

Filip Wygoda 212829 Wednesday, 13:15, odd weeks	BookHandler
	Software Requirements Specification

1. Introduction

The following documentation introduces the concept for web application serving as a platform for sharing and reading books and graphical novels that can be described as online library. This application should be especially useful for book enthusiasts that are looking for rare prints, not available in internet bookstores and on general auction sites and people interested in expanding their personal collections.

1.1 Purpose

The reason for creating this specification is to introduce the vision of BookHandler web application and provide description for technical aspects of implemented web service. This documentation is intended for a team of software engineers, developers and testers that will implement and test the application based on a requirements and constraints described in the document. Any decisions regarding development of the service should be based on the information provided in this specification and the development process should follow guidelines provided in further sections.

1.2 Scope

This software will consist of database and server application localized on external server and end user client application intended for web browsers. The database is necessary for storing the data generated by users while using the service. that will provide graphical user interface. The role of server application is to handle client requests and apply business logic to the information provided by those requests. Finally, the browser application implemented as a single-page application will provide clear and definite graphical user interface for easier interaction with the system. Such partition of software comes with important benefits, as it allows for development of multiple end user client applications intended for different platforms that would communicate with the one server application using the same API.

1.3 Definitions, acronyms and abbreviations

Term	Explanation
Title/Publication	A book that is available for reading in the system
Favorites list	Personal list of titles that the user has particular interest in
User account	Location with personal user data and history of user's transactions
User rating	Option for users on both ends of transaction to assess the quality of transaction after it's completion
ORM	Object relational mapping – technique for converting data between different type systems
API	Application programming interface – computing interface for communication between different software/ hardware parts of system

