

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light greenish-blue. They are positioned diagonally, with the blue one partially covering the green one.

# Virtual Bookshelf

Group #5



# Software Process Model

Our team has chosen an agile-based approach to developing our virtual bookshelf. Our goal is first to create a functioning prototype and then iteratively add features. We anticipate encountering various challenges, and an agile methodology will allow us to adapt by adjusting or removing features that no longer align with our vision. Additionally, this approach enables us to focus on specific aspects of the project through sprints, breaking the workload into smaller, more manageable tasks. This makes development more efficient and less overwhelming. Lastly, since this approach is iterative, it allows us to continuously refine our user interface, ensuring a more user-friendly experience.



# Requirements

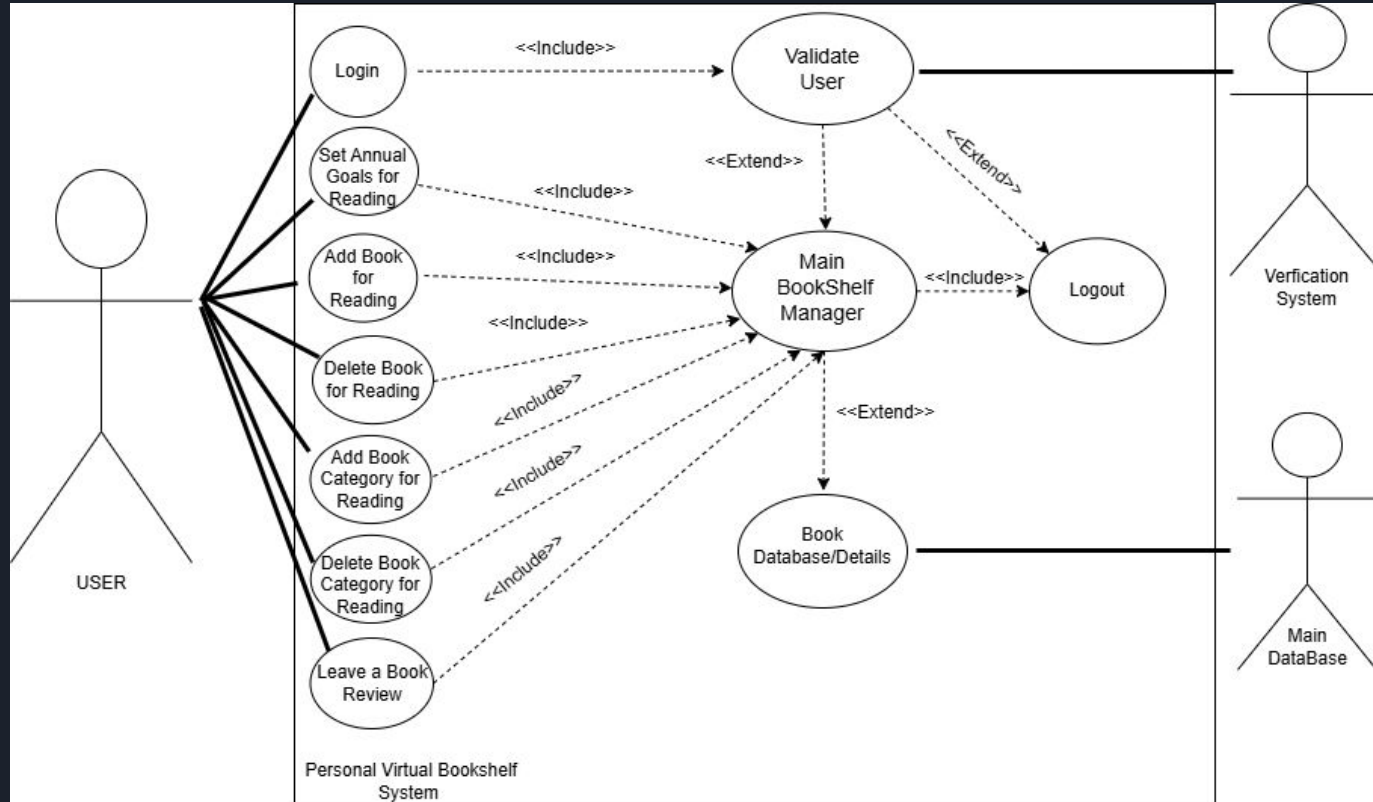
## Functional Requirements:

- Users shall be able to add and delete books.
- Users shall be able to create, edit, and delete categories for books.
- Users shall be able to set annual reading goals (number of books they want to read per year).
- Users shall be able to search for books they have added.
- Users shall be able to leave thoughts on how they felt about their book and rank them.

