|  |
| --- |
| **Chadlist NoBrainNoPain** |

**[Name(s) of student(s), student number, photographs]**

**Supervisor: [Name of supervisor(s)]**

**[Name and logo of educational institution]**

**[Logo of companies included]**

**[Number of characters]**

**[Study program]**

**[Semester]**

**[Date]**

**Table of content**

[Abstract iii](#_Toc18659739)

[1 Introduction 1](#_Toc18659740)

[2 Analysis 2](#_Toc18659741)

[2.1 Requirements 2](#_Toc18659742)

[2.2 Functional Requirements 2](#_Toc18659743)

[2.3 Non-Functional Requirements 3](#_Toc18659744)

[3 Design 4](#_Toc18659745)

[4 Implementation 5](#_Toc18659746)

[5 Test 6](#_Toc18659747)

[5.1 Test Specifications 6](#_Toc18659748)

[6 Results and Discussion 7](#_Toc18659749)

[7 Conclusions 8](#_Toc18659750)

[8 Project future 9](#_Toc18659751)

[9 Sources of information 10](#_Toc18659752)

[10 Appendices 1](#_Toc18659753)

**List of figures and tables**

Optional

# Abstract

*An abstract is a shortened version of the report and should contain all information necessary for the reader to determine:*

1. *What are the aim and objectives of the project*
2. *What are the main technical choices*
3. *What are the results*

*Frequently, readers of a report will only read the abstract, choosing to read at length those reports that are most interesting to them. For this reason, and because abstracts are frequently made available to engineers by various computer abstracting services, this section should be written carefully and succinctly to have the greatest impact in as few words as possible.*

*Although it appears as the first section in a paper, most report writers write the abstract section last.*

Cf. (Dawson 2009, p.195).

# Introduction

Our customer is a Non-profit organization that focuses on waste management. It has offices all around Northern Europe and Scandinavia. The Headquarters of the organization is in Copenhagen, Denmark.

We were contacted by the company with the offer to develop a system which would help its users to sell and exchange their belongings which they no longer need. Other users can then see their offer and if it interests them, they can contact the seller and buy it off of them.

The purpose of the introduction is to provide background information and set the scene for your project. Within which business or organization are you doing the project? Who are the stakeholders and who is the customer?

The background information is adapted from your project description where you have already described the problem domain. Describe the current situation and existing context. Your statements must be supported by references to reliable and relevant sources.

This should lead to why this project is relevant and outline your aim and objectives. Which technical problems and challenges will be presented in this report, again taken from your project description. System illustrations and rich pictures are welcome here.

State delimitations relevant for your project in the introduction. Delimitations include what the project will not cover in relation to your project description, i.e. what could have been expected in your project. Remember that you can only make delimitations to aspects mentioned in the project description and you must argue well for your delimitations.

The last sentences of the introduction should be an overview of the sections to follow. This will be a good transition to the next sections.

Remember: You must ensure a clear connection between sections in the project report, from Project Description, Analysis, Design, Implementation to Test. This means that everything that is implemented can be found in design, everything that is designed is based on the analysis, and anything that is found in analysis has a clear link to requirements, etc.

# Analysis

The purpose of ChadList is to reduce wasting of reusable items throughout the world by giving everyone access to a platform, where they can sell and buy those items quickly and without any fees.

The system will have two types of users: regular users and administrators. Regular users will be able to sell and buy items and report to Administrator if an item that is being sold is forbidden from selling on the platform or if there was a problem with the seller or buyer during a transaction. The Administrator can then decide to ban the post and user being reported.

## Functional Requirements

## 

### Critical Priority

1. As a user I want to register, by specifying my first name, last name, phone number, email, date of birth and gender and log in specifying the email and password I used for the registration, in order to access my account.
2. As a user I want to be able to post an item for sale specifying its name, description of the item, price, currency and status of the item, in order to sell an item I don’t need anymore.
3. As a user I want to message the user over an item they are selling in order to negotiate and ask details about the item.
4. As a user I want to be able to search for a specific item specifying the part of name, part of the description and min and max price in order to find an item that I am looking for.
5. As a user I want to be able to view details of a specific item in order to decide about the purchase.
6. As a user I want to be able to see all my notifications in order to keep track of messages and item updates.
7. As a user I want to be able to see all my active and previous chats with the sellers, to be able to verify information between what the seller said and what the item is about.
8. As a user I want to be able to report an item in case the item does not respect the rule about forbidden items.