1.4 References

^[1] IEEE Recommended Practice for Software Requirements Specifications

<http://www.cse.msu.edu/~cse870/IEEEExplore-SRS-template.pdf>

^[2] Deployment Diagram: UML Tutorial with EXAMPLE

<https://www.guru99.com/deployment-diagram-uml-example.html>

^[3] Diagram klas - Class diagram

<http://zasoby.open.agh.edu.pl/~09sbfraczek/diagram-klas%2C1%2C11.html>

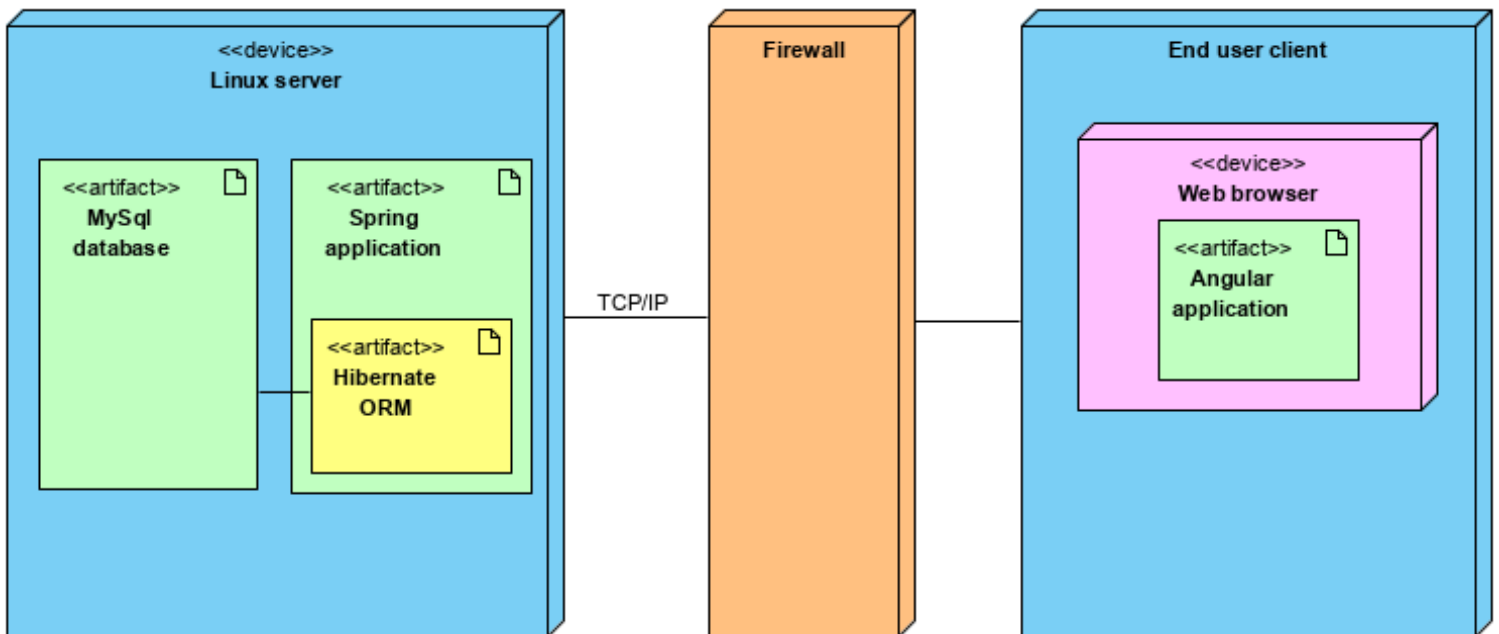
1.5 Overview

The following section of the document will bring general overview of the system without delving into detailed specification. Product perspective, including diagram of major system components and mockups of user interface, will be described and subsection about product functions will provide necessary insight into the system. Moreover user perspectives will be brought to attention for better envisionment of the end users role end functions.

The third section will provide more specific description of the service's operations that will consist of functional and non-functional requirements as well as business. Additionally, this part will contain a class diagram that describes the structure of the system and shows relationships between system's objects in object-oriented modelling.

2.1 Product perspective

This project is a standalone service, independent of any external systems. It consists of two applications: Spring application that resides on the server side, responsible for executing http requests and React application, provided to the end user, running in the browser. Additionally, part of the system is a MySQL database required to store user and service data. Following deployment diagram presents hardware that is necessary for the system's execution and determines how the software is deployed on the underlying hardware.

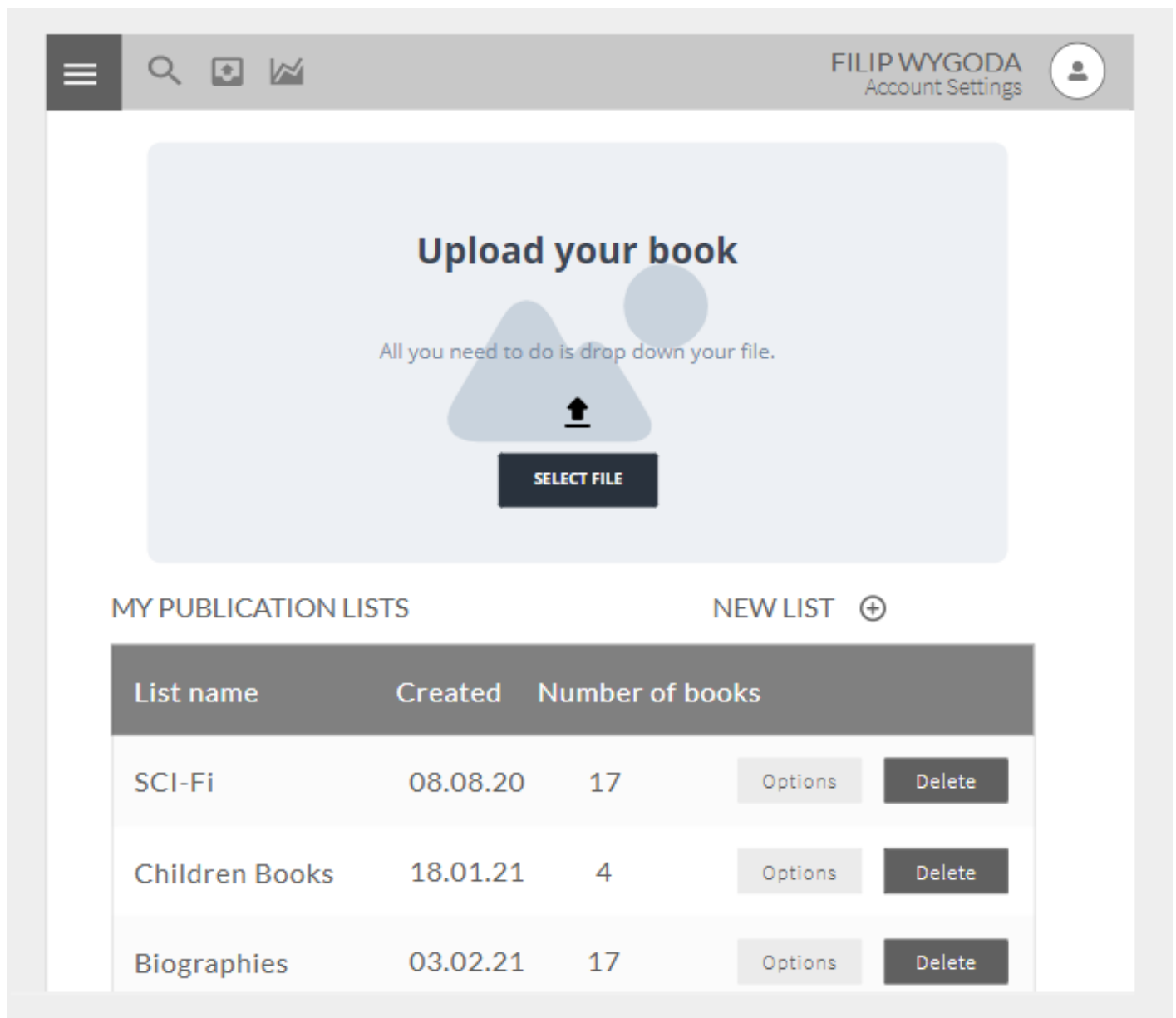


Picture 2.1. Deployment diagram of the service.

