

# System Specification

Name of the Project

Project Name	Template
Project Leader	P. Bauer
Document state	In process
Version	V. 1.0

## Revisions

Date	Author	Change
November 03, 2019	P. Bauer	First version

# Inhaltsverzeichnis

# 1 Initial Situation and Goal

*Zu der Idee kamen wir über mehrere Personen, die uns mitteilten das sie so ein Programm schon öfters benötigt haben und uns fragten ob wir nicht so eines Programmieren könnten. Also haben wir uns im Internet darüber Informiert ob es so eine Bewertungs-Website wirklich noch nicht in so einer Form gibt. Wir erkannten, dass es in diesem Bereich ein Defizit gibt und dass solche Bewertungswebisten nur für AirBnB und der Gleichen zur Verfügung gestellt wurden.*

*Aktuell kann beim Anmieten einer Yacht oder eines Segelbootes auf das Angebot verschiedener Vermieter (Vercharterer) nur so zugegriffen werden, dass der Mieter die einzelnen Websites der Vercharterer besucht und sich das Angebot ansieht. Eine globale Suche von Booten in einer bestimmten Region ist derzeit nicht möglich.*

*Weiters ist es noch nicht möglich, eine unabhängige Bewertung der Geschäftstransaktion einer Vermietung (Zustand des Boots, Qualität der Betreuung, etc.) einzusehen. So ist man derzeit als Mieter noch von den Angaben des Vermieters abhängig und es ist schwierig bis unmöglich, sich ein eigenes und unabhängiges Bild zu machen. Dies gilt sowohl für die textuelle Beschreibung des Bootes als auch für die zur Verfügung gestellten Bilder.*

*Auf der anderen Seite ist es für die Vercharterer auch sehr schwierig ihr Angebot an Booten zu kommunizieren. Jeder einzelne Vermieter muss sein Angebot auf seiner eigenen Website bewerben. Dies impliziert einen signifikanten Aufwand, der vor allem von Kleinvermietern nicht geleistet werden kann.*

*Im Bereich der Ferienzimmervermietung oder auch Ferienwohnungsvermietung gibt es bereits ein Angebot an zentralen Plattformen (Airbnb, HomeAway, etc), welche einen niederschwellig zugänglichen Marktplatz für die Vermietung von Wohnungen und Zimmer ermöglicht.*

*Vorteile: Mieter bekommt für sein Geld das bestmögliche Boot, weil er von anderen Usern die Bewertungen der Boote einsehen kann.*

*Aufgrund der abgegebenen Bewertungen der User zu den Booten und dem Charterer, sinkt das Betrugsrisiko beträchtlich. Des weiteren helfen einem die hochgeladenen Bilder und Kommentare sich ein gutes Bild über das Boot zu machen.*

*In weiterer Folge ist es sicherlich auch eine Erleichterung für den Vermieter. Dieser bekommt einen weiteren Platz sein Boot zu bewerben und erhält gleichzeitig ein konstruktives, hoffentlich gutes Feedback.*

*Aufgrund reger Nachfrage kann man dann auch Werbung auf unserer Website schalten. Potentielle Werbepartner wären, Bootszubehör- bzw. Bootshändler und Städte mit Häfen.*

*Risiko: Bewertungen können durch fehlerhafte Bilder bzw. Kommentare verfälscht*

*werden. Aufgrund der Anmeldung versuchen wir den durch Bots verursachbaren Schaden möglichst klein zu halten*

## **1.1 Initial Situation**

*Der Einsatzbereich unserer Website liegt bei allen Menschen, die sich ein Boot mieten wollen, oder sich ein Boot gemietet haben und eine Bewertung dazu abgeben möchten. Auch können die Vermieter diese Website gut gebrauchen, um ihre Boote anzupreisen. Der Besitzer kann Aufgrund der Kommentare einsehen ob etwas, mit dem zur Vermietung bereitgestellten Boot nicht passt. Somit ist unsere Website vielseitig einsetzbar.*

