Documentation for Delib, course project on ITP II

Team members:

Daniel Kalinin
Elena Patrusheva
Anastasia Pichka
Susanna Gimaeva
BS1-5 students

Innopolis, 2018

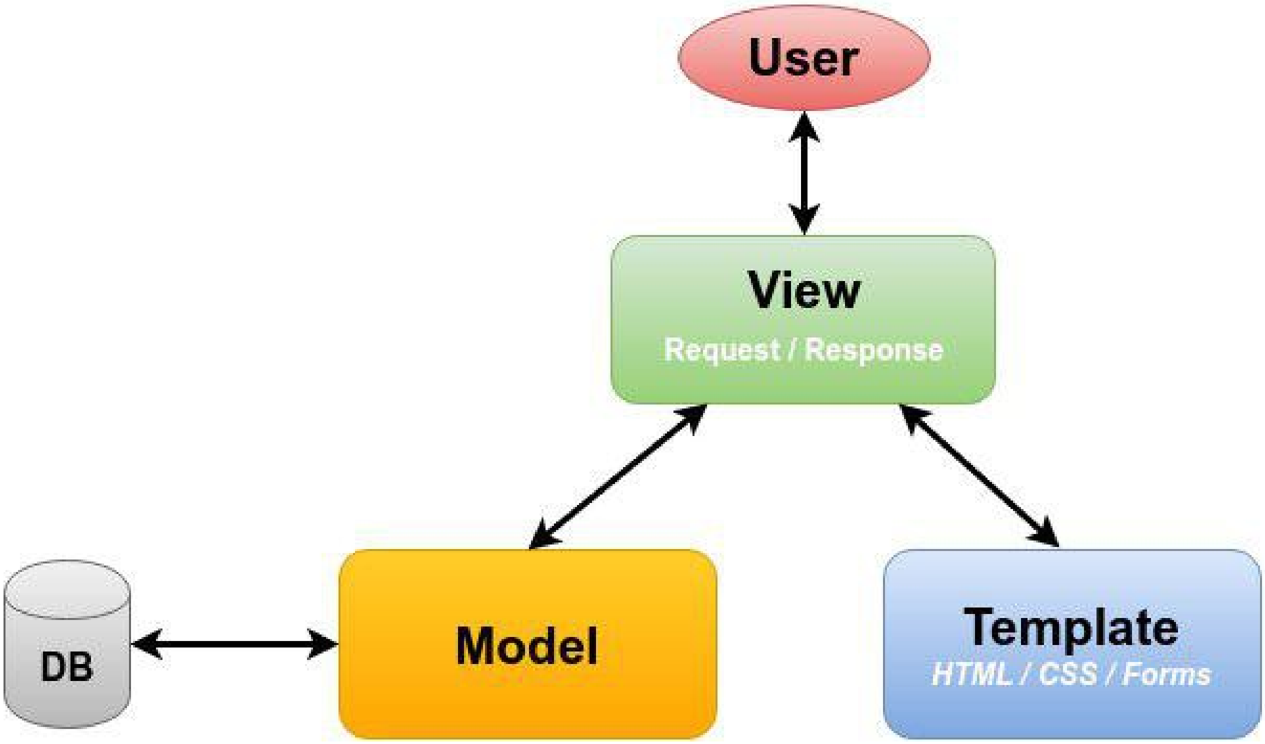
Introduction and Purpose

This document describes the architecture and design for the Delib project. Delib is a web site project for all users who are fond of reading books and magazines and would like to have access to the IU library. Everyone can register, see all the books our library can offer and check them out. A registered user will get the response if the library has available copies of the document you need and if there is a book you are searching for.
Unlike many systems making a waiting list of users for the book, Delib allows to be sure once a user has checked out the copy of the book online, he or she can take it in the real library at any convenient time and not hurry to be the first to come and take it.