2.1.1 User interface



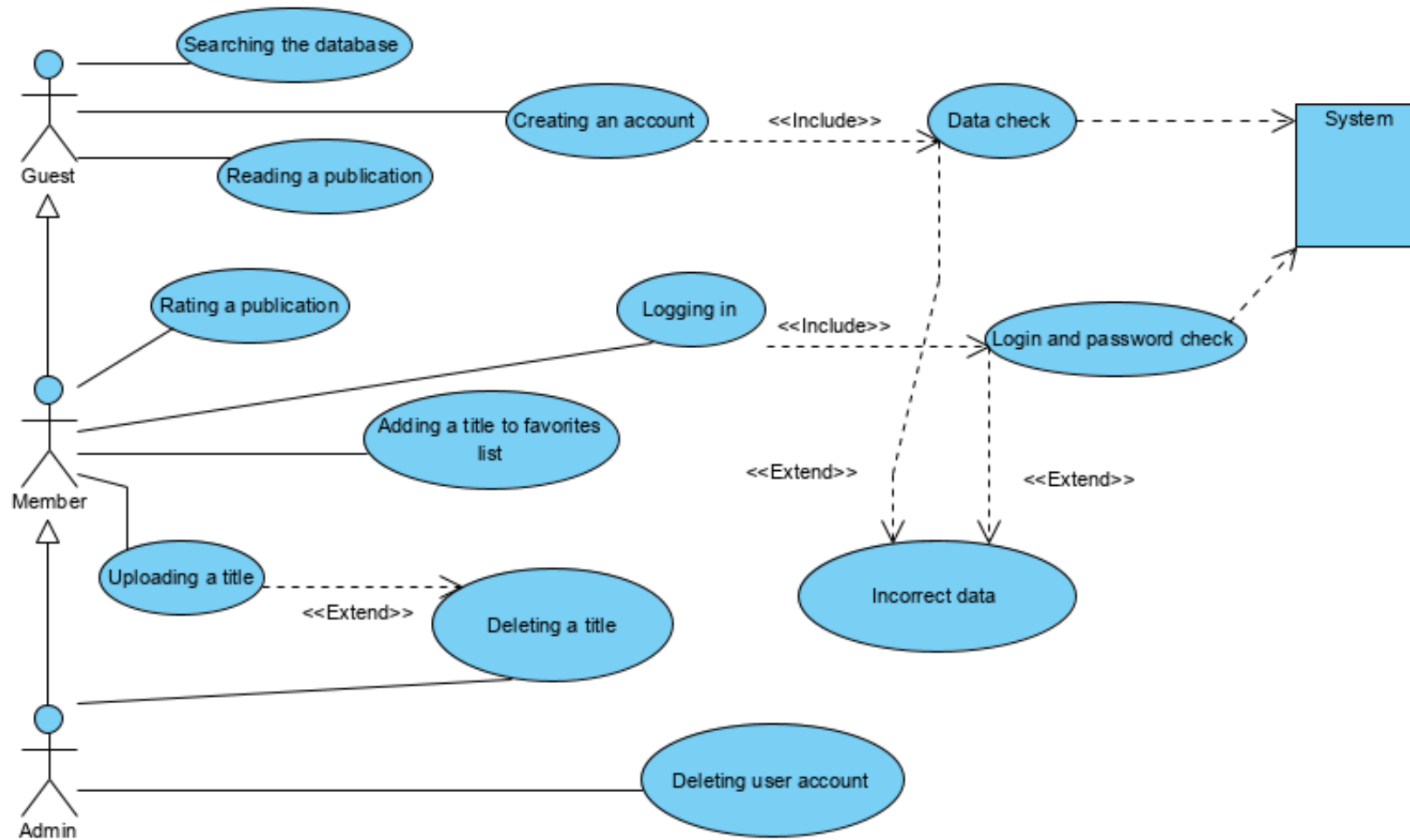
Picture 2.2. Mockup interface of a publication page view.



Picture 2.3. Mockup interface of a member account view.

2.2 Product functions

Product functions will be best represented with a following use case diagram:



Picture 2.2. Use case diagram.