### **1.1.1 Application Domain**

*Aktuell kann beim Anmieten einer Yacht oder eines Segelbootes auf das Angebot verschiedener Vermieter (Vercharterer) nur so zugegriffen werden, dass der Mieter die einzelnen Websites der Vercharterer besucht und sich das Angebot ansieht. Eine globale Suche von Booten in einer bestimmten Region ist derzeit nicht möglich.*

*Weiters ist es noch nicht möglich, eine unabhängige Bewertung der Geschäftstransaktion einer Vermietung (Zustand des Boots, Qualität der Betreuung, etc.) einzusehen. So ist man derzeit als Mieter noch von den Angaben des Vermieters abhängig und es ist schwierig bis unmöglich, sich ein eigenes und unabhängiges Bild zu machen. Dies gilt sowohl für die textuelle Beschreibung des Bootes als auch für die zur Verfügung gestellten Bilder.*

*Auf der anderen Seite ist es für die Vercharterer auch sehr schwierig ihr Angebot an Booten zu kommunizieren. Jeder einzelne Vermieter muss sein Angebot auf seiner eigenen Website bewerben. Dies impliziert einen signifikanten Aufwand, der vor allem von Kleinvermietern nicht geleistet werden kann.*

*Unser Programm sollte somit diesen fehlerhaften Marktanteil verschwinden lassen.*

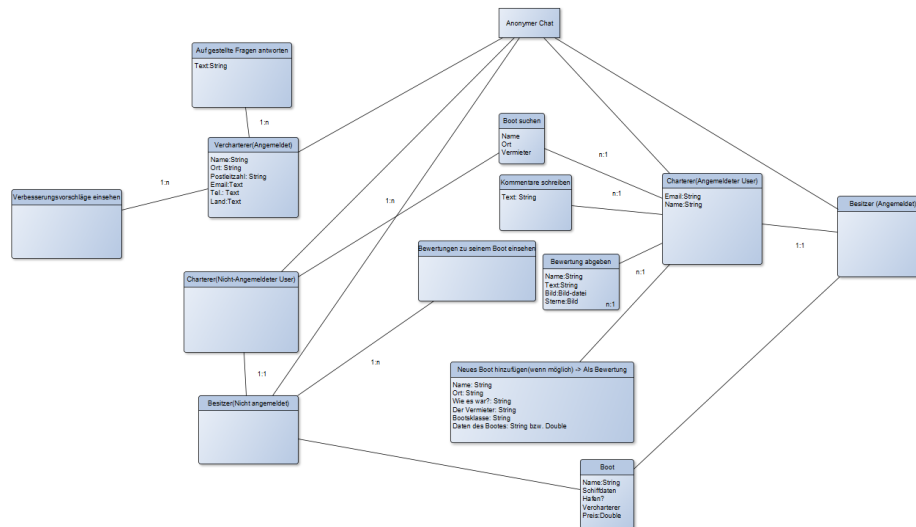
### **1.1.2 Glossary**

*Der Vermieter (Vercharterer) kümmert sich um die Vermietung der Boote. Jedoch gehören diese Schiffe ihm nicht selber, sondern bekommt diese von den Besitzern zur Verfügung gestellt.*

*Der Besitzer selber stellt sein Boot dem Vermieter zur Verfügung, der sich um die Vermietung bzw. um das Bewerben der Boote kümmert. Dieser erhält vom Vermieter natürlich eine Provision dafür (Betrifft unsere Website jedoch nicht). Der Mieter*

der sich gerne ein Boot mieten möchte, kann auf unserer Website zu jenem Boot die Bewertungen einsehen. Des Weiteren kann er nach bestimmten Booten suchen.

### 1.1.3 Model of the Application Domain



Die verschiedenen Userklassen(Gruppen) sind der Besitzer des Bootes (angemeldet bzw. nicht angemeldet), der Vercharterer (angemeldet) und der Charterer bzw. Ex Charterer oder auch normaler Website Besucher (angemeldet und nicht angemeldet) Der Besitzer des Bootes hat das Boot mit Namen, Schiffdaten, Hafen, Vercharterer und Preis. Dieser vermietet das Boot an den Vercharterer, der das Boot dann wieder zum Vermieten freigibt. Der Besitzer kann sein Schiff und die Bewertungen dazu natürlich einsehen, um etwas zu kommentieren, muss er jedoch wie ein normaler Benutzer angemeldet sein.