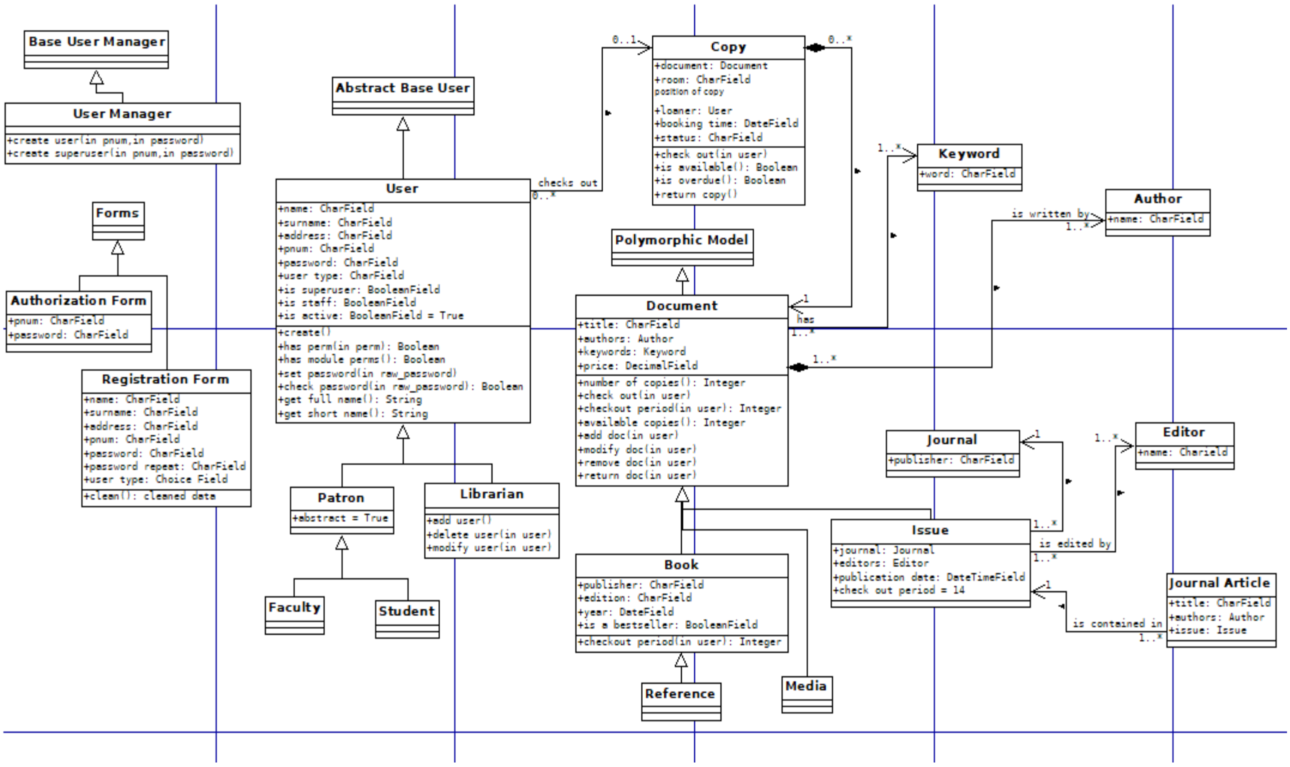
The purpose of the document is to describe the project in a way that addresses the interests and concerns of both users and developers. The document will identify major system components and describe their static attributes and dynamic patterns of interaction.

Django framework was chosen to implement the project as a web site because it is suitable for projects like this and it met the developers needs.

High-Level System Design



Detailed Class Design (uml-diagram)



Components description

Users

There are two types of users (your implementation should be flexible enough to allow the extension to more types of users)

Patron:

Library patrons can search for, check out and return documents. Documents can be checked out for certain time periods. There are two sub-types of patrons: Faculty, representing all the teaching body (e.g. professors, instructors, TAs) and Students of the university. A student cannot be a faculty member.

Librarians:

Librarians can check overdue documents, manage patrons, and add/delete/modify documents.

Documents

The main asset of the library are documents. The application contains the following type of documents, although, can be extended to more types:

Books:

Books are written by one or more authors and published by a publisher. Books have a title and may exist in different editions – each published in a certain year. For example, "Introduction to Algorithms" is a book written by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein. It was published by the MIT Press. The third edition was published in 2009.

