

SEP Projektmappe

PROJEKTMAPPE DES PROJEKTES

Super AirlinE Management Program

Dokumentation des Projektes

Gruppe I:

Chen, Bowen

Constantin, Melanie

Eichner, Dennis

Gohres, Kevin

Utkun, Gülistan

Hinweis

An vielen Stellen findet ihr im Dokument folgendes Kästchen:

Dies ist eine Hilfestellung.

Diese Kästen dienen dazu, euch kurze Informationen über Ziele und Inhalte der jeweiligen Abschnitte zu geben. Auch die Beispiele und Templates dienen dazu, euch bei der Dokumentation eures Projektes zu unterstützen. **Sowohl die Kästchen als auch die Beispiele und Templates sind spätestens zur finalen Abgabe der Projektmappe vollständig zu entfernen.** Betrachtet dieses Dokument bitte nicht als Aufgabe, die man von oben nach unten abarbeiten soll; es soll vielmehr als durchgängige Dokumentation eurer Projektarbeit dienen und fortlaufend erweitert bzw. angepasst werden, sodass am Ende des SEPs der Entwicklungsprozess eurer Software vollständig dokumentiert ist.

Das SEP-Team wünscht euch
viel Erfolg
bei der Bearbeitung der Hauptaufgabe!!!

Inhalt

Projektbeschreibung	4
Iteration I	6
User-Stories	6
Papierprototypen	10
Aktivitätsdiagramm	17
Datenbankschema	21
Strukturdiagramm (Klassendiagramm oder Objektdiagramm)	21
Funktionalitätsplanung	23
Iteration II	25
User-Stories	25
Papierprototypen	29
Aktivitätsdiagramm	35
Datenbankschema:	38
Strukturdiagramm (Klassendiagramm oder Objektdiagramm)	40
Funktionalitätsplanung	45
Unittests	46
Systemtests	47
Iteration III	51
User-Stories	51
Papierprototypen	52
Aktivitätsdiagramm	53
Strukturdiagramm (Klassendiagramm oder Objektdiagramm)	53
Funktionalitätsplanung	54
Unittests	55
Systemtests	55
Nutzerhandbuch	56
Technische Anforderungen	56
Installationsanleitung	56
Bedienungsanleitung	56

Projektbeschreibung

In diesem Abschnitt soll die Projektbeschreibung abgedruckt werden, die ihr als Aufgabenbeschreibung von eurem Betreuer erhalten habt. Sie dient als initiales Anforderungsdokument für eure Spezifikationsaktivitäten.

In der ersten Iteration werden die grundlegenden Strukturen implementiert, auf deren Grundlage das Programm basiert. Hierzu gehört das Einlesen der Datensätze für die Flugzeuge und die Flughäfen sowie die persistente Speicherung aller Daten.

Zunächst soll ein Administratorkonto angelegt werden, über das es möglich ist, FluggesellschaftsManager anzulegen. Jedes Konto soll hierbei über einen Benutzernamen und ein Passwort verfügen. Passwörter sollen grundsätzlich gesalzt² und gehasht gespeichert werden.

Im Folgenden werden die zu implementierenden Funktionen des Fluggesellschafts-Managers beschrieben. Ein Fluggesellschafts-Manager soll genau eine Fluggesellschaft anlegen können. Eine Fluggesellschaft verfügt hierbei über einen Namen, ein Land (Hauptstandort) und ein Budget. Das Budget soll genutzt werden, um Flugzeuge (siehe Datensatz) kaufen zu können. Dabei kann davon ausgegangen werden, dass jedes Flugzeug in beliebiger Zahl zur Verfügung steht. Der Fluggesellschafts-Manager soll Fluglinien anlegen können, die im späteren Verlauf als Vorlage für die Instanziierung von Flügen genutzt werden. Eine Fluglinie verfügt hierbei über einen Start- und Zielflughafen, ein Startdatum (Jungfernflug), ein Intervall (täglich, alle drei Tage, wöchentlich), ein Flugzeug (welches natürlich im Besitz der Fluggesellschaft sein muss), eine Sitzplatzverteilung (Anzahl Economy/Business, wobei ein Business-Sitz zwei Sitzen gemäß Datensatz entspricht und nur maximal 25% aller Sitze als Business-Sitze deklariert werden können) und konkrete Preise für die Sitze in den verschiedenen Klassen. Bei der Zuweisung eines Flugzeuges soll berücksichtigt werden, ob dieses für die gewählte Strecke geeignet ist (z.B. anhand der Reichweite).

Ein Fluggesellschafts-Manager soll sich zudem eine Gesamtübersicht über alle Fluglinien seiner Fluggesellschaft anzeigen lassen können, sowie einzelne Fluglinien auswählen und bearbeiten (Änderung des Flugzeugs, Sitzplatzverteilung, Preise, Intervall) können.

In der zweiten Iteration sollen die Funktionalitäten aus Iteration 1 wie folgt erweitert werden: Es sollen auf Basis der Fluglinien konkrete Flüge instanziert werden können, die maximal 6 Monate nach Startdatum angesetzt sind. Zudem soll dem Fluggesellschafts-Manager eine Gesamtübersicht über alle Flüge zur Verfügung stehen.

Außerdem sollen in dieser Iteration die Funktionalitäten des Kunden wie folgt implementiert werden: Ein Kunde soll sich im Super Airline Management Program registrieren können. Ein Kundenkonto soll hierbei über einen Benutzernamen, Passwort, Vor- und Nachname, Adresse, Telefonnummer und ein Postfach verfügen. Auch dieses Passwort soll gesalzt und gehasht gespeichert werden. Ein Kunde soll Flüge buchen können. Hierzu gehört das Suchen von Flügen unter Angabe eines Start- und Zielflughafens, eines Datums, sowie einer gewünschten Anzahl an Sitzplätzen. Dem Kunden sollen

gemäß seinen Angaben Direkt- und Multistopp-Flüge angezeigt werden, welche maximal 3 Tage vor bzw. nach dem Datum, das er angegeben hat, stattfinden. Darüber hinaus soll ein System für die Auswahl und Reservierung von Sitzplätzen implementiert werden. Jeder Kunde soll zudem über ein Flugkilometerkonto für die Anzeige der Kilometer die geflogen wurden, sowie ein CO2-Konto für die Anzeige des CO2-Verbrauchs verfügen. Die Gesamtkilometeranzahl ergibt sich aus der Summe der Kilometer aller Einzelflüge, der CO2-Gesamtausstoß ergibt sich aus der Summe der CO2-Ausstöße der Einzelflüge:

$$\text{Flugdistanz (km)} * 0,0571 \text{ kg CO}_2 / \text{km} * \text{Anzahl Sitzplätze (Buchung)}$$

Zuletzt soll eine Buchungsübersicht implementiert werden, über die ein Kunde alle Informationen über die von ihm gebuchten Flüge gemäß seinen Buchungen einsehen kann. Diese Fluginformationen soll er sich zudem als *.pdf-Datei exportieren lassen können.

In der dritten Iteration sollen die Funktionalitäten des Fluggesellschafts-Managers wie folgt erweitert werden: In der Gesamtübersicht über alle Fluglinien einer Fluggesellschaft soll die Möglichkeit bestehen, sich eine Karte anzeigen zu lassen, in der alle Strecken der Fluglinien eingezeichnet sind.

Außerdem soll die Gesamtübersicht über alle Flüge aus Iteration 2 wie folgt erweitert werden: Für jeden Flug soll die Rentabilität berechnet werden. Hierzu soll für jeden Flug der erwartete Gewinn auf Basis folgender Faktoren berechnet werden:

$$\text{Kerosinkosten: } 3,58\text{€} * \text{Distanz (km)}/100 * \text{Anzahl Sitze (Flugzeug)}$$

$$\text{Einnahmen: Anzahl Sitzplätze Buchungen} * \text{Preis der entsprechenden Klasse}$$

$$\text{Flughafenpauschale: } 0,05\text{€}/\text{m} * (\text{Länge Startbahn (m)} + \text{Länge Landebahn (m)})$$

Ist ein Flug nicht rentabel, soll der Fluggesellschafts-Manager diesen stornieren können. Da im Falle einer Stornierung die Kosten für sich erübrigende Zubringer- sowie Anschlussflüge übernommen werden müssen, ergeben sich die Stornierungskosten aus den Preisen der wegfallenden Sitzplätze auf eben solchen Flügen. Im Falle einer Stornierung eines Flugs sollen alle betroffenen Kunden benachrichtigt und ihre Konten entsprechend aktualisiert werden. Hierzu soll eine Nachricht an das Postfach des betroffenen Kunden und via Telegram-Messenger 5 an die Handynummer des Kunden gesendet werden.

Die Funktionalitäten des Kunden sollen wie folgt ergänzt werden: Ein Kunde soll Informationen über seinen Abflughafen abrufen können. Hierzu gehört das „Berechnen“ einer ÖPNV- und Autoroute zum Flughafen. Außerdem soll er folgende Informationen zu seinem Zielort abrufen können: Hotels in der Nähe, Attraktionen/Sehenswürdigkeiten in der Nähe, Restaurants in der Nähe, Impressionen des Zielorts (Fotos und Videos).

Iteration I

User-Stories

User Story-ID	User Story 1
User Story-Beschreibung	Als Nutzer des Systems möchte ich Datensätze für Flugzeuge einlesen können, damit ich diese im System persistent speichern kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Kevin Gohres
Abhängigkeiten zu anderen User Stories	8,9,11

User Story-ID	User Story 2
User Story-Beschreibung	Als Nutzer des Systems möchte ich Datensätze für Flughäfen einlesen können, damit ich diese im System persistent speichern kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Kevin Gohres
Abhängigkeiten zu anderen User Stories	9,11

User Story-ID	User Story 3
User Story-Beschreibung	Als Admin möchte ich Fluggesellschafts-Manager anlegen können, damit diese ebenfalls mit dem System arbeiten können.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Bowen Chen
Abhängigkeiten zu anderen User Stories	

User Story-ID	User Story 4
User Story-Beschreibung	Als Nutzer des Systems möchte ich, dass mein Konto einen Benutzernamen und ein Passwort hat, damit ich mich authentifizieren kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Bowen Chen
Abhängigkeiten zu anderen User Stories	5

User Story-ID	User Story 5
User Story-Beschreibung	Als Nutzer des Systems möchte ich, dass mein Passwort gehashed und gesaluted gespeichert wird, damit dies sicherer ist.
Geschätzter Realisierungsaufwand	1 PT
Priorität	mittel
Autor(en)	Kevin Gohres
Abhängigkeiten zu anderen User Stories	4

User Story-ID	User Story 6
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich eine Fluggesellschaft anlegen können, damit ich diese im System verwalten kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	1,2,3,7

User Story-ID	User Story 7
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich, dass eine Fluggesellschaft über einen Namen, ein Land und ein Budget verfügt,

	damit ich z. B. das Budget beim Kaufen von Flugzeugen berücksichtigen kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	6

User Story-ID	User Story 8
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich Flugzeuge kaufen können, damit meine Fluggesellschaft ausreichend Flüge anbieten kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	1,7

User Story-ID	User Story 9
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich Fluglinien anlegen können, damit diese als Vorlage für die Instanziierung von Flügen genutzt werden können.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Melanie Constantin
Abhängigkeiten zu anderen User Stories	1,2

User Story-ID	User Story 10
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich, dass eine Fluglinie über einen Startflughafen, Zielflughafen, ein Startdatum, ein Intervall (täglich, alle drei Tage, wöchentlich), ein Flugzeug (welches natürlich im Besitz der Fluggesellschaft sein muss), eine Sitzplatzverteilung

	(Anzahl Economy/Business, wobei ein Business-Sitz zwei Sitzen gemäß Datensatz [1] entspricht und nur maximal 25% aller Sitze als Business-Sitze deklariert werden können) und konkrete Preise für die Sitze in den verschiedenen Klassen verfügt, damit ich diese entsprechend verwalten kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	9

User Story-ID	User Story 11
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich, dass bei der Zuweisung eines Flugzeuges berücksichtigt wird, ob dieses für die gewählte Strecke geeignet ist (z. B. anhand der Reichweite), damit auch nur Flugzeuge, die den Anforderungen eines Fluges entsprechen, diesem zugewiesen werden können.
Geschätzter Realisierungsaufwand	1 PT
Priorität	mittel
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	1,2

User Story-ID	User Story 12
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich mir eine Gesamtübersicht anzeigen lassen können, in der ich einzelne Flüge auswählen und bearbeiten (Änderung des Flugzeugs, Sitzplatzverteilung, Preise, Intervall) kann, damit ich die Flüge meiner Fluggesellschaft verwalten kann.
Geschätzter Realisierungsaufwand	1 PT
Priorität	hoch
Autor(en)	Melanie Constantin
Abhängigkeiten zu anderen User Stories	

Papierprototypen

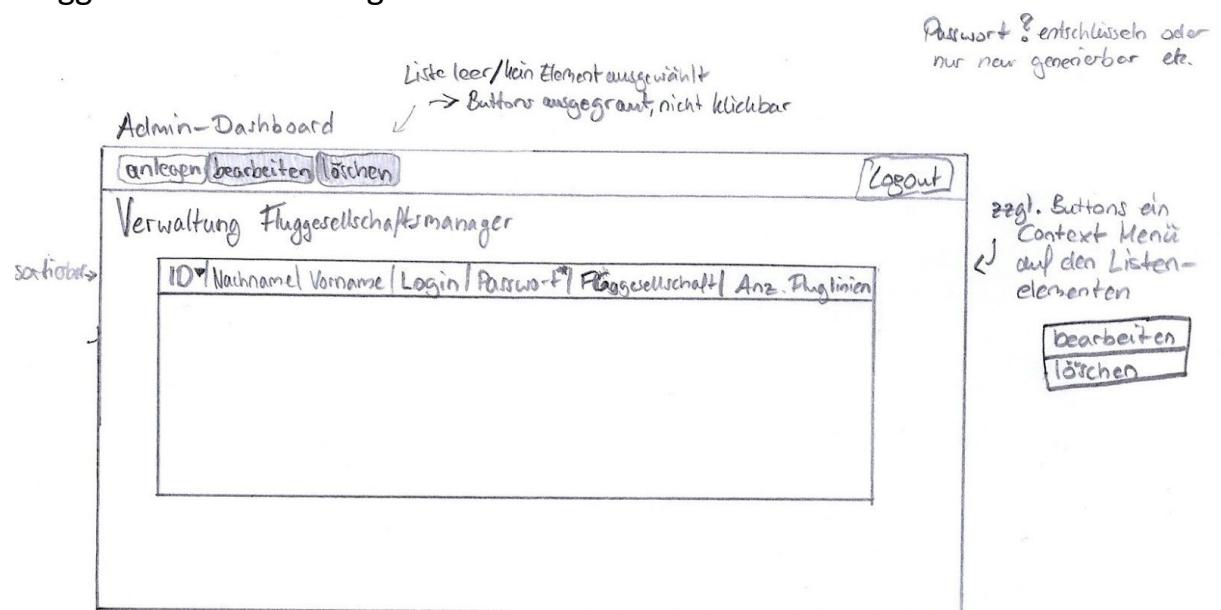
01 - Login-Screen ist der erste Screen bei Programmstart:



02 - Login-Screen mit Fehlermeldung bei falscher Nutzernamne und Passwort Kombination:



03 - Admin-Dashboard zum Anlegen und Verwalten der Fluggesellschafts-Manager:



04 - Admin - Fluggesellschafts-Manager anlegen/bearbeiten:

The form has the following fields and buttons:

- Section Header:** "neuer Fluggesellschafts-Manager" with a close button "X" on the right.
- Text Fields:** "Vorname:" with an input field, "Nachname:" with an input field.
- Generated Fields:** "Login : vn.nn" and "Passwort : pwaaa!".
- Buttons:** "abbrechen", "anlegen", "löschen", and "speichern" at the bottom.