### High Priority

1. As a user I want to be able to edit my profile in order to modify any information that became outdated.
2. As a user I want to be able to delete or edit an item, in case I did any mistake during posting or I decided not to sell the item, or I sold the item.

### Medium Priority

1. As a user I want to be able to have a list of items that might interest me in order for me to find better items.
2. As a user I want to be able to add an item to my wish list in order to save items that I am not sure yet that I want.
3. As a user I want to get a notification whenever there is an update on an item in my wish list in order to be up to date about my saved items.
4. As a user I want to be able to see the sellers rating in order to see if he is a trustable person.
5. As a user I want to be able to comment/rate a seller to help other users trust the seller or not.
6. As a user I want to be able to see a seller’s profile in order to see any details that interest me such as address, rating or the amount of sold products.
7. As a user I want to be able to delete items from my wish list, in order to remove items that do not interest me anymore.
8. As a user I want to be able to change the password so that I don’t get locked out of my account.
9. As a user I want to be able to delete my account in case I do not want to use the service anymore.

### Low Priority

1. As a user I want to be able to sort the items by categories, in order to find what interests me.
2. As a user I want to be able to track how many people added a specific item to their wish list for me to take a faster decision about the item.
3. As a user I want to be able to see related items when I look at a specific post, in order to have a better chance of getting a better deal.

## Non-Functional Requirements

1. The system must be able to work on Mozilla Firefox v106.0 and previous, Google Chrome v107.0 and previous and Microsoft Edge 107.0 and previous

## Actor Description

### User can sell items on the platform as well as search for items they would like to buy and contact other users in order to buy them. Users can also report items and users if the items they are selling are banned on the platform or if the transaction did not go as agreed.

**Administrator** can look at the reports from users and decide depending on the reports to either ban the post or the post and the user from the platform.

