

---

# **Software Requirements Specification**

**for**

# **PawTravel**

**Version 1.3**

**Authors:**      **Dawid Grobert**  
                     **Julia Boczkowska**  
                     **Michał Zieliński**  
                     **Rafał Synoradzki**  
                     **Bartosz Ryczek**  
                     **Daniel Marek**

**Modified: 11.03.2022**

**Created: 11.03.2022**

# Table of Contents

<b>Table of Contents</b>	<b>ii</b>
<b>Revision History</b>	<b>ii</b>
<b>1. Introduction</b>	<b>1</b>
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References	1
<b>2. Overall Description</b>	<b>2</b>
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	2
2.5 Design and Implementation Constraints	2
2.6 User Documentation	2
2.7 Assumptions and Dependencies	3
<b>3. External Interface Requirements</b>	<b>3</b>
3.1 User Interfaces	3
3.2 Hardware Interfaces	3
3.3 Software Interfaces	3
3.4 Communications Interfaces	3
<b>4. System Features</b>	<b>4</b>
4.1 System Feature 1	4
4.2 System Feature 2 (and so on)	4
<b>5. Other Nonfunctional Requirements</b>	<b>4</b>
5.1 Performance Requirements	4
5.2 Safety Requirements	5
5.3 Security Requirements	5
5.4 Software Quality Attributes	5
5.5 Business Rules	5
<b>6. Other Requirements</b>	<b>5</b>
<b>Appendix A: Glossary</b>	<b>5</b>
<b>Appendix B: Analysis Models</b>	<b>5</b>
<b>Appendix C: To Be Determined List</b>	<b>6</b>

## Revision History

Name	Date	Reason For Changes	Version
Dawid Grobert	11.03.2022	The document has been created	1.0
Michał Zieliński, Rafał Synoradzki, Daniel Marek	23.03.2022	Partially updated section 1, 2 and 3	1.1
Michał Zieliński	28.03.2022	Updated document with new data	1.2
Rafał Synoradzki	30.03.2022	Modified section 3 of the document, performed an overall document refactoring	1.3

# **1. Introduction**

## **1.1 Purpose**

The purpose of this project is to create a website which allows its users to find travel offers as well as post their own.

## **1.2 Intended Audience and Reading Suggestions**

The website is targeted to anyone looking for cheap vacations or interesting travel guides. Users can create free accounts which unlock many functions including the possibility to connect with other users, posting offers and guides and evaluating others.

## **1.3 Product Scope**

The purpose of this platform is to provide a space for individuals seeking a better deal on their holiday trips and them sharing their own discoveries. To store the offers, a relational database system is going to be used, and an account system will be implemented.

## **1.4 References**

- <https://www.pepper.pl/> The biggest website aggregating offers in Poland.
- <https://www.wakacyjnipiraci.pl/> Website with many offers related to travelling.

# **2. Overall Description**

## **2.1 Product Perspective**

This website is a standalone product, it is not a part of a larger application ecosystem. It is being provided as an alternative to already existing services of a similar type (i.e. a browser for travel offers).

## **2.2 Product Functions**

The product's functionality is based around offers related to traveling. Therefore, the main functionalities available to the users are:

- Browsing existing offers
- Filtering offers to match user criteria
- Posting offers found by the user
- Rating offers of other users

## **2.3 User Classes and Characteristics**

- Unregistered user - can browse and display any available offer, is able to create an account.
- Registered user - can post their own offers, can leave feedback on other offers in the form of comments and votes, can use the internal messaging system to contact other registered users.
- Administrator - can moderate entire user posted content on the website

## **2.4 Operating Environment**

The project is a web based application, therefore It only requires a Django-compatible web server to operate. From the user's side, a modern web browser is required.

## **2.5 Design and Implementation Constraints**

The team is composed of young software developers without real experience. The team must plan carefully during the design process to go through the implementation phase smoothly.

## **2.6 User Documentation**

As a website application, we don't plan to prepare technical documentation for the user. However, we will prepare some simple guides which will help use our product.

