData	PurchaseData of a purchase
-----------------------------	----------------------------

Methods

+ SearchPurchaseEachPane(PurchaseData purchaseData, int number, SearchPurchasePane purchasePane)	Initialize SearchPurchaseEachPane with the given PurchaseData, the number(Booking number) and SearchPurchasePane
+ <u>setTopLeftAnchor(Node node, double left, double top)</u>	AnchorPane.setTopAnchor(node, top); AnchorPane.setLeftAnchor(node, left);
+ Getter	-

7. Package utils

7.1 Class Goto

Methods

+ <u>void goToMainPage(Boolean bool)</u>	Go to main page
+ <u>void goToFlightAvailablePage(RequestData)</u>	Go to available flights page
+ <u>void goToPurchasePage()</u>	Go to purchase page
+ <u>void goToAboutUsPage()</u>	Go to about us page
+ <u>void goToLoginPage()</u>	Go to login page
+ <u>void back()</u>	Go back to previous page

7.2 Class IOReaderWriter

Fields

+ <u>Scanner sc</u>	A scanner for reading file
---------------------	----------------------------

Methods

+ <u>ArrayList<String></u> <u>getStringsFromTextFile(String path)</u>	Read text file to ArrayList<String> internally (Inside a jar file)
+ <u>ArrayList<String></u> <u>getStringsFromTextFileExternally(String fileName) throws Exception</u>	Read text file to ArrayList<String> externally (Outside a jar file)
+ <u>ArrayList<FlightData></u> <u>getListOfFlightData(String path)</u>	Read text file to ArrayList<FlightData> internally
+ <u>ArrayList<FlightData></u> <u>getListOfFlightDataExternally(String fileName) throws Exception</u>	Read text file to ArrayList<FlightData> externally
+ <u>ArrayList<PurchaseData></u> <u>getListOfPurchaseDataExternally(String fileName) throws Exception</u>	Read text file to ArrayList<PurchaseData> externally
+ <u>Map<String, String></u> <u>getMap(String path)</u>	Read text file to Map<String, String> internally
+ <u>Map<String, String></u> <u>getMapExternally(String fileName)</u>	Read text file to Map<String, String> externally
+ <u>void</u> <u>writeStringsExternally(ArrayList<Object> objectsList, String filename)</u>	Write text file externally from ArrayList<Object>

7.2 Class UIComponent

Fields

+ <u>double USER_MAX_SCREEN_WIDTH</u>	Maximum screen's width
+ <u>double USER_MAX_SCREEN_HEIGHT</u>	Maximum screen's Height

Methods

+ <u>Label getLabel(String name, int fontSize)</u>	Return Label with the "name" text and font size set to "fontSize"
+ <u>Text getText(String name, int fontSize)</u>	Return text with the "name" text and font size set to "fontSize"
+ <u>Text getText(String name, Font font)</u>	Return text with the "name" text and set font to "font"
+ <u>ImageView getImageVlew(String url_string, double fitHeight, Boolean isPreserveRatio)</u>	Return ImageView located at "url_string" and set it fitHeight to "fitHeight" and set PreserveRatio to "isPreserveRatio"
+ <u>Image getImage(String url_string)</u>	Return Image located at "url_string"
+ <u>Image getImage(String url_string, double requestedWidth, double requestedHeight, boolean preserveRatio, boolean smooth)</u>	Return Image located at "url_string" and set it width to "requestedWidth", set it height to "requestedHeight", set preserveRatio to "preserveRatio" and set smooth to "smooth"
+ <u>TextField getTextField(String text, int prefWidth, int prefHeight)</u>	Return TextField with the "text" text and set it prefWidth to "prefWidth" and set it prefHeight to "prefHeight"

+ <u>void drawLine(double x1, double y1, double x2, double y2, GraphicsContext gc)</u>	Draw line in canvas between (x1, y1) and (x2, y2)
+ <u>int strHourToIntMin(String time)</u>	Return the minutes required to go from 0:00 to time (e.g. time = 18:00 -> return 18 * 60 + 0 * 0)
+ <u>String intMinToStrHour(int time)</u>	Reverse of strHourToIntMin(String time)
+ <u>int getDuration(String departTime, String arrivalTime)</u>	Return the difference in min between two points of time (e.g. departTime = 18:00, arrivalTime = 18:50 -> return 50)
+ <u>boolean isBefore(String time1, String time2)</u>	Return true if time1 is before time 2. Otherwise, return false.
+ <u>void setFont(Label[] arr, Font font)</u>	Set every label's font in arr to "font"
+ <u>DatePicker getDatePicker()</u>	Return new DatePicker()
+ <u>CheckBox getCheckBox()</u>	Return new CheckBox()
+ <u>ChoiceBox<String> getChoiceBox()</u>	Return new ChoiceBox<String>()
+ <u>Button getButton(String name)</u>	Return new Button(name)