## Use Case Diagram

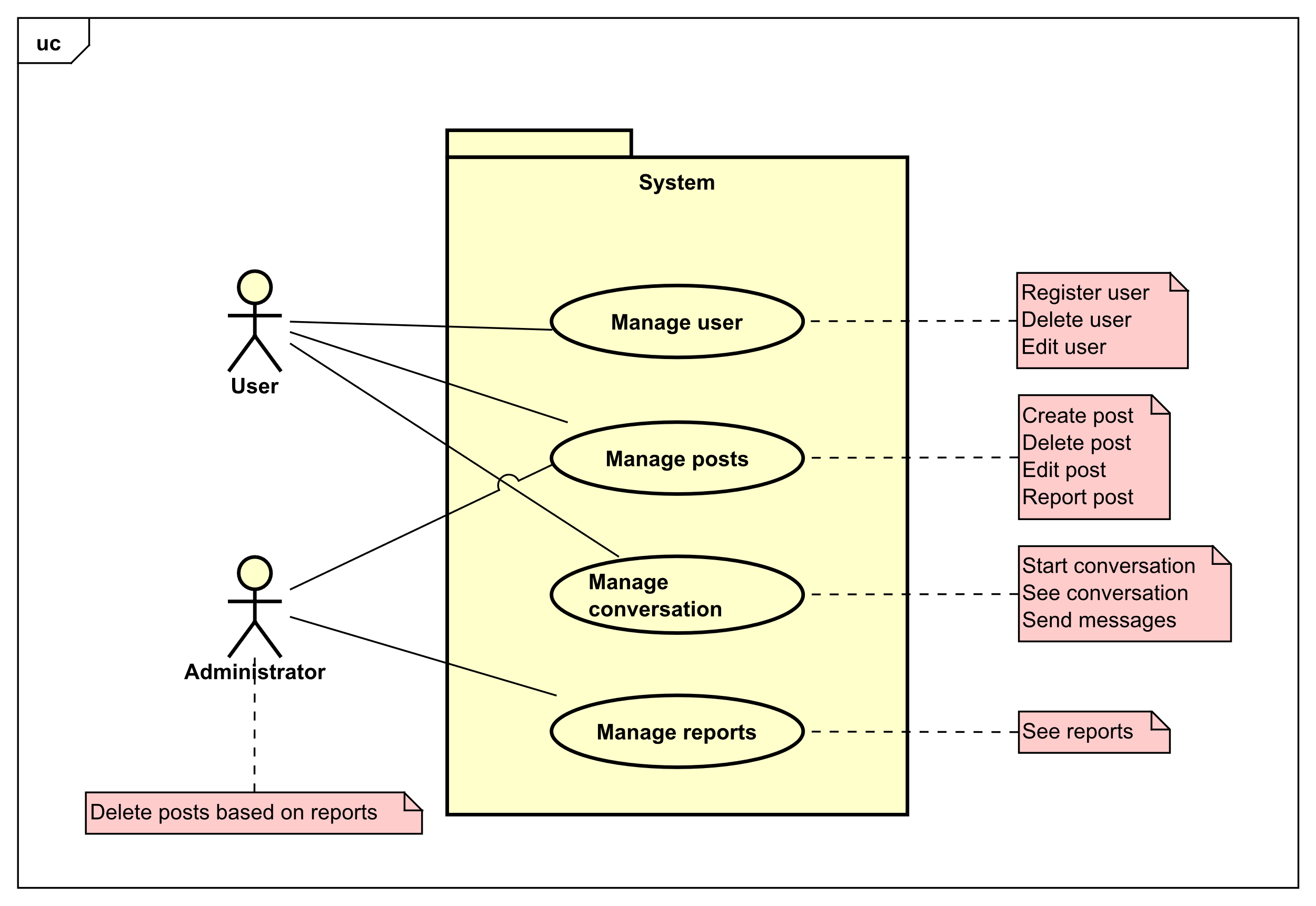


Figure 1 – Use Case Diagram

## Use Case – Brief Description (Manage Posts)

**Manage posts** – A user will need an account to access most of the functions. The user presses the new post button and starts introducing all the necessary information required to create a post. Another user logs on to the website, sees the post and starts talking to the seller. The user ends up buying the product, and goes to the seller’s profile, writes a nice comment, and gives the seller a positive rating. Now the seller goes to his post marks it as SOLD or deletes the post entirely so that he won’t be contacted again for a product he sold.