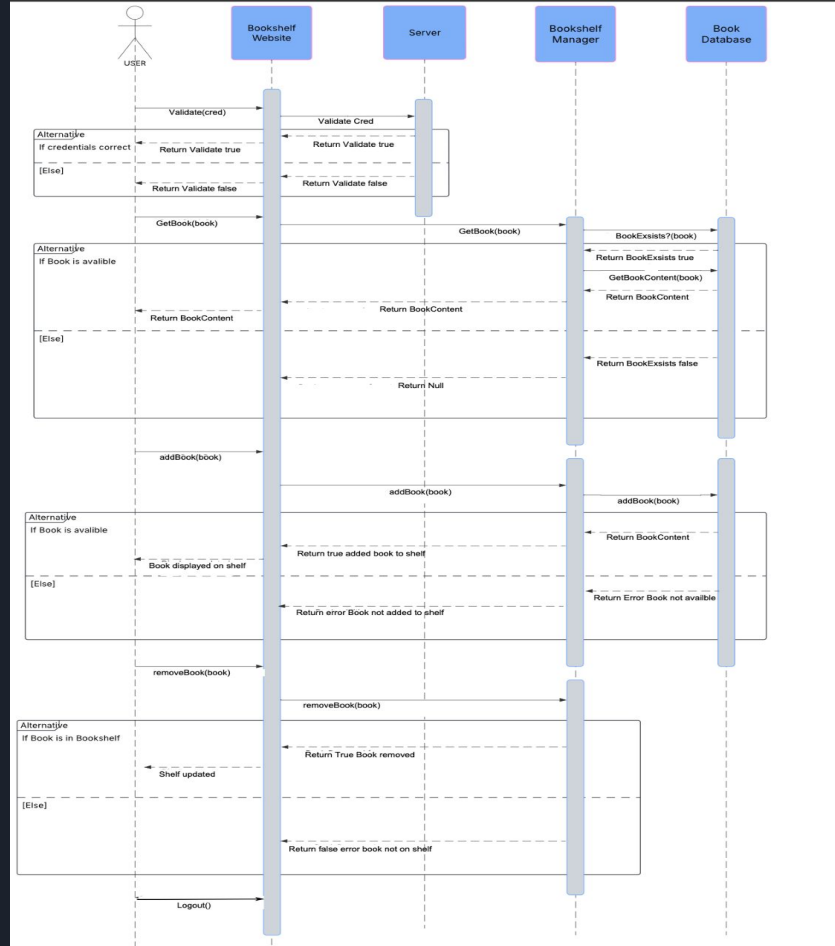
## Non-Functional Requirements:

- The system shall have a user-friendly interface making it easy to move around the website.
- The system shall be responsive to different devices and easily accessible through all.
- The system shall respond quickly to user actions, like adding or deleting books.
- Data shall be memorized and persisted so the user can keep track of their progress, previous books, etc.
- The system shall be scalable to handle many books for users.
- The system shall follow agile methods so that new features may be added into the future and the system can be up to date.

