Cairo University

Faculty of Computers and Information

CS251

Software Engineering I

Online University Library System

Software Requirements Specifications

Malak Ahmed Saber

Touka Mohamed Korany

Menna Tullah Hisham Mohamed

CS251: Phase 1 – Team Name
Project: <Online Library System>

Software Requirements Specifications

March 2024

Contents

CS251: Phase 1 – Team Name
Project: <Online Library System>

Software Requirements Specifications

Team

ID	Name	Email	Mobile
20220341	Malak Ahmed Saber	duskee1234@gmail.com	01200970897
20220094	Touka Mohamed Korany	touka.agra@gmail.com	01205291454
20220344	Menna Tullah Hisham Mohamed	mennahesham533@gmail.com	01140799279

Document Purpose and Audience

- **This document is a Requirements Document includes the requirements lists, use case model, use cases description. We will use this document for Requirements analysis in the upcoming SDLC steps.**
- **The Target Audience is : Customer/client (he is supposed to confirm on this document to continue the project, all his requirements have to be satisfied)**

Introduction

Software Purpose

- **Satisfy the customer requirements, developing a system that can handle the changes therefore developing a system that does not decay.**

Software Scope

We need to ensure and facilitate the online borrowing of library books by university members, authentication, certification, payment, and reward points management so the components are :

- **Borrowing Books :** requires authentication and integration with join waiting list functionality
- **Authentication :** supports authentication with a login page or google account, requires integration with an external identity provider service
- **Certification :** lets user be able to certify (login) if uncertified, get extended borrowing periods after certification based on syndicates organizations.

CS251: Phase 1 – Team Name
Project: <Online Library System>

Software Requirements Specifications

- o **Payment :** supports payment with credit card or reward points, requires integration with external bank/credit card service.
- o **Reward Points Managment :** Earn reward points through various actions like borrowing books, participating in events, or contributing to the library community.

Definitions, acronyms, and abbreviations

- **Identity Provider = the one who supplies/stores university members' google accounts**

Requirements

Functional Requirements

- **The Identity Provider :**
  - o **Should be able to authenticate the user as a certified university member**

- **The University Member :**
  - o **Should be able to explore books**
  - o **Should be able to search for books**
  - o **Should be able to browse books (find and see available books) and Should be able to browse books (find and see available books/ recommended books) while borrowing.**
  - o **Should be able to certify if not certified**
  - o **Should be able to get an extended borrowing period service after he certifies if he is a member of some syndicates-organizations**
  - o **Should be able to return borrowed books**
  - o **Should be able to join a waiting list when he tries to borrow a book if he is authorized as a certified member**
  - o **Should be able to borrow books if he is authorized as a certified member**
  - o **Should be able to checkout after he borrows a book if he is authorized as a certified member**
  - o **Should be able to pay by reward points or pay by credit card**
  - o **Should be able to earn reward points if he borrowed books or participated in library events or contributed in library events**
  - o **Should be able to authenticate through a login page or authenticate through user's google based university account**

CS251: Phase 1 – Team Name
Project: <Online Library System>

Software Requirements Specifications

Non Functional Requirements

|	Details
Performance	• System should have 1 millisecond response time during browse and search.
Scalability	• System should be able to handle high usage during peak usage periods.
Availability	• System should be available 24 hours during day. • System should not be down more than 5 minutes when updating the system.
Security	• System should encrypt user data during authentication by password hashing (store them in database not plain text) , and encrypt payment information by tokenization.
Maintainability	• System should be easy to modify and provides adding new features.
Usability	• System should include clear functionality and suitable places for buttons to be easy to use (Browsing Button, Borrowing button, Returning button).
Compatibility	• System should support different operating systems (mac , Linux, Windows)
Supportability	• System must be able to add new books without any modification or system crash.