2.3 User perspectives

Role	Description	Functions
Guest	A person that is not logged in or is not a member of a service	<ul style="list-style-type: none">- an option to create account or log in- can search the database of available publications- is able to read publications
Member	A person with a valid account in the service	<ul style="list-style-type: none">- has every privilege of a guest- might create a list of his favorite titles- is able to rate publications
Admin	A person holding an administrative rights in the service	<ul style="list-style-type: none">- has every privilege of a member- is able to delete an account of a user that is breaking the rules of the service- is able to delete publication that is non-complying with rules of the service

3. Specific requirements

3.1 Functional requirements

Requirement	Priority	Release
Every user must be able to enter the service	must	1.0
Every guest must be able to create an account	must	1.0
Every member must be able to login in to his/her account	must	1.0
Every user should be able to search the database	should	1.0
Every user must be able to open a publication and read it	must	1.0
Members may be able to upload a new publication	could	1.0

Every user must be able to search the database	must	2.0
Members may be able to rate publications	could	2.0
Members may be able to add titles to their favorites list	could	2.0
Admins should be able to delete publications	should	2.0
Admins should be able to delete members accounts	should	2.0
Members should be able to upload a new publication	should	2.0
Members should be able to delete their own publications	should	2.0
Members must be able to rate publications	must	3.0
Members must be able to add titles to their favorites list	must	3.0
Admins must be able to delete publications	must	3.0
Admins must be able to delete members accounts	must	3.0
Members must be able to upload a new publication	must	3.0
Members must be able to delete their own publications	must	3.0

3.2 Non-functional requirements

Requirement	Priority	Release
The service must be compatible with latest release of Google Chrome and Mozilla Firefox browsers	must	1.0
Database search query should last less than 5 seconds	should	1.0
The process of submitting logging data should last less than 10 seconds	should	1.0
The service must handle publications uploaded in .pdf format	must	1.0

The service must handle files smaller than 100MB	must	1.0
User interface should be responsive and automatically adjust to browser window	should	2.0
Database search query must last less than 5 seconds	must	3.0
The process of submitting logging data must last less than 10 seconds	must	3.0
User interface must be responsive and automatically adjust to browser window	must	3.0

3.3 Business logic

STRUCTURAL CONSTRAINTS

- The publication is associated with only one user
- Member names cannot be the same
- Members must have an e-mail address and password assigned to the account
- Password for user account cannot be recovered, only changed

OPERATIONAL CONSTRAINTS

- Publications displayed after using search bar must fulfill the search requirements
- Members can delete a publication only if it was uploaded by them
- Admins can delete any publication available in the service
- Members can assign a rating to a publication only once, however the rating can be corrected
- Publication rating must be an integer between 1 and 10

CONCLUSIONS

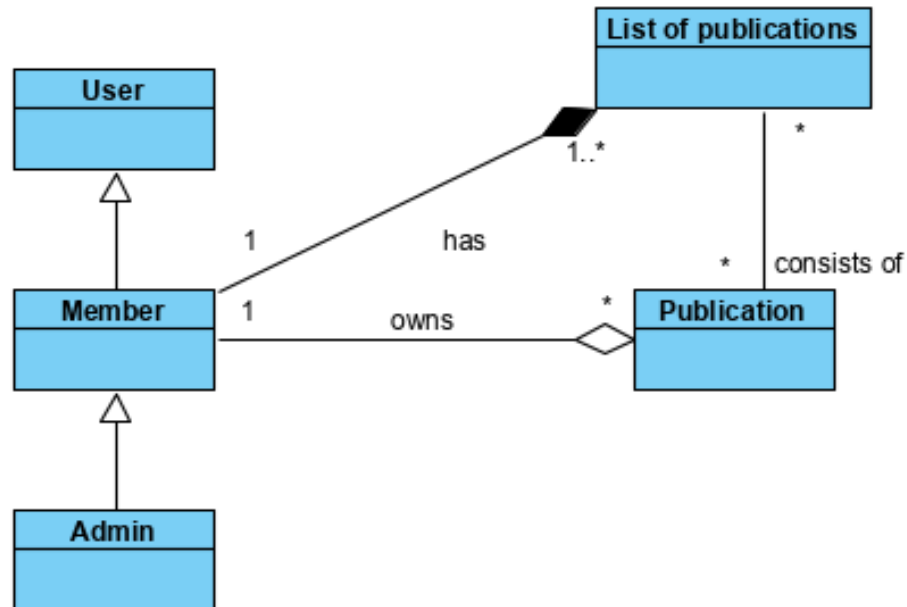
- Publications with a worse rating are presented further in search results
- A member frequently violating terms of service might have his/her account terminated
- In case of an account deletion all user publications are deleted as well
- A publication that violates terms of service might be deleted by admin

CALCULATIONS

- Rating for a publication is a mean of every rating that was submitted for this specific publication

3.4 Class diagram

Class diagram should further exemplify the business logic of the application. Class diagram serves as an important part of object-oriented modelling that maps application entities to classes and models relationships between them.



Picture 3.1 Class diagram.