System Requirements Document

BoatDocker Application

Group 4

|  |  |  |
| --- | --- | --- |
| Name | Email | Roles |
| Muhammad Mohsin Qamar Khan | [muqm22@student.bth.se](mailto:muqm22@student.bth.se) | Requirement Elicitation,  Domain analysis,  SRS documentation |
| Syed Ali Hassan | [syal22@student.bth.se](mailto:syal22@student.bth.se) | Observation of domain,  Project Goals, stakeholder analysis |
| Muawaz Ayyaz | [muay22@student.bth.se](mailto:muay22@student.bth.se) | Quality requirements, Project scope |
| Sai Prakash Chakla |  | Quality Requirements, Functional Requirements |
| Muhammad Shahzaib | [musz22@student.bth.se](mailto:musz22@student.bth.se) | Data Requirements, Domain level Requirements |
| Hafiz Muhammad Sultan Afridi | [haaf22@student.bth.se](mailto:haaf22@student.bth.se) | Observation of domain,  Project Goals, stakeholder analysis |

Course: Requirement Engineering and Product Management

Project supervisor: Krzysztof Wnuk, Vi

VERSION: Version V1 REVISION DATE: 18-11-2022

Contents

[Introduction: 3](#_Toc119673006)

[Propose: 3](#_Toc119673007)

[Scope: 4](#_Toc119673008)

[Definitions, acronyms, and abbreviations 4](#_Toc119673009)

[Overview 4](#_Toc119673010)

[The above document gives the details specification for the BoatDockApp. 4](#_Toc119673011)

[Goals of the product 5](#_Toc119673012)

[Context diagram for the system 5](#_Toc119673013)

[Stakeholder Identification and analysis 6](#_Toc119673014)

[Requirements Elicitation Techniques 9](#_Toc119673015)

[Elicitation Technique 1 (Observations): 9](#_Toc119673016)

[Elicitation Technique 2 (Interview): 9](#_Toc119673017)

[Elicitation Technique 3 (Brainstorming): 10](#_Toc119673018)

[Elicitation Technique 4 (Reverse brainstorming): 10](#_Toc119673019)

[System Requirements 11](#_Toc119673020)

[Domain Level Requirements: 11](#_Toc119673021)

[Functional Product Level Requirements: 11](#_Toc119673022)

[Data Requirements 17](#_Toc119673023)

[Product Quality Requirements 20](#_Toc119673024)

[References: 21](#_Toc119673025)

[Document Revision History 21](#_Toc119673026)

## Introduction:

This Requirement specification document for a Boat Dock App for RE group (REG). REG is a new in Sweden IT market. A company working on sustainable software system including ecommerce, sales and marketing. The proposed Boat Dock App management system will provide customer and moor owner to publish advertisement and moor dock place for customer, the user need to publish Add, search advertisement, browsing the moor catalog and ability to complete mooring hiring on-line with payment system. This document describes the scope, objectives and goal of the Boat Dock App. In addition to describing non-functional requirements, this document models the functional requirements with use cases, interaction diagrams, and class models. This document is intended to direct the design and implementation of the Boar Dock system.

## Propose:

The purpose of this document is to provide all parties involved with a clear understanding of the scope of that is needed for the ABC Company– Boat Parking Web App. Upon ABC Company internal approval and signoff, this document will serve as an official project implementation document. When necessary, the document may be updated upon agreement. This document is based upon the following points that define the scope of the project. These points are critical to the overall Boat Parking Web App project schedule and budget. They are reflected in the intermediate project deadlines established by the project team. The following is a list of the major project points to be addressed by both the IT project team and ABC Company personnel.

Main functionality of Boat Mooring App:

* Possibilities to registration boat with length, width, height
* Search for the area for moor and must be specified price, distance time date and length, width and depth if not registered.
* Show available alternatives that are compatible with boat
* Information about the mooring price ,distance, revenue
* Costumer option to review the experience with rating
* In App Payment via various gateways
* System should be secure with login with OTP authentication
* Chat Option between Mooring Owner and customer

## Scope:

The Scope of this system is an interactive web based system that support the marketing of location based Boat mooring and hospitality industry of Sweden and Nordic region. The system support directly redirect customer to mooring dock and its existing sales agent network.

The system invoices and account billing are not part of this project. The mobile app is also not part of BoatDockApp. The existing manual system will be replaced by BoatDockApp. In addition, changes to the logical and physical design of the databases are expected and cost will bear by the client. A web search engine and language translator will be obtained as purchased components for the BoatDockApp. Their internal details are not part of this project. Issues of website authentication and security are not part of this project.

# Definitions, acronyms, and abbreviations

|  |  |
| --- | --- |
| **Term** | **Meaning** |
| ABC Company | ABC Company LLC: software company who is developing the application |
| Moor Owner | Boat Parking place owner |
| RE | Requirement Engineering |
| MC | Master card Scheme |
| Visa | Visa Card Scheme |
| SOA | Service Oriented Architecture |
| SOAP | Simple Object Access Protocol |
| ORM | Object Relational Mapping |

# Overview

The above document gives the details specification for the BoatDockApp.

**Section 2:** **Stakeholder Identification and analysis:**  **t**his section we lists the client for development the system. List of all stockholder and the group of interest of importance.

**Section 3:** **Requirements Elicitation Techniques: t**his section lists the requirement elicitation techniques that you used and brief summary of particular technique.

**Section 4:** **System Requirement: t**his section states the requirement at different levels domain and product design with data functional and quality in each level.