Der Vercharterer mit Namen, Ort, Postleitzahl, Email, Telefonnummer und Land, kann Verbesserungsvorschläge einsehen und auf diese eingehen. Des weiteren kann er auf gestellte Fragen eingehen.

Der Charterer(ohne Anmeldung) kann ein Boot nach den Suchkriterien (Name, Ort, Vermieter) suchen. Über einen anonymen Chat (siehe Willhaben.at) kann er mit dem Vercharterer bzw. dem Besitzer des Bootes in Kontakt treten. Er kann auch Bewertungen einsehen. Wenn der Charterer angemeldet ist, hat er die Möglichkeit, Kommentare zu schreiben und Bewertungen zu den Booten abzugeben.

## 1.2 Goal

*Benutzer können sich anmelden, sind aber nicht dazu verpflichtet. Durch eine Anmeldung bekommen sie mehrere Features wie:*

- *Boote hochladen (Details der Boote, Bilder, ...)*
- *Beschreibung abgeben*
- *Boote mieten*
- *Bewertungen schreiben, bearbeiten und löschen*
- *Anonymen Chat starten*

*Als unangemeldeter User kann man:*

- *nur Bewertungen lesen*
- *nach Booten suchen (nicht mieten)*

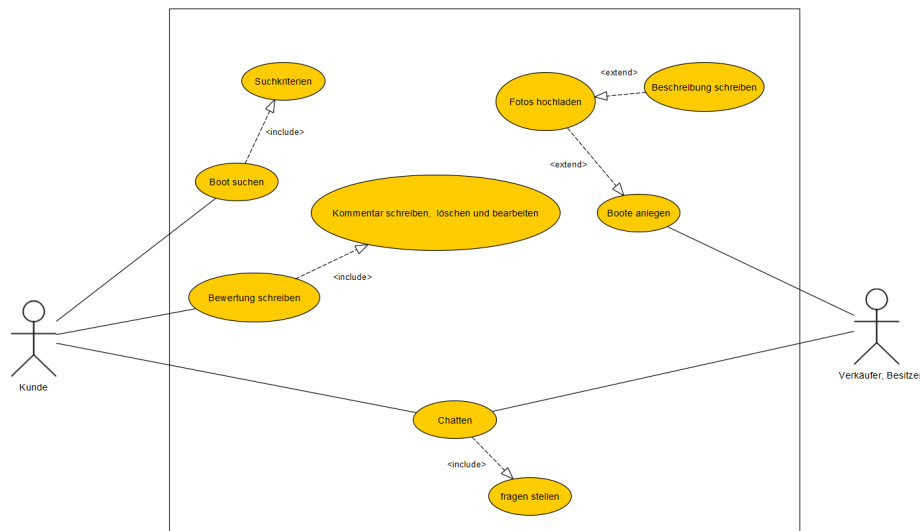
*Gesucht werden kann nach Ort, Vermieter, Schiffstyp und Schiffsname. Weiters ist es möglich Boote nach Preis, Größe, Bewertung und Namen zu filtern. Für Fragen steht ein anonymer Chat zur Verfügung, wo sich Vermieter und Mieter austauschen können beziehungsweise bei Problemen um einen Support Anfragen können. Wenn es weitere Infos gibt können diese über Emails an das Team geschickt werden. Das Ziel dieses Projekts ist es, möglichst viele Boote zur Verfügung zu stellen, damit sich Menschen, die gerne ein Boot mieten möchten, sich ein klares Bild daraus machen können. Mithilfe der Zusammenarbeit ehrlicher Vermieter können klar strukturierte Aussagen gemacht werden, wie zum Beispiel ob ein Boot beschädigt ist, für eine bestimmte Anzahl an Personen geeignet ist oder das Boot dem angegebenen Preis entspricht. Mit der Möglichkeit selbst Boote hochzuladen und eine Beschreibung und Bewertung abzugeben soll anderen Personen helfen, sich für das richtige Boot zu entscheiden. Es besteht weiterhin das Risiko, dass falsche Details zu Booten abgegeben werden. Zur schnellen Übersicht werden die Boote mit der besten Bewertung angezeigt.*