05 - FG-Manager - Fluggesellschafts-Tab ohne FG:

<input type="button" value="anlegen"/>	<input type="button" value="bearbeiten"/>	<input type="button" value="löschen"/>	<input type="button" value="Logout"/>
<u>Fluglinien</u> / <u>Fluggesellschaft</u>			
Es wurde noch keine Fluggesellschaft angelegt.			
<u>Fluggesellschaft anlegen</u>			

06 - FG-Manager - Fluggesellschaft anlegen/bearbeiten:

Fluggesellschaft anlegen,
bearbeiten analog:

neue Fluggesellschaft		X
Name : <input type="text"/>		
Hauptsitz : <input type="text"/>		
Budget : <input type="text"/> €		
<input type="button" value="abbrechen"/>	<input type="button" value="anlegen"/>	<input type="button" value="löschen"/>
		<input type="button" value="speichern"/>

07 - FG-Manager - FG-Tab mit FG und Flugzeug-Liste:

The wireframe shows the FG tab selected. At the top, there are buttons for 'Anlegen' (Create), 'Bearbeiten' (Edit), 'Löschen' (Delete), and 'Logout'. Below these are tabs for 'Fluglinien' and 'Fluggesellschaft'. The 'Fluggesellschaft' tab is active, displaying information: Fluggesellschaft: Air-SEP, Hauptstz: Deutschland, Budget: 10.780.000€. A section titled 'Flugzeuge' lists a single entry: A370, Reichweite: 5.000 km, Hersteller: Alpha, Geschwind.: 2.000, Verbrauch: viel, Sitzplätze: 300, in Sitz: 1, Preisflz: 42. To the left of the table, handwritten notes say: 'Verkauf nicht möglich kleinste Zahl = Besitz' with an arrow pointing to the table, and '↳ im Pop-up mit Gesamtsumme oder doch im selben? z.B. hier' with an arrow pointing to the table. To the right, handwritten notes say: 'gsl. Filterbutton "zeige Flugzeuge in Besitz"' with an arrow pointing to the table, 'steige alle Flugzeuge, Tabelle sortierbar' with an arrow pointing to the table, and 'Kaufen' and 'verkaufen' buttons at the bottom.

08 - FG-Manager - Flugzeug kaufen (Alternative zu Kauffunktion integriert im FG-Tab):

The wireframe shows a modal dialog box for buying an aircraft. It has a header with a close button. Inside, there are columns for 'Modell', 'Reichweite', 'Sitzpläne', 'Preis', and 'in Betrieb?'. Below the table, it says 'In Bestell: 0'. At the bottom are buttons for 'Kauf' (Buy) with a quantity input field set to 1, and 'Warten' (Wait). Handwritten notes on the left say 'sortierbar (von Haus)' with an arrow pointing to the table.

((09 - FG-Manager - FG löschen Warnung:))

The wireframe shows a confirmation dialog box. It has a header with a close button. The title is 'Löschen der Fluggesellschaft'. The text inside says: 'Mit dem Löschen der Fluggesellschaft löschen Sie alle Fluglinien und geplante Flüge. Sind Sie sicher?'. At the bottom are buttons for 'Abbrechen' (Cancel) and 'Löschen' (Delete). Handwritten notes on the left say 'Button: löschen' with an arrow pointing to the 'Löschen' button, and 'Bedingung: Tab "Fluggesellschaft" offen' with an arrow pointing to the 'Fluggesellschaft' tab in the header.

10 - FG-Manager - Fluglinien-Tab:

<input type="button" value="anlegen"/>	<input type="button" value="bearbeiten"/>	<input type="button" value="löschen"/>	<input type="button" value="Logout"/>			
<input type="button" value="Fluglinien"/> <input type="button" value="Fluggerellschaft"/>						
ID	Start	Ziel	Entfernung	Flugzeug	Intervall	Sitzplätze (Business/Economy)
1	DUS	MOW	3200 km	A370	1x tägl.	120 (10/100)
				<input type="button" value="Flüge instanzieren"/>		

11 - FG-Manager - Fluglinie anlegen/bearbeiten:

+ Kontingenztabelle
Gruppiert nach Land
tippen und filtern?

vorgefiltert →
+ Reichweite mit
in DropDownList
anzagen

Schieberegler
blockiert oder
Text unrot,
wenn > 25%.

Fluglinie anlegen

Start Entfernung: 3200 km

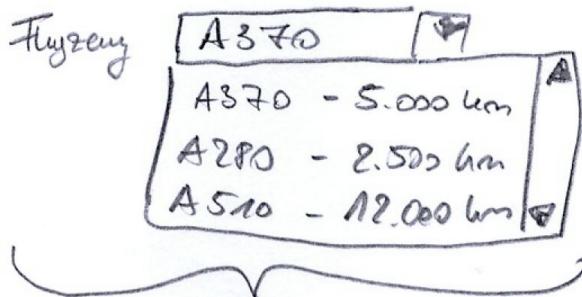
Ziel

Intervall x

Flugzeug 120 Sitzplätze
5.000 km Reichweite

Sitzplatzverteilung Business Economy
10 100

12 - FG-Manager - Fluglinie anlegen/bearbeiten Flughafen-Auswahl - Alternative 1 als gruppierte Drop-Down:



Alternativ kleineres Popup?
Vll bei Flughafenwahl
schließt sofort bei einer Auswahl
:)

13 - FG-Manager - Fluglinie anlegen/bearbeiten Flughafen-Auswahl - Alternative 2 als gruppierte Achordion-Pane/Liste, Alternative 3 als gruppierte Button-Auswahlübersicht (statt Buttons klickbare Labels?):

Flughafen Start/Ziel → schließt bei Auswahl

Wähle einen Flughafen

Europa

- Deutschland
 - DUS (Düsseldorf International)
 - NDR (Düsseldorf Weeze)

Deutschland

Belgien

Asien

- China
 - GZO (Guangzhou)

vs.

Europa

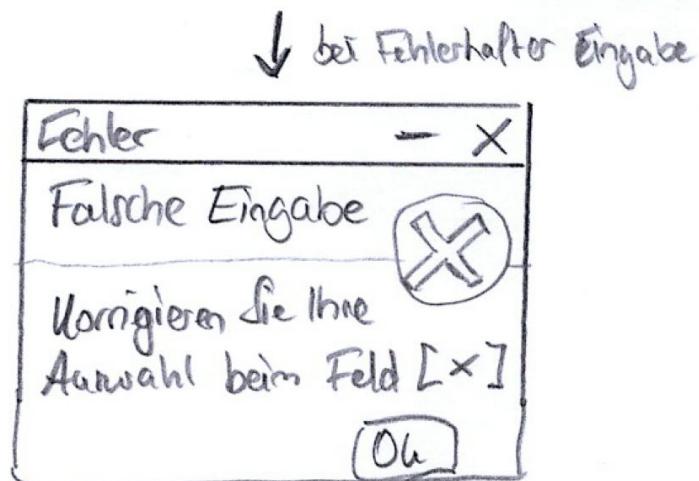
Deutschland

DUS NDR BER Weeze

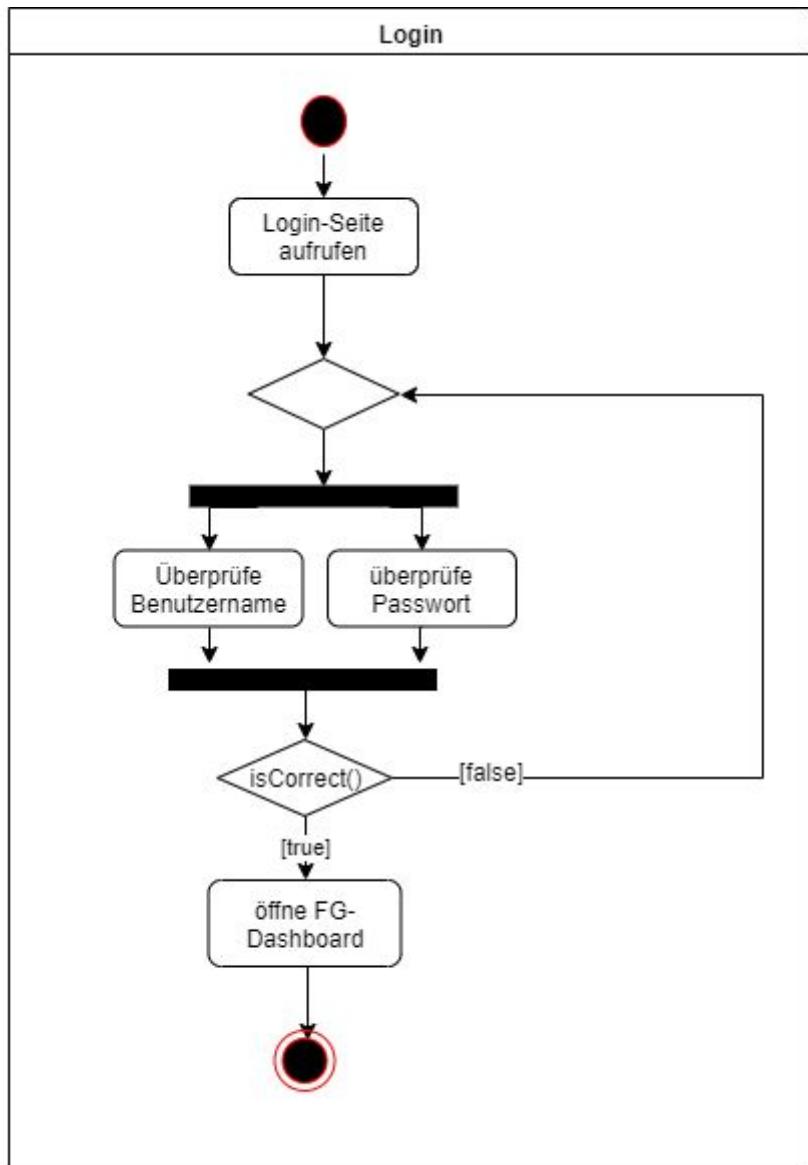
Kontinente ↗ entweder weitere jron oder ohne?

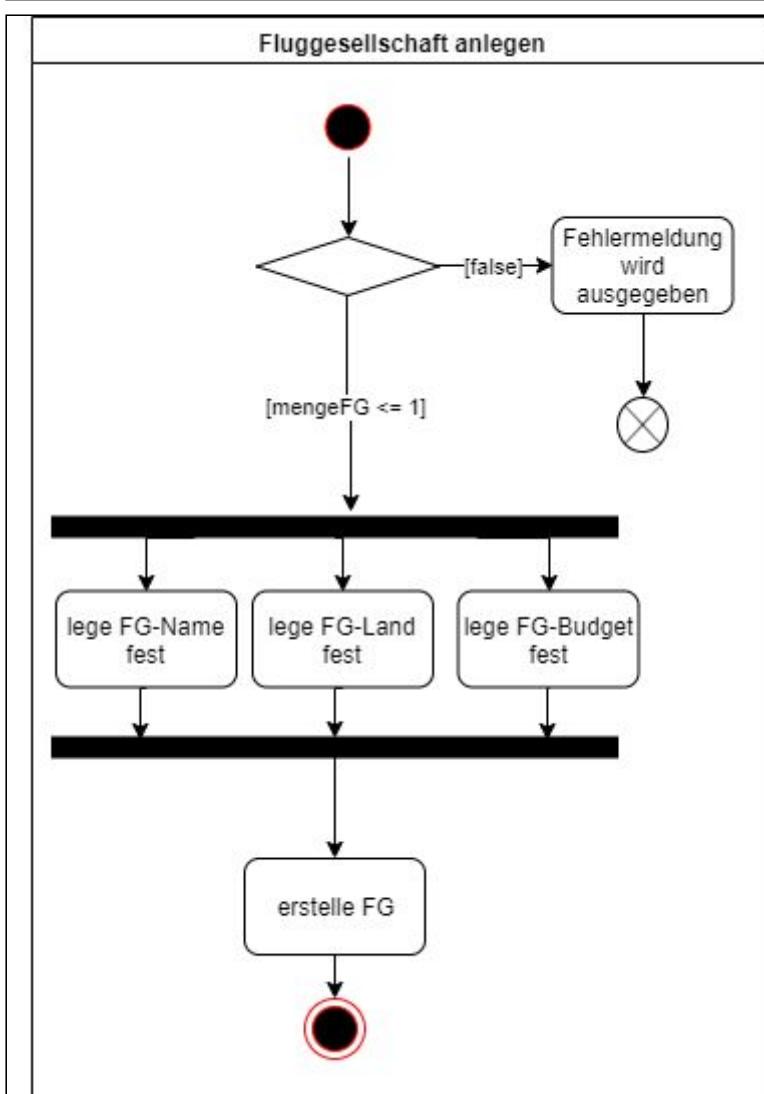
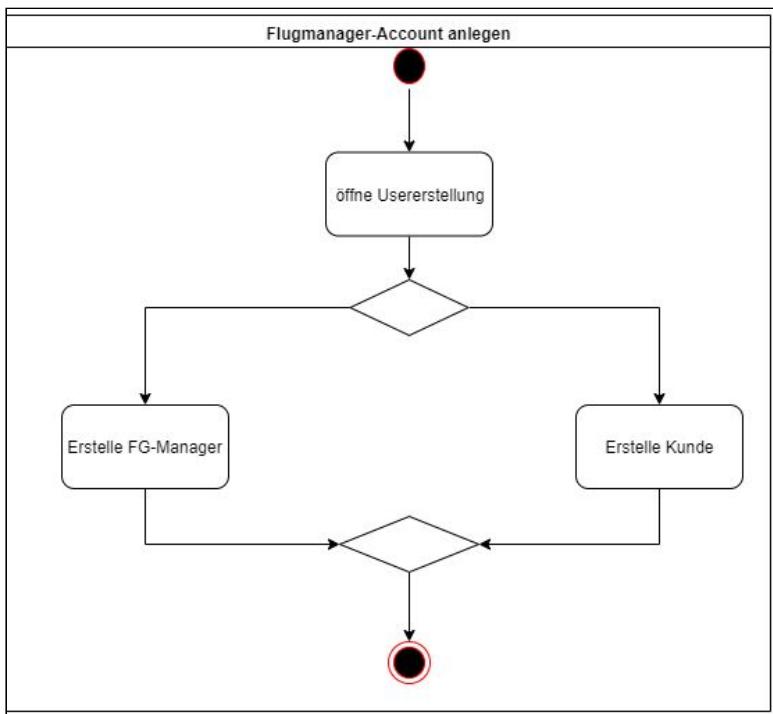
klickbares Raster...

