

Pflichtenheft Hotelbuchung V1.0

Version	Autor	Matrikel-Nr.	Datum	Status	Kommentar
1.0	Linus Englert	41150721	13.01.2023	Fertig	

PFLICHTENHEFT HOTELBUCHUNG

Inhalt

Pflichtenheft Hotelbuchung V1.0.....	1
1 Zielbestimmung.....	3
1.1 Musskriterien	3
1.2 Wunschkriterien.....	3
1.3 Abgrenzungskriterien.....	3
2 Produkteinsatz	4
2.1 Anwendungsbereiche	4
2.2 Zielgruppen	4
2.3 Betriebsbedingungen.....	4
3 Produktübersicht	5
3.1 Anwendungsfalldiagramm	5
3.2 Komponentendiagramm	6
4 Produktfunktionen	7
4.1 Geschäftsprozesse	7
4.2 Listen	9
4.3 BPMN-Modell.....	9
5 Produktdaten	11
5.1 Stammdaten.....	11
5.2 Bewegungsdaten.....	11
5.3 Klassendiagramm	12
6 Benutzungsoberfläche	13
7 Nicht-funktionale Anforderungen.....	13
7.1 Produktleistungen.....	13
7.2 Qualitätsanforderungen.....	13
7.3 Technische Produktumgebung	14
7.4 Spezielle Anforderungen an die Entwicklungsumgebung.....	14
8 Gliederung in Teilprodukte	14
9 Ergänzungen.....	14
9.1 Docker Image	14
Glossar Hotelbuchung V1.0.....	15
Anhang	17
Selbständigkeitserklärung.....	17

1 Zielbestimmung

Ein Reiseunternehmen bietet Hotelzimmerbuchungen (Einzel- und Doppelzimmer) sowie weitere Dienstleistungen, die zu einem Hotelzimmer hinzugebucht werden können (Flug, Mietwagen oder Wellness-Angebot).

Die zugrunde liegende Architektur besteht derzeit aus einem Monolithen, der auf einer klassischen 3-Schichten-Architektur und einem Client/Server-Modell basiert. Es soll eine moderne, komponentenorientierte Architektur der Webanwendung entwickelt werden, die die Anfragelast über eine skalierbare Plattform verteilt.

Damit die notwendige innovative Technologie und die Skalierbarkeit gegeben sind, soll Docker in Verbindung mit der Kubernetes Plattform verwendet werden.

1.1 Musskriterien

- Buchung Hotelzimmer
- Buchung Flug
- Buchung Mietwagen
- Buchung Wellnessangebot
- Preiskalkulation
- Selbstregistrierung Kunde
- Verwaltung Kundendaten
- Persistieren der Daten in einer Datenbank

1.2 Wunschkriterien

- Statistische Auswertungen
- Historisierung von Daten

1.3 Abgrenzungskriterien

- Kein Missbrauch der Daten

2 Produkteinsatz

Das Produkt ermöglicht Kunden das Registrieren ihrer Daten und Buchen von Zimmern und Dienstleistungen. Des Weiteren ermöglicht es eine Übersicht über alle vorhandenen Daten und deren Verwaltung, sowie statistische Auswertungen.

2.1 Anwendungsbereiche

Das Produkt ist in erster Linie für die Kunden des Reiseunternehmens, damit diese sich selbst registrieren und Hotelzimmer in Verbindung mit Dienstleistungen buchen können.

Zudem bietet das Produkt die Möglichkeit der Verwaltung von Kundendaten sowie statistische Auswertungen darüber. Dies bietet dem Management essenzielle Einblicke und eine solide Basis für Entscheidungen

2.2 Zielgruppen

Benutzer der Anwendung lassen sich in folgende Funktionsgruppen gliedern:

- Kunde
 - Registrierung und Buchungen
- Administrator
 - Verwaltung von Daten im System
- Support
 - Hilfestellung bei Problemfällen, ist notwendig da Kunden selbstständig auf der Web-Oberfläche agieren, aber nicht unbedingt mit der Technik vertraut sind
- Führungsebene
 - Kontrolle anhand von Statistiken und generierten Berichten