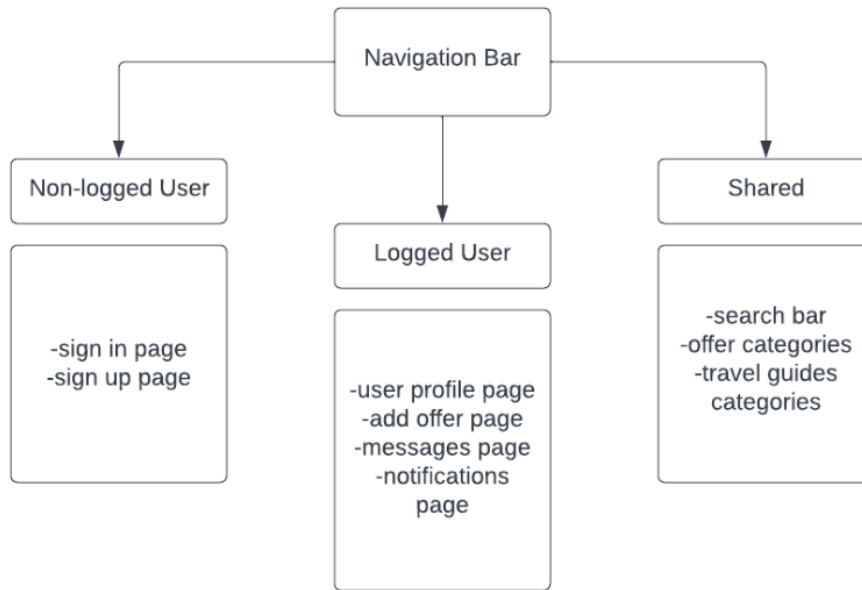
# **3. External Interface Requirements**

## **3.1 User Interfaces**

### **3.1.1 Navigation Bar**

The navigation bar appears on every page, therefore it is an important part of the user interface. It holds some top level functions:

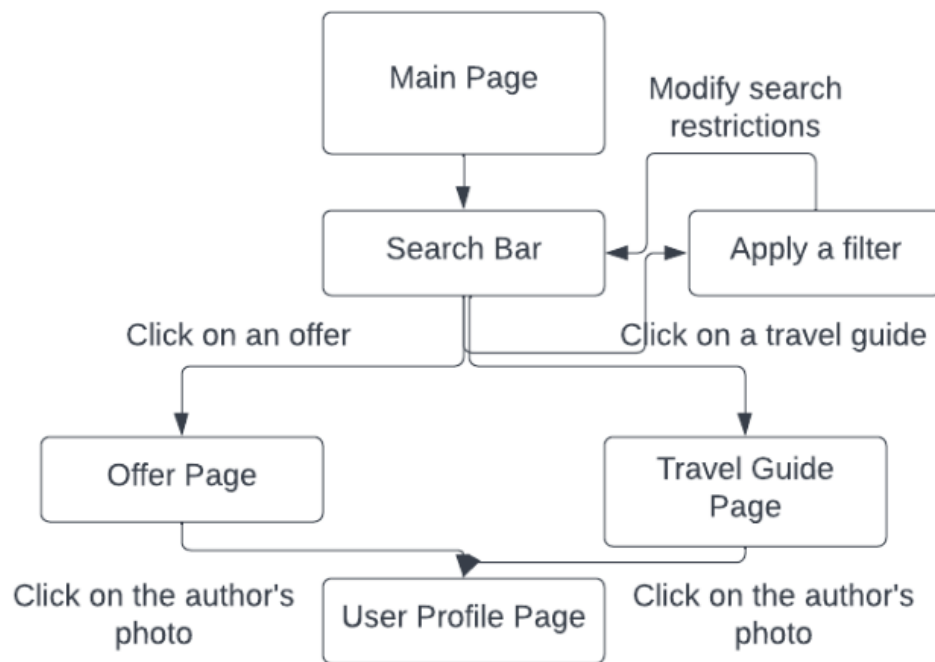
- option to log into an existing account
- option to create a new account
- displaying information about new notifications
- redirecting to the user profile
- redirecting to travel guides