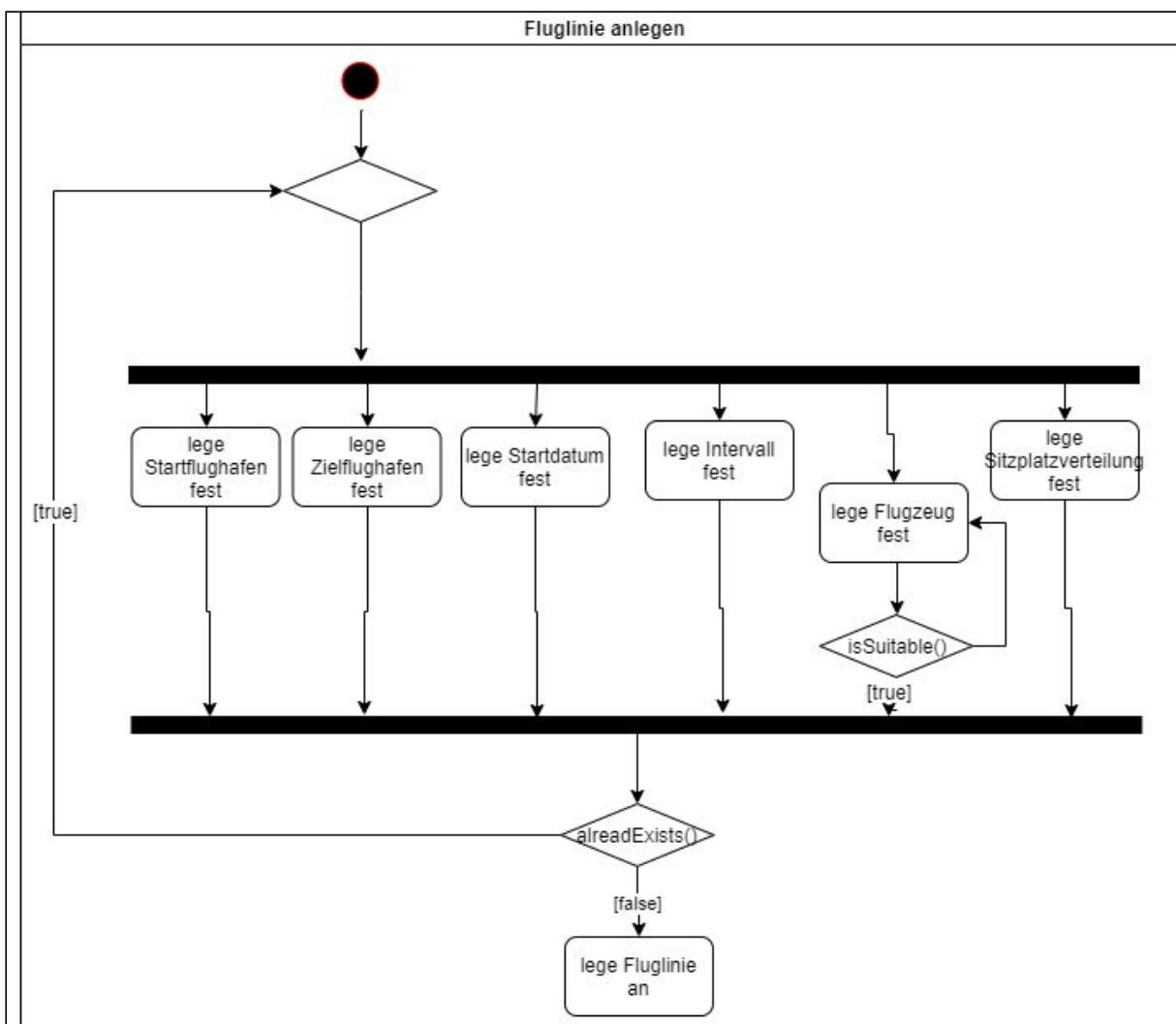
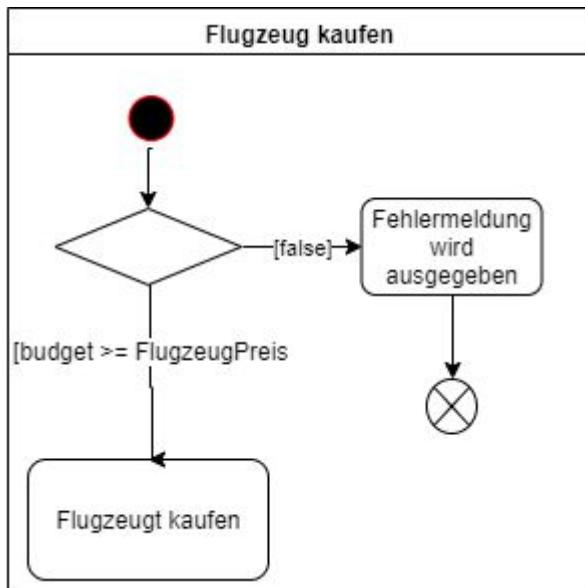
14 - Fehlermeldung bei fehlerhafter/fehlender Eingabe unter Angabe des betroffenen Feldes:

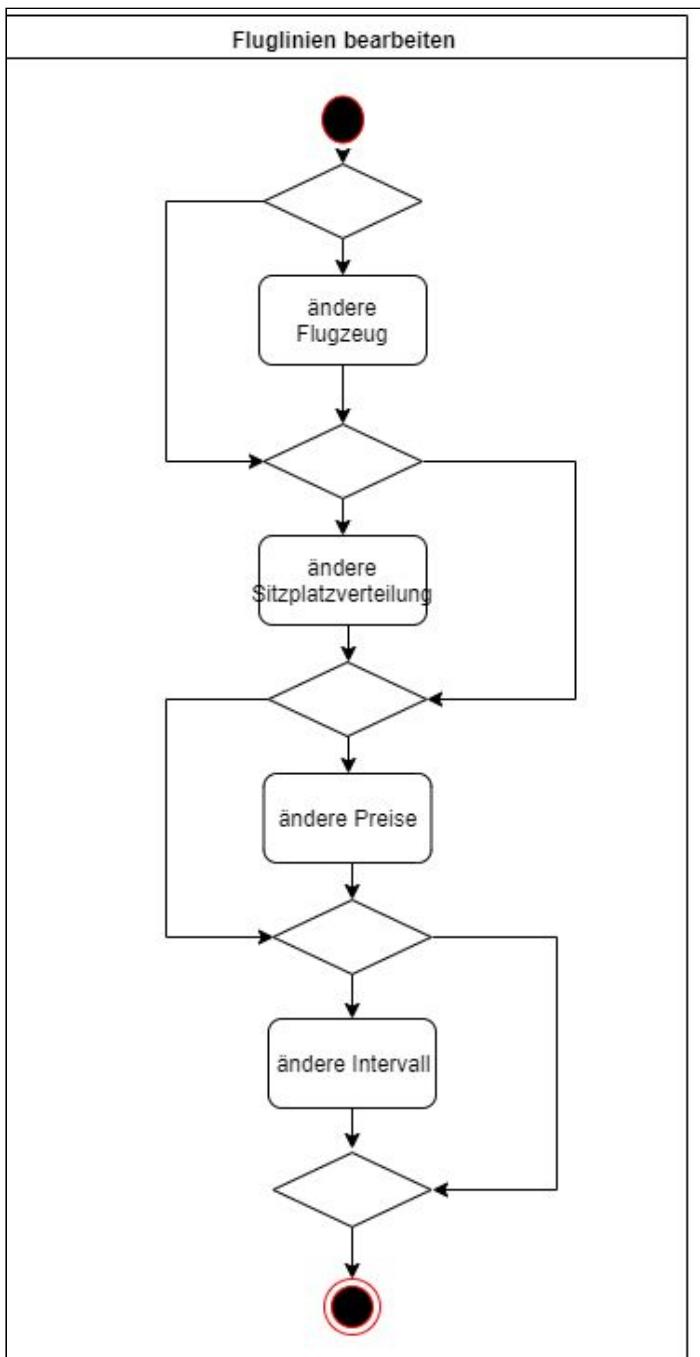


Aktivitätsdiagramm

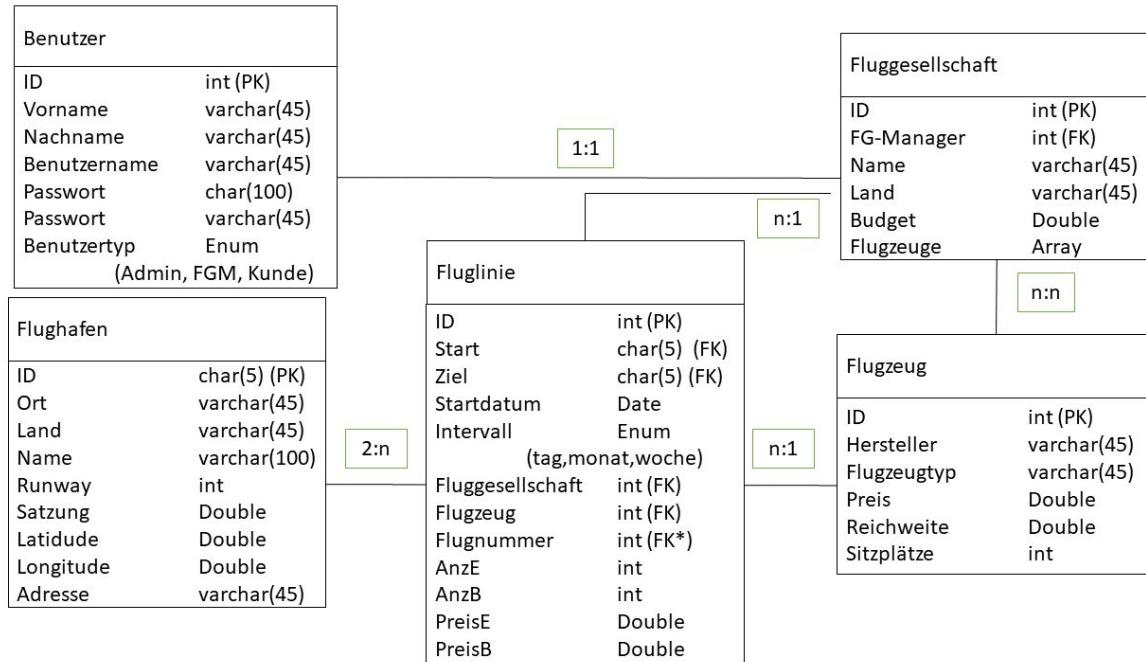




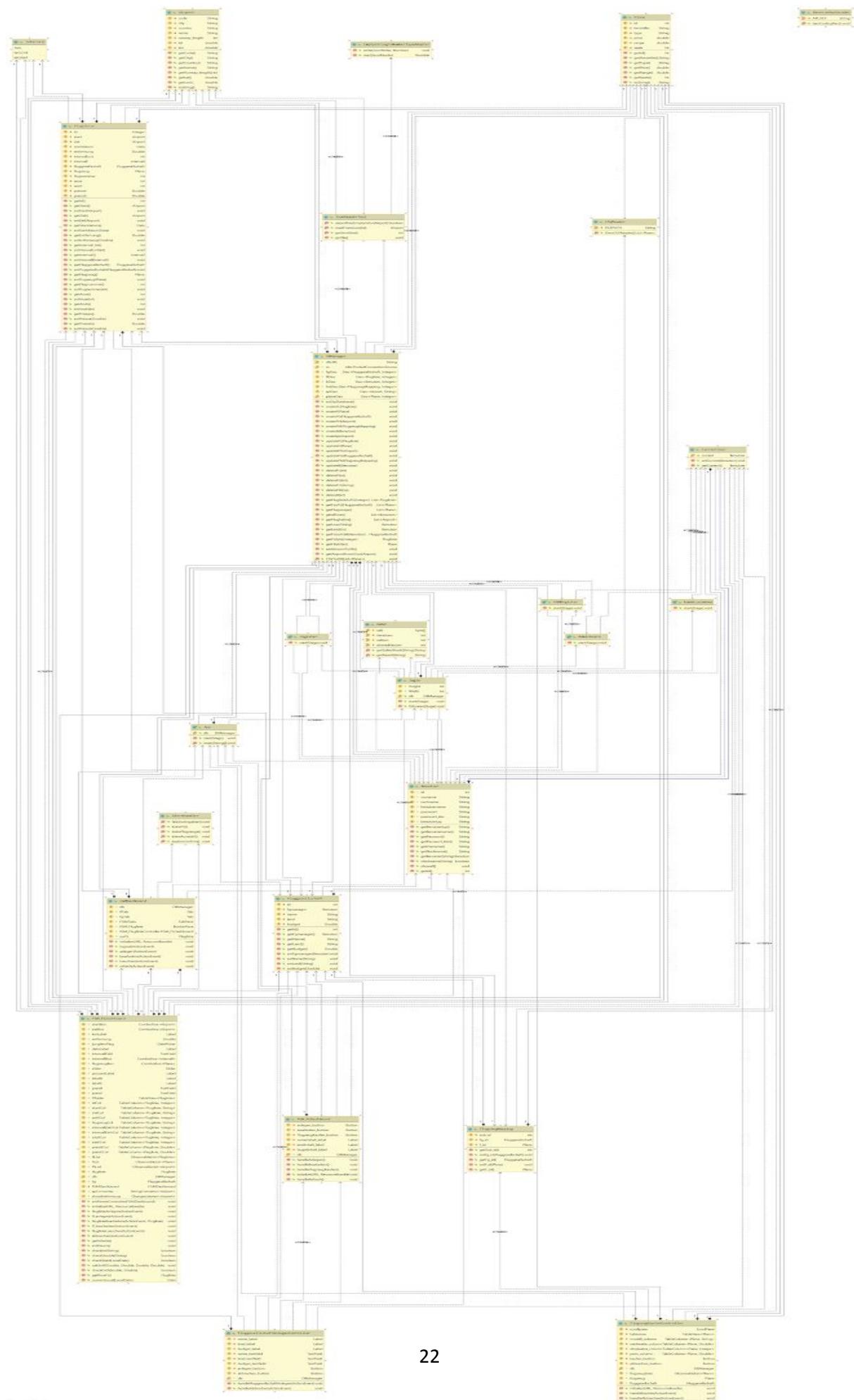




Datenbankschema



Strukturdiagramm (Klassendiagramm oder Objektdiagramm)



