

Cairo University
Faculty of Computers and Informal



CS251

Software Engineering I

Assignment 1

Software Requirements Specifications

Submitted By:

- Maya Fouad Fathy 20220270
- Roaa bassem 20220136
- Nadra Mahmoud 20220355
- Lojayn Khaled Farouk 20220258



Team

ID	Name	Email	Mobile
20220270	Maya Fouad Fathy		
20220258	Lojayn Khaled farouk	lojaynk@gamil.com	01060909223
20220355	Nadra Mahmoud Saad	Oshasaad968@gmail.com	01152529724
20220136	Roaa bassem		01155480577

Document Purpose and Audience

- To have a clear understanding of how the system works and what functions the system perform
- For everyone involved in the project has a clear understanding of what needs to be developed and how it will function.

Introduction

Software Purpose

The software is designed to simplify the borrowing and management of library books for university members. It offers easy access to resources and includes features like certification, authentication, and reward points to improve user experience and engagement.

Software Scope

1. **User Authentication:** Enable university members to authenticate themselves either through the library's login page or their university Google account.
2. **Book Browsing:** Allow users to browse available books in the library, including searching, exploring, and viewing book details.
3. **Book Borrowing and Returning:** Facilitate the borrowing and returning of books by university members.
4. **Certification Process:** Implement a process for university members to certify themselves as certified members, enabling access to extended borrowing periods and other benefits.
5. **Waiting List Management:** Enable users to join waiting lists for books that are currently unavailable.
6. **Payment Functionality:** Integrate a payment function for checking out books, allowing payment through credit card services or university member reward points.
7. **Reward Points System:** Implement a system for university members to earn reward points through various actions such as borrowing books, participating in library events, or contributing to the library community.



Requirements

Functional Requirements

1. The university member could **certify** as a certified university member.
2. The University member could **browse books**, he could use it if the university member only wants to find and see the currently available books within the library to borrow.
3. The university member could **browse recommended books** based on past browsing actions.
4. The university member could **borrow books**, borrowing books required to be authenticated.
5. The university member could **return** a previously borrowed book.
6. A certified university member can get **extended borrowing periods** based on belonging to specific institutes offered by the Web site borrowing period extension service.
7. The University Member could perform **checkout** on borrowed books. To checkout, the university member needs to be certified, and a **payment** function should be involved. The payment could be done either through some credit card payment service, or using certified university member reward points.
8. University member **earn reward points** from reward points party through various actions such as borrowing books, participating in library events, or contributing to the library community in other ways.
9. University member could **join waiting list** for books that are not currently available, the waiting list requires authentication.
10. User **authentication** could be done through the library book borrowing system's login page or through the user 's googles based university account. Using university accounts requires external identity provider participation.

