Requirements and Analysis Document for Hyro

TDA367, Chalmers tekniska högskola

Eimer Ahlstedt, Emil Lindblad, Sebastian Kvaldén, Timothy Nilsson, Erik Larsson

10 oktober 2021

Table of contents

1 Introduction

The purpose of this application is to enable people to participate in sharing economy, by allowing users to both rent and rent out different objects. This is achieved by creating a virtual marketplace, where users can create an account, browse listings and also add their own items for rent.

People who can benefit from this application can be divided into two categories: the ones who rent items and the ones who supply items for rent.

Imagine that you have just bought some new shelves and realizing that you do not have a hammer drill for mounting them to your concrete wall. Instead of spending a lot of on a new one or leasing one with improper and complicated agreements, you can just check if anyone nearby has a hammer drill for rent on Hyro.

In the other case, you might have a tool or item that you only use a few times a year, like a pressure washer or a trailer. Instead of letting these items collect dust in your garage, you can rent them out on Hyro. Making them available for more people, decreasing their environmental footprint.

1.1 Definitions, acronyms and abbreviations

- **Det funkar på min branch"** - A way of saying that a bug has been fixed only locally, not avaliable to the rest of the developers.
- **Jira** - A tool used for Agile software management.
- **JSON** - JavaScript Object Notation. A human readable data format usually used for data interchange with webservers. We use it in our application for storing and persisting data between sessions.
- **JavaFX** - A Java framework used for creating graphical users interfaces.
- **GitHub** - A tool used for storing code repositories.

2 Requirements

2.1 User Stories

The following user stories is implemented or will be implemented in the future. The stories in this chapter are named with a number to be trackable across documents and platforms such as Jira.

2.1.1 OOP-3

Status: DONE
Story Name:OOP-3

Description
As a user I want to be able to run the program, so that I can start using it.

Confirmation

Functional:
Program compiles and basic window opens.

Non-Functional:
Project created and accessible through GitHub.
Configure maven and JavaFX.

2.1.2 OOP-27

Status: DONE
Story Name:OOP-27

Description
As a User I want to be able to see the same information when I restart the application so that I can continue renting later.

Confirmation
Functional:
A User should be able to see the same listings after restarting the application.

Non-Functional:
JSON-files created and tested for User and Listing-classes.
Fully functional read/write-methods to accsess the JSON-files

2.1.3 OOP-18

Status: DONE
Story Name:OOP-18

Description
As a User I want to be able to Authenticate, so that other users know who add products for rent and who rents their products.

Confirmation
Functional:
Functionality to create new user is available.
Functionality for existing users to authenticate and access their information.

Non-Functional:
Only logged in users can access their information.

2.1.4 OOP-41

Status: DONE
Story Name:OOP-41

Description
As a User, I want to be able to enter my login credentials, so that I can be logged in.

Confirmation
Functional:
Accessible Log-In page for users to enter their credentials or create account.

Non-Functional:

2.1.5 OOP-22

Status: DONE
Story Name:OOP-22

Description
As a User I want to be able to navigate through the different pages, so that I can explore the application.

Confirmation
Functional:
Access different views as a user. Such as Log-In and Settings screen.

Non-Functional:

2.1.6 OOP-47

Status: TESTING
Story Name:OOP-47

Description
As a User I want to be able to change my personal information (address, password etc).

Confirmation
Functional:
View that shows and enables changes to the logged in users information.

Non-Functional:
Check for correct syntax in user input and logged in user.

2.1.7 OOP-4

Status: IN-PROGRESS
Story Name:OOP-4

Description
As a User I want to be able to view a product, so that I can determine if I want to rent it.

Confirmation
Functional:
A user should be able to view a pre-defined product in the application.

2.1.8 OOP-19

Status: IN-PROGRESS
Story Name:OOP-19

Description
As a User I want to be able to add a product for rent. so that I can show other users that I have products for rent.

Confirmation
Functional:
Logged-in user can create a product and let other users rent it.

2.1.9 OOP-19

Status: IN-PROGRESS
Story Name:OOP-19

Description
As a User I want to be able to add a product for rent. so that I can show other users that I have products for rent.

Confirmation
Functional:
Logged-in user can create a product and let other users rent it.

2.1.10 OOP-50

Status: IN-PROGRESS
Story Name:OOP-50

Description
As a User, I want to be able to create an account so that I can keep track of my data and log in.

Confirmation
Functional:
Functional "Create Account-button on log-in page for new users to create account.

2.1.11 OOP-20

Status: IN-PROGRESS
Story Name:OOP-20

Description
As a User I want to be able to request to rent a product, so that can I show the owner that I am interested.

Confirmation
Functional:
A logged-in user can request to rent a product that another user has advertised as available.

2.1.12 OOP-45

Status: IN-PROGRESS
Story Name:OOP-45

Description
As a User, I want to be able to see a list of all products that are available for renting, so that I can browse and decide what I want to rent.