Funktionalitätsplanung

Id	Funktionalität	Geschätzter Aufwand	Verantwortlicher	Abhängige Funktionalitäten	Quellcode-referenz	Status
1.	Datengrundlage					
1.1	Einrichten und Kommunikation Datenbank	2 Tage	Melanie Constantin	1,2,3,4	DBManager.java (außer Methoden: addAirportToDb(),get AirportFromJson(),CSV ToDB())	fertig
1.2.1	Modellklassen	1 Tag	Bowen	2,3,4	Benutzer.java,	fertig
1.2.2	Modellklassen	1 Tag	Dennis	2,3,4	Fluggesellschaft.java	fertig
1.2.3	Modellklassen	1 Tag	Kevin	2,3,4	Flugzeug.java, Flughafen.java	fertig
1.2.4	Modellklassen	1 Tag	Melanie		Fluglinie.java	fertig
1.3	Import csv Flugzeuge	1 Tag	Kevin	1.1, 1.2, 3	CSVReader.java, Methode in DBManager.java: CSVToDB()	fertig
1.4	Import json Flughäfen	2 Tage	Kevin	1.1, 1.2, 3	JsonReaderTool.java, EmptyStringToNumberTypeAdapter.java, Methoden in DBManager.java: addAirportToDb(),get AirportFromJson()	fertig
2.	Nutzerverwaltung					
2.1	Hashen und Salten der Passwörter	2 Tage	Kevin	2.3,2.4	HASH.java	fertig
2.2	Login-GUI und -Controller	2 Tage	Bowen	1.2.1, 2.1	login.java,	fertig
2.3	Admin FGM Verwaltung GUI und Controller	2 Tage	Bowen	1.2.1	Adminboard.java FGMRegister.java	fertig
2.4	Register GUI für neuen Admin oder Kunde	2 Tage	Bowen	1.2.1, 2.1	register.java	fertig
2.5	Gleiche Benutzername vermeiden	1 Tag	Bowen	2.3,2.4	Benutzer.java	fertig
2.6	Currentuser übergeben und speichern	1 Tag	Bowen	3, 1.2	Currentuser.java	fertig

3.	Fluggesellschaftsmanager					
3.1	Dashboardgrundlege (Tab-Pane)	1 Tag	Melanie		FGMDashboard.java, FGMDashboardMain.fxml	fertig
3.1.1	Dashboard Fluglinie					fertig
3.1.1.1	Fluglinie anzeigen	1 Tag	Melanie	3.1, 1.2.2, 1.2.3	FGM_FLDashboard.java, FGM_Fluglinie.fxml	fertig
3.1.1.2	Fluglinie anlegen	1 Tag	Melanie	3.1, 1.2.2, 1.2.3	FGM_FLDashboard.java, FGMDashboard.java, FGM_Fluglinie.fxml, FGM_FLneu.fxml, FGMDashboardMain.fxml	fertig
3.1.1.3	Fluglinie bearbeiten	1 Tag	Melanie	3.1, 1.2.2, 1.2.4	FGM_FLDashboard.java, FGMDashboard.java, FGML_Fluglinie.fxml, FGM_FLedit.fxml, FGMDashboardMain.fxml	fertig
3.1.1.4	Fluglinie löschen	1 Tag	Melanie	3.1.1.1,	FGM_FLDashboard.java, FGMDashboard.java, FGML_Fluglinie.fxml, FGMDashboardMain.fxml	fertig
3.1.2	Dashboard Fluggesellschaft					fertig
3.1.2.1	Fluggesellschaft anzeigen	1 Tag	Dennis	1.2.2	FGM_FGDashboard.java, fluggesellschaftsmanager.fxml	fertig
3.1.2.2	Fluggesellschaft anlegen	1 Tag	Dennis	1.2.2	FluggesellschaftAnlegenController.java, fluggesellschaft_anlegen.fxml	fertig
3.1.2.3	Flugzeug kaufen	1 Tag	Dennis	1.2.3	FlugzeugKaufController.java, flugzeug_kaufen.fxml	fertig
4.	Zusatzfunktionen					
4.1	AlertHandler	1 Tag	Melanie		AlertHandler.java	fertig
4.2	Adresse aus Koordinaten	1 Tag	Kevin		ReverseGeoEncoder.java	in Bearbeitung

Iteration II

User-Stories

User Story-ID	User Story 1
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich Flüge auf Basis der Fluglinien, die maximal 6 Monate nach dem Startdatum angesetzt sind, instanziieren können, damit die Kunden mit Hilfe des Systems Tickets für die einzelnen Flüge kaufen können.
Geschätzter Realisierungsaufwand	1PT
Priorität	Hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	

User Story-ID	User Story 2
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich eine Übersicht über alle Flüge meiner Fluggesellschaft haben, damit ich meine Fluggesellschaft dementsprechend verwalten kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Gering
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 1

User Story-ID	User Story 3
User Story-Beschreibung	Als Kunde möchte ich mich registrieren können, damit ich das System nutzen kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	

User Story-ID	User Story 4
User Story-Beschreibung	Als Admin möchte ich, dass ein Kunde einen Benutzernamen, ein Passwort, einen Vor- und Nachnamen, eine Adresse und eine Telefonnummer besitzt, damit das Konto eines Kunden mit diesen Informationen verwaltet werden kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 3

User Story-ID	User Story 5
User Story-Beschreibung	Als Kunde möchte ich, dass ich ein Postfach besitze, damit ich wichtige Mitteilungen nachlesen kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Mittel
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 3, 4

User Story-ID	User Story 6
User Story-Beschreibung	Als Kunde möchte ich Flüge buchen können, damit ich meine Reisen mit Hilfe des Systems planen kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 1, 3, 4

User Story-ID	User Story 7
User Story-Beschreibung	Als Kunde möchte ich Flüge unter der Angabe von eines Start- und Zielflughafens sowie eines Datums und der Anzahl an gewünschten Plätzen suchen können, damit ich mir nur passende Flüge anzeigen lassen kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Mittel
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 1, 3, 4

User Story-ID	User Story 8
User Story-Beschreibung	Als Kunde möchte ich, dass ich bei meiner Suche zwischen Direkt- und Multistopp-Flügen, welche maximal drei Tage vor bzw. nach dem Datum, das ich angegeben habe, stattfinden, entscheiden kann, damit ich auch dies bei meiner Suche berücksichtigen kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Mittel
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 1, 3, 4

User Story-ID	User Story 9
User Story-Beschreibung	Als Kunde möchte ich Sitzplätze für einen Flug auswählen und reservieren können, damit ich die Möglichkeit habe, gezielt Plätze eines Fluges zu kaufen.
Geschätzter Realisierungsaufwand	1PT
Priorität	Mittel
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 1, 3, 4

User Story-ID	User Story 10
User Story-Beschreibung	Als Kunde möchte ich ein Flugkilometerkonto und ein CO2-Konto haben, damit ich meinen persönlichen Konsum überblicken kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Gering
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 1, 3, 4, 6

User Story-ID	User Story 11
User Story-Beschreibung	Als Kunde möchte ich eine Übersicht über alle meine Buchungen haben, damit ich auf einem Blick sämtliche Buchungen von mir überblicken kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Gering
Autor(en)	Dennis Eichner

Abhängigkeiten zu anderen User Stories	User Story 1, 3, 4, 6
---	-----------------------

User Story-ID	User Story 12
User Story-Beschreibung	Als Kunde möchte ich die Übersicht über alle meine Buchungen als PDF. exportieren können, damit ich diese Übersicht als Datei auf meinem PC abspeichern und weiterverwenden kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Gering
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	User Story 1, 3, 4, 6

User Story-ID	User Story 13
User Story-Beschreibung	Als Fluggesellschafts-Manager möchte ich verschlüsselte Nachrichten an den Kunden versenden können, um Datenschutz zu gewährleisten.
Geschätzter Realisierungsaufwand	1PT
Priorität	Hoch
Autor(en)	Dennis Eichner
Abhängigkeiten zu anderen User Stories	

User Story-ID	User Story 14
User Story-Beschreibung	Als Kunde möchte ich eine Option für die Buchung von Hin- und Rückflug nutzen können, da ich in der Regel von Zuhause aus fliege und wieder zurück und so schneller buchen kann.
Geschätzter Realisierungsaufwand	1PT
Priorität	Niedrig
Autor(en)	Melanie Constantin
Abhängigkeiten zu anderen User Stories	

User Story-ID	User Story 15
User Story-Beschreibung	Als Kunde möchte ich Tickets für mehrere Personen buchen können, um Tickets für meine Familie direkt in einem Schritt buchen zu können.
Geschätzter Realisierungsaufwand	1PT
Priorität	Niedrig
Autor(en)	Melanie Constantin

Abhängigkeiten zu anderen User Stories	
---	--

User Story-ID	User Story 16
User Story-Beschreibung	Als Kunde möchte Buchungen stornieren können, falls ich meine Meinung ändere.
Geschätzter Realisierungsaufwand	1PT
Priorität	Niedrig
Autor(en)	Melanie Constantin
Abhängigkeiten zu anderen User Stories	

Papierprototypen

01 - Flug instanziiieren

The image shows a hand-drawn paper prototype of a flight booking application interface. The prototype is contained within a rectangular frame with rounded corners. It includes the following elements:

- Flight Details:**
 - Startflughafen:** DUS - Düsseldorf
 - Zielflughafen:** FRA - Frankfurt a.M.
 - Entfernung:** 100 km
 - Flugdauer:** 2:00 Std.
- Flight Type:** Flugzeug
- Aircraft:** Airbus A320 - 520 Sitzplätze
- Startzeit:** [1. Flug am 23.10.2015]
- Time:** 12 : 30 Uhr **Täglich**
- Buttons:**
 - abbrechen** (cancel) button
 - instanziieren** (instance) button
 - bis wann?** (until when?) question with a small input field.

02 - Übersicht instanziierter Flüge

Flugübersicht								
Start	Ziel	Intervall	1. Flug	Flugzeug	Economy	Business		
DUS	EDL	Täglich	27.12.20	Airbus 310	200	50€	20	200€
DUS	FRA	3-Tägig	13.01.20	Airbus 310	198	50€	19	200€
DUS	OTP	Wöchentl.	14.01.20	Airbus 330	150	60€	15	200€
FRA	OTP	Täglich	01.01.20	Airbus 380	172	50€	18	200€

(abbrechen)

03 - Übersicht Fluglinien mit Angabe bis wann Flüge instanziert

Fluglinien			Fluggesellschaft	Refresh	Logout
Start	Ziel	...			
DUS	EDL				instanziert bis
DUS	FRA				18.01.2020
DUS	OTP				23.06.2020

Flüge instanzieren!

