

# Pflichtenheft

01.05.2022

Version 0.1

## Historie der Dokumentenversion

Version	Datum	Autor	Bemerkung
0.1	01.05.2022	Lukas Heiming	Ersterstellung
0.2	07.05.2022	Lukas Heiming	Komplette Überarbeitung

## Inhaltsverzeichnis

<b>1</b>	<b>Anforderungen .....</b>	<b>3</b>
----------	----------------------------	----------

# 1 Anforderungen

## A1 Website Kunde Reservierung

### A1.1 Maske zum Reservieren eines Angebots

A1.1.1 Die Übersichtsseite eines Angebots zeigt alle Daten an, welche in dem entsprechenden Eintrag in der „Angebote“ Tabelle und in der „Ferienwohnung“ Tabelle eingetragen wurden (Adresse, Bilder, Beschreibung der Wohnung, Anzahl Räume, Garten, Wohnfläche, Anzahl Betten, Balkon, Für wie viel Personen, WLAN, Auktion\_EndDatum, Mietzeitraum\_Start, Mietzeitraum\_End, Mietpreis, Stornierbar, Aktueller\_Tokenpreis)

A1.1.2 Wenn ein angemeldeter User auf den Button „Reservieren“ in einem Angebot klickt wird mithilfe des Felds „Tokenstand“ in der Tabelle „User“ überprüft ob der User genügend Tokens hat um ein Gebot zu erstellen.

### A1.2 Maske zum Bewerten einer Wohnung

A1.2.1 Mieter können in der Rechnungshistorie eine Wohnung bewerten, vorausgesetzt das Datum im Feld „Mietzeitraum\_End“ in der Tabelle „Angebote“ liegt in der Vergangenheit.

A1.2.2 Bewertungen sind auf der Übersicht der Wohnung öffentlich einzusehen (Tabelle: „Bewertung“ verbunden mit Tabelle „Ferienwohnung“).

A1.3 Wenn das Gebot eines Interessenten überboten wurde, wird eine E-Mail zu der in seinem Konto eingetragenen E-Mail-Adresse (Tabelle: „User“ -> Feld: „E-Mail“) geschickt. In dieser befindet sich ein Link um direkt ein neues Gebot abzugeben

A1.4 Wenn die vom Vermieter bestimmte Zeit abgelaufen ist (Tabelle: „Angebote“ -> Feld: „Auktion\_EndDatum“), erhält der aktuell Höchstbietende die Wohnung für den im Angebot festgelegten Zeitraum (Tabelle: „Angebote“ -> Felder: „Mietzeitraum\_Start“, „Mietzeitraum\_End“), Gewinner wird per E-Mail über den Sieg informiert

## A2 Website Kunde Stornierung

A2.1 In der Rechnungshistorie hat der Kunde die Möglichkeit ein gewonnenes Angebot zu stornieren mittels eines Buttons vorausgesetzt der Vermieter hat die dem Angebot das Feld „Stornierbar“ auf „True“ gesetzt. Der Button ist ausgegraut, wenn das Startdatum des Vermietungszeitraums weniger als 7 Tage in der Zukunft liegt/in der Vergangenheit liegt oder das Feld „Stornierbar“ auf „False“ gesetzt wurde.

A2.2 Wird ein gewonnenes Angebot storniert wird dies in der Rechenhistorie von dem Vermieter des Angebotes und dem Mieter als storniert markiert indem das Feld „Storniert“ in dem entsprechenden Eintrag der Tabelle „Rechnungshistorie“ auf „True“ gesetzt wird.

## A3 Website Kunde Rechnungshistorie

A3.1 In der Rechnungshistorie werden alle Einträge der Tabelle „Rechnungshistorie“ in einer Liste angezeigt, welche die gleiche „User\_ID“ haben wie der eingeloggte User und alle Rechnungen welche ein Angebot beinhalten welches der User erstellt hat.

## A4 Website Kunde Registrierung

### A4.1 Maske für Neuerfassung von Kunden

A4.1.1 E-Mail-Adresse dient zwar nicht als Primary Key, darf aber trotzdem noch nicht in der Datenbank (Tabelle: User Feld: E-Mail) vorhanden sein

A4.1.2 Bei der Registrierung müssen gültige Kreditkartendaten angegeben werden, werden in Tabelle Kreditkartendaten gespeichert

A4.1.3 Es muss ein Username angegeben werden, dieser dient zwar nicht als Primary Key, muss trotzdem eindeutig sein, also darf noch nicht in der Datenbank (Tabelle: User Feld: Username) vorhanden sein

A4.1.4 Es müssen Name, Vorname angegeben werden, werden in Tabelle: User Feld: Name und Tabelle: User Feld: Vorname gespeichert

A4.2 Nachdem sich ein Kunde registriert hat muss er sein Konto über eine automatisch versendete E-Mail bestätigen

## **A5 Website Kunde Login**

A5.1 Maske zur Erfassung der Anmeldedaten (E-Mail, Passwort), zuerst wird geprüft ob die E-Mail-Adresse in der Datenbank (Tabelle: User Feld: E-Mail) vorhanden ist, dann wird geprüft ob das Passwort (Tabelle: User Feld: Passwort) dieser E-Mail zugeordnet ist, wenn ja wird der User eingeloggt. Wenn nicht, wird er nicht eingeloggt.