Software Requirements Specifications

System Models

Use Case Model
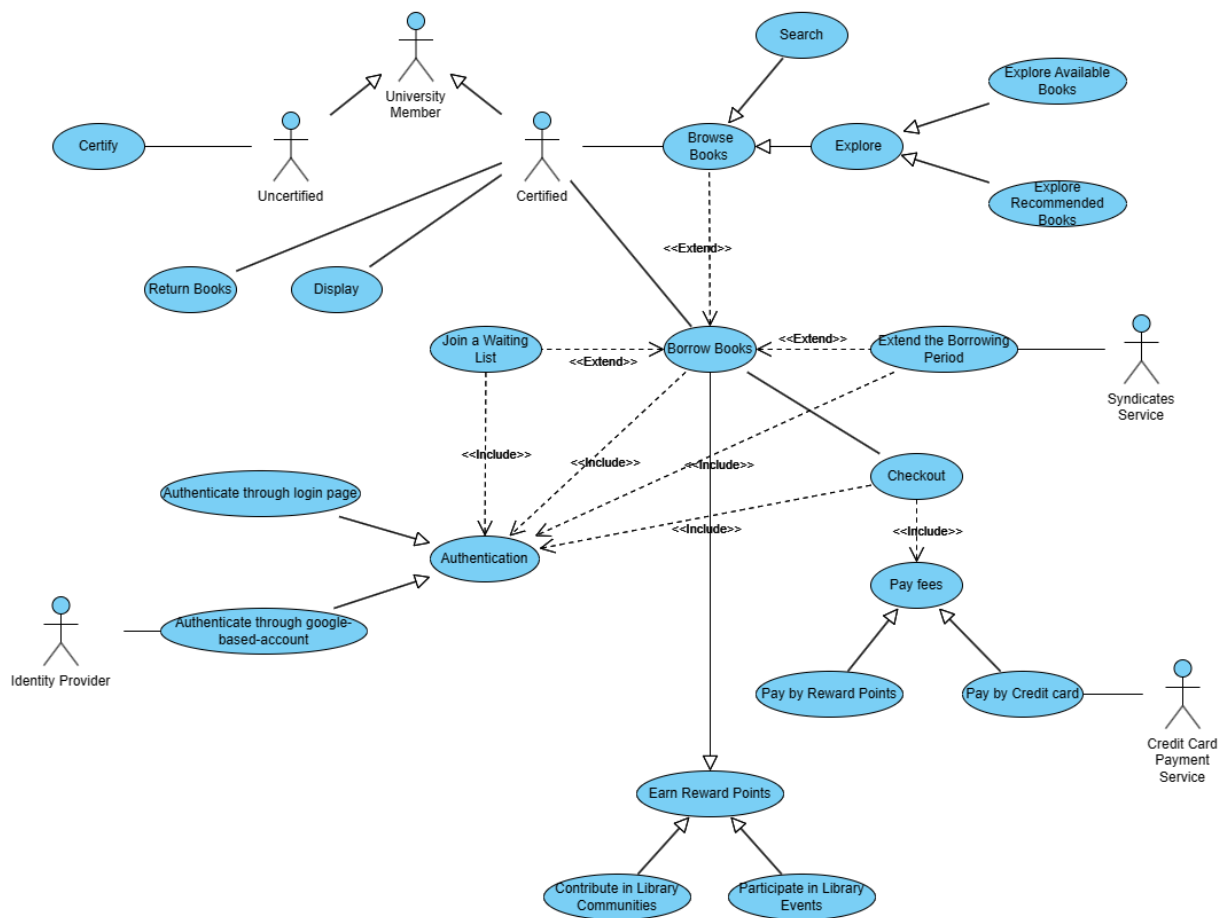
CS251: Phase 1 – Team Name
Project: <Online Library System>

Software Requirements Specifications

Use Case Tables

Use Case ID:	1	
Use Case Name:	Checkout	
Actors:	BookBorrower	
Pre-conditions:	The BookBorrower is certified. The BookBorrower asks to borrow a book. The BookBorrower has an amount of reward points from borrowing previous books, participating in library events, contributing in library events.	
Post-conditions:	The BookBorrower's reward points is decremented. The BookBorrower's brrowed books list includes the new borrowed book.	
Flow of events:	**User Action**	**System Action**
	1- The BookBorrower clicks on checkout button.	
		2- Calls authenticateifcertified
	3- The BookBorrower authenticate as certified.	
		4- Library responds by offering a choise between payment by reward points or credit card.
	5-The BookBorrower selects payment by reward points.	

Software Requirements Specifications

		6-Library checks amount of reward points AND announces the new amount with an acknowledgment.
	7-The BookBorrower receives acknowledgment and confirm it.	
		8-Library adds book to borrowed books list AND calculates new reward points AND stops the interaction.
Exceptions:	**User Action**	**System Action**
	1- The BookBorrower clicks on checkout button.	
		2- Calls authenticateifcertified
	3- The BookBorrower authenticate as certified. [invalid]	
		4-Library announces failure due to unauthority AND stops the interaction.
	5-The BookBorrower selects payment by reward points.	
		6-[invalid]Library annonces failure due to lack of reward points AND

Software Requirements Specifications

		stops the interaction.
	7- The BookBorrower receives acknowledgment and confirm it. [invalid]	
		8-[invalid]Library stops the interaction.
Includes:	Authentication, payment function	
Notes and Issues:	---	

CS251: Phase 1 – Team Name
Project: <Online Library System>

Software Requirements Specifications

Use Case ID:	2
Use Case Name:	Authentication
Actors:	Member university, identity provider
Pre-conditions:	The member university as certified is already signed up in library website OR already have google account.
Post-conditions:	The member university as certified can join waiting list, borrow books (extended borrowing periods) and checkout.