Non-Functional Requirements

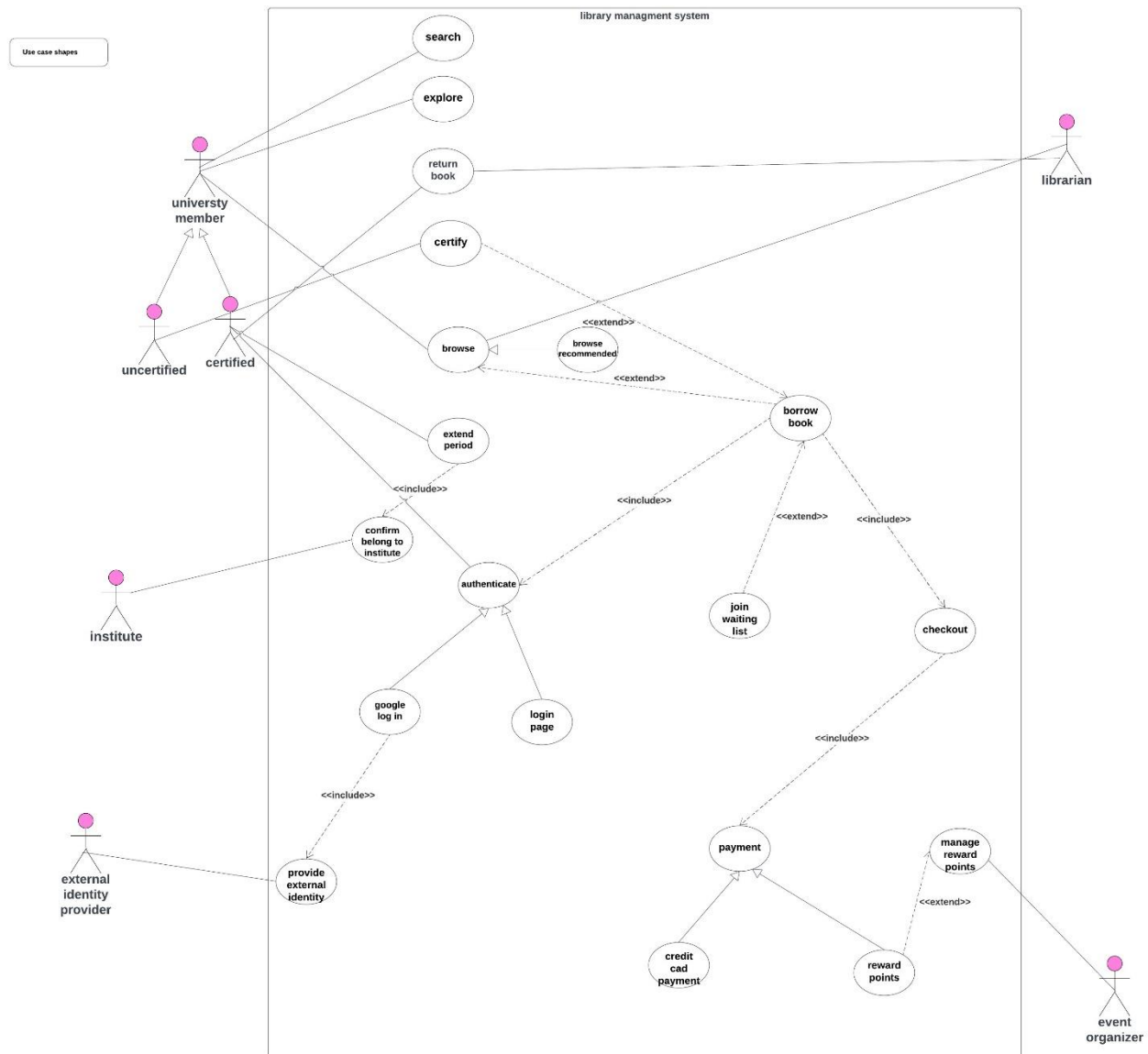
1. Payment transactions should be encrypted and comply with Payment Card Industry Data Security Standard for secure financial transactions (**Security**)
2. Response time for search should be done in no more than 10 sec (**Performance**)
3. If an error occurred in checking out after payment the system must recover Without data loss (**Reliability**)
4. The website should be able to handle at least 150 users concurrently during working Hours (**Performance**)
5. Clear and consistent layout, labeling, and navigation elements should be provided to guide users through the system (**Usability**)
6. User authentication processes should comply with industry-standard security protocols to prevent unauthorized access. (**security**)



7. The library management system shall be designed and implemented to ensure accessibility across all user devices, including but not limited to desktop computers, laptops, tablets, and smartphones, without compromising functionality or user experience (**portability**)
8. It should have built-in fault tolerance mechanisms to handle unexpected failures without impact on user experience (**Reliability**)

System Models

Use Case Model





Use Case Tables

Use Case ID:	1	
Use Case Name:	Borrow book	
Actors:	Initiated by university member, communicates with librarian	
Pre-conditions:	The university member has found a book he wants to borrow	
Post-conditions:	The books are borrowed, and added to the list of books of university member	
Flow of events:	User Action	System Action
	1- The university member presses borrow book button on the book he likes.	
		2- System checks if he is certified 3- System checks if he is authenticated. 4- system checks for book availability 5-System checks book age range if university member is less than 12 years old 5-system checks for allowed limit for borrowing books 6- system checks the range for accumulated Fines 7- system displays a checkout button
	8- University member presses on the checkout button	
		9-after checkout, system updates the borrowed book list For the university member and library
		10- system displays available slots form
	11- university member chooses a slot	
		12- system adds slot to library delivery slots
Exceptions:	User Action	System Action