# Goals of the product

* The system shall allow for online rent a moor either by customer or sales agent /moor owner. This will eliminate the current delay between their decision to customer and the location owner this will reduce the time a sale agent spend on dead for an online order by 100% this will also cost the process of an sale dead will be reduce to 500 SEK.
* Moore detail and description update within 30 seconds of the database being updated by the product owner. This will reduce the number incorrect location with Google Map API and this will also eliminates the redundant update of customer information.
* The system shall display all information of location, mooring place and price and other facilities associated with company .this feature will improve service by reducing the mean number of web pages a user must navigate per session to 10000 / user. It should reduce unnecessary phone calls to sales agents and staff by 98%.
* The system allows REG Company to view all owner of moor location. An customer / moor owner should able to contact to REG company in one call/email to save time for correct any information
* The system should provide accounting with actual amount of transaction. this will improve the customer service reducing billing complain by 100% in correcting inaccurate account. Reports facilities provide for future uses.
* The system provides accurate location and places and agreement details so this will allow the order to be processed in intently and details updated within 10 seconds.

# Context diagram for the system

Banner, deals, photo update, payment

Banner, deals, information

Approval, updates confirmation

User management, Agreement

Search, Agreement, payment details

Approve Agreement

Payment via debit credit card

Payment via debit credit card

Advertiser/Sales Agent

Web App Admin Team

Customer

Payment System

# Stakeholder Identification and analysis

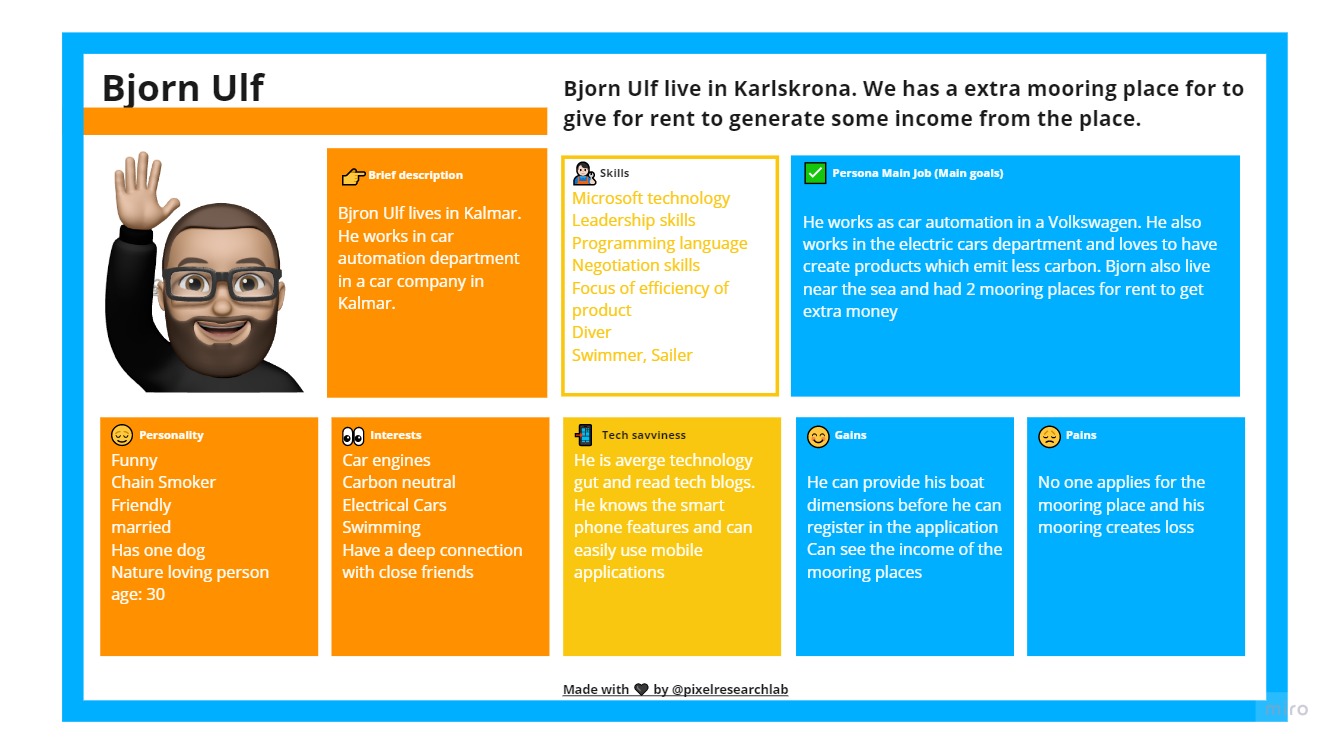
We had an interview with our customer in which we gathered customers’ needs and customer expectations for the application. Through this, we analyzed and understand different type of user for our web application. The list of stakeholders are listed below:

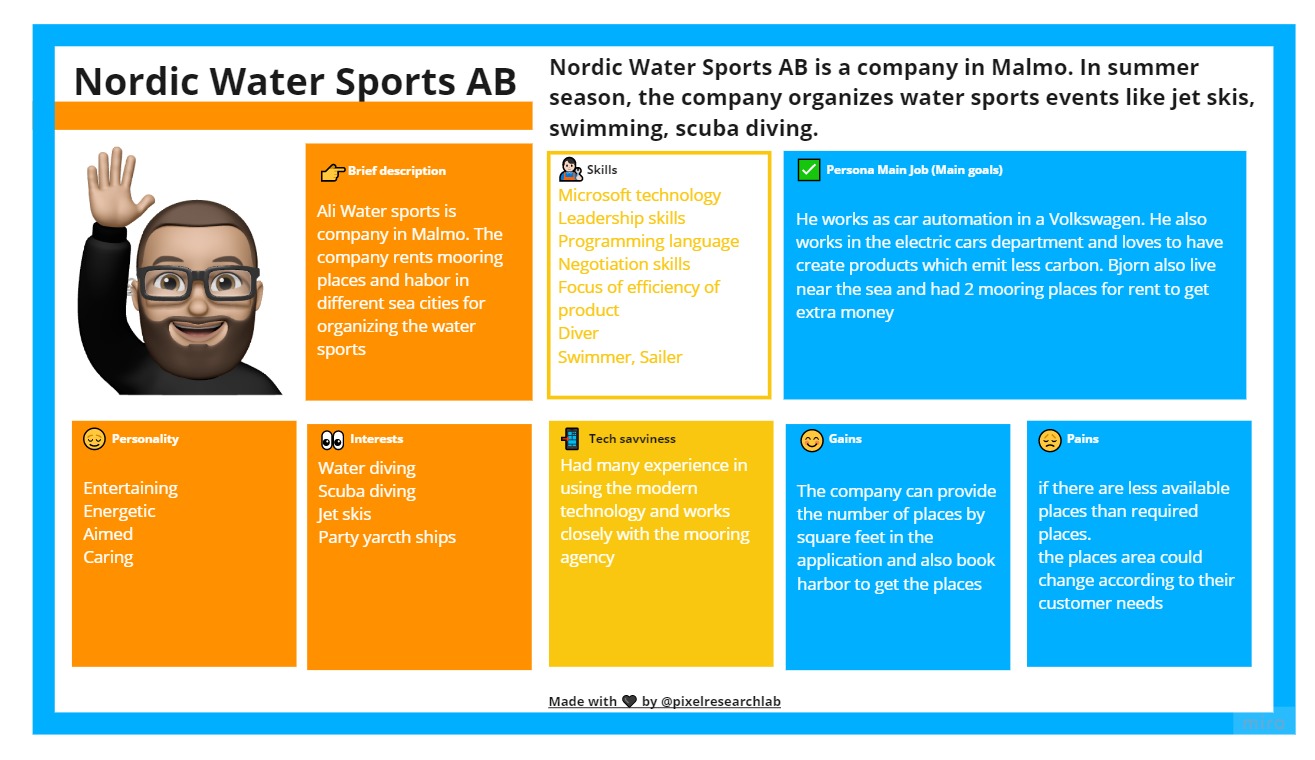
* **Travelers**: The user group travel on their personal boat around the cities and want to rent a mooring place for a short stay. These user need a mooring place near a hotel, restaurant and grocery store where they can buy items, rest and eat. The user can register and login as a customer to rent a mooring place for a short interval of time. Travelers can talk to the mooring place owner through chat after applying for the rental place in the application.
* **Mooring place owners**: the user category has a place to rent out for mooring place. The user can register and upload details of the mooring place he wants to rent out, and duration availability of the mooring place.
* **Harbor owner**: Harbor owner have a place of a big mooring place for rental where he has multiple mooring place for the boats. Big and small boats are moored in the harbor.
* **Boat owner**: Boat owner is a living resident in a city and wants a mooring place to his boat for a yearly basis.
* **Water Event Organization Company:** Water event Organization Company is a company that organizes water sports and events for a particular season in a year. The company rents us multiple places for boats.

Harbor Owners, Mooring place owners are stakeholders which give out the place for rent and act as tenant. Boat owner, Water Event Organization Company, travelers are the stakeholders which register for applying for a mooring place to rent.

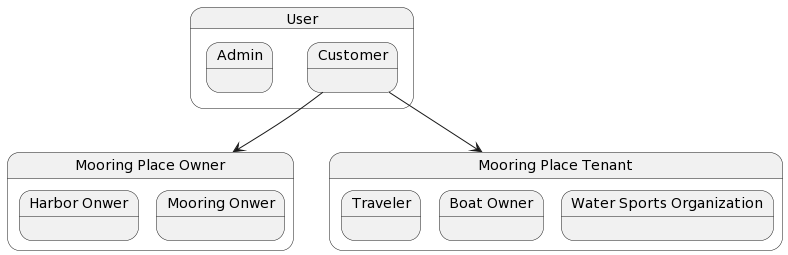
We also developed personas as to identify the user needs.







The total categorized into are defined below:



# Requirements Elicitation Techniques

## Elicitation Technique 1 (Observations):

Through initial requirement one-pager, we started to observe the business requirements of the system. We understand the stakeholder and observed how the system and business value would be developed in the system.

Why we used observation:

After getting project description one-page, we booked a face-to-face meeting with project owners to understand and get requirements. Before taking the interview, we understand the system and made some queries to better understand the system.

Requirements we get from this technique:

The list of requirements we get from this technique are:

* DL1,DL4,FR1,FR2, FR5,FR7,FR10

## Elicitation Technique 2 (Interview):

We interview the product owners on 8 November 2022. We asked about the product business aspects and asked queries which we developed through observation elicitation technique. The interview was done in structured way, the agenda and the process of the interview was determined by us.

Why we used this technique:

Initially, we developed some queries in order to better understand the needs of the business domain. As we know using observation and interview technique together would provide us better understanding of the system and resolve our questions. Through interviews, we also defined the scope of the system by asking the product owners.

Requirements we get from the system:

* DL2,DL3,FR3,FR4, FR6,FR8 ,QR1,QR2

## Elicitation Technique 3 (Brainstorming):

After completing the interviews, we elicit the business level, domain level and planned 2 brainstorming session of one hour in which every group member is involved. We had a session of 40 minutes in which we discussed the solutions from the requirements and developed the design of the system.

Why we used this technique:

After conducting the interview from the product owner, we needed to brainstorm the possible solutions of the requirements and design a model of the application.

Requirements we get from the system:

* DL5,DL6,FR10,FR14

## Elicitation Technique 4 (Reverse brainstorming):

We did a session as team in reverse brainstorming for our project in order identify the potential problem and address them and think about the possible solutions.