## Use Case – Fully Dressed Description (Manage Posts)

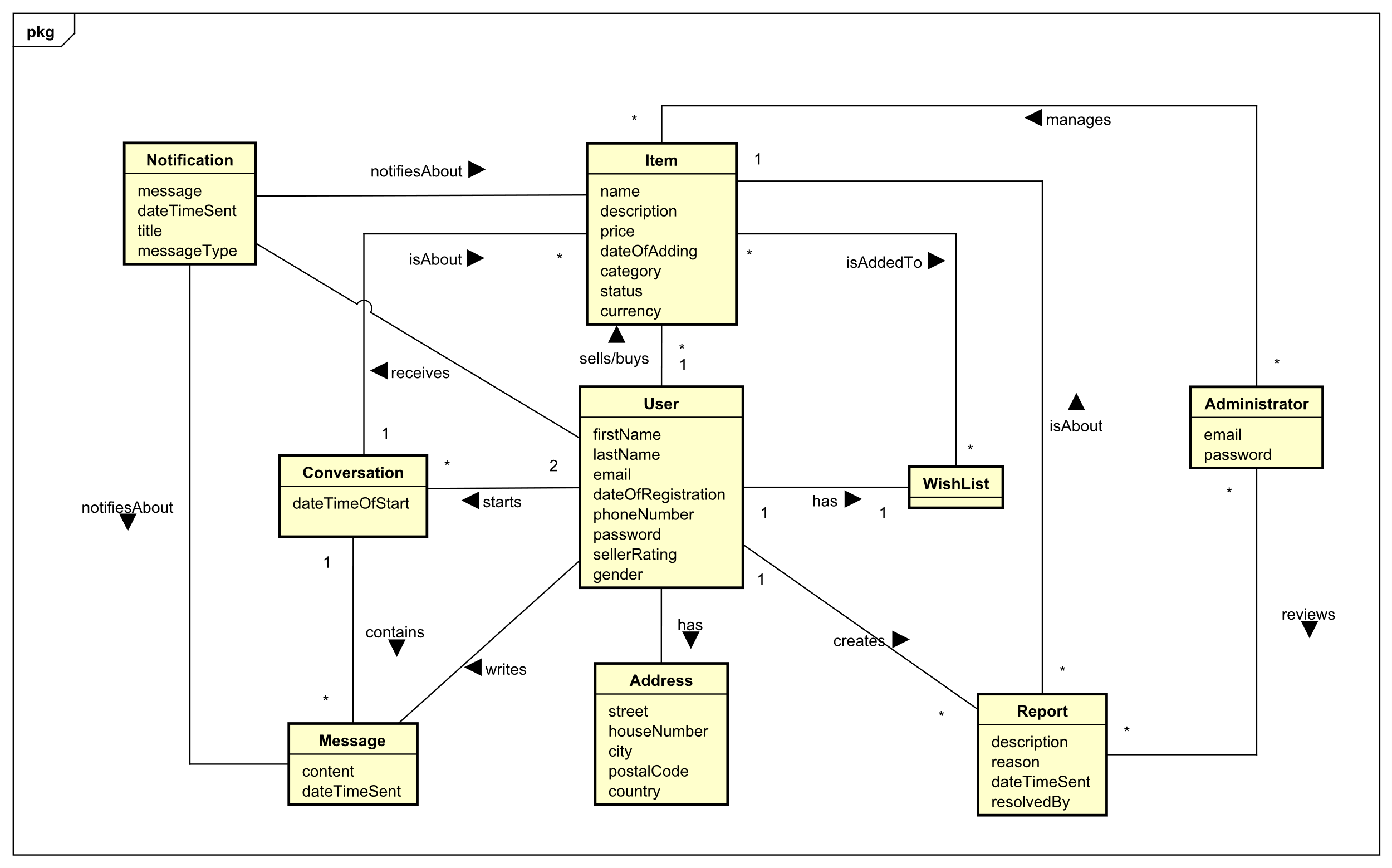
|  |  |
| --- | --- |
| Use Case | Manage posts |
| Summary | The user creates new posts with items, updates and deletes them |
| Actor | User |
| Precondition | The user must be logged in. For delete and update, user has to have created at least one post |
| Postcondition | The post will be created, updated or deleted |
| Base scenario | Create a post   1. User logins to his account. 2. User presses the marketplace button. 3. A window will be shown with all items that are for sale and their description. 4. If the user wants to sell an item, he needs to press the add post button. 5. A window will be shown, where the user can add all the necessary information about the item or edit it.   Delete Posts   1. User deletes the post by pressing the edit button. 2. A window will appear will all the information about post, and a delete button below.   The user presses the delete button and the post will be deleted |
| Alternative scenario |  |
| Exception | If one of the fields or more are empty, an error will be shown. |

## Activity Diagram (Manage Posts)

## System Sequence Diagram

## Domain Model

A **User**, after registering and creating an account and providing their **Address**, can decide to sell their **Item** (things they own and do not need anymore). After that, another **User**s can save the item to their **Wishlist.** If they wish to buy that item, they can start a **Conversation** with them to negotiate the price and delivery by sending a **Message**. After a message is sent, the seller will get a **Notification**. If an **Item** which is not legal to sell, is posted on the platform, another users can **Report** them and the **Administrator** can then ban the **User** and delete the **Item**.



# Design

The team has decided to split the system into 3 tiers. The decision to choose 3 tier architecture over regular client server application had many different reasons. One of the reasons was that if the client decides to change the database provider to another in the future, the application tier will not have to be changed. Another reason was easier testability of each tiers, because they are separated, we can test business layer, without relying on presentation layer working.

## Presentation Tier

The presentation tier will be mainly responsible for interaction with the users. The technology used for the presentation tier is Blazor WASM and C#. The tier will fetch data from and send data to the business tier using REST Web API and HTTP protocol.

## Business Tier

The business tier will be responsible for handling user requests and responses. The technology used for this tier is C# and .NET 6.0. The tier will communicate with the database tier using gRPC.

## Database Tier

The database tier will be responsible for storing user data. The technology used for this tier is Java 17, JPA/Hibernate and PostgreSQL.