Confirmation
Functional:
A main page where a user can browse products.

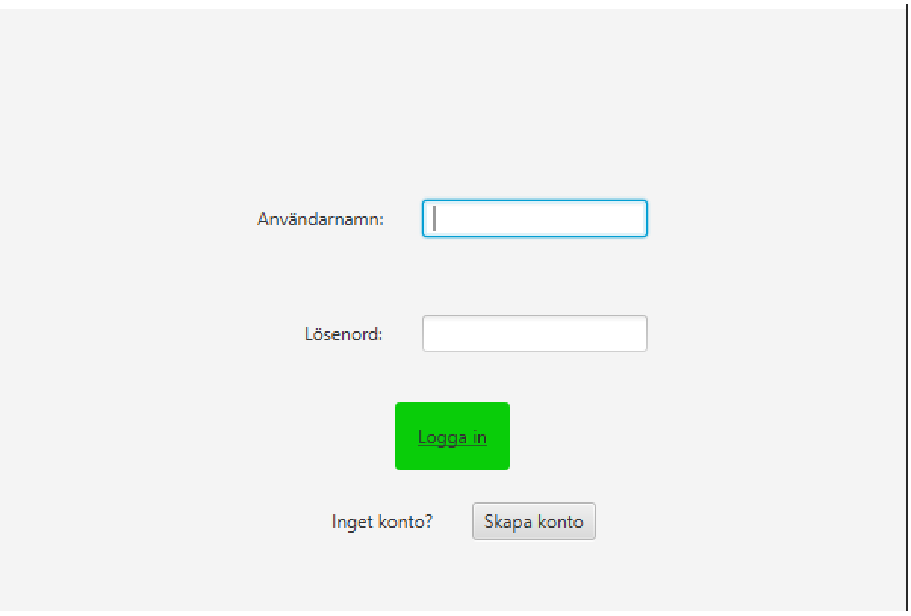
2.2 Definition of Done

New features are worked on in a new branch

Code is commented
90% Test Coverage
All checks pass and merge to master is successful

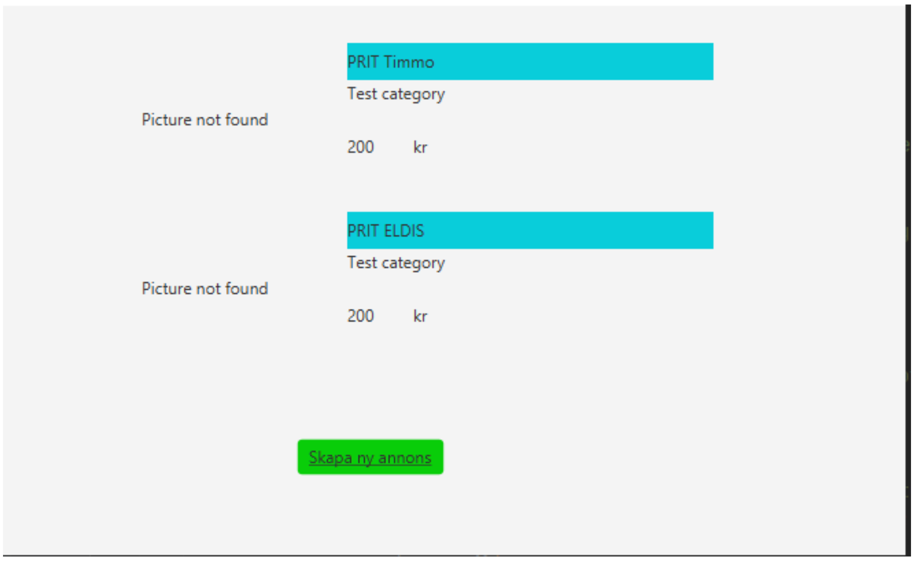
2.3 User interface

The User Interface is created using JavaFX and consists of multiple fxml files. Each view that the user can navigate to is a Sceneåhich is where all the data gets passed through, both input and the output. Each Scene has its own controller that is in charge of changing the scene that user wants to see. So if for example a users presses Log in button and all log in checks passes, the controller will request the SceneHandler to then switch the scene the browse view. The Scene Handler then loads the respective fxml file and then displays it in the application.



Figur 1: Login in view where users can log in to an existing account or create a new one

dab (Låt stå)