Why we used this technique:

We used this technique because we have a large system which are many user types. To remove any conflict of the user group’s requirements, we identified the issue through reverse brainstorming.

Requirements we get from the technique:

* FR9,FR11,FR12,FR13,QR4,QR5,QR6,DL4,DL2



# System Requirements

We used different techniques for system requirements. Through this techniques, we developed requirements of different levels.

## Domain Level Requirements:

DL1: Product shall have a chat box for interaction between tenant and mooring place owner.

DL2: The product support an interactive map from where user can search the areas and mooring places in the area along with restaurants and hotels

DL3: The product support payment procedure from the users.

DL4: The mooring tenant able to give length, width and height of their boat in registration.

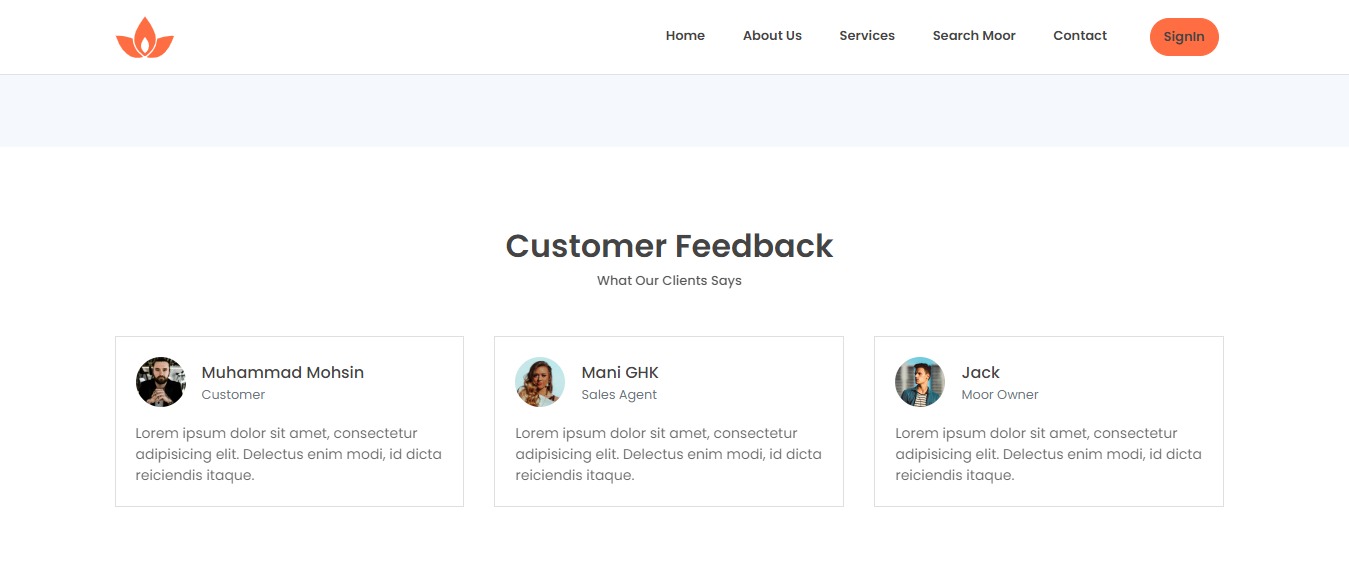
DL5: The system shows highlighted mooring places where hotels and restaurants are more rated

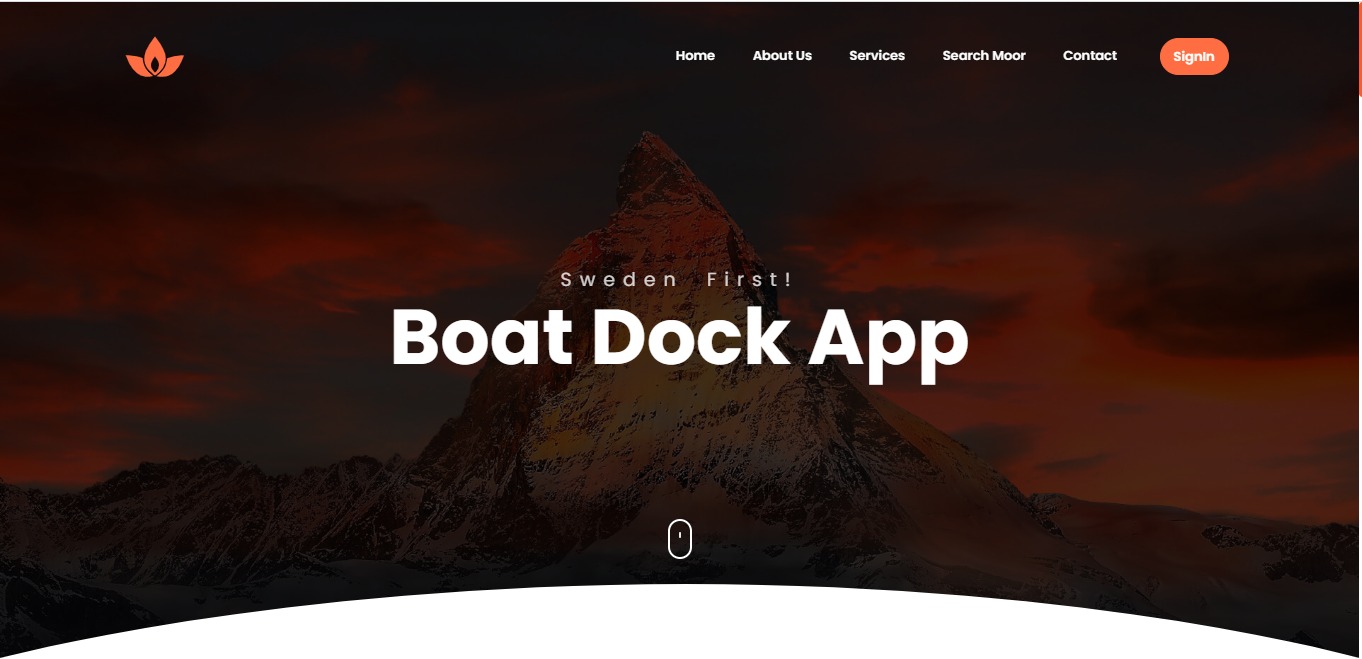
DL6: the system allow tenant to rate the experience of the mooring place and mooring place owner to rate the tenant