04 - Flugbuchung Suche

Kunde Flüge suchen

1 Nachricht (Reise) | (Zug auf)

Flugkarte /Konto/Puchungen

mit Rückflug

DUS - Düsseldorf FRA - Frankfurt

Hin: 18-01-19 ±3 Rücker: 23-01-19 ±1

2 Personen

Economy Business

Zwischenziel einfügen

Suche

Suchergebnisse

10:30 18.01.2019 2:30 Nd. DUS-FRA ...
 - 13:00

Airline P Airbus A320 400 km 200 m³
 ... 200€ 400€ kaufen

05 - Flugbuchung Suche Multistopp

Kinde Multi-Stop

Flugsuche Konto/Buchungen

mit Rückflug
DUS - Düsseldorf → nach **FRA - Frankfurt**
 Hin: [18-01-17] [] ± 3

2 Personen

Economy Business

Zwischenziel einfügen
FRA - Frankfurt → nach **GUA - Guatemala Stadt**
 Hin: [24-08-19] [] ± 1

Economy Business

Zwischenziel einfügen
GUA - Guatemala Stadt → nach **DUS - Düsseldorf Int.**
 Hin: [02-11-19] [] ± 3

Economy Business

Zwischenziel einfügen

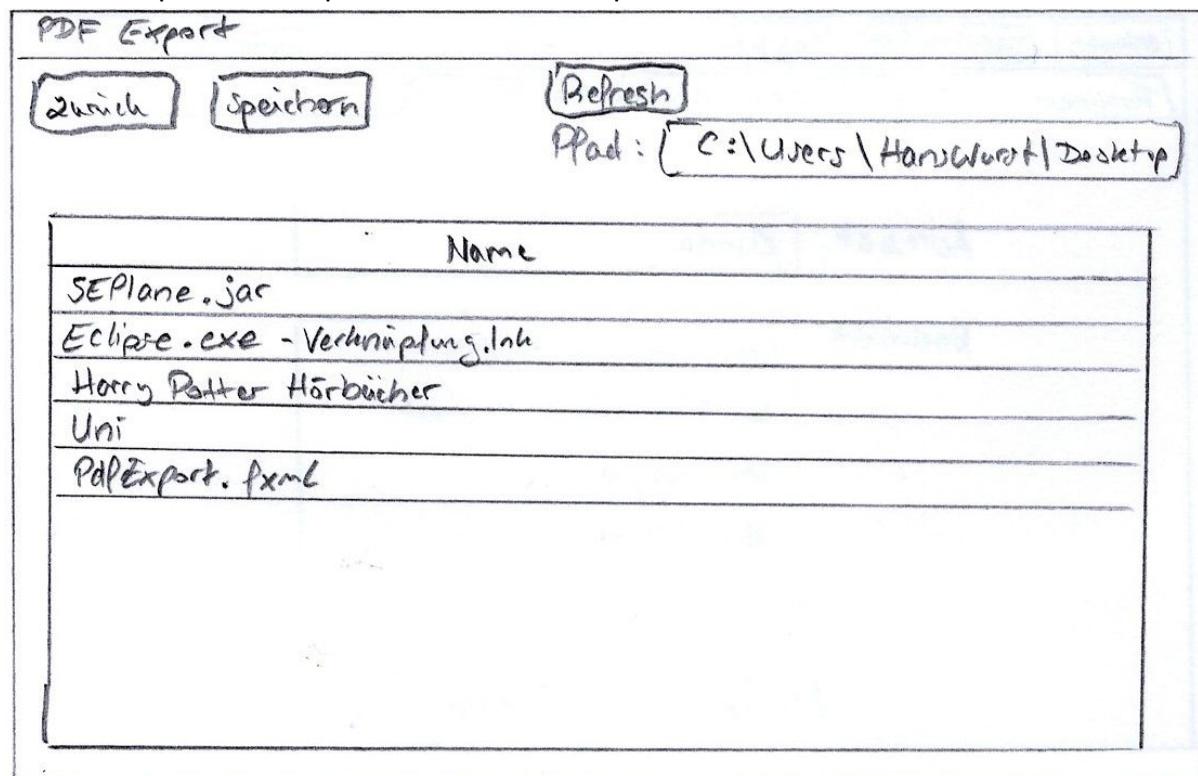
06 - Buchungswischenschritte: Sitzplatzreservierung, Bestätigungsdialog

Buchungswischenschritt	Buchungsbestätigung
<p>Sitzplatzreservierung</p> <p>Reservieren Sie Ihren Sitzplatz für Ihren Flug von DUS nach FRA</p> <p>Ihre gewählte Klasse ist Economy (8A)</p> <p><input type="button" value="Abbrechen"/> <input type="button" value="Kaufeal"/></p>	<p>Sie haben folgenden Flug gebucht zu einem Gepäckpreis von 135,00 €</p> <p>Bitte überweisen Sie den zu zahlenden Betrag an DE 8 —</p> <p><input type="button" value="Export PDF"/> <input type="button" value="Ok"/></p>

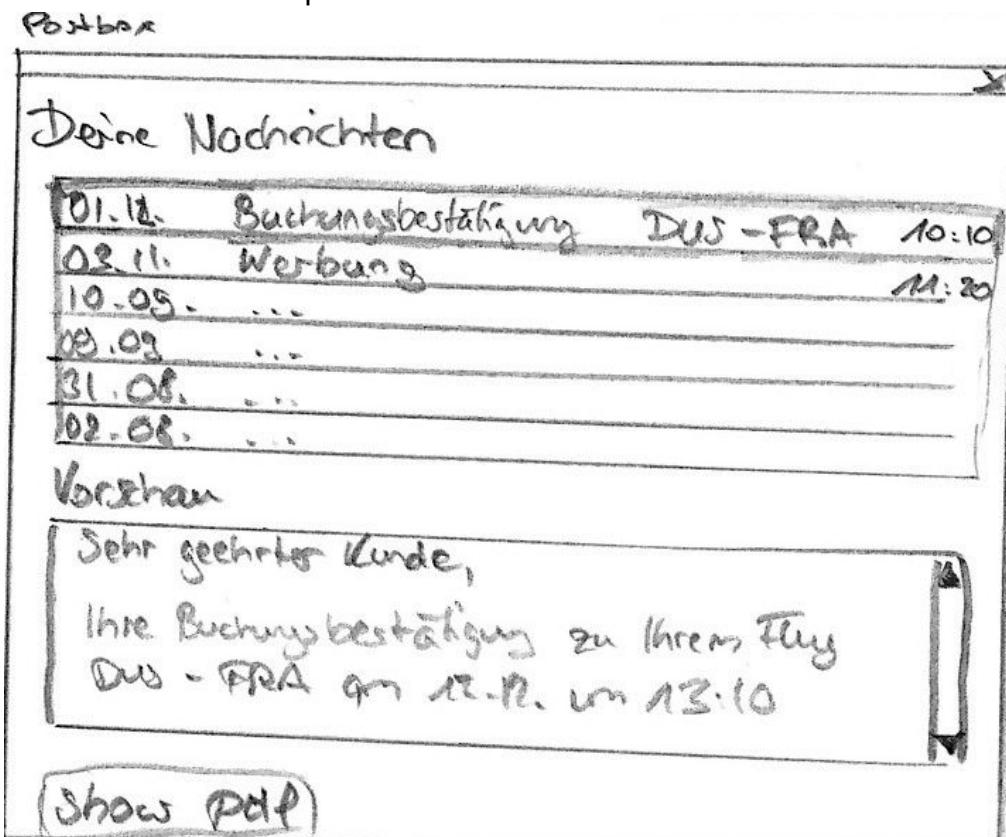
07 - Buchungsübersicht (mit geplanten Features aus IT 3) mit PDF Export

Kunde: Buchungs-/Kontoübersicht		<input type="button" value="Nachrichten"/> <input type="button" value="Postbox"/> <input type="button" value="Logout"/>																													
<p>Flugreise Konto/Buchungen</p> <p>Kundenname: <input type="text"/> CO₂-Konto: 1002 m³</p> <p>Daten als Label: Flug-km: 3582 km</p> <p><input type="button" value="Keine Buchungen"/></p> <table border="1"> <thead> <tr> <th>Dat.</th> <th>Uhrz.</th> <th>Start</th> <th>End</th> <th>Sitz</th> <th>Preis</th> </tr> </thead> <tbody> <tr> <td>15.11.</td> <td>10:30</td> <td>DUS</td> <td>FRA</td> <td>83C</td> <td>350€</td> </tr> <tr> <td>08.02.</td> <td>14:45</td> <td>AHM</td> <td>MEX</td> <td>32B</td> <td>100€</td> </tr> <tr> <td>01.03.</td> <td>23:20</td> <td>DUS</td> <td>EDI</td> <td>1A</td> <td>282€</td> </tr> <tr> <td>05.04.</td> <td>07:10</td> <td>—</td> <td>—</td> <td>28</td> <td>293€</td> </tr> </tbody> </table>	Dat.	Uhrz.	Start	End	Sitz	Preis	15.11.	10:30	DUS	FRA	83C	350€	08.02.	14:45	AHM	MEX	32B	100€	01.03.	23:20	DUS	EDI	1A	282€	05.04.	07:10	—	—	28	293€	<p><input type="button" value="pdf Export"/></p> <p>Infos zum Zielort</p> <p>— O: 33°C a 1 12:30</p>
Dat.	Uhrz.	Start	End	Sitz	Preis																										
15.11.	10:30	DUS	FRA	83C	350€																										
08.02.	14:45	AHM	MEX	32B	100€																										
01.03.	23:20	DUS	EDI	1A	282€																										
05.04.	07:10	—	—	28	293€																										

08 - PDF Export: File Explorer zur Auswahl Speicherort



09 - Postbox Kunde Option 1



10 - Postbox Kunde Option 2

Buchungsdashboard

Buchung Postfach

Sender	Date	Nachricht
Pgms	22.10.20	Sehr geehrter Kunde, Ihr Flug wurde storniert. Pech gehabt.

[Open] [Refresh]

11 - Nachrichten senden Fluggesellschaftsmanager

FCM-Dashboard

[Anlegen] [bearbeiten] Kündigung [Refresh] [Logout]

Fluglinien Kunde Fluggesellschaft Nachrichten

Absatzat: Kunde

Nachricht:

Sehr geehrter Kunde,
 Ihr Flug wurde storniert.
 Pech gehabt.
 Ihr FGH

file Ex Senden

12 - Kundenregistrierungsfenster

Kunde registrieren

Username:

Nachname:

Benutzename:

BenutzerTyp: kunde

Email:

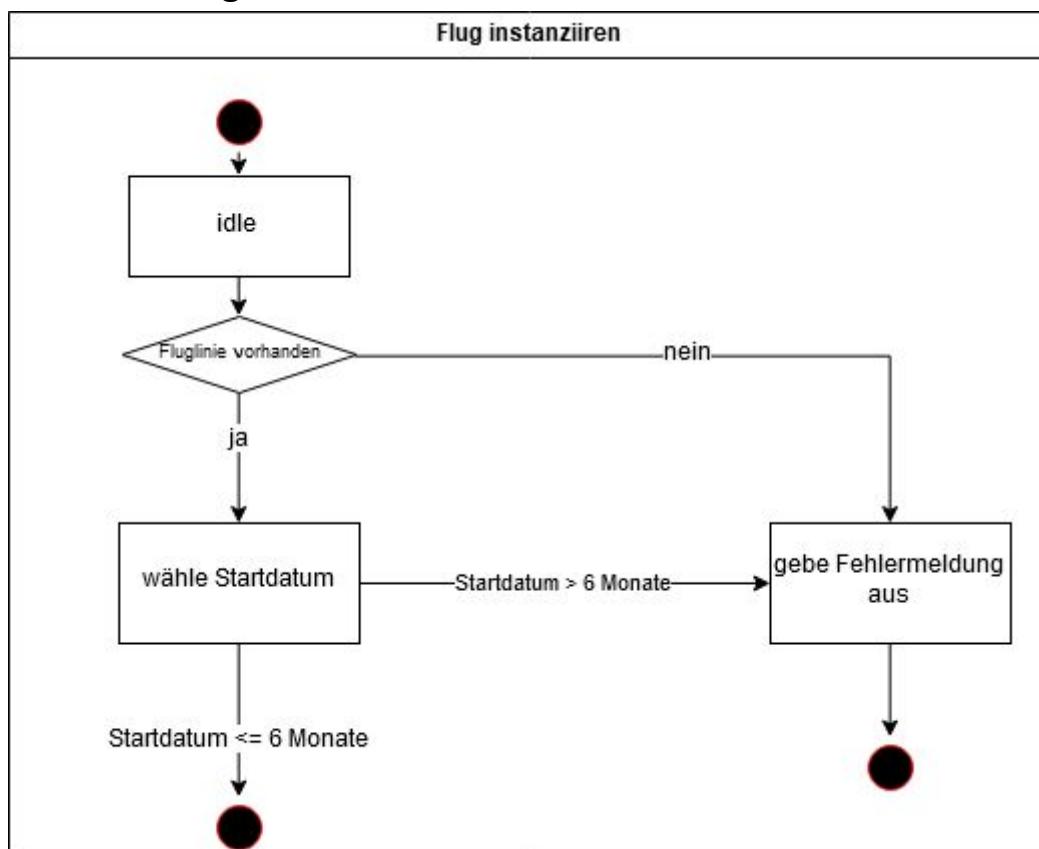
Adresse:

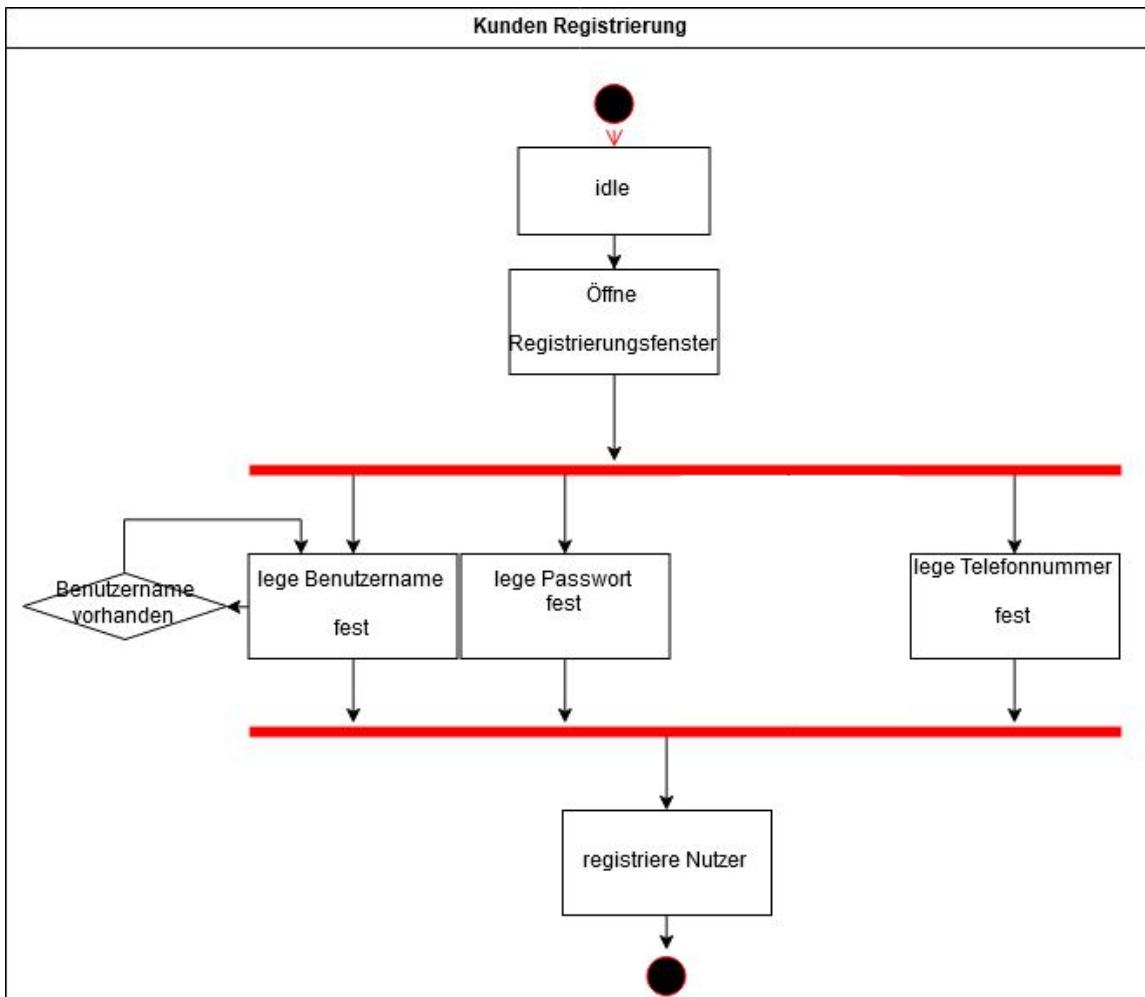
Telephone number:

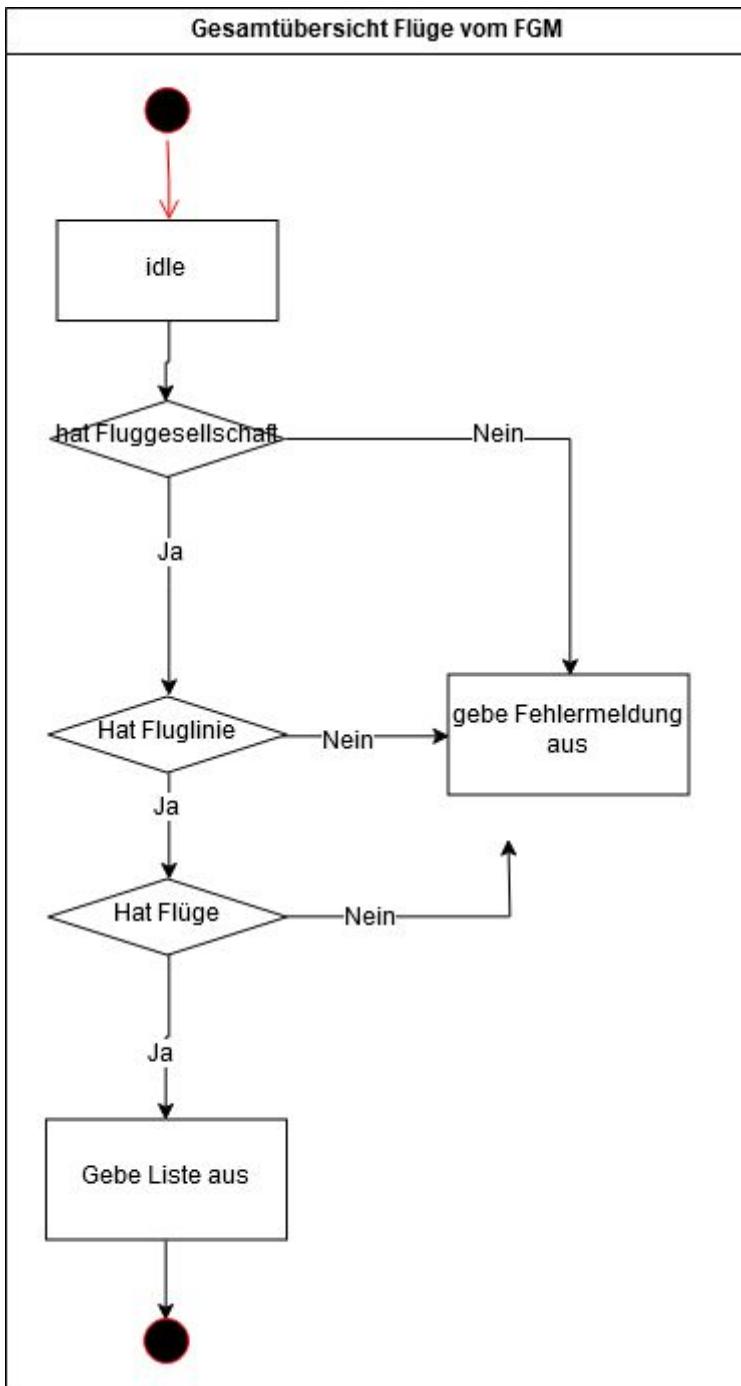
Passwort:

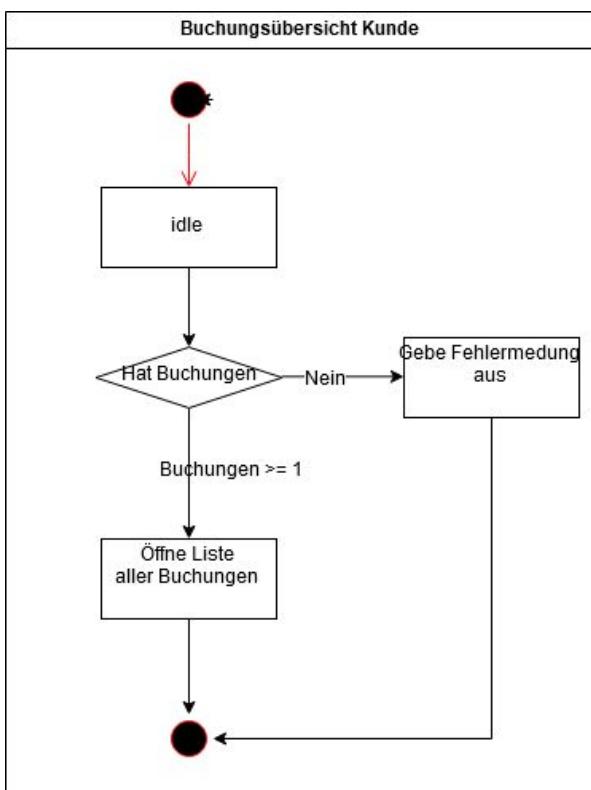
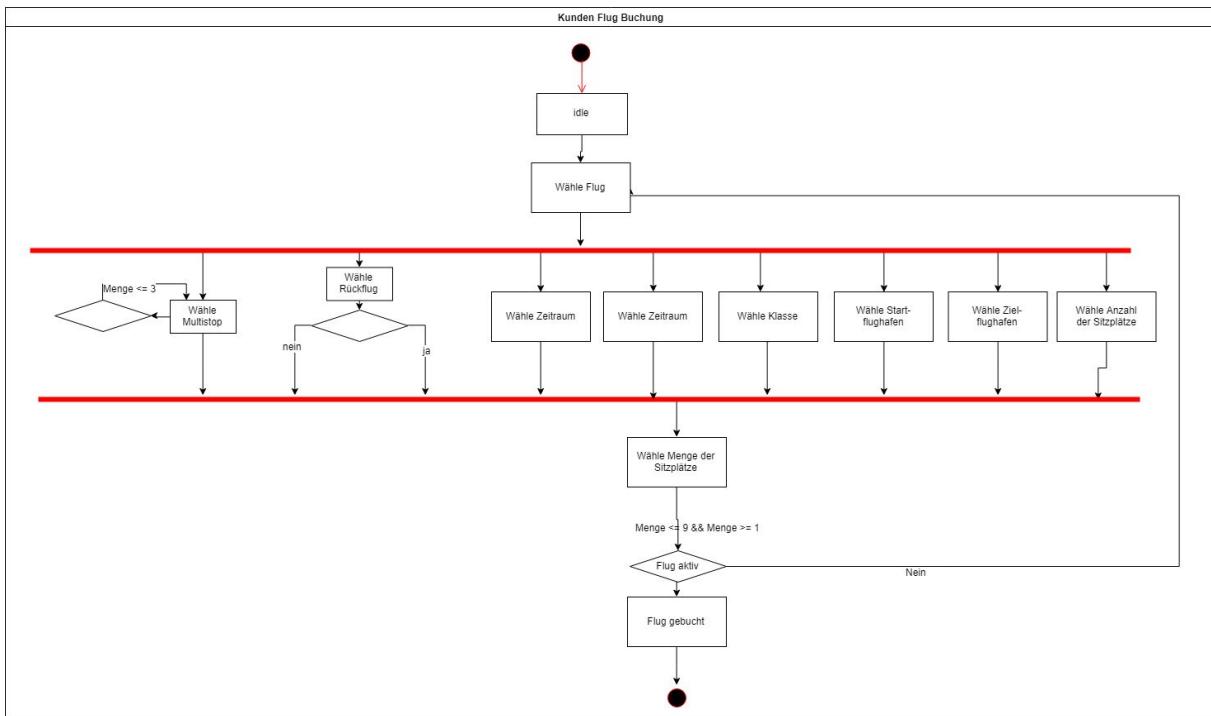
Register