Picture not found

PRIT Timmo

Test category

200 kr

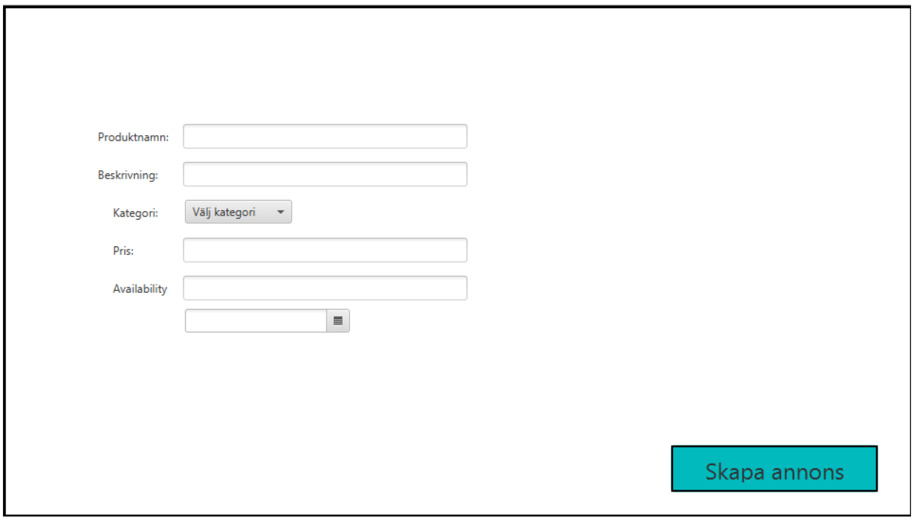
Picture not found

PRIT ELDIS

Test category

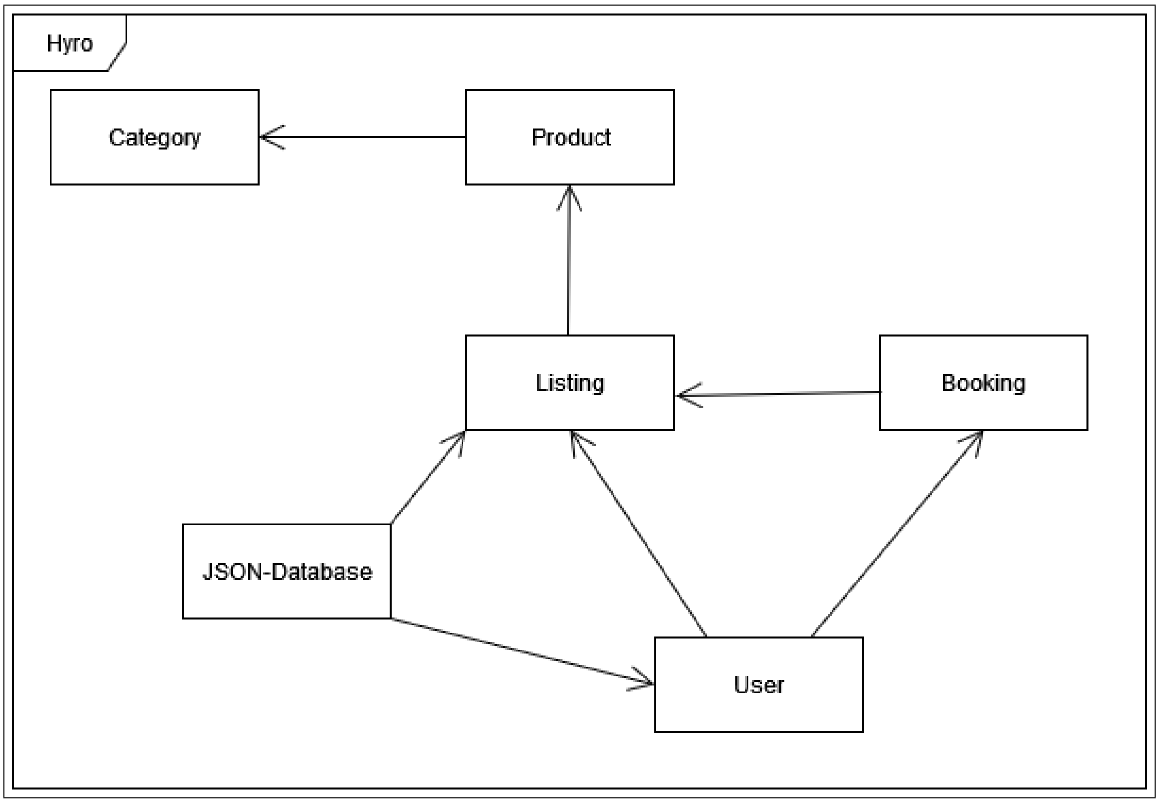
200 kr

Figur 2: Browse view where logged in users can see available listings for rent and also create a new listing



Figur 3: Create listing view where users can post new listings from one of their own belongings

3 Domain Model



Figur 4: The current Domain Model of Hyro

3.1 Class responsibilities

Above is the current state of the Hyro Domain Model. Functionality to show GUI has been left out.

The User class is for keeping track of an individual users contact information, currently listed products and active booking requests.

Listing exists to contain information on what product which user has listed. User and Listing are both tracked by the JSON database.

The purpose of Booking is to be the middle ground between a customer wanting to rent a product and the owner of the product. It keeps track of the state of the booking and controls the payment.

Product and Category are simple. A product is the item available for rent, and the Product class contains information on the item, such as its category. The Category is used to filter searches, for example if a user wants to rent any grill, it could browse a hypothetical grill-category.

References

NA