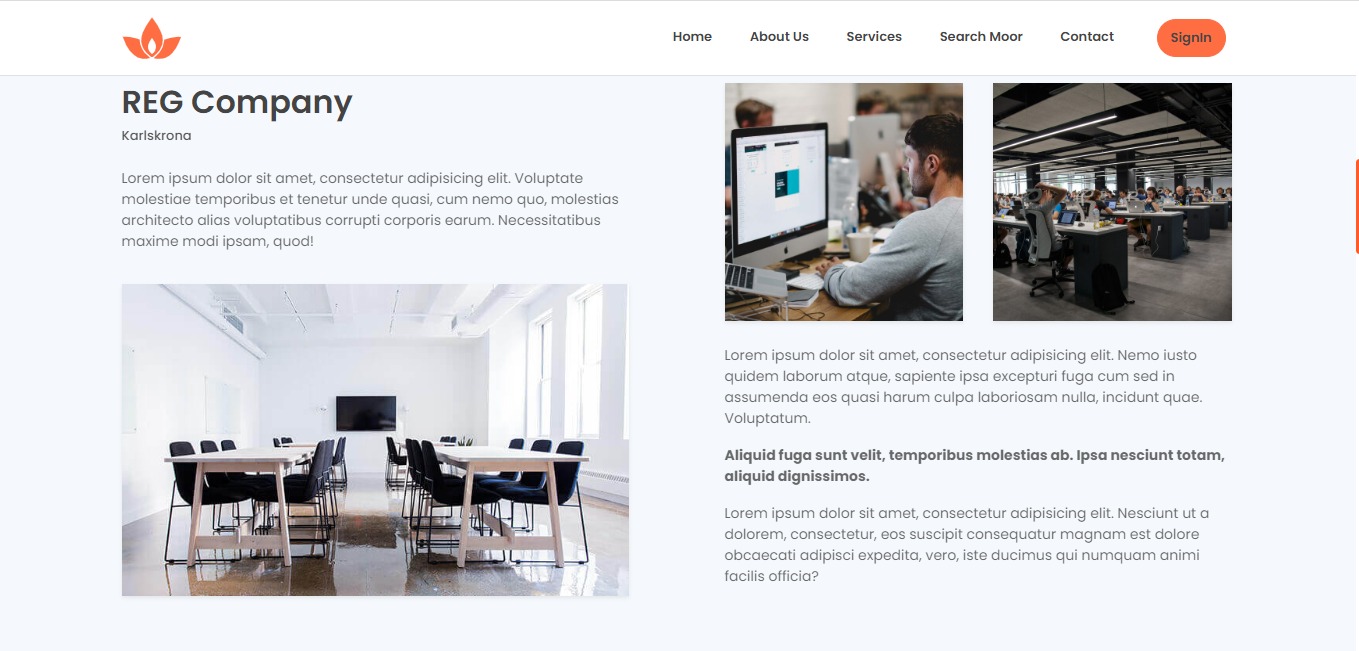
## Functional Product Level Requirements:

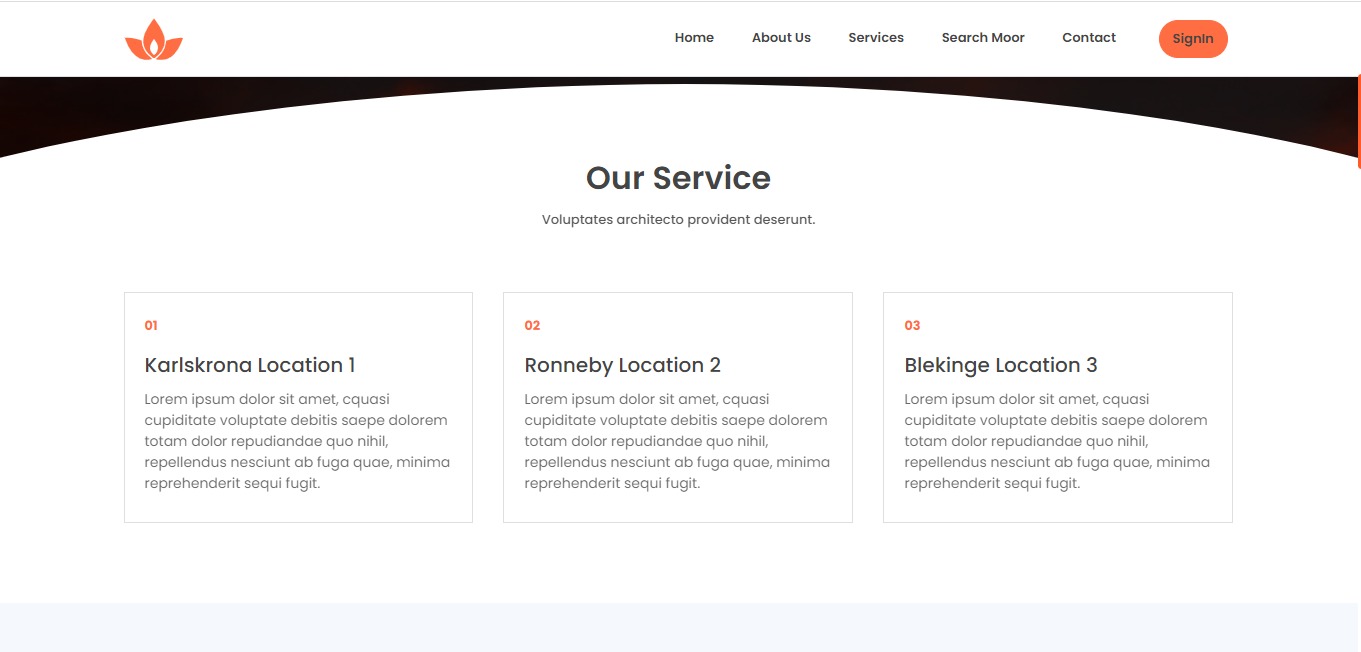
We did 2 task descriptions and 2 use cases in order to define the functional product level requirements and added screens and prototypes