A5.2 Unterhalb des Loginfensters gibt es einen „Passwort vergessen“ Button, um das Passwort zurückzusetzen muss eine Mail Adresse angegeben werden, welche in der Tabelle User vorhanden ist. Von einem Mailserver wird dann eine automatisch generierte E-Mail an diese Adresse versendet, welche einen Link beinhaltet mit dem das Passwort zurückgesetzt werden kann.

## **A6 Website Tokenshop**

A6.1 Im Tokenshop wird von Kunden angegeben wie viele Token sie kaufen möchten.

A6.2 Der aktuelle Tokenpreis wird aus der Datenbank ausgelesen (Tabelle: Tokens Feld: Tokenpreis) und dem Kunden wird der Gesamtpreis angezeigt.

A6.3 Dem Kunden werden alle Kreditkarten angezeigt, welche in der Datenbank mit seinem Konto über den Fremdschlüssel User\_ID verknüpft sind. Vor dem Bezahlen muss er eine davon auswählen.

A6.4 Nach dem Bezahlen erhält der Kunde von einem Mail Server eine Bestätigungsmail mit Rechnung und die Tokens werden gutgeschrieben (Tabelle: User Feld: Kontostand).

## **A7 Website Kontoverwaltung**

A7.1 WKK\_009 wird spezifiziert.

Angemeldete Kunden ändern ihre E-Mail über ein Formular, das aufgerufen wird, wenn man den Link (E-Mail ändern) anklickt.

Auf diesem Formular (Mail-Ändern-Formular) geben die Kunden folgendes an:

- Alte E-Mail (Zur Verifizierung, Textfeld)
- Passwort (Zur Verifizierung, Textfeld)
- Neue E-Mail (Textfeld)
- Neue E-Mail erneut (zur Kontrolle, ob die Neue E-Mail korrekt zweimal eingegeben wurde, Textfeld)

Bei Erfolg:

Der absende-Button für das Formular kann gedrückt werden, dadurch wird eine Mail an die neue E-Mail abgeschickt, in welcher Sie E-Mail bestätigen können, welches eine Mail zurücksendet.

Erst bei empfang der bestätigten Bestätigungsmail werden die Daten an das backend weitergegeben diese Werte in den entsprechenden Datenbank-Tabellen ab.

Bei Misserfolg:

Solange das Formular nicht komplett und korrekt (alte E-Mail, Passwort, neue E-Mail und neue E-Mail erneut) ist werden die Fehler angezeigt und das Formular kann nicht abgesendet werden.

A7.2 WKK\_008 wird spezifiziert.

Angemeldete Kunden ändern ihrem Benutzernamen über ein Formular, das aufgerufen wird, wenn man den Link (Benutzernamen ändern) anklickt.

Auf diesem Formular (Benutzernamen-Ändern-Formular) geben die Kunden folgendes an:

- Passwort (Zur Verifizierung, Textfeld)

- Neuer Benutzernamen (Textfeld, #100 Zeichen)

Neue Bei Erfolg:

Der absende-Button für das Formular kann gedrückt werden, dadurch werden die Daten an das backend weitergegeben diese Werte in den entsprechenden Datenbank-Tabellen ab.

Bei Misserfolg:

Solange das Formular nicht komplett und korrekt (Passwort, Neuer Benutzername) ist werden die Fehler angezeigt und das Formular kann nicht abgesendet werden.

#### A7.3 WKK\_010

Angemeldete Kunden ändern Ihr Passwort über ein Formular, das aufgerufen wird, wenn man den Link (Passwort ändern klickt).

Auf diesem Formular (Passwort-Ändern-Formular) geben die Kunden folgendes an:

- Passwort (Zur Verifizierung, Textfeld)
- Neues Passwort (beinhaltet mindestens einen Großbuchstaben, Sonderzeichen und Zahl, Textfeld)
- Neues Passwort erneut (zur Kontrolle, ob das Passwort korrekt zweimal eingegeben wurde, Textfeld)

Bei Erfolg:

Der absende-Button für das Formular kann gedrückt werden, dadurch werden die Daten an das backend weitergegeben und dieses speichert die neuen Werte in den entsprechenden Datenbank-Tabellen ab.