2.3 Betriebsbedingungen

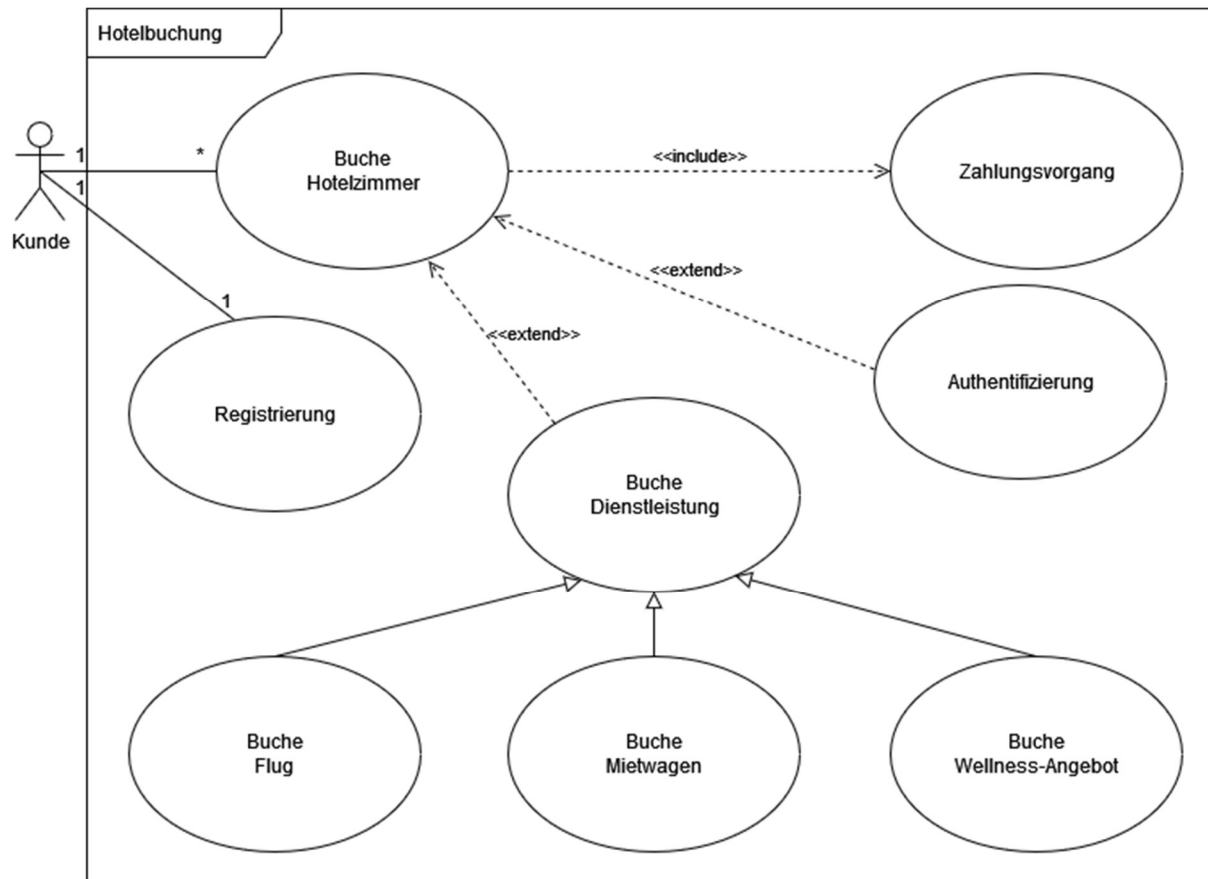
Die Anwendung ist über das Internet für jeden zugänglich und muss daher gegen jegliche Art von Angriff geschützt werden, unter anderem durch Firewall, Intrusion-Detection-System, etc.

Aktivitäten in der Anwendung sind zu protokollieren und gemäß der gesetzlichen Aufbewahrungsfristen sicher aufzuheben.