Aktivitätsdiagramm

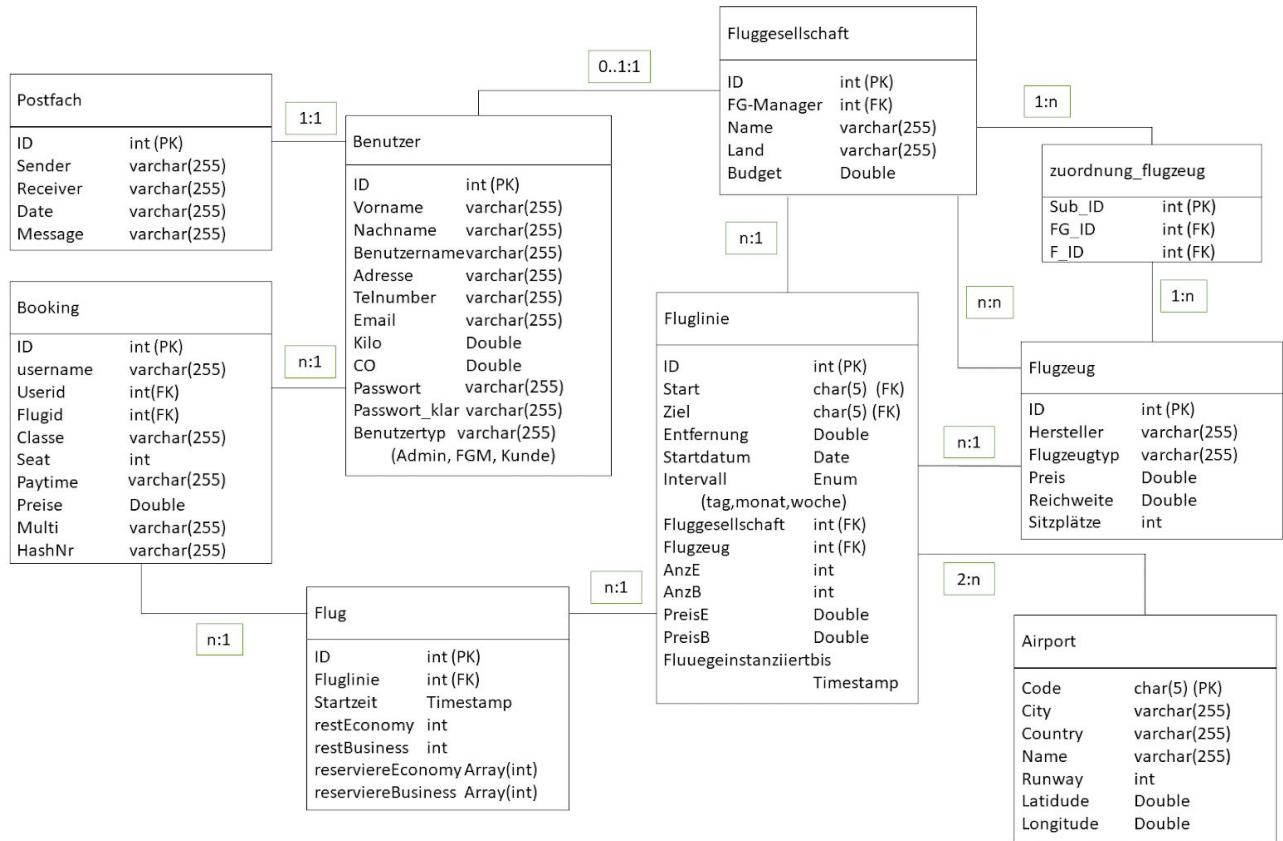




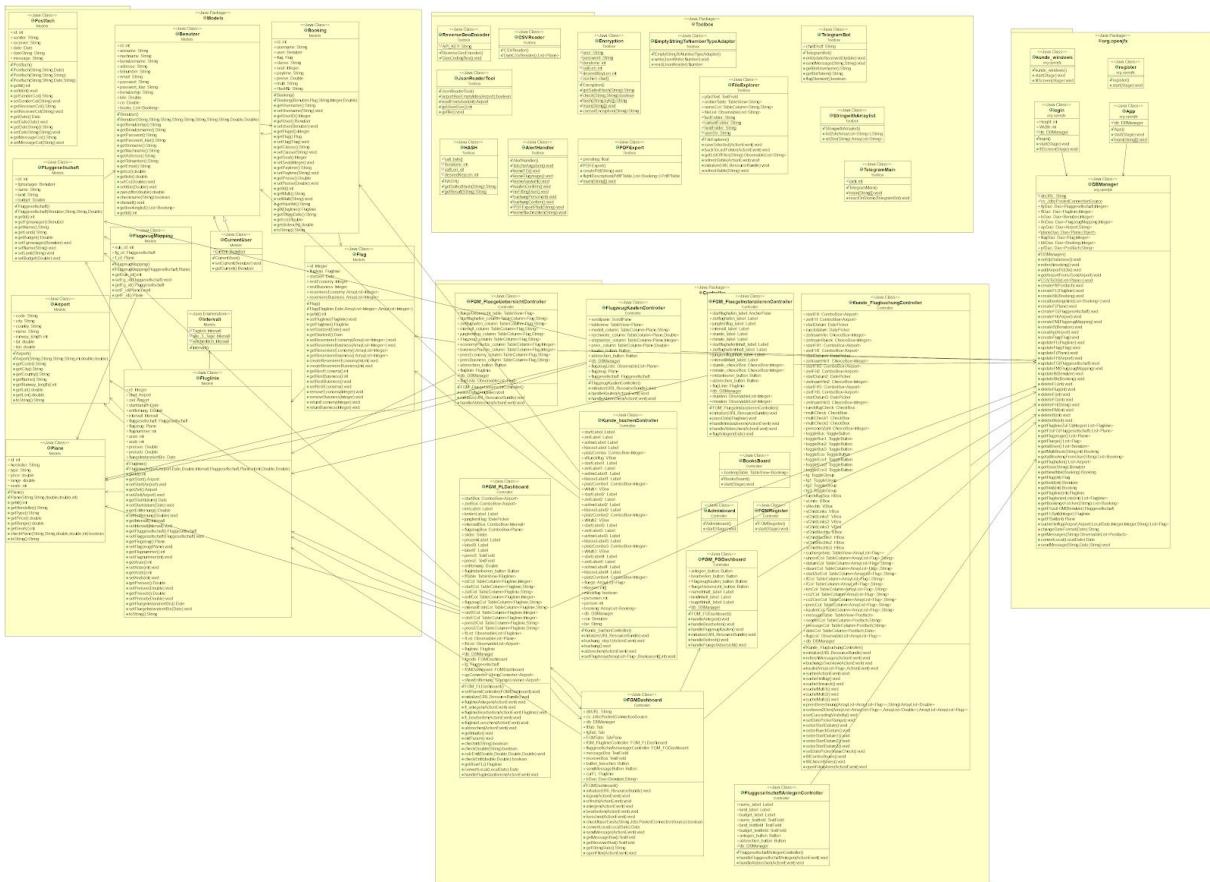




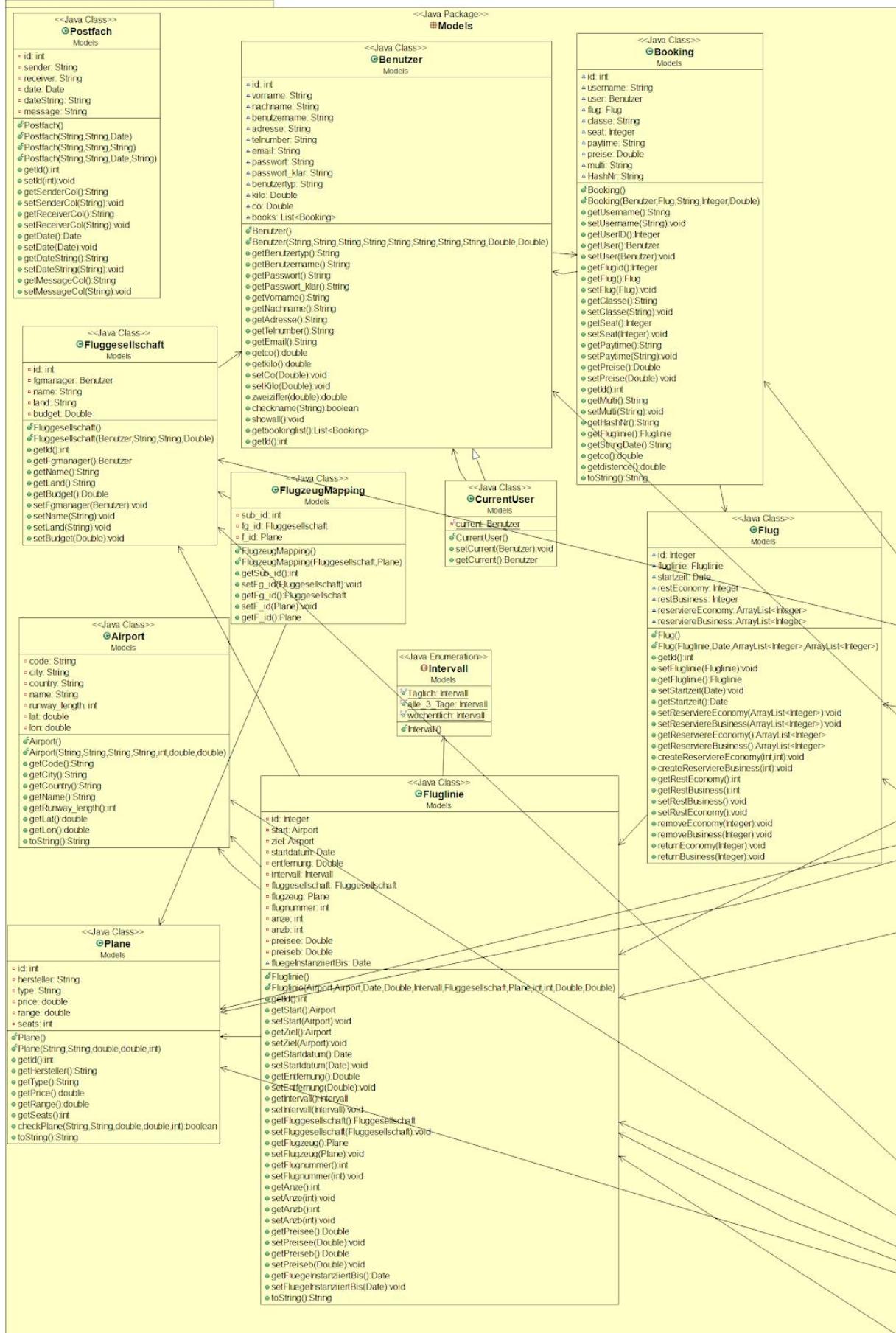
Datenbankschema:



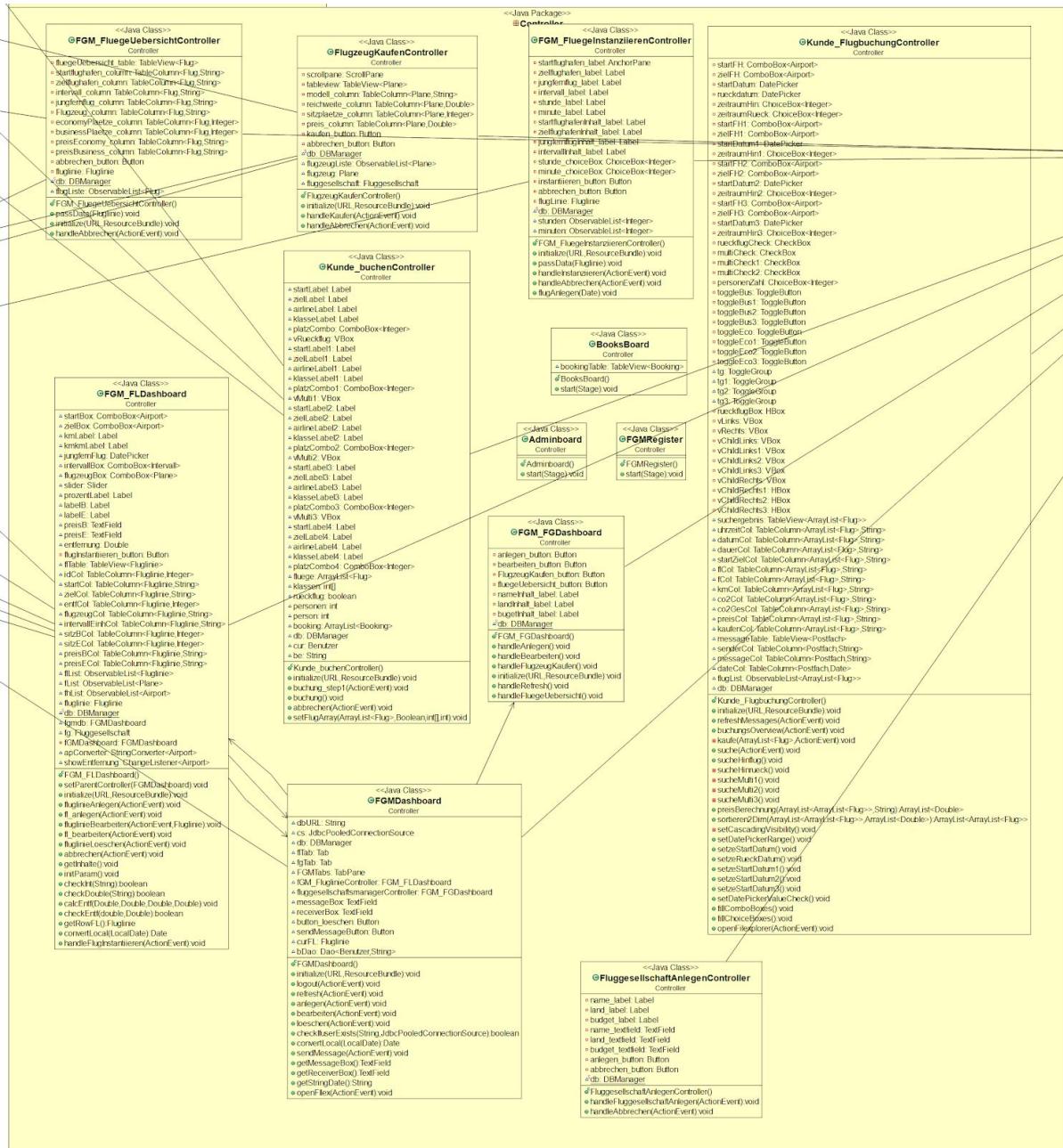
Strukturdiagramm (Klassendiagramm oder Objektdiagramm)



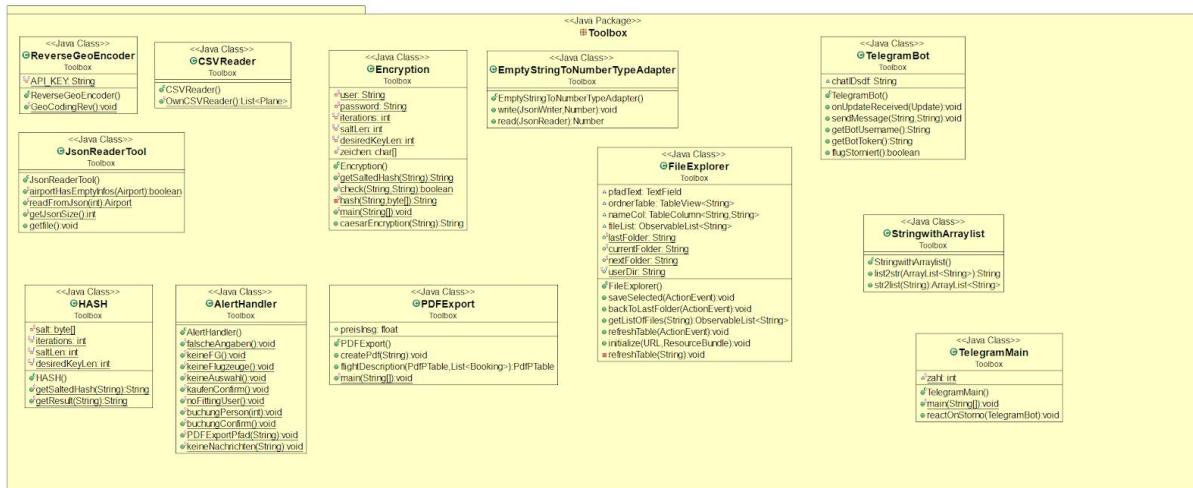
Models

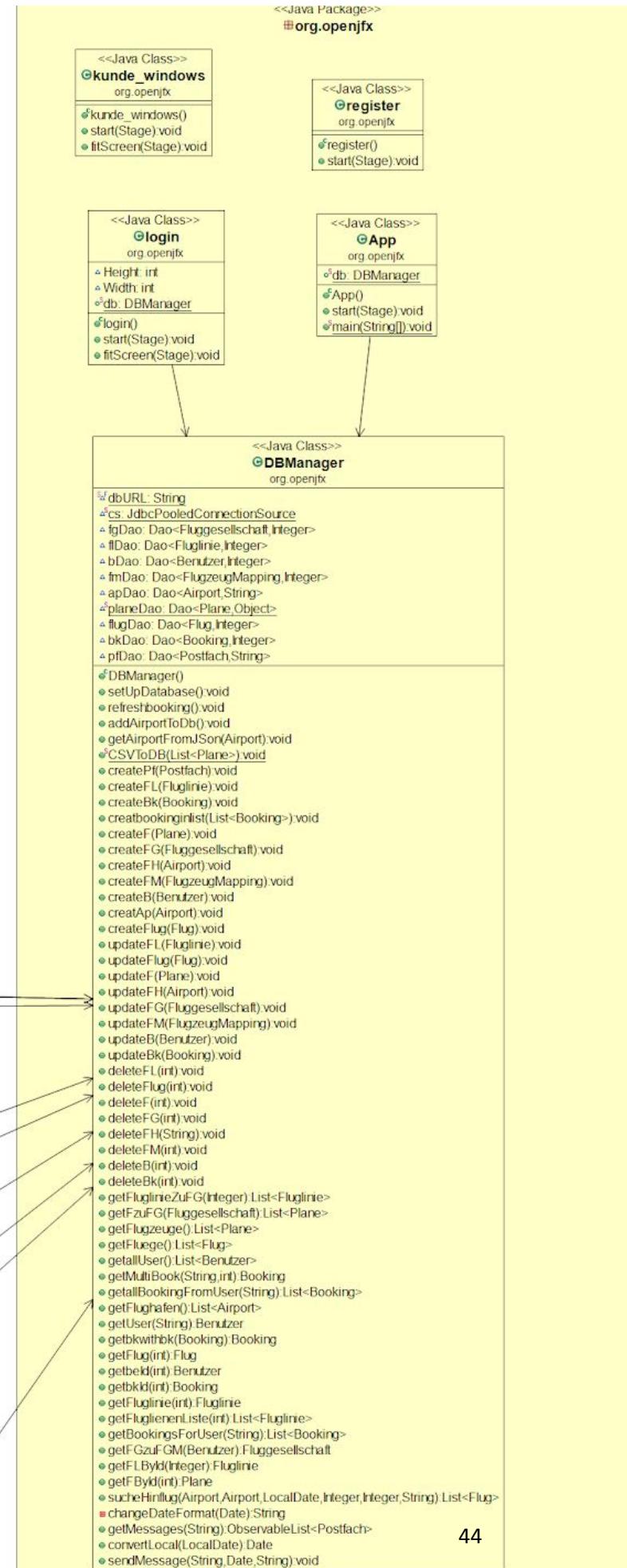


Controller



Toolbox und org.openjfx





Funktionalitätsplanung

Id	Funktionalität	Geschätzter Aufwand	Verantwortlicher	Abhängige Funktionalitäten	Quellcode-referenz	Status
1.	Fluginstanziierung					
1.1	Flüge instanziieren 6 Monate im Voraus	1 Tag	Dennis Eichner			fertig
1.2	Übersicht instanzierte Flüge	1 Tag	Dennis Eichner			fertig
2.	Nutzerverwaltung					
2.1	Kundenregistrierung	2 Tage	Bowen Chen			fertig
3.	Flugbuchung					
3.1	Buchungsübersicht Singlestop Booking Mulit-Booking	1 Tage	Bowen Chen			fertig
3.1.1	PDF-Export der Buchung mit Dateiexplorer	2 Tage	Kevin Gohres			fertig
3.2	Flugsuche Hinflug - Rückflug - Multistop, für mehrere Personen, Preisklassen Business und Economy, +/- 3 Tage Zeitraum	2 Tage	Melanie Constantin		Flug.java, Kunde_FlugbuchungController.java, Kunde_buchen.fxml, Kunde_Flugbuchung.fxml DBManager.java (sucheHinflug())	fertig
3.3	CO2 und KM Konto	1 Tag	Bowen Chen			fertig
3.4	Reservierung Sitzplätze und Buchung	1 Tag	Melanie Constantin		Kunde_buchenController.java	fertig
3.5	Speicherung der Buchungen als Single/Multi	1 Tag	Bowen Chen			fertig