Bei Misserfolg:

Solange das Formular nicht komplett und korrekt (Passwort, neues Passwort und neues Passwort erneut) ist werden die Fehler angezeigt und das Formular kann nicht abgesendet werden.

#### A7.4 WKK\_012 wird spezifiziert

Angemeldete Kunden ändern Ihr Zahlungsmittel über ein Formular, das aufgerufen wird, wenn man den Link (Zahlungsmittel ändern) klickt.

Auf diesem Formular (Zahlungsmittel-Ändern-Formular) geben die Kunden folgendes an:

- Passwort (Textfeld)
- Altes Zahlungsmittel (Textfeld)
- Alte CVC (Zahlenfeld, akzeptiert nur #3 Zahlen)
- Neues Zahlungsmittel (Textfeld)
- Neues CVC (Zahlenfeld, akzeptiert nur #3 Zahlen)

Bei Erfolg:

Der absende-Button für das Formular kann gedrückt werden, dadurch werden die Daten an das backend weitergegeben und dieses speichert die neuen Werte in den entsprechenden Datenbank-Tabellen ab.

Bei Misserfolg:

Solange das Formular nicht komplett und korrekt

(Passwort, Altes Zahlungsmittel, Alte CVC, Neues Zahlungsmittel, Neues CVC) ist werden die Fehler angezeigt und das Formular kann nicht abgesendet werden.

#### A7.5 WKK\_013 wird spezifiziert

Angemeldete Kunden fügen ein Zahlungsmittel über ein Formular hinzu, welches aufgerufen wird, wenn man den Link (Zahlungsmittel hinzufügen) klickt.

Auf diesem (Zahlungsmittel-Hinzufügen-Formular) geben die Kunden folgendes an:

- Passwort (Zur Überprüfung, Textfeld)
- Neues Zahlungsmittel (Textfeld)
- Neues CVC (Zahlenfeld, akzeptiert nur #3 Zahlen)

Bei Erfolg:

Der absende-Button für das Formular kann gedrückt werden, dadurch werden die Daten an das backend weitergegeben und dieses speichert die neuen Werte in den entsprechenden Datenbank-Tabellen ab.

Bei Misserfolg:

Solange das Formular nicht komplett und korrekt (Passwort, Neues Zahlungsmittel, Neues CVC) ist werden die Fehler angezeigt und das Formular kann nicht abgesendet werden.

#### A7.6 WKK\_014 wird spezifiziert

Angemeldete Kunden löschen ein Zahlungsmittel über ein Formular, welches aufgerufen wird, wenn man den Link (Zahlungsmittel löschen) klickt.

Auf diesem Formular wird eindeutig (oben im Formular/Groß) darauf hingewiesen, dass man nur Zahlungsmittel löschen kann, wenn man mehr als eins bereits hinzugefügt hat.

Auf diesem (Zahlungsmittel-Löschen-Formular) geben die Kunden folgendes an:

- Passwort (Zur Überprüfung, Textfeld)
- Altes Zahlungsmittel (Textfeld)
- Altes CVC (Zahlenfeld, akzeptiert nur #3 Zahlen)

Bei Erfolg:

Der absende-Button für das Formular kann gedrückt werden, dadurch werden die Daten an das backend weitergegeben und dieses löscht die Werte in der entsprechenden Datenbank-Tabellen ab.

Bei Misserfolg:

Solange das Formular nicht komplett und korrekt

(Passwort, Neues Zahlungsmittel, Neues CVC) ist werden die Fehler angezeigt und das Formular kann nicht abgesendet werden.

Solange nur ein Zahlungsmittel angegeben ist, kann dieses nicht gelöscht werden.

#### A7.7 WKK\_015 - nicht im Lastenheft beschrieben

Angemeldete Kunden löschen ihr Konto über ein Formula, welches aufgerufen wird, wenn man den Link (Konto löschen) klickt.

Auf diesem (Konto-Löschen-Formular) geben die Kunden folgendes an:

- Passwort (Zur Überprüfung, Textfeld)

Bei Erfolg:

Der absende-Button für das Formular kann gedrückt werden, die Entsprechenden Daten werden nach einem Tag gelöscht.

Bei Misserfolg:

Solange das Formular nicht komplett und korrekt

(Passwort) ist werden die Fehler angezeigt und das Formular kann nicht abgesendet werden.

### **A8 Website Administratoransicht**

#### A8.1 WA\_003 wird spezifiziert.

Administratoren haben auf allen Websites die Möglichkeit Dinge zu editieren über ein Icon rechts neben den entsprechenden Feldern (Unangebrachte Kommentare/Beschreibungen, geklaute Bilder usw.) Er fungiert auch als Moderator. Das Editieren wird jeweils überprüft, ob der jeweilige User in der Datenbank bei der Tabelle User, das Feld Admin als wahr gesetzt ist.