### 3.1.2 Main Page

The main page is the entry point of the application. It contains functionalities such as:

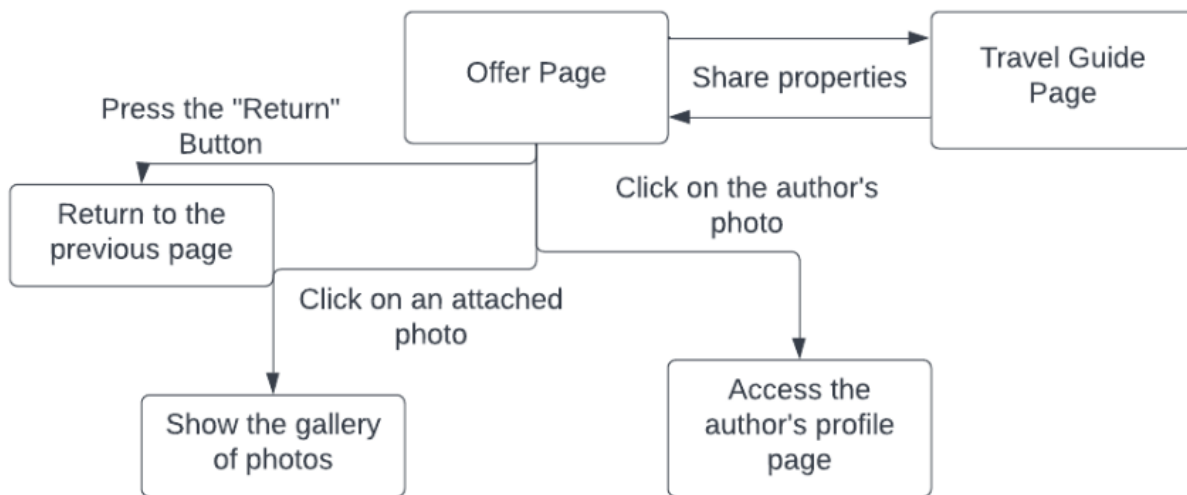
- displaying trending offers
- displaying regular offers below the trending ones
  - redirecting to a more detailed offer page upon clicking on the offer title or photo
- a search bar with filters for the location and offer type
- redirecting to a logged-in user's profile page upon clicking on their photo



### 3.1.3 Offer Page

The offer page contains a more detailed view of an offer. It is accessed through the main page, upon clicking on an available offer. It contains functionalities such as:

- extended description of the offer
- option to turn on notifications for similar offers
- option to leave a vote on the offer
- redirecting to the offer author's profile page upon clicking on their photo
- option to message the offer author by clicking on a message icon
- option to participate in discussion regarding the offer



### 3.1.4 User Profile

The user profile displays general information regarding a specific user account. The user can access their own profile he's currently logged in or other users' accounts. It contains information such as:

- All offers added by the user
- All travel guides added by the user
- All comments added by the user

Additionally, concerning their own account, the user can:

- browse saved offers
- redirecting to a chat page upon clicking on a message icon

### 3.1.5 Messages page

The message page displays all previous conversations with other registered users. Additionally, it gives the user an option to add certain users to their own blacklist (thus blocking communication with them).

### 3.1.6 List of Travel Guides

Travel guides allow users to share blog-like guides about ways of saving money during holidays. The general list of guides allows the user to:

- Search for a specific guide (filtering by its title name or country)

### 3.1.7 Detailed Travel Guide description

The detailed view of a travel guide contains an extended description of it, alongside functionalities such as:



- Rating the guide
- Adding a comment to the guide
- Visiting the guide author's profile
- (Optional) a gallery of photos attached to the guide

## 3.2 Hardware Interfaces

Due to being a web application, the product is designed to be available to use on both computer and mobile devices' browsers with JavaScript enabled. The internal communication is going to be performed between the website and a PostgreSQL database.