3 Produktübersicht

3.1 Anwendungsfalldiagramm

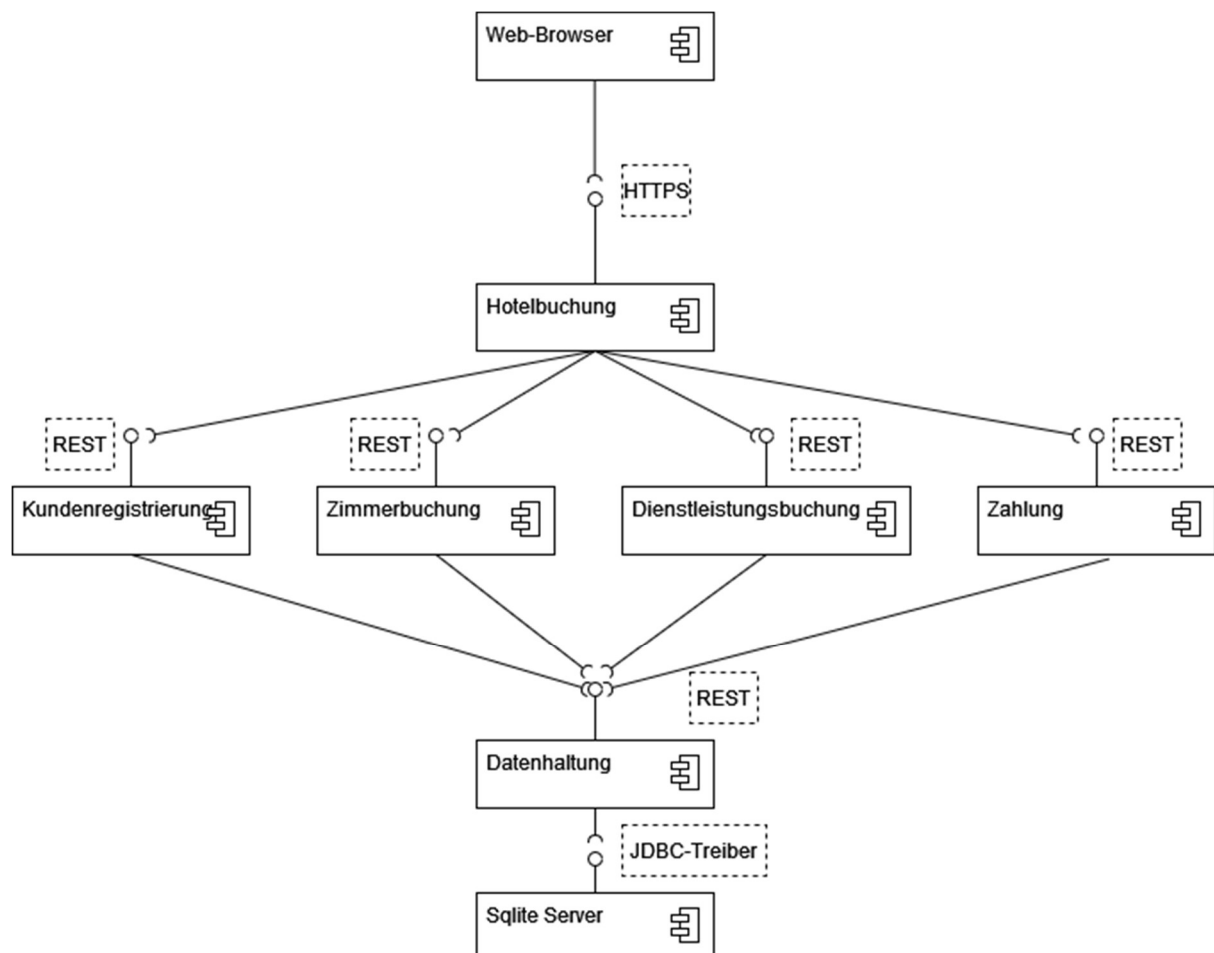
Das folgende Diagramm gibt eine Übersicht über alle Kernprozesse der Anwendung.



PFLICHTENHEFT HOTELBUCHUNG

3.2 Komponentendiagramm

Das folgende Diagramm zeigt das Zusammenspiel aller geplanten Komponenten, die in separaten Containern ausgeführt werden sollen.



4 Produktfunktionen

4.1 Geschäftsprozesse

Geschäftsprozess: Buchung Hotelzimmer

Ziel: Ein registrierter Kunde reserviert und bezahlt ein bestimmtes Hotelzimmer.

Kategorie: primär

Vorbedingung: Der Kunde muss registriert sein. Außerdem muss ein Zimmer verfügbar sein.

Nachbedingung Erfolg: Das betreffende Zimmer ist für den betreffenden Kunden für die vom Kunden gewünschte Zeit reserviert und kann für diese Zeit nicht mehr gebucht werden. Der Zahlungsvorgang war zudem erfolgreich.

Nachbedingung Fehlschlag: Das betreffende Zimmer kann weiterhin gebucht werden oder der Zahlungsvorgang ist fehlgeschlagen.

Akteure: Kunde

Auslösendes Ereignis: Kunde beginnt Hotelzimmer-Buchungsvorgang durch Knopfdruck

Geschäftsprozess: Buchung Flug

Ziel: Ein registrierter Kunde reserviert und bezahlt einen Flug.

Kategorie: sekundär

Vorbedingung: Der Kunde muss registriert sein. Außerdem muss ein Flug verfügbar sein. Flugbuchung ohne Hotelzimmerbuchung ist nicht möglich.

Nachbedingung Erfolg: Der betreffende Flug ist für den betreffenden Kunden für die vom Kunden gewünschte Zeit reserviert und kann für diese Zeit nicht mehr gebucht werden. Der Zahlungsvorgang war zudem erfolgreich.

