**Object-oriented Analysis & Design with UML**

**Advanced Web Programming**

# Project Report

Project Name：

Group Name：

Date：

Dept.: Computer Sciences & Engineering

# Contents

Project Report 1

Contents 2

1. Introduction 3

1.1 Background of the project 3

1.2 Division of work 3

2. Requirements 4

2.1 User stories 4

2.2 Alternative flows 4

2.3 Use cases 4

2.4 Activity diagram 5

2.5 Classes 6

3. Design 7

3.1 Architectural design 7

3.2 The workflow of the system (if needed) 8

3.3 Class Design 8

4. Implementation 8

4.1 Module 1 8

5. Testing (optional) 9

# 1. Introduction

## 1.1 Background of the project

*Briefly introduce the project’s background, which should include: the users of the system, the objectives of the system...*

For example:

SEU needs a web-based teaching system, which will be used by students, teachers, and an administrator. The objectives of different users are:

(1) The students can log in to the website to browse, search, and download files.

(2) The teachers can log in to the website to upload and delete course-related files.

(3) The system administrator can approve a user registration application.

The theme of the project can be one of the following topics. You can also propose a topic that interests you and develop it after evaluation by the teacher.

1. A course selection system: After logging in, students can select their required and elective courses for the current semester based on their department and grade level, and view their own schedule. Administration teachers can maintain course information and students’ information after logging in. Administrator can help the students to select courses too.
2. A Library Management System: After logging in, students can search, borrow, and return books, and view their historical borrowing records. Administrators can maintain book information and students’ information after logging in. Administrator can help the students to borrow and return books too.
3. An e-commerce system: The main functions are like Amazon.com or Taobao.
4. A Restaurant ordering system: After scanning the code to confirm the table number, customers can browse dishes, select dishes, and pay bills. After the waiter logs in, they can help customers place orders. After logging in, the restaurant manager can see sales statistics.

## 1.2 Division of work

*Briefly introduce your team members and how the work is divided between each one.*

# 2. Requirements

## 2.1 User stories

*Describe user stories of the project.*

*For example:*

(1) As a student, I can register on the website. After the administrator's approval, I can log in to the system.

(2) As a student, I need to log in to the website before I can use it.

(3) As a student, after login, I can browse the files that teachers uploaded.

(4) As a student, after login, I can search for files by keywords, and download found files.

(5) As a teacher, I can register on the website. After the administrator's approval, I can log in to the system. This story is similar to (1).

(6) As a teacher, I need to log in the website before I can use it. This story is similar to (2).

(7) As a teacher, after I’ve logged in, I can upload files to the website.

(8) As the administrator, I need to log in to the website before I use it. This story is similar to (2).

(9) As the administrator, after I login, I can approve the user's registration application.

## 2.2 Alternative flows

*Describe alternative flows of the user stories.*

*For example:*

(1) When a teacher uploads a file, the upload will be rejected if the file is larger than 20 Mo.

## 2.3 Use cases

*Give use case diagrams to model your user stories, as Example 1 shows. If your system is too large, please separate your use cases into several diagrams.*

Example 1:

(1) Use case diagram

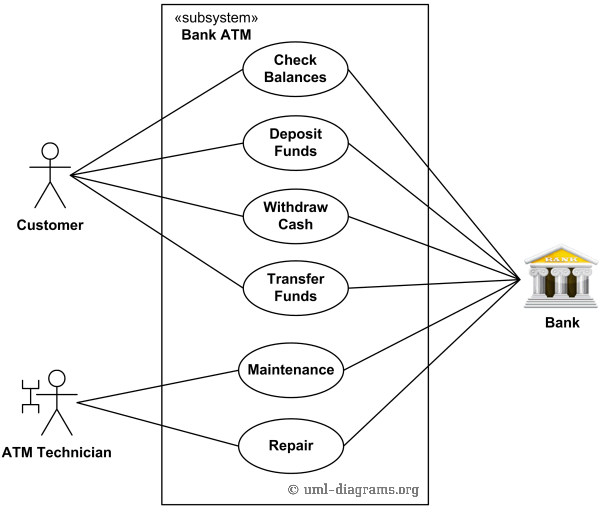


Figure 1. example of a use case diagram

(2) use case specification

Table 1. specification for the Register use case

|  |  |  |
| --- | --- | --- |
| **Items** | **Descriptions** | **Notes** |
| Name of the use case | **Register** |  |
| Brief description | Users (students and teachers) can create an account in the website. |  |
| Basic flow | 1. a user can click a button called "Register" in the welcome user interface of the website.  2. a dialog will show, the user inputs the "username", selects "student" role or "teacher" role, inputs the password two times, then submits.  3. registration finished, go back to the welcome user interface. |  |
| Alternative flows | 1. If the username is not acceptable, display why.  2. If the passwords are not consistent, tell the user.  3. If the user's password is not acceptable, display why. |  |
| Precondition | 1. Nobody is logged in to the system. |  |
| Postcondition | 1. Add a new user record in database.  2. The user just registered is now logged in. |  |

## 2.4 Activity diagram

*Use activity diagrams to illustrate the steps of a use case.*

*The following examples are activity diagrams for two use cases.*

(1) For the “download file” use case



Figure 2. Activity diagram for the "download file" use case

## 2.5 Classes

*Using CRC，design classes for your system by abstracting objects and refining them.*

For example:

We designed 4 classes for our system.

(1) BookList class: explanation of BookList class.

(2) BorrowList class: explanation of BorrowList class.

....



Figure 3. The initial class diagram of the system

# 3. Design

## 3.1 Architectural design

*Give your architectural design and explanations to each module.*

*For example:*

XXXX System

Sub-System 1

Sub-System 2

Sub-System 3

Module 4

Module 3

Module 2

Module 1

Figure 4. The architecture of the system

*Explanations:*

(1) Module 1: the functions of Module 1.