## 2 Functional Requirements

*Functional requirements describe the features of a system which are expected by a user in order to solve a specific problem. The requirements are derived from the business processes and work flows which are supported by the system.*

*The description of functional requirements is accomplished by means of use cases. A use case describes a concrete and self-contained process. The sum of all use cases describes the system behaviour. Describe use cases in plain text and support it by provide clear and illustrative use case diagrams. In case of a very data-oriented application provide a first version of a data model (business domain model). This model is the basis for the data base design in a later project stage. The data model is derived from the entities of the domain model.*

### 2.1 Overview



*1. UseCase: Der Kunde kann ein Boot suchen 2. UseCase: Der Kunde kann eine Bewertung schreiben 3. UseCase: Der Kunde kann chatten bzw. mit dem Support schreiben 4. UseCase: Der Verkäufer bzw. der Besitzer kann chatten 5. UseCase: Der Verkäufer bzw. der Besitzer kann ein Boot anlegen*

### 2.2 Use Case 1: <Boot suchen>

*Der Kunde kann anhand von Suchkriterien ein Boot aus der Datenbank suchen.*



### 2.2.1 General Description

*Give a general description of where a tabular form might help:*

ID:	1
Goal:	The goal of the use case
Precondition:	Under which condition is the user case triggered?
Postcondition:	What conditions are true after the use case was successfully executed?
Involved Users:	Role name: Description of users interacting with the system. "Users" can be other systems, too.

### 2.2.2 UI to call the use case

*Give a sketch of the UI and describe the UX controls.*

### 2.2.3 The Standard Use

*Describe the happy path.*

- *UI*
- *Description*
- *Activity, Sequence, or State Diagram to visualize the workflow*

### 2.2.4 The Non-Standard Use

*Describe the corner cases, possible errors and how the system reacts on them*

- *UI*
- *Description*
- *Activity, Sequence, or State Diagram to visualize the workflow*

### 3 Non-Functional Requirements

*Non-functional requirements describe all aspects of a system that cannot be mapped to a specific feature. Nevertheless these requirements are essential for the system itself. Non-functional requirements are, e.g., quality requirements, security requirements, or performance requirements.*

*Non-functional requirements define basic features of a system which also have an impact on the architecture. They also influence the development costs and, therefore should be formulated in a measurable way. WRONG: The system shall be fast responding. CORRECT: The response time shall be within 500 ms.*

*The following section has to be copied for each non-functional requirement.*

#### 3.1 NFR 1: <Name>

ID:	Identifier of the NFR
Name:	The name of the NFR
Type :	Type as described below
Descritpion:	

The type of the NFR shall be taken from one of the following: The table below shall finally be deleted.

MAINT	Maintenance and portability requirement	Which maintenance or porting effort is expected in the future? Internationalization expected? Porting to different hardware platform?...
SEC	Security requirement	Security requirements comprise confidentiality, data integrity, and availability. How much do we have to consider that data is not accessible to unauthorized persons? Is the correctness and/or consistency of data to be guaranteed? How severe are total system faults?
LEGAL	Legal requirement	Are there any standards or legal constraints to be considered?
USE	Usability Requirement	Usability covers all aspects to make the targeted user like to work with the software.
EFFIC	Efficiency Requirement	Runtime and/or memory efficiency of the program.

## 4 Quantity Structure

*Describe the number of expected records in master data as well as business cases. This assessment is basis to make proper decisions concerning the form of data persistence (e.g., XML or data base) and data base product. Furthermore the quantity structure gives you a better idea about special requirements (e.g. the GUI) for your system because of hight quantity data.*

## 5 System Architecture and Interfaces

*To illustrate how your system is embedded in it's environment list all interfaces to surrounding systems. Interfaces to users, supporting systems, logistics, peer-systems are to be listed and described.*

## 6 Acceptance Criteria

*The acceptance criteria define which criteria the system has to fulfill in order to be accepted. Describe, what has to be checked such that the system can be accepted. Give at least one acceptance test for each functional requirement described in this document. For each acceptance criterion one subsection has to be written.*