Flow of events:	User Action	System Action
	1- The member university clicks on borrow button.	
		2-the library system shows two options login AND google account.
	3- The member university selects google account.	
		4- The library system shows page form.
	5-The member university enters his data.	
		6- The library system sends the data of the member university to identity provider.
	7- The identity provider authenticates the member university as certified.	

Software Requirements Specifications

		8-the library system sends acknowledgement authentication as certified member university.
	9-the member university receives acknowledgement and accepts it.	
Exceptions:	**User Action**	**System Action**
	1- The identity provider authenticates the member university as certified. [invalid]	
		2-Library system stop interacting with member university as certified.
	3- the member university receives acknowledgement and accepts it. [invalid]	
		4-library system announces failure and stop the interaction.
Includes:	Borrow book	
Notes and Issues:		

Software Requirements Specifications

Use Case ID:	3	
Use Case Name:	Borrow Book	
Actors:	The Book Borrower	
Pre-conditions:	The Book Borrower is certified. The Book Borrower click on Borrow Button	
Post-conditions:	The Borrowed Book goes to the Wishlist cart	
Flow of events:	**User Action**	**System Action**
	1- The user clicks on Borrow Button	
		2-The System check the user authentication
		3-if the user authenticated then check the availability of the book
		4- check if the book is suitable for user's age
		5-if all conditions met and the book is available, ask the user if he is in a specific institute
	6- the user responses and if he is in a specific institute he will write the institute name	
		7- if he is in a specific institute, The website borrowing system will offer an extended borrowing period

Software Requirements Specifications

	User Action	System Action
	8- the user accepts the offer	
		9-the system extends the borrowing periods 10-the system sends an acknowledgment
	11-the user confirms the acknowledgment	
		12-the system add the book to the Wishlist cart
Exceptions:	**User Action**	**System Action**
	The user doesn't authentication [invalid condition]	1- The system announces failure because the user doesn't authenticate 2-the system requests the user to authenticate
	The user requests a book to borrow [invalid book]	1- The system announces failure because the book is unavailable 2- The system recommends him other books and to joins the waiting list
	The user input his institute name [invalid institute]	1- The system announces failure because the institute that the user has inputted isn't exist 2- the system requests the user to input the institute name properly or to cancel the offer of the extended borrowing period

Software Requirements Specifications

	The user age isn't suitable for the book	1- The system announces reject borrowing the book 2- The system recommends him other books that are suitable for him
Includes:	Authentication	
Notes and Issues:		

Software Requirements Specifications

Use Case ID:	4	
Use Case Name:	Return book	
Actors:	Bookborrower	
Pre-conditions:	The member university already authenticated as certified and he already borrowed books.	
Post-conditions:	1-Returning book adds in available book list , remove from list of borrowed books . 2-overdue will update.	
Flow of events:	**User Action**	**System Action**
	1- the bookborrower clicks on return button.	
		2-the library system shows the borrowed book list.
	3- the bookborrower chooses the book.	
		4- the library system increases overdue fines if bookborrower exceed the borrowing duration and send acknowledgement .
	5-the bookborrower recieves acknowledgement and accepts it.	
		6- the library system removes return book from borrowed books

Software Requirements Specifications

		list.
Exceptions:	**User Action**	**System Action**
		1-In case the bookborrower don't exceed the borrowing duration , the overdue fines don't changes.
	2- -the bookborrower recieves acknowledgement and accepts it.[invalid]	
		3-the return book remains in the borrowed books list.
Includes:	Borrow book	
Notes and Issues:		

CS251: Phase 1 – Team Name
Project: <Online Library System>

Software Requirements Specifications

Ownership Report

Item	Owners
Functional Requirements	*Malak*
Non-Functional Requirements	*Touka (4 functions)* *Malak (2 functions)* *Menna (2 functions)*
Use Case Model	*Malak (Draft, Modifications)* *Menna (shared in Modifications)* *Touka (shared in Modifications)*
Checkout Use Case Description	*Malak*
Authorization Use Case Description	*Menna*
Borrow Use Case Description	*Touka*
Return Use Case Description	*Menna (All)* *Malak (shared in Flow of events)* *Touka (shared in Flow of events)*

Policy Regarding Plagiarism:

1. تشجع الكلية على مناقشة الأفكار و تبادل المعلومات و مناقشات الطلاب حيث يعتبر هذا جوهريا لعملية تعليمية سليمة
2. ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم فى الكود و لكن تبادل الحلول غير مقبول و يعتبر غشا.
3. أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.
4. قد توجد على النت برامج مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.
5. إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.
6. فى حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و فى حالة تكرار الغش سيرسب الطالب فى المقرر.