(2) Module 2: the functions of Module 2.

.....

## 3.2 The workflow of the system (if needed)

*Use an activity diagram to illustrate the operating processes of your system & give some explanations about them.*

## 3.3 Class Design

*Design details for your classes and explain them.*

For example:

....

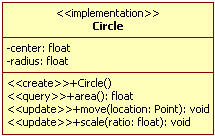


Figure 5. some detailed classes and their relationships

# 4. Implementation

## 4.1 Module 1

(1) Briefly introduce the main functions of Module 1.

(2) The classes involved in Module 1 should be specified and explained.

(3) If there are message transmissions between the classes, use a sequence diagram to illustrate them.

For example:

The sequence diagram (for the "Download File" case)



Figure 6. The sequence diagram of Module 1

(4) If you use databases to store your data, please describe the design of your tables.

For example:

Table2. Sys\_XXXXX\_Info

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Explanation** | **Data Type** | **Note** |
| Id | 编号 | Int | Primary Key |
| ProjectName | 工程名称 | Varchar |  |
| VersionID | 项目版本号 | Varchar |  |
| ProjPath | 项目路径 | Varchar |  |
| McCabe | 圈复杂度 | Int |  |

# 5. Testing (optional)

For the main scenarios, please show how your system runs with a sequence of user interfaces.

(1) Briefly introduce the scenario of the system usage.

(2) Show how your system runs according to the scenario.

For example:

For the use case xxxx, the users need to do xxxx, and the system will do xxxx.

The steps of this scenario are:

(1)The first step of xxxx is to do xxxx, and the system will show a user interface as Figure 7 shows:

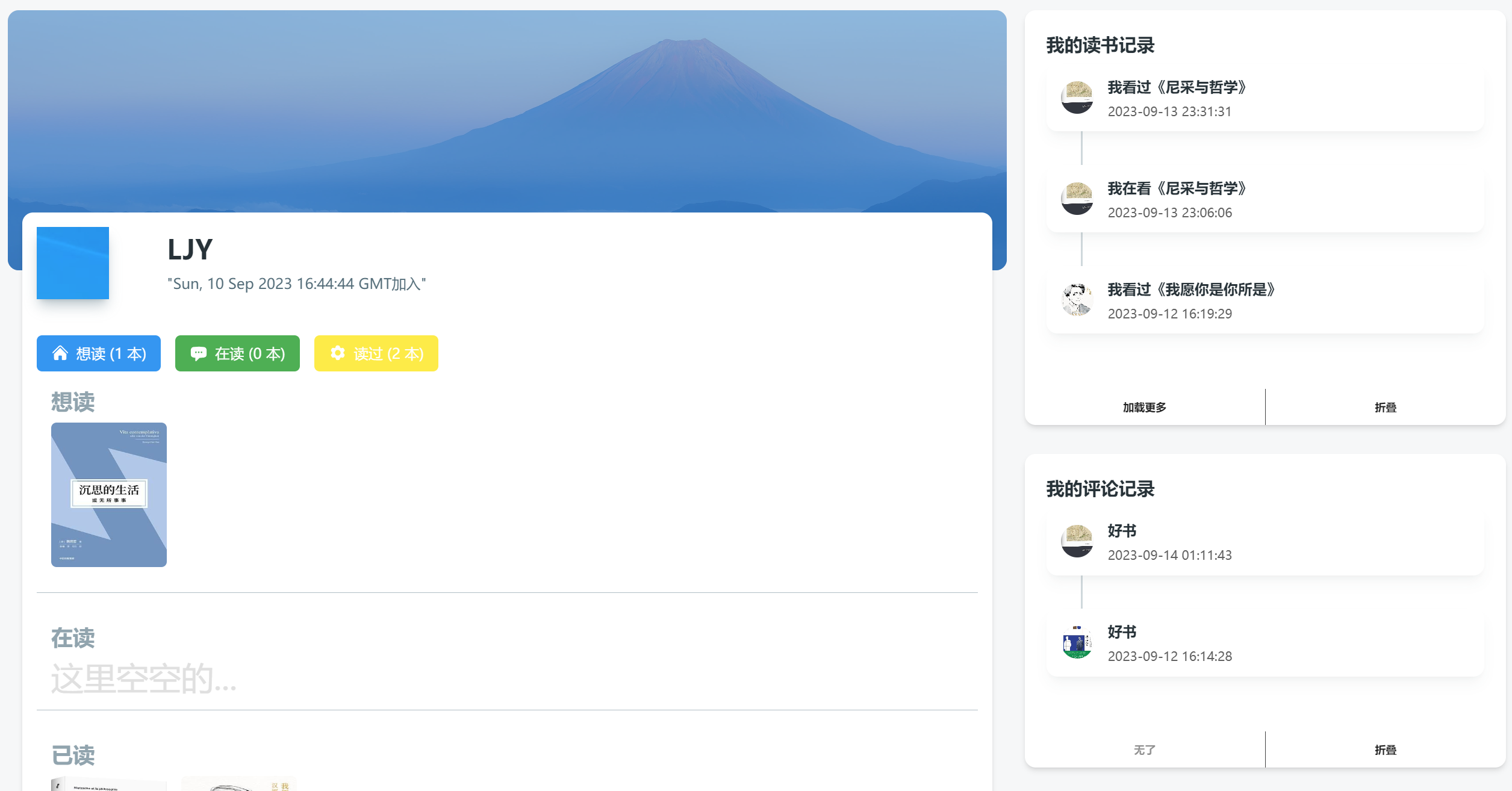
(2)

Figure 7. The user interface of xxx