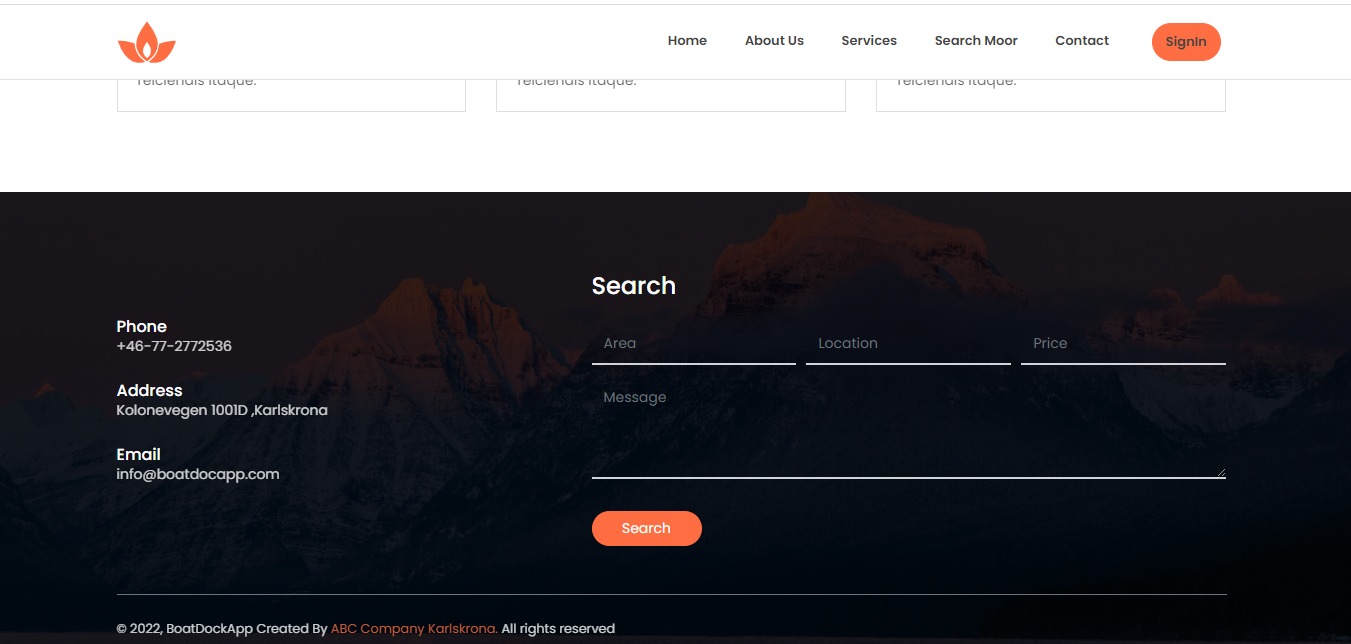
**Screens and prototypes:**

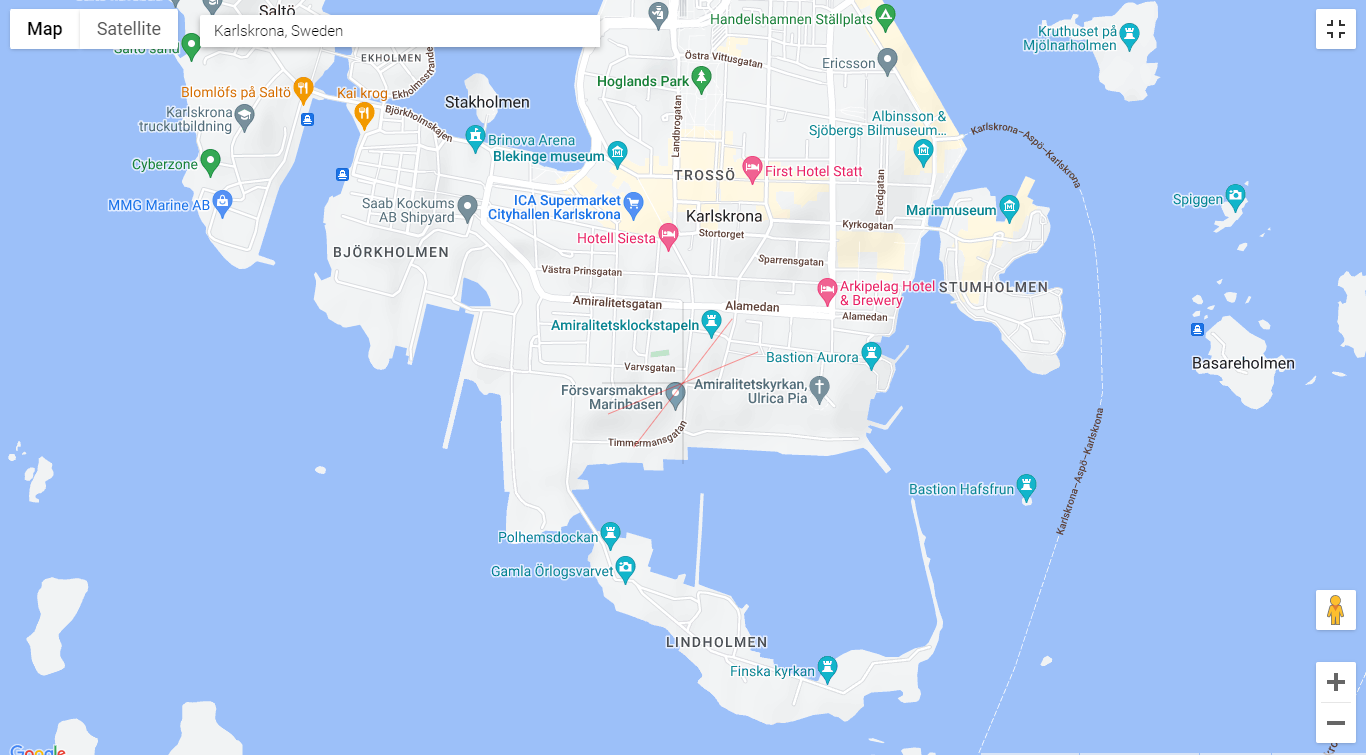
****

****

****

****

****

****

**Task Descriptions:**

|  |
| --- |
| **Task Name**: 1.1 Registration |
| **Purpose**: Registered customer to Mooring App  **Trigge**r:  **Precondition**: customer looks to mooring boats and registered first  **Frequency**: 0.6 customers / minute (customer use the application)  **Critical**: user already exsist/ user block /account marked suspecious |
| **Sub-tasks:**   1. Registered to application 2. Gives personal information 3. Geographical information |
| **Variants:**   1. User already exsist 2. False information 3. Authentication not confirm |

|  |
| --- |
| **Task Name**: 2.2 Searching for boat |
| **Purpose**: searching for the mooring place  **Trigge**r:  **Precondition**: customer client search for mooring places by region/ county  **Frequency**: 0.6 average per day  **Critical**: no mooring places available /booking full |
| **Sub-tasks:**  1) first search the moor by using boar specification  2) contact to owner about more details  3) provide document for contract and other fomalaties  4) Deal confirmation |
| **Variants:**   1. Moor arleady accupied by others 2. Places is not good for mooring boat 3. Hotels / facilities too far away from boat |

**Use Cases:**

|  |  |
| --- | --- |
| **Use Case Name** | 1.1 Customer Registration |
| **Brief Description** | Customer/user registered to applicaiton / login to App |
| **Actors** | Customer/ Moor Owner/ client |
| **Precondition** | User need user id and password to access the App |
| **Basic flow** | 1)Customer download the applicaton  2) Customer registered to application by providing user email / password  3) Cusotmer recieved confirm |
| **Alternative flow** | 1)Customer call to Support  2)provide information and registered. |