Nachbedingung Fehlschlag: Der betreffende Flug kann weiterhin gebucht werden oder der Zahlungsvorgang ist fehlgeschlagen.

Akteure: Kunde

Auslösendes Ereignis: Kunde beginnt Flug-Buchungsvorgang durch Knopfdruck

Geschäftsprozess: Buchung Mietwagen

Ziel: Ein registrierter Kunde reserviert und bezahlt einen bestimmten Mietwagen.

Kategorie: sekundär

Vorbedingung: Der Kunde muss registriert sein. Außerdem muss ein Mietwagen verfügbar sein.

Nachbedingung Erfolg: Der betreffende Mietwagen ist für den betreffenden Kunden für die vom Kunden gewünschte Zeit reserviert und kann für diese Zeit nicht mehr gebucht werden. Der Zahlungsvorgang war zudem erfolgreich.

Nachbedingung Fehlschlag: Der betreffende Mietwagen kann weiterhin gebucht werden oder der Zahlungsvorgang ist fehlgeschlagen.

Akteure: Kunde

Auslösendes Ereignis: Kunde beginnt Mietwagen-Buchungsvorgang durch Knopfdruck

Geschäftsprozess: Buchung Wellness-Angebot

Ziel: Ein registrierter Kunde reserviert und bezahlt ein bestimmtes Wellness-Angebot.

Kategorie: sekundär

Vorbedingung: Der Kunde muss registriert sein. Außerdem muss ein Wellness-Angebot verfügbar sein.

Nachbedingung Erfolg: Das betreffende Wellness-Angebot ist für den betreffenden Kunden für die vom Kunden gewünschte Zeit reserviert und kann für diese Zeit nicht mehr gebucht werden. Der Zahlungsvorgang war zudem erfolgreich.

Nachbedingung Fehlschlag: Das betreffende Wellness-Angebot kann weiterhin gebucht werden oder der Zahlungsvorgang ist fehlgeschlagen.

Akteure: Kunde

Auslösendes Ereignis: Kunde beginnt Wellness-Angebot-Buchungsvorgang durch Knopfdruck

Geschäftsprozess: Preiskalkulation

Ziel: Um das gleichzeitige Buchen mehrerer Angebote zu ermöglichen sollen die einzelnen Preise aufsummiert werden.

Kategorie: sekundär

Vorbedingung: Es muss mindestens eine Buchung im Warenkorb sein.

Nachbedingung Erfolg: Der kalkulierte Preis wurde fehlerfrei aus den Einzelpreisen berechnet.

Nachbedingung Fehlschlag: Der berechnete Preis stimmt nicht.

Akteure: System

Auslösendes Ereignis: Eine Buchung wird dem Warenkorb hinzugefügt.

Geschäftsprozess: Selbstregistrierung Kunde

Ziel: Ein nicht registrierter Kunde erstellt ein Benutzerkonto unter Verwendung seiner persönlichen Daten.

Kategorie: primär

Vorbedingung: -

Nachbedingung Erfolg: Der Kunde hat ein funktionierendes Benutzerkonto mit der Möglichkeit Buchungen zu tätigen.

Nachbedingung Fehlschlag: Die Erstellung des Benutzerkontos für den Kunden ist fehlgeschlagen oder das erstellte Benutzerkonto ermöglicht keine Buchungen.

Akteure: Kunde

Auslösendes Ereignis: Kunde beginnt Registrierung durch Knopfdruck

4.2 Listen

Hotel-Liste für alle Benutzer mit folgenden Daten:

Name, Telefon, E-Mail, Adresse, Anzahl Zimmer

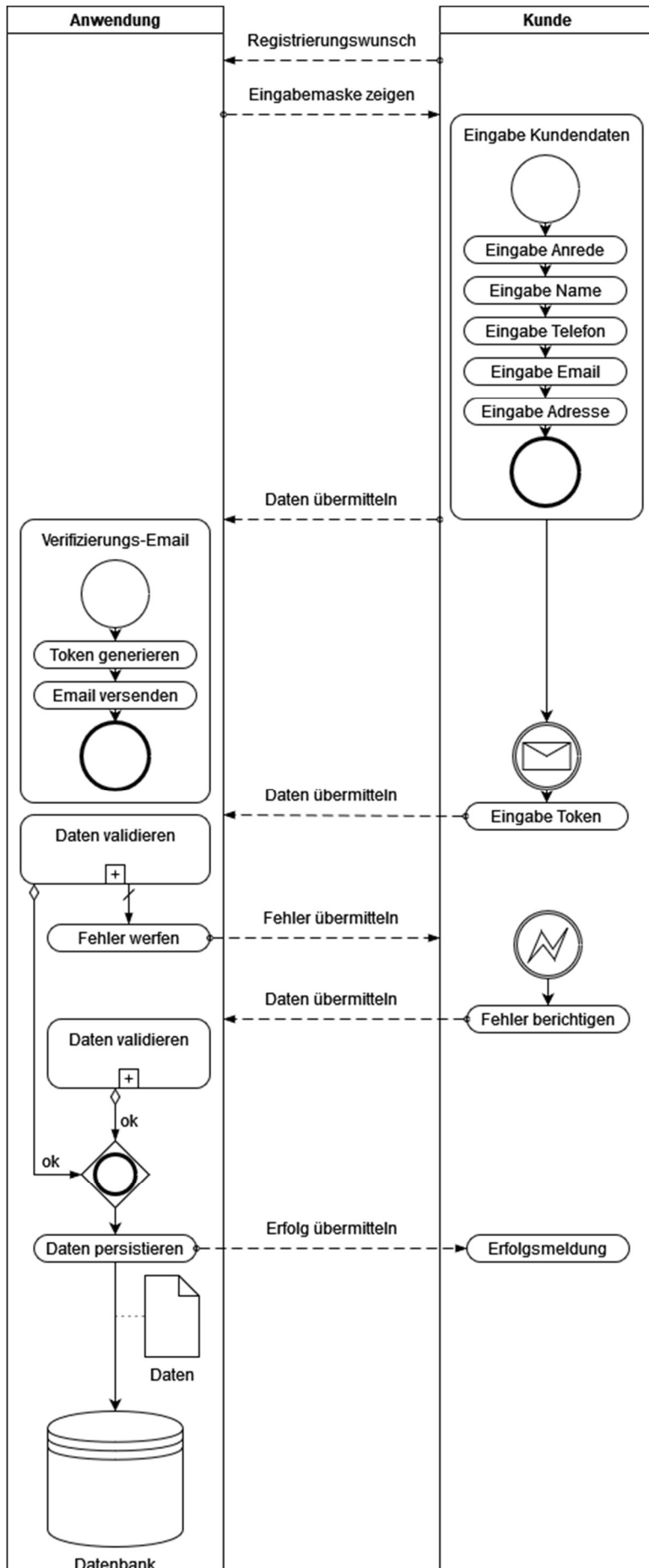
Kunden-Liste für das Reiseunternehmen mit folgenden Daten:

Name, Vorname, Telefon, E-Mail, Adresse, Anzahl Buchungen

4.3 BPMN-Modell

Das folgende Diagramm veranschaulicht beispielhaft, wie ein Selbstregistrierungsprozess in der Anwendung ablaufen könnte. Dabei gibt es eine Kommunikation zwischen Backend (genannt „Anwendung“) und dem Frontend in Form einer Weboberfläche auf der der Kunde agiert.

PFLICHTENHEFT HOTELBUCHUNG



5 Produktdaten

5.1 Stammdaten

Adressdaten: *Straße, Hausnummer, Postleitzahl, Ort, Adresszusatz, Land (Name, Amtssprache, ISO-2-Code, Zeitzone*

Ein *Land* kann mehrere *Zeitzone*n haben.

Hoteldaten: *Name, Anschrift (Anrede (hier: „Hotel“), Telefon, E-Mail)*

Eine *Anschrift* verweist auf mindestens eine *Adresse*. Ein *Hotel* verweist auf beliebig viele *Zimmer* (*Nummer, Zimmer-Typ (Name, Preis, Personenzahl)*).

Kundendaten: *Nummer, Vorname, Nachname, Geburtsdatum, Anschrift (Anrede, Telefon, E-Mail)*

Eine *Anschrift* verweist auf mindestens eine *Adresse*. Ein *Kunde* kann beliebig viele *Zahlungsmethoden* (*Bar, Bankverbindung oder Kreditkarte*) hinterlegen. Ein *Kunde* kann beliebig viele *Buchungen* tätigen.

Dienstleistungsdaten:

Eine *Dienstleistung* kann ein *Flug (Flugnummer)* sein.

Eine *Dienstleistung* kann ein *Mietwagen (Modell)* sein.

Eine *Dienstleistung* kann ein *Wellness-Angebot (Name, Dauer, Wellness-Typ)* sein.