Elements that may be relevant in this section:

* Architecture: Find architecture patterns here (Leszek Maciaszek 2004, chap.9).
* Technologies: Describe technologies used, also alternative technologies. Argue for choice of technology according to the project aim.
* Design Patterns: Describe which design patterns (GoF (Gamma et al. 2002) etc.) you are using and why.
* Class Diagrams
* Interaction Diagrams
* UI design choices
* Data models, persistence, etc.

You must explain all diagrams in the report. These diagrams including descriptions are the blueprints for the implementation.

Hint: One way to figure out which objects/classes are needed in the design is to apply the General Responsibility Assignment Software Patterns/principles (GRASP) (Larman 2004, chap.17).

Hint: Consider how to design your system to make it testable.

# Implementation

The purpose of the implementation section is to explain interesting code snippets. An idea is to explain the complete path through your system from UI to database etc.

Remember that your implementation must be consistent with your design (Larman 2004, chap.20).

Which standard libraries are used? How are design patterns implemented, etc.

Hint: Implement your code in a testable manner.

# Test

The purpose of the test section is to document the result of your testing; to verify if the content of the requirements section has been fulfilled. How is the system tested, which strategy has been used; e.g. White Box (Unit Test), Black Box, etc.

## Test Specifications

For functional requirements, test specifications must be listed. These test specifications can be described as soon as the functional requirements have been completed (Use Cases including descriptions).

IEEE can be used as a template for test specification (IEEE Computer Society 2008). VIA Library can give you access to this standard.

# Results and Discussion

The purpose of the results and discussion section is to present the outcome and achieved results of the project.

# Conclusions

The purpose of the conclusion section is to compile the results from each section in the report. What is the conclusion? Did the project fulfil the requirements? Etc.

You can only comment on report contents, no new topics or content can be introduced in this section.

# Project future

Reflect on your project from a technical viewpoint and describe what you would change if you could.

Suggest how the project could be improved or made ready for production. Discuss scalability, suggest possible spin offs, what is needed, missing, etc.?

# Sources of information

**Note: Use the standard reference method: Harvard Anglia. A very good reference tool is Mendeley** (Mendeley.com 2016), **ask VIA Library if you need help.**

Banger, D., 2014. A Basic Non-Functional Requirements Checklist « Thoughts from the Systems front line.... Available at: https://dalbanger.wordpress.com/2014/01/08/a-basic-non-functional-requirements-checklist/ [Accessed January 31, 2017].

Business Analyst Learnings, 2013. MoSCoW : Requirements Prioritization Technique — Business Analyst Learnings. , pp.1–5. Available at: https://businessanalystlearnings.com/ba-techniques/2013/3/5/moscow-technique-requirements-prioritization [Accessed January 31, 2017].

Dawson, C.W., 2009. *Projects in Computing and Information Systems*, Available at: http://www.sentimentaltoday.net/National\_Academy\_Press/0321263553.Addison.Wesley.Publishing.Company.Projects.in.Computing.and.Information.Systems.A.Students.Guide.Jun.2005.pdf.

Gamma, E. et al., 2002. *Design Patterns – Elements of Reusable Object-Oriented Software*, Available at: http://books.google.com/books?id=JPOaP7cyk6wC&pg=PA78&dq=intitle:Design+Patterns+Elements+of+Reusable+Object+Oriented+Software&hl=&cd=3&source=gbs\_api%5Cnpapers2://publication/uuid/944613AA-7124-44A4-B86F-C7B2123344F3.

IEEE Computer Society, 2008. *IEEE Std 829-2008, IEEE Standard for Software and System Test Documentation*,

Larman, C., 2004. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*,

Mendeley.com, 2016. Homepage | Mendeley. Available at: https://www.mendeley.com/ [Accessed February 2, 2017].

YourCoach, S.M.A.R.T. goal setting | SMART | Coaching tools | YourCoach Gent. Available at: http://www.yourcoach.be/en/coaching-tools/smart-goal-setting.php [Accessed August 19, 2017].

# Appendices

The purpose of your appendices is to provide extra information to the expert reader. List the appendices in order of mention.

Examples of appendices

* Project Description
* User Guide
* Source code – source documentation
* Diagrams
* Data sheets
* Etc.

**Appendix A Project Description**

Insert the original Project Description here