Journal Articles:

Journal articles are written by one or more authors, have a title, and are published in a certain journal. Journals have a title and are published by a publisher in issues. Issues have editors and a publication date. For example, "Communication ACM" is a journal. The article "Go to Statement Considered Harmful" written by Edsger W. Dijkstra was published in the "March 1968" issue of this journal and Edward Nash Yourdon was the editor of the issue.

Audio/Video (AV) materials:

AV materials have a title and the list of authors. Documents also have the price value (in Rubles) – used when a patron gets a fine, and a list of keywords (e.g. "Programming Languages", "course material") – used when a patron is searching for documents.

Copies:

A library may have several copies of each document. Copies are stored in a certain place inside the library, e.g., a room, level. For each copy we need to know whether it is currently checked out and by whom. Patrons are only allowed to check out documents for a certain time (e.g., 2 weeks).

Requirements:

1. Each user of the library (librarians and patrons) has one unique card for as long as they are in the system.
2. The library needs to know at least the name, address, phone number, and library card number for each person of the system.
3. In addition, at any particular point in time, the library may need to know or to calculate the items a patron in the system has checked out, when they are due, and any outstanding overdue fines.
4. A patron can check out copies of documents. The system makes sure that it is not possible to check out a copy of a document if all copies of this document are currently checked out by other patrons.
5. Books are checked out for three weeks, unless: • they are current best sellers, in which case the limit is two weeks or • they are checked out by a faculty member, in which case the limit is 4 weeks (regardless the book is best seller)
6. AV materials and journals may be checked out for two weeks.
7. The overdue fine is a hundred rubles per item per day, but cannot be higher than the value of the overdue item.
8. The library also has reference books and magazines, which cannot be

checked out.

9. A patron can renew an item once (and only once), unless there is an outstanding request for the item, in which case the item needs to be returned immediately.
10. Users are able to access detailed information for each document or refine/modify his search.
11. Librarians can add/delete/modify documents and patrons to the library system. For example, a librarian can add a new copy for a book to the library and record where it is kept within the library.

For users and customers

In order to have access to the local library management system you need to run the site on a local server. As soon as you are here, you will be offered to register and after that all the functionality of the system will be provided to you. For better understanding of the scope of actions the system allows for, here is the list of user stories which illustrate common situations. This list will give you an idea of how it can be used, however the options are not limited to the list provided below.

1. As a student, I want to check out book b1 – which is not a best seller nor a reference book – for 2 weeks.
2. As a faculty, I want to check out book b1 – which is not a best seller nor a reference book – for 1 week.
3. As a librarian, I want to add a student s – with his respective information – to the system so he can check out books.
4. As a librarian, I would like to see the list of the already registered patrons in the library system.
5. As a librarian, I would like to modify documents in the library system.
6. As a librarian, I would like to add to the system a new copy of a book that is already in the system (the information data should be stored).
7. As a librarian, I would like to add a new document (e.g book, journal), being able to specify the required fields (e.g. title, authors, etc.).
8. As a librarian, I want to modify a user card, so that I can see the change after reopening the card.
9. As a librarian, I would like to have the information of all documents, users and their relation (who has booked what) even if the system is restarted.
10. As a librarian, I want to add any number of user cards, so that later I

can delete them in any order.

11. As a librarian, I would like to modify some information about a particular user such as a student has been upgraded to faculty.
12. As a student, I will checkout a book and I would like to see the return date for the book calculated by the system.
13. As a patron, I would like to return a book.
14. As a librarian, I would like to delete a book from the storage system (e.g. because it has been lost).
15. As a librarian I want to add a new document and specify the amount of copies of the document
16. As a librarian I want to see the list of checked out documents and request the return of some specific document.
17. As a librarian I would like to accept the documents returned by patrons

Team members contribution for the delivery:
Backend - Daniel Kalinin and Elena Patrusheva
Frontend and documentation - Anastasia Pichka and Susanna Gimaeva