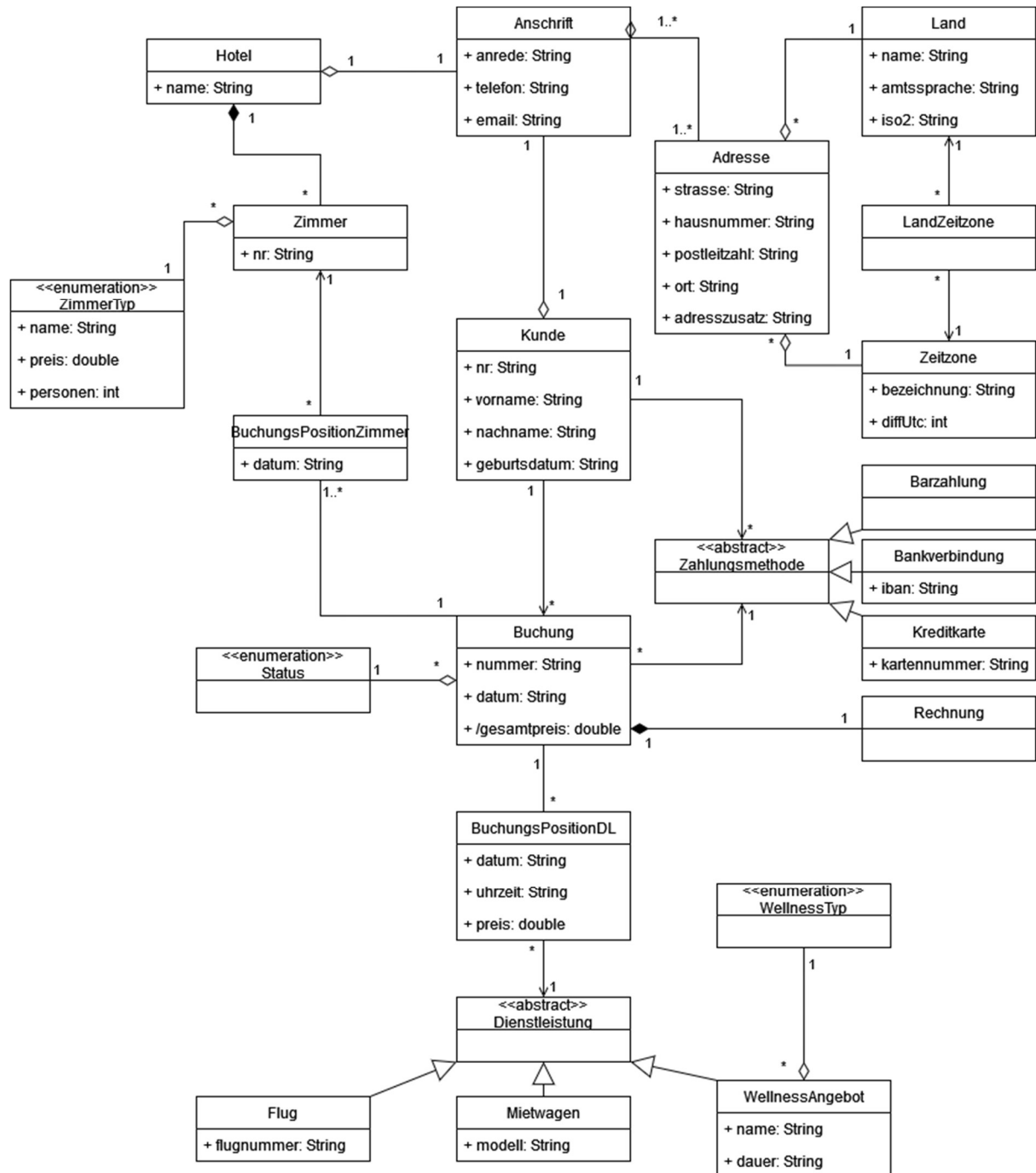
5.2 Bewegungsdaten

Buchungsdaten: *Nummer, Datum, dynamisch berechneter Gesamtpreis, Status, Zahlungsmethode, Rechnung*

Eine *Buchung* verweist auf mindestens eine *BuchungsPositionZimmer (Datum)*, der jeweils ein *Zimmer* zugeordnet ist. Eine *Buchung* verweist auf beliebig viele *BuchungsPositionDL (Datum, Uhrzeit, Preis)*, denen jeweils eine konkrete *Dienstleistung* zugeordnet ist. Eine *Buchung* hat eine zugeordnete *Zahlungsmethode*. Einer *Buchung* ist eine *Rechnung* zugeordnet.

PFLICHTENHEFT HOTELBUCHUNG

5.3 Klassendiagramm



6 Benutzungsoberfläche

Die Web-Oberfläche soll „responsive“ sein, also jedes mögliche Endgerät unterstützen. Eingabe sollen sowohl über Tastatur und Maus als auch per Touchscreen möglich sein.