	1-university member presses borrow book button	
		2-System checks certification, university member is not certified, system asks him to certify first
		3- The university member is not authenticated so the system supplies him with a form
	4- university member fills the form	
		5- system checks the book range and it is not suitable so the system rejects borrowing and informs the user
		6-system checks for the allowed borrowing limits, and its exceeded so the system rejects borrowing and informs the user
		7- system checks the range for accumulated Fines, and its exceeded so the system rejects borrowing and informs the user
		8-system checks for book availability, book is not found so the system puts him on waiting list and informs him
Includes:	Checkout, join waiting list, authentication, certification	

Use Case ID:	2	
Use Case Name:	Return	
Actors:	University Member, Librarian	
Pre-conditions:	<ul style="list-style-type: none"> • The University Member Logged into Library web Site (authentication) • The University Member must have previously borrowed books. 	
Post-conditions:	<ul style="list-style-type: none"> • Books are returned and Marked in Web Site as Available. • Receive calculated fines if there. 	
Flow of events:	User Action	System Action



	1- University Member request to return book.	
		2- System asks Member to enter the book ID.
	3- University Member enter the book ID.	
		4- System validates the book as being borrowed by the user. 5- System calculates fines for exceeding the borrowing duration. 6- Inform the University Member of the fines.
	7- University Member pays the fines.	
		8- System updates university member fines. 9- System records the time to return the book.
	10- Librarian receives the book from University Member. 11- Librarian confirms the return of the book.	
		12- System updates book status as available. 13-System removes book from university member borrowing list
Alternate Flow	If no fines, skip steps 5,6,7,8.	
Exceptions:	User Action	System Action
	3- University Member enter the book ID.	
		4- The book ID is not found or not borrowed by the user. System displays an error message, and asks Member to enter correct ID.



Use Case ID:	3	
Use Case Name:	Browsing Books	
Actors:	University Member, Librarian	
Pre-conditions:	Library website is opened	
Post-conditions:		
Flow of events:	User Action	System Action
	1- navigates to the "Browse Books" section	
		2- Displayed available books
	3- user filter books by recommendation	
		4-Displayed available books with in category
	5- navigates to the "explore Books details" section	
		6-Displayed books details

Use Case ID:	4	
Use Case Name:	Checkout	
Actors:	University member	
Pre-conditions:	1) The Uni-member logged in as a certified member 2) The Uni-member select one or more books to borrow	
Post-conditions:	1) The total Income of the library increase and the Uni-member be ready to complete the borrowing prosses	
Flow of events:	User Action	System Action
	1- the uni-member click the checkout books bottom	



		<p>2- the system calculates the fees based on the number of the the borrowed books and overdue fines less than a specific amount of money</p> <p>3- the system displays the borrowed books , the fees and the available borrowing date and the total fees of each one.</p>
	4 - the uni-member confirm the checkout and click the payment bottom	
		<p>5 – the system completes the process of the payment following its steps</p> <p>“another use case”</p>
		<p>6 -The system updates the total library income</p> <p>7- The system generates confirmation of the checkout prosses and display on it the success massage</p>
Exceptions:	User Action	System Action
	1- The total fines is more than the determined amount	
		2- the system asks the member to eliminate the number of borrowed books
	1-University member click the checkout books bottom without borrowing books	
		2- the system asks the member to eliminate the number of borrowed books
Includes:		



Ownership Report

Item	Owners
Borrow Use case	Lojayn khaled
Return use case	<i>Maya fouad</i>
Check out use case	<i>Nadra mahmoud</i>
Browsing books use case	<i>Roa bassem</i>