| **Exit conditions** | Logout from application |

|  |  |
| --- | --- |
| **Use Case Name** | 1.2 Search for Mooring Boat |
| **Brief Description** | Customer search for the mooring place |
| **Actors** | Cusotmer /client /moor owner |
| **Precondition** | Already search for the boat /User access the application via user id/ password |
| **Basic flow** | 1. Customer search for mooring place by filteres 2. Find places by customer deals/ cheap prices 3. After find place need to contact owner for details 4. Sign contract and payment |
| **Alternative flow** | Customer email to Mooring agency for registration and mooring place by mooring place ID /code |
| **Exit conditions** | Logout from application |

PR1: User shall be able to search mooring places by entering details by search by city/location/area name in the search box

PR2: User shall be able to view the hotels, restaurants, leisure places and petrol pump near the mooring place in the maps.

PR3: User shall be able to select the mooring place and apply for registering

PR4: User shall be able to enter payment details in the application

PR5: User shall be able to pay the mooring place rent through in-app payment gateway

PR6: User shall be able to select download the rent contract

PR7: User shall be able to add mooring places in the system

PR8: User shall be able to approve renting request in to the mooring place owner

PR9: User shall be able to rate the mooring place experience after the rent time

PR10: User shall be able to rate the mooring place tenant

PR11: User shall be able to register as a tenant

PR12: User shall be able to enter boat dimensions (length, width and height) in its user profile

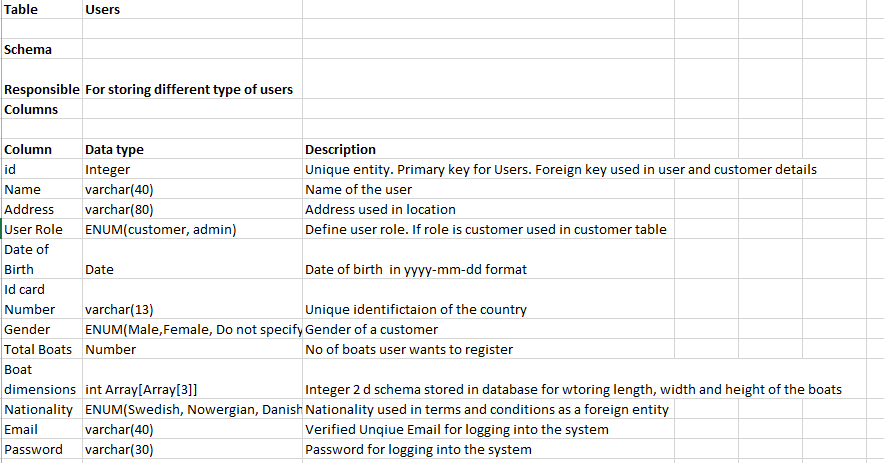
PR13: User shall be able to select the duration of rent by days to rent out mooring

PR14: User as a mooring place owner shall be able to view the rental request of the mooring place.