## 3.3 Software Interfaces

The web application can be divided into three sections: frontend, backend and database. Each part use different libraries:

- Vue.js for the frontend
- Django for the backend
- PostgreSQL for the database

## 3.4 Communications Interfaces

The most used communication protocol is going to be HTTPS due to the web nature of this application. It will serve the majority of user and server requests. On top of that, the SMTP protocol may be used to send email notifications to users.

# 4. System Features

## 4.1 User Accounts

### 4.1.1 Description

The website supports an account system for users. They can create their own account through the login page. Owning such an account comes with extra functionalities, such as:

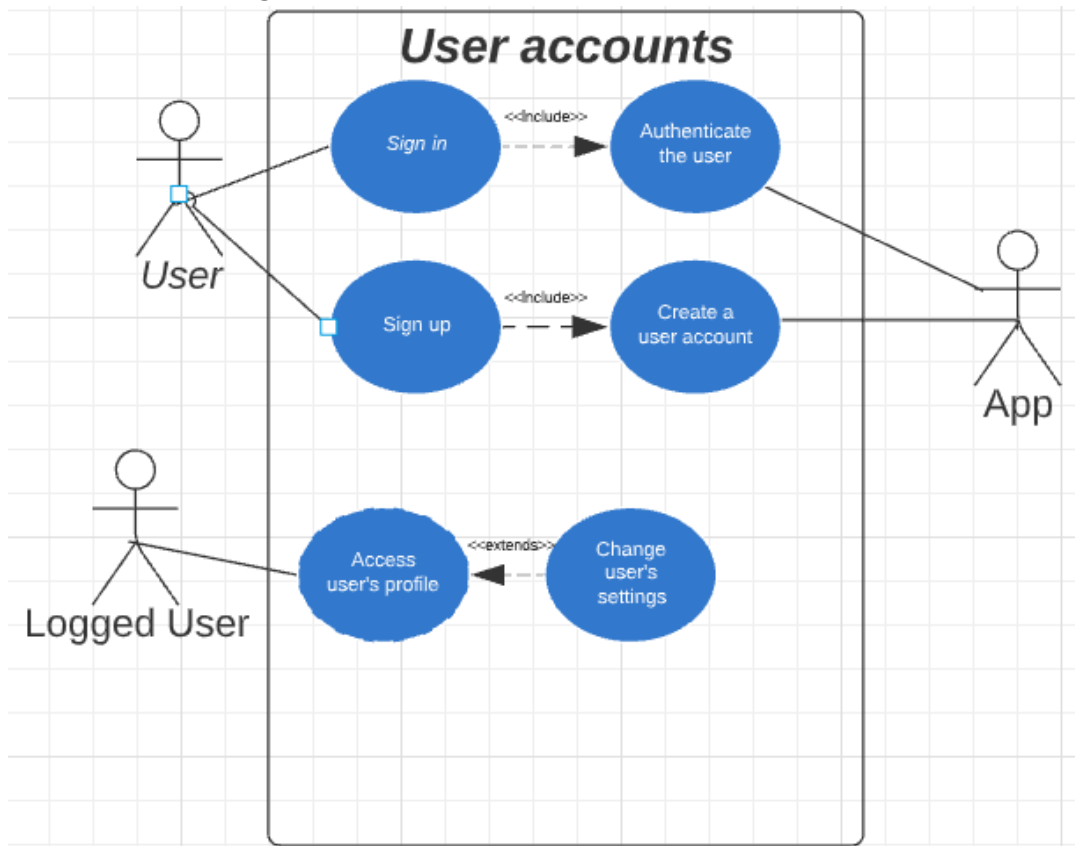
- adding new offers and guides,
- ability to comment on existing offers and guides,
- enabling notifications about future offers
- interacting with other registered users through messages
- changing account properties

**Priority: High**

### 4.1.2 Technical Details

In order to support this feature several data tables have to be incorporated into the database. Next, an authentication system has to be inserted - this is done using Django's integrated authentication feature. On top of that, a logged-in user's view of the site has to be different from a regular visitor's one - it should be extended by features such as the ability to comment and vote on offers.