7 Nicht-funktionale Anforderungen

7.1 Produktleistungen

Alle Reaktionszeiten auf Benutzeraktionen müssen unter 2 Sekunden liegen.

7.2 Qualitätsanforderungen

7.2.1 Funktionalität

Richtigkeit: Das Produkt verhält sich zuverlässig und getätigte Buchungen sind auch wirklich mit den angegebenen Daten abgespeichert. Jeder Fehler wirkt sich direkt negativ auf die Kundenzufriedenheit aus und muss daher vermieden werden. Daher ist diese Anforderung als **sehr wichtig** einzustufen.

Sicherheit: Da es sich um Software handelt, gilt es auf die entsprechende **IT-Sicherheit** und den **Datenschutz** zu achten. Die Anwendung arbeitet mit personenbezogenen Daten, dies geschieht zu gewerblichen Zwecken und innerhalb der EU. Daher sind die Regelungen der DSGVO anwendbar. Um diese zu gewährleisten sind Sicherheitsmaßnahmen notwendig, wie eine Verschlüsselung der Daten und Sichere Kommunikation, beispielsweise über HTTPS. Eventuelle Datenlecks könnten für Kunden sehr unangenehm sein und dem eigenen Ruf erheblich schaden. Daher ist diese Anforderung als **sehr wichtig** einzustufen.

7.2.2 Zuverlässigkeit

Wiederherstellbarkeit: Es ist entscheidend, dass es nicht zu Datenverlusten kommt, damit beispielsweise bereits bezahlte Buchungen nicht einfach „vergessen“ werden können. Für eine Ausfallsicherheit der Festplatten kann ein RAID-System verwendet werden. Des Weiteren sollte mindestens ein Backup am Tag erstellt werden. Daher ist diese Anforderung als **sehr wichtig** einzustufen.

7.2.3 Benutzbarkeit

Verständlichkeit: Die Benutzeroberfläche muss intuitiv sein. Es darf keine Verständnisprobleme geben, wenn Kunden die Anwendung auch ein zweites Mal verwenden sollen. Daher ist diese Anforderung als **wichtig** einzustufen.

7.2.4 Effizienz

Zeitverhalten: Webseitenbesucher sind tendenziell eher ungeduldig. Daher ist es **wichtig**, dass Antworten schnell kommen und längere Prozesse mit einem Ladebalken, der verständlich den Fortschritt widerspiegelt, überbrückt werden. Für das Frontend könnte eine Single-Page-Webanwendung verwendet werden, da dabei Inhalte dynamisch nachgeladen werden, was die Antwortzeiten erheblich verkürzen kann.

7.2.5 Änderbarkeit

Analysierbarkeit: Die Anwendung sollte so geschrieben sein, dass der Code gut lesbar ist, also mit vielen Kommentaren, sprechenden Variablennamen, etc. Andere Entwickler sollten sich problemlos einarbeiten können, um eventuell Änderungen vorzunehmen. Diese Anforderung ist als **normal** einzustufen.

Modifizierbarkeit: Die Anwendung sollte für die Zukunft auch anpassbar sein. Es wäre auch denkbar dafür Schnittstellen zur Verfügung zu stellen, die dann bestimmte Erweiterungen zulassen. Da die Komponenten über Schnittstellen kommunizieren, könnten auch einfach neue Komponenten hinzugenommen werden. Diese Anforderung ist als **normal** einzustufen.

7.2.6 Übertragbarkeit

Austauschbarkeit: Einzelne Komponenten sollten austauschbar sein, um Weiterentwicklungen oder Anpassungen des Angebots zu ermöglichen. Dies wird durch die komponentenorientierte Architektur gewährleistet. Da es sich um eine der Kernanforderungen handelt, ist diese als **sehr wichtig** einzustufen.

7.3 Technische Produktumgebung

Das Produkt ist Client/Server-fähig und Internet-fähig.

7.3.1 Software

Server-Betriebssystem: beliebig, empfohlen wird Linux Debian oder Ähnliches

Client-Betriebssystem: beliebig

7.3.2 Hardware

Server: Skalierbares Kubernetes-Cluster

Client: browserfähiges Endgerät

7.3.3 Orgware

Ein Benutzerhandbuch für Wartungsarbeiten wird mitgeliefert.

7.3.4 Produkt-Schnittstellen

Die Kommunikation der Komponenten erfolgt über REST-Schnittstellen.

7.4 Spezielle Anforderungen an die Entwicklungsumgebung

Keine Abweichungen von der Produktumgebung.

8 Gliederung in Teilprodukte

Jede Komponente hat ihren eigenen Docker-Container. Diese können zwar einzeln ausgeführt werden, sollten aber immer gemeinsam gestartet werden, um eine volle Funktionsfähigkeit zu gewährleisten.

9 Ergänzungen

9.1 Docker Image

Zur Veranschaulichung der Docker-Technologie wird ein Docker-Image mitgeliefert, das eine REST-Schnittstelle mit Java Spring Boot darstellt. Nähere Informationen dazu finden sich in der mitgelieferten README-Datei.

Glossar Hotelbuchung V1.0

Version	Autor	Matrikel-Nr.	Datum	Status	Kommentar
1.0	Linus Englert	41150721	13.01.2023	Fertig	

Begriff	Definition
Bank	Unternehmen, das Geld- und Kreditgeschäfte betreibt und den Zahlungsverkehr vermittelt.
Bewegungsdaten	Dynamische Daten, die aus Transaktionen stammen.
BPMN	Die „Business Process Model and Notation“ ist eine grafische Spezifikationssprache in der Wirtschaftsinformatik und im Prozessmanagement. Sie stellt Symbole zur Verfügung, mit denen Fach-, Methoden- und Informatikspezialisten Geschäftsprozesse und Arbeitsabläufe modellieren und dokumentieren können.
Docker	Docker ist eine freie Software zur Isolierung von Anwendungen mit Hilfe von Container-Virtualisierung. Docker vereinfacht die Bereitstellung von Anwendungen, weil sich Container, die alle nötigen Pakete enthalten, leicht als Dateien transportieren und installieren lassen.
Glossar	Ein Glossar ist eine Liste von Wörtern mit beigefügten Bedeutungserklärungen oder Übersetzungen. Als Anhang eines Werkes wird ein Glossar auch als Wörterverzeichnis bezeichnet, ein eigenständiges Glossar als Wörterbuch.
Hotelbuchung	Eine Reservierung und Bezahlung eines bestimmten Hotelzimmers in Verbindung mit beliebig vielen zusätzlichen Dienstleistungen.
Kubernetes	Kubernetes ist ein von Google entwickeltes Open-Source-System zur Verwaltung von Container-Anwendungen. In diesem Umfeld wird die Verwaltung der technischen Container-Infrastruktur auch als Orchestrierung bezeichnet.
Kunde	Eine natürliche Person, die in der Anwendung registriert, oder dabei sich zu registrieren, ist und die Möglichkeit hat Buchungen über Hotelzimmer und Dienstleistungen zu tätigen.
Monolith	Eine unzertrennbare Anwendung. Das erschwert die Wartung und ermöglicht keine Lastenverteilung bei vielen Anfragen.
RAID	Ein RAID-System dient zur Organisation mehrerer physischer Massenspeicher zu einem logischen Laufwerk, das eine höhere Ausfallsicherheit oder einen größeren Datendurchsatz erlaubt als ein einzelnes physisches Speichermedium.
Reiseunternehmen	Ein Dienstleister, der als Schnittstelle zwischen Urlaubern und Dritten Leistungsträgern (Hotels, Fluggesellschaften, etc.) fungiert, indem touristische Angebote (Pauschalreisen, Bausteinreisen, etc.) veröffentlicht werden.
REST	Representational State Transfer ist ein Paradigma für die Softwarearchitektur von verteilten Systemen, insbesondere für Webservices. REST ist eine Abstraktion der Struktur und des Verhaltens des World Wide Web. REST hat das Ziel, einen

PFLICHTENHEFT HOTELBUCHUNG

Architekturstil zu schaffen, der den Anforderungen des modernen Web besser genügt.

Single-Page-Webanwendung

Als Single-Page-Webanwendung oder Einzelseiten-Webanwendung wird eine Webanwendung bezeichnet, die aus einem einzigen HTML-Dokument besteht und deren Inhalte dynamisch nachgeladen werden. Im Gegensatz dazu bestehen klassische Webanwendungen aus mehreren, untereinander verlinkten HTML-Dokumenten.

Stammdaten

Gespeicherte Daten, die für einen langen Zeitraum gültig bleiben und mehrmals verarbeitet werden.

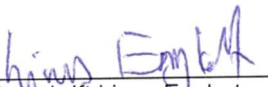
Zimmer

Eine Buchbare Einheit eines Hotels, die eine gewisse Anzahl an Betten und einen bestimmten Preis hat.

Anhang

Selbständigkeitserklärung

Ich, Linus Englert, habe die Studienarbeit zur Veranstaltung Software Engineering I im Wintersemester WS2022/23 bei Prof. Dr. Torsten Zimmer selbständig erstellt, ohne Unterstützung durch andere Personen angefertigt und (nur ich) habe an der Erstellung mitgewirkt.


Unterschrift Linus Englert