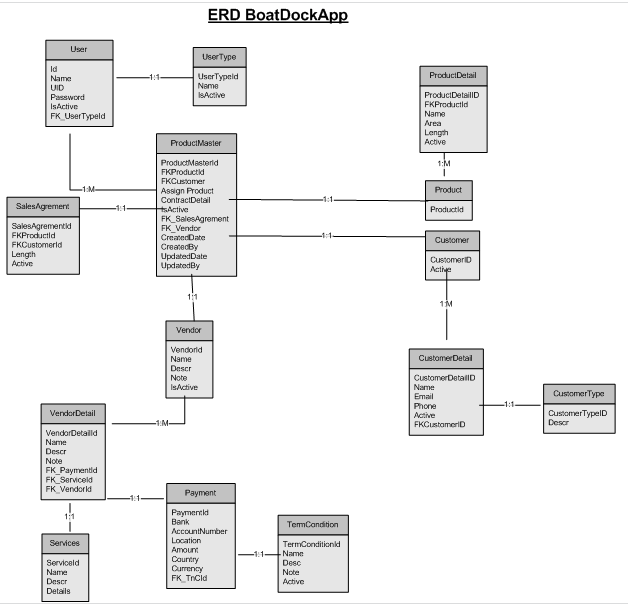
## Data Requirements

We used data dictionary, one prototype and ER model diagram to get data requirements of the system. We are doing rational data base as we can define class ad entity relation model which suits our Object oriented approach.

**Data Dictionary:**



**E/R Model:**



DR1: User with User-Role as Admin, shall be able to view all the user with their roles in the system

DR2: User with User-role as mooring place tenant, shall be able to provide boat dimensions in the system

DR3: User have following required attributes (name, Id number, Date of birth, nationality, phone number, email and password).

DR4: User as user Role: Lesser shall be able to add the terms and conditions of the mooring place

DR5: User as User Role: Renter shall be able accept the terms and conditions of the mooring place.

## Product Quality Requirements

We used QUPER Model for getting the performance requirements. We also measured the cost/benefit of developing features in house or integrating it with 3rd party.

**QUPER Model:**

**Creation:**

1. **Quality Aspect:**

* **Performance:** System response time for each page (5 Sec)

1. **Reference list/ competitors**

* **Canal & River Trust: (**Take 2 sec)

1. **Quality Break points**

* **Utility:**  4 Sec: All page load time
* **Differentiation:** 3Sec : Filtration activate
* **Saturation:** 2 sec: Mooring Place booked

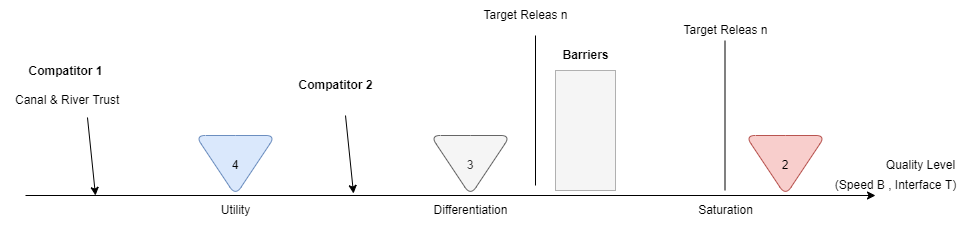
1. **Barriers**

* **Steep cost: 5 sec:** payment system
* **Steep Cost: 3 sec**: new architecture

1. **Target**

* **Good: 3 Sec:** This target is possible to create an own payment system without using third party service.
* **Stretch: 3 Sec:** If new S/w Architecture is feasible.

**Road Map:**

****

**Reliability:**

QR1: System must be reliable to maintain its working for At least 1 billion users at a time.

QR2: System must be fault Tolerance and ensure that work properly during fault detection.

QR3: System should be available 22 hours every day 97.3%.

**Availability:**

QR4: System must be available 24\*7 hour and must have less than 1 mint down time within a week/month.

**Performance:**

QR5: System response time of every page must not exceed 5 sec.

QR6: All the applied filters must generate result within 5 sec

**Security:**

QR7: User cannot perform any contractual agreement about the mooring place without explicit user authorization.

**Maintenance:**

QR8: System must be modifiable at any time in future to be existing features.

# References:

|  |  |
| --- | --- |
| [1] | M. Suiab, "Requirment Elicitation Technique :- A Review Paper," *International Journal of Computer & Mathematical Sciences,* vol. 3, no. 9, pp. 2347-8527, 2014. |

# Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Name | Description |
| 1.0 | 10-11-2022 | Muhammad Mohsin, Syed Ali Hassan, Muawaz Ayyaz, Hafiz Muhammad Sultan | First version of the SRS document. We started working on section 1 on the document |
| 1.1 | 12-11-2022 | Sai Parkas Chakla, Syed Ali Hassan, Muhammad Shahzaib, Hafiz Muhammad Sultan | We did analysis of the domain and written the elicitation techniques we are going to use in the document |
| 1.2 | 14-11-2022 | Muhammad Mohsin, Syed Ali Hassan, Muhammad Shahzaib, Sai Prakash, Muawaz Ayyaz, Hafiz Muhammad Sultan | After interview with the product owners, we started writing system requirements |
| 1.3 | 17-11-2022 | Muhammad Mohsin, Syed Ali Hassan, Muhammad Shahzaib, Sai Prakash, Muawaz Ayyaz, Hafiz Muhammad Sultan | We finalized the version 1 with complete system requirements |