#### 4.1.3 Use case diagram



## 4.2 Offers

### 4.2.1 Description

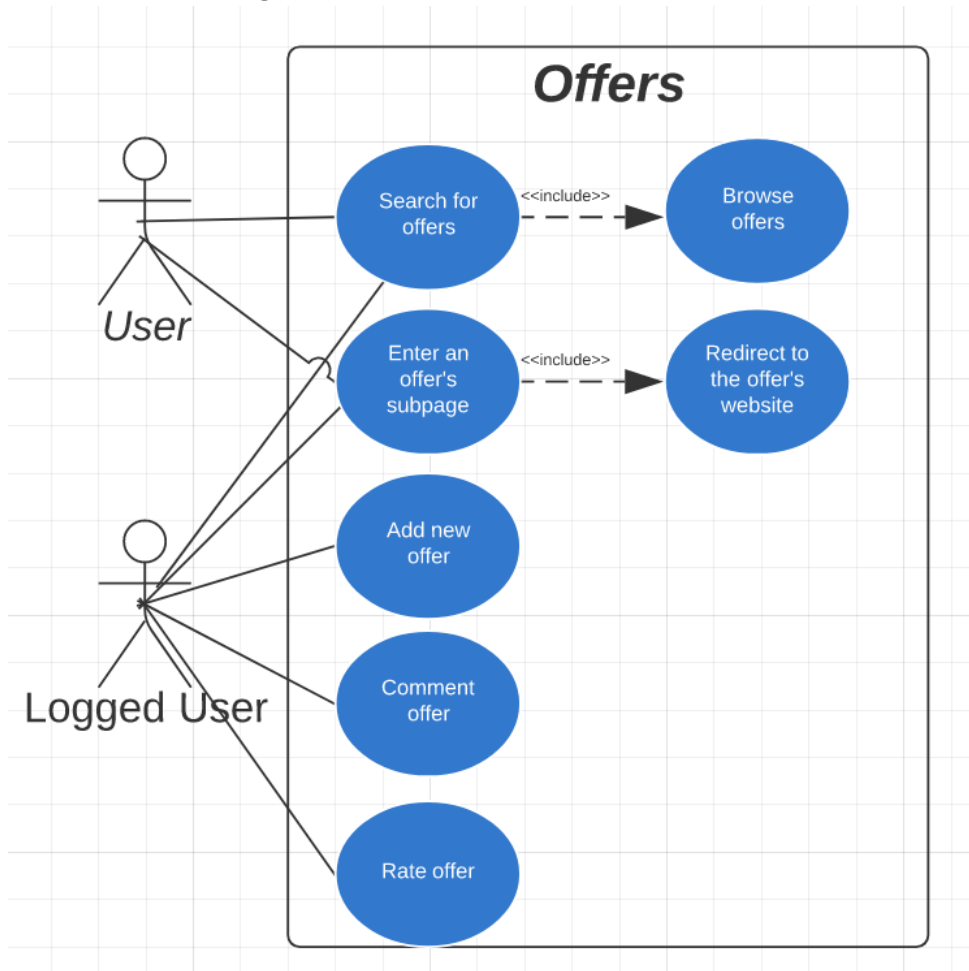
The website's main attraction are the offers posted on it by users. Upon visiting the site, other users can access them. Each offer should consist of a description and a link to the original offer. One of the ways to find them would be through the usage of the search bar (searching by title, type, price, etc.). On top of that, there should be an option to sort a presented list of offers by the aforementioned properties, among others.

**Priority: High**

### 4.2.2 Technical Details

In order to support this feature, several data tables have to be incorporated into the database. Sorting and filtering has to be implemented.