### **A9 Website Wohnung registrieren**

#### A9.1 Formular um neue Wohnungen zu registrieren

A9.1.1 Folgende Daten müssen angegeben werden: Adresse, Bilder, Beschreibung der Wohnung, Anzahl Räume, Garten, Wohnfläche, Anzahl Betten, Balkon, Für wie viel Personen, WLAN -> dies geschieht über verschiedene Eingabe Felder (Textfelder, Drag and drop (blob storage), Zahlenfelder)

A9.2 Button zum absenden des Formulars(erst wenn alle Felder ausgefüllt sind), die Daten werden dann an das backend geschickt und von dort in die Datenbank übernommen

### **A10 Website Angebot für Wohnung erstellen**

#### A10.1 Formular um Angebot für eine Wohnung zu erstellen

A10.1.1 Vermieter muss über Dropdown Menü auswählen für welche Wohnung er ein Angebot erstellen will

A10.1.2 Zusätzlich muss er Buchungszeitraum, Auktionszeitraum, Preis und ggf. Stornierungsmöglichkeiten angeben

## **A11 Token System**

A11.1 Es gibt einen Tokenshop, in diesem kann man mit Echtgeld Tokens kaufen. Man wählt die gewünschte Anzahl an Tokens aus und wird dann an einen Zahlungsdienst weitergeleitet (Kreditkarte -> altersverifizierung).

A11.2 Nachdem man die Tokens im Shop gekauft hat, erhält man diese auf sein Konto (Tokenguthaben wird in der NavBar angezeigt)

A11.3 Die Tokens werden verwendet um sich das Mietrecht einer Immobilie für einen gewissen Zeitraum zu ersteigern

A11.4 Kunden die länger als 1 Jahr nichts gemietet haben, bekommen eine kleine Menge an Tokens umsonst (Kauflust anreizen)

## **A12 Server Applikation**

A12.1 Rest API/ C# Backend stellt Verbindung mit SQL-Datenbank her.

A12.2 Entgegennehmen von Nutzerinformationen aus dem Registrierungsformular, (Username, Name, Passwort, Vorname, E-Mail, Vermieter) in UserDB. User\_ID wird automatisch erzeugt und Admin Feld wird mit „false“ initialisiert.

A12.3 Admin schaltet bei anderen Nutzern Admin Fähigkeiten in der UserDB frei, dafür wird ein rest API-Aufruf geschaffen der die Identität.

A12.4 Kreditkarteninformationen des Nutzers werden in die KreditkartenDB gespeichert werden (User\_ID, Kartenummer, CVV).

A12.5 Ferienwohnungen werden ebenso per RestAPI in die Datenbank übertragen, die Bilder werden in einer gesonderten DB (Bilder) als BLOB gespeichert, diese werden per Bild\_ID dann mit der Ferienwohnung verknüpft. Weitere Felder welche per Form in der FerienwohnungDB gespeichert werden sind: FW\_ID, User\_ID, Straße, Hausnummer, Ort, PLZ, Wohnfläche, AnzZimmer, AnzBetten, AnzBaeder, WiFi, Garten, Balkon, Beschreibung.

A12.6 Eine weitere api Funktion speichert Bewertungen zu einer Wohnung in der BewertungenDB, dafür wird geprüft ob der user bereits eine Buchung auf dieser Ferienwohnung hat und falls diese Überprüfung erfolgreich ist wird seine User\_ID, FW\_ID, AnzSterne & der Kommentar gespeichert.

A12.7 Wenn ein Nutzer ein Gebot abgibt wird geprüft ob der gebotene Wert den aktuellen Wert überschreitet und ob genügend Credits auf dem Konto des Nutzers sind, falls beide Prüfungen erfolgreich sind wird der Bieter (User\_ID) das Angebot (Angebot\_ID) und der Preis in der GebotDB gespeichert.

12.8 Periodisch werden alle laufenden Angebote auf ihr Ablaufdatum überprüft, wird dieses überschritten wird der aktuelle Höchstbietende (User\_ID) das Angebot (Angebot\_ID) und eine automatisch erstellte RH\_ID in die RechnungshistorieDB gespeichert.

12.9 Ein API call nimmt neue Tokenpreise entgegen, bevor diese eingetragen werden wird allerdings geprüft ob der ausführende Nutzer als Administrator in der UserDB hinterlegt ist.

## **A13 Datenbank**

A13.1 In der Datenbank werden sämtliche Daten gespeichert.

A13.1.1 Accountdaten speichern

A13.1.2 Angebote speichern

A13.1.3 Gebote speichern

A13.1.4 Wohnungsdaten speichern

A13.1.5 Bewertungen speichern

A13.1.6 Rechnungshistorie speichern

#### A13.1.7 Administrator Aktivitätslog speichern