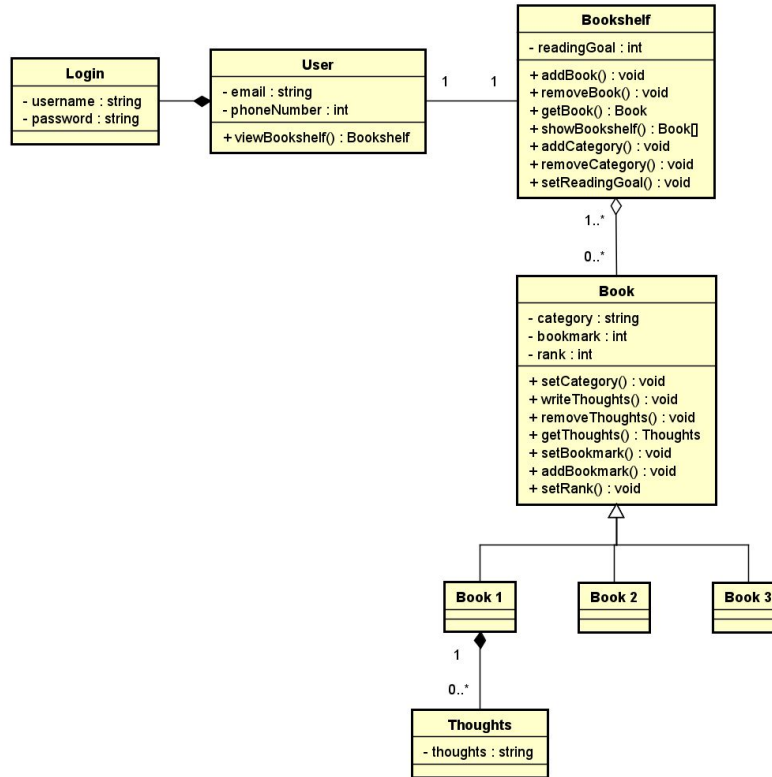
# Use Case Diagram



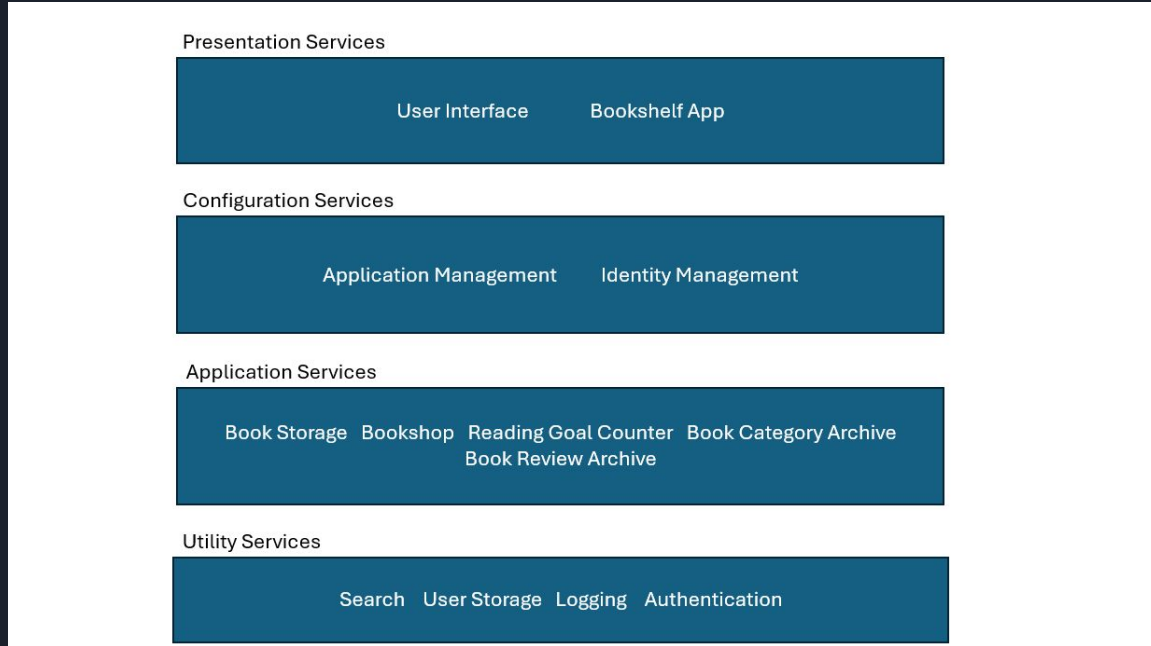
# Sequence Diagram



# Class Diagram



# Architectural Design



Our team chose the Layered Architecture Pattern because it can be split so that each section can be tinkered with independently. The layered pattern also allows us to add or remove different new features, so the system stays up to date. The layered pattern can also have the same systems in each layer that help increase the dependability of the system.

# Project Scheduling

Function Category	Count	Simple	Average	Complex	Complexity
User input	10	3	4	6	40
User output	6	4	5	7	30
User Queries	7	3	4	6	28
Data files / relational tables	20	7	10	15	300
External Interfaces	4	5	7	15	28

$$PCA = (45 * 0.1) + 0.65 = 1.1$$

$$GFP = 426$$

$$FP = 486.6$$

$$E = 486.6 / 30 = 15.6 \text{ or } 16$$

$$\text{Duration} = 16 / 4 = 4 \text{ weeks}$$

Estimated Cost of Hardware Products:  
\$350/yr