#### 4.2.3 Use case diagram



### 4.3 Travel Guides

#### 4.3.1 Description

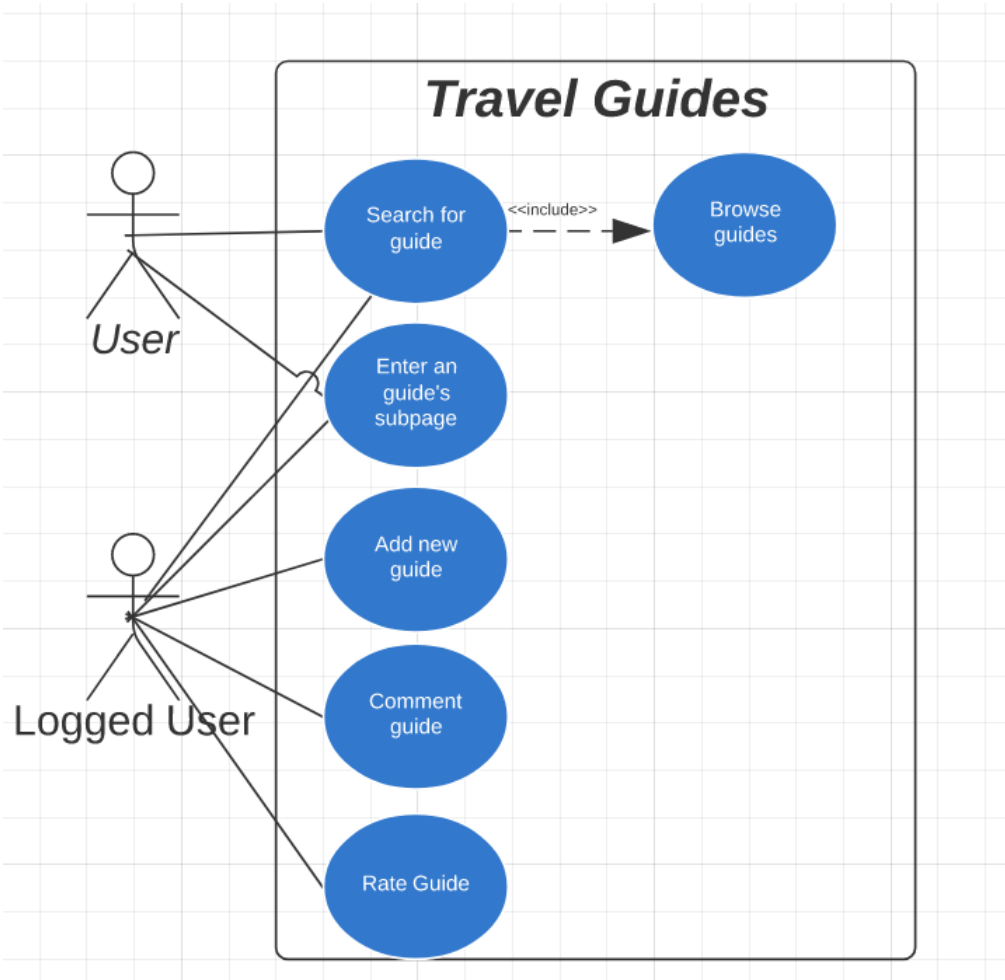
Alongside the regular offers presented on the site, travel guides are available on the website. They can be posted by users in the same manner. Such guides are presented to a visitor in the form of a small blog.

**Priority: Medium**

#### 4.3.2 Technical Details

Technical details specified in 4.2.2 apply also to this section.