3.6	Kunde Buchung stornieren	1 Tag	Bowen Chen			fertig
4.	Nachrichten					
4.1	Nachrichten via Telegram	2 Tage	Kevin Gohres			fertig
4.2	Postbox Kunde	1 Tag				fertig
4.3	Nachrichtenversand FGM	1 Tag	Kevin Gohres			fertig
4.	Modultests					
4.1	Modultest ArrayList2String	1 Tag	Bowen Chen		StringwithArrayListTest	fertig
4.2	Modultest Flüge nach Preisen sortieren	1 Tag	Melanie Constantin		Kunde_FlugbuchungControllerTest.java	fertig
4.3	Modultest	1 Tag	Kevin Gohres			fertig
4.4	Modultest	1 Tag	Dennis Eichner			fertig

Unitests

Unitest oder auch Modultests sind Komponententests. Diese werden in der Softwareentwicklung angewendet, um die funktionalen Einzelteile (Units) von Computerprogrammen zu testen.

Id	Getestete Funktionalität	Quellcode Referenz	Status
1	sortieren von zwei Flugbuchungskombinationen nach Business oder nach Economy Preis	Kunde_FlugbuchungControllerTest.java	bestanden
2	ArrayList wird als String umgeformt	ArrayList2String.java	bestanden
3			

Systemtests

Systemtests sind Tests des Gesamtsystems gegen die Anforderungen nach erfolgreicher Integration. Eingaben und Sollverhalten werden dabei aus der Anforderungs-spezifikation abgeleitet.

Datum	09.12.2019		
Tester	Nicolai Haferkamp		
SW-Version	7c15e50e		
Vorbedin-gung(en)	Nutzer muss als Manager eingeloggt sein und es sollten noch keine Flüge zu der Ausgewählten Fluglinie existieren.		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / x
1	Nutzer wählt eine Fluglinie aus und drückt auf dem Button Flug Instanziieren	Es sollte sich ein Fenster öffnen, in dem Alle wichtigen daten, die für die Instanziierung wichtig sind ausgewählt werden können. Eine Erfolgreiche Instanziierung ist nur dann möglich, wenn der Flug bis maximal 6 Monate nach Jungfernflug Datum erfolgt und ein Datum/Uhrzeit festgelegt ist.	✓
2	Nutzer wählt den Tab/drückt den Knopf „Gesamtübersicht über alle Flüge“	System zeigt alle Flüge, die in der Datenbank gespeichert sind, in Form einer Tabelle an.	✓
Nachbe-dingung(en)	Die Instanzierten Flüge werden nun in der Fluggesamtliste angezeigt		
Testurteil	Test Bestanden		

Datum	09.12.2019		
Tester	Nicolai Haferkamp		
SW-Version	7c15e50e		
Vorbedin-gung(en)	Nutzer hat keinen Account Vorher gehabt oder den alten gelöscht		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / x
1	Öffnet Programm und wählt die Option Registrieren	Es öffnet sich ein Fenster in dem der Nutzer einen Benutzernamen, Passwort, Vor- und Nachname, Adresse, Telefonnummer und E-mail eintragen kann.	✓
2	Nutzer gibt seine Daten in die passenden Textfelder im Interface ein	Das Passwort wird bei der Eingabe, anders als bei den anderen Textfeldern, mit Sternen unkenntlich gemacht	✓
3	Bestätigen der Registrierung	Alle Daten werden in Datenbank gespeichert und das Passwort wird gehasht und gesalzt	✓
Nachbe-dingung(en)	Nutzer hat nun einen Kunden Account erstellt und kann sich jetzt im System Anmelden		
Testurteil	Test Bestanden		

Datum	09.12.2019		
Tester	Nicolai Haferkamp		
SW-Version	7c15e50e		
Vorbedin-gung(en)	<p>Nutzer braucht einen Gültigen Account und muss mit diesem Angemeldet sein.</p> <p>Wenn der Nutzer keine Buchungen hat, landet er automatisch im Such Tab</p>		
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / X
1	Kunde will einen Flug suchen und nutzt die Suchfunktion	Im selben Fenster, kann der Kunde Start- und Zielflughafen, Datum/Uhrzeit eingeben um den Passenden Flug zu finden	✓
2	Bestätigt suche	System zeigt alle passenden Flüge, die in dem gesuchten Zeitraum liegen in einer Liste an.	✓
3	Kunde drückt bei einem Flug auf den Button Kaufen	Es öffnet sich ein Fenster, in dem der Kunde die Sitzplätze auswählen kann die er Belegen will.	✓
4	Kunde drückt Button Buchungsübersicht	System zeigt alle Gebuchten Flüge an und kann außerdem mit dem Button Export PDF die ausgewählte Buchung exportieren kann	✓
Nachbe-dingung(en)	Der Gebuchte Flug wird inklusive der Sitzplätze in der Datenbank gespeichert		
Testurteil	Test Bestanden		

Iteration III

User-Stories

Template:

User Story-ID	<Eindeutiger Identifizierer>
User Story-Beschreibung	<Text der User Story mittels Satzschablone: Als <Rolle> möchte ich <Ziel> [, um/sodass <Nutzen>] (s. Foliensatz „Anforderungen“)>
Geschätzter Realisierungsaufwand	<Einschätzung der Zeit, die benötigt wird, um die Userstory zu implementieren>
Priorität	<Wichtigkeit der User Story hinsichtlich der Aufgabenstellung z.B. hoch, mittel niedrig>
Autor(en)	<Hier bitte nur einen Zuständigen eintragen z.B. Max Mustermann>
Abhängigkeiten zu anderen User Stories	<Auflistung verwandter User Stories>
Zugehörige Szenarien	<Zuordnung zu Szenarien> <ul style="list-style-type: none">● Hauptscenario:● Alternativszenarien:● Ausnahmeszenarien:

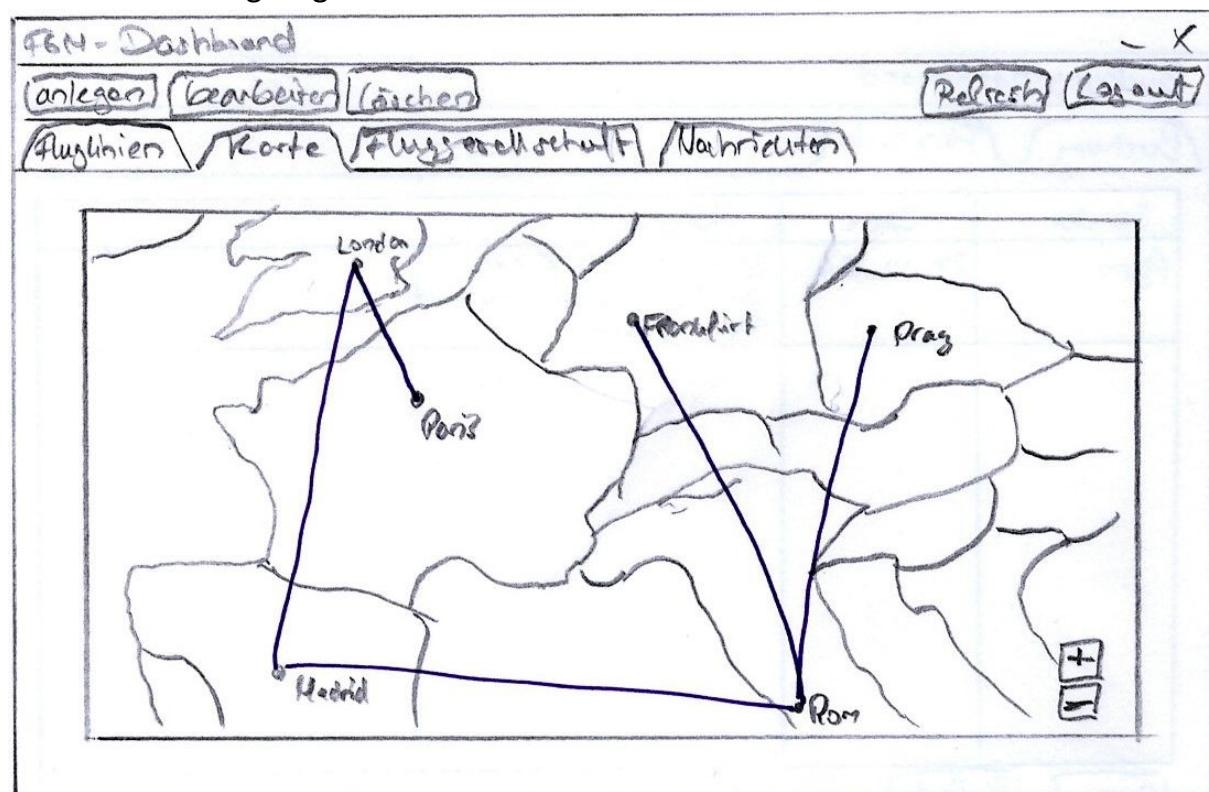
Papierprototypen

01 - Flugübersicht mit Stornierungsoption und Übersicht Rentabilität (Gewinn) und Stornierungskosten

Flugübersicht									- X
Start	Ziel	1. Flug	Interval	Flugzeug	Economy	Business	Rentabilität	Stornokosten	
Augs	DUS	1.10.19	Täglich	Airbus 350	101	10	+ 1080 €	2500 €	
DUS	OTP	2.10.19	Täglich	Airbus 350	198	15	- 300 €	850 €	

schließen **Flug stornieren**

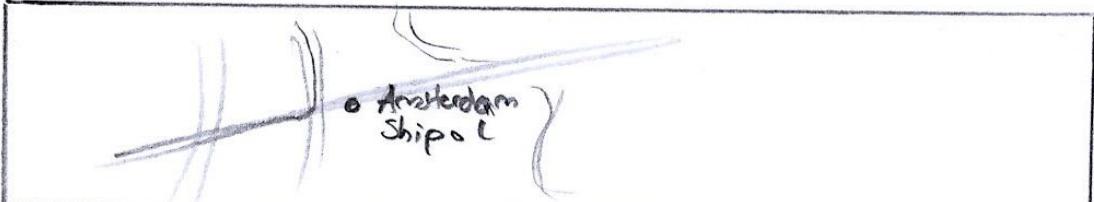
02 - Visualisierung Fluglinien



03 - Informationen zum Zielort mit Option zur Routenberechnung (müsste natürlich zum Startflughafen sein, //TODO)

Informationen zum Zielort

Informationen zu Ihrem Flug nach Amsterdam



Hotels • Restaurants • Impressionen • Attraktionen •

Coffee Shop Flower Power
★★★★★

Indian Restaurant Kumbai
★★★★★

Route berechnen mit:

04 - Pop-Up Start für Routenberechnung

Auto-Route

Gebe deinen Startpunkt an:

Aktivitätsdiagramm

Strukturdiagramm (Klassendiagramm oder Objektdiagramm)

Funktionalitätsplanung

Id	Funktionalität	Geschätzter Aufwand	Verantwortlicher	Abhängige Funktionalitäten	Quellcode-referenz	Status
1.	Fluglinien graphisch darstellen					
1.1	Einbindung GMaps API	1 Tag	Melanie Constantin			
1.2	Mapping Koordinaten auf GMaps, Polylines	1 Tag	Melanie Constantin			
2.	Rentabilität					
2.1	Berechnung Rentabilität pro Flug					
3.	Stornierung					
3.1	Kostenberechnung bei Stornierung					
3.2	Nachrichtenversand bei Stornierung Postbox und Telegram					
3.3	Stornierung Flug					
4.	Informationen zum Zielort					
4.1	Routenberechnung ÖPNV, Auto	1 Tag	Kevin Gohres			
4.2	Anzeige von Hotels, Restaurants, Sehenswürdigkeiten, Fotos, Videos	1 Tag	Kevin Gohres			

Unitests

Id	Getestete Funktionalität	Quellcode Referenz	Status
...			

Systemtests

Datum			
Tester			
SW-Version			
Vorbedin-gung (en)			
Schritt	Aktion (User)	Erwartete Reaktion (System)	✓ / x
1			
2			
3			
Nachbe-dingun g(en)			
Testurteil			

Nutzerhandbuch

Technische Anforderungen

Technische Mindestanforderungen, welche das Programm benötigt, um wie gewünscht bedienbar zu sein.

Installationsanleitung

Genaue Erläuterung, wie das entwickelte Programm vollkommen funktionsfähig auf einem Rechner in Betrieb genommen werden kann.

Bedienungsanleitung

Genaue Erläuterung, wie das entwickelte Programm zu bedienen ist.