#### 4.3.3 Use case diagram



## 4.4 Messaging Board

### 4.4.1 Description

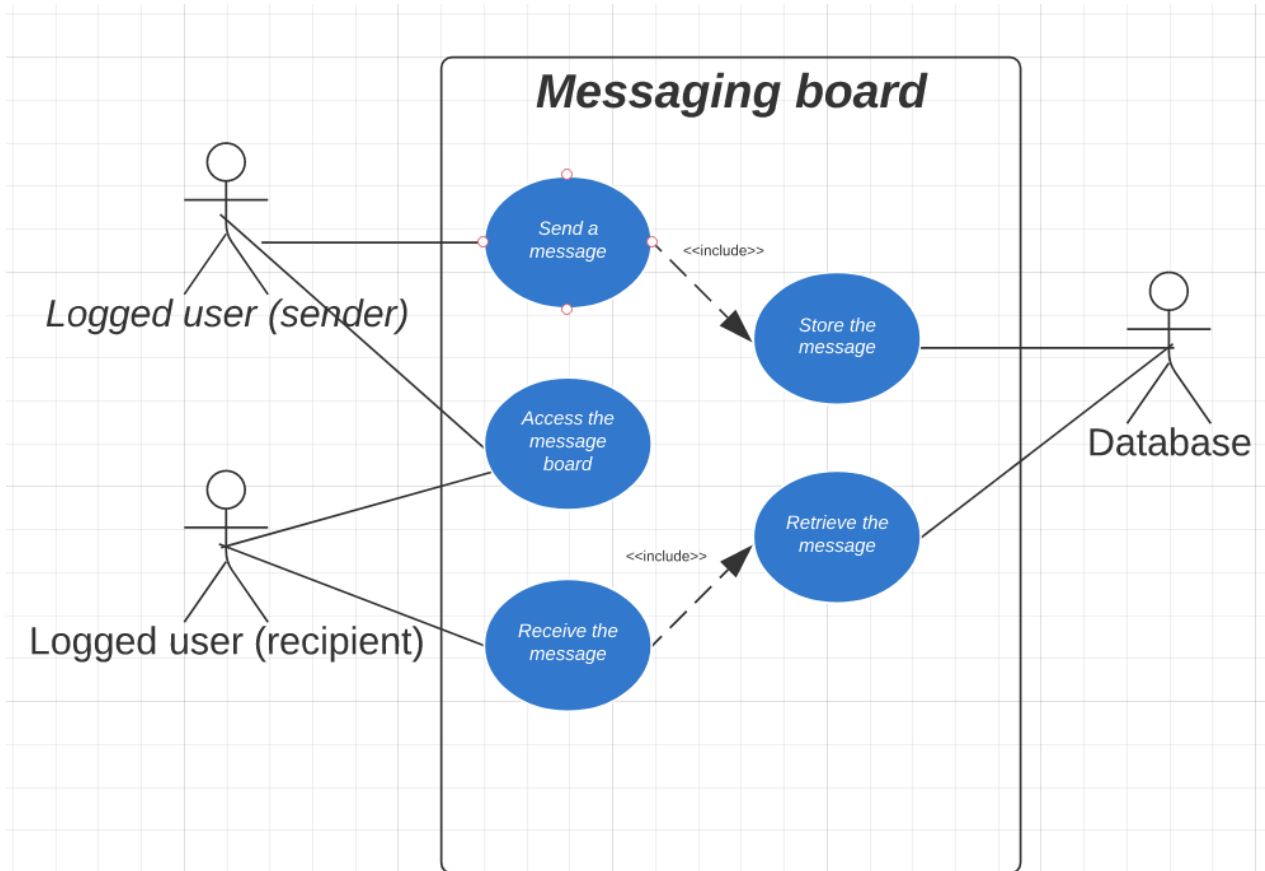
This feature allows registered users to exchange private messages between each other.

**Priority: High**

### 4.4.2 Technical Details

In order to support this feature, additional data tables have to be created to contain the history and other properties of an exchange.

### 4.4.3 Use case diagram



## 5. Other Nonfunctional Requirements

### 5.1 Performance Requirements

Server-side, the application should work efficiently for its users, regardless of the traffic to it. On the client's side, the website should not consume more system memory than modern systems can provide.

### 5.2 Safety Requirements

The website will have social features (comments, guides etc.) therefore users may post inappropriate content. Moderation and filtering solutions must be implemented in order to keep the website safe. Additionally, the website's Terms of Service must be compliant with local regulations.

### **5.3 Security Requirements**

The application will store user's credentials to serve the account system. Therefore, the safety of provided data must be ensured. Users' passwords have to be hashed using modern standards. Additionally, a 2FA system may be introduced to protect users from losing their accounts.

### **5.4 Software Quality Attributes**

Every team member who wishes to update the code within the repository's main branch will need to send a pull request, which will be reviewed by other developers and then accepted by the team leader. Additionally, each pull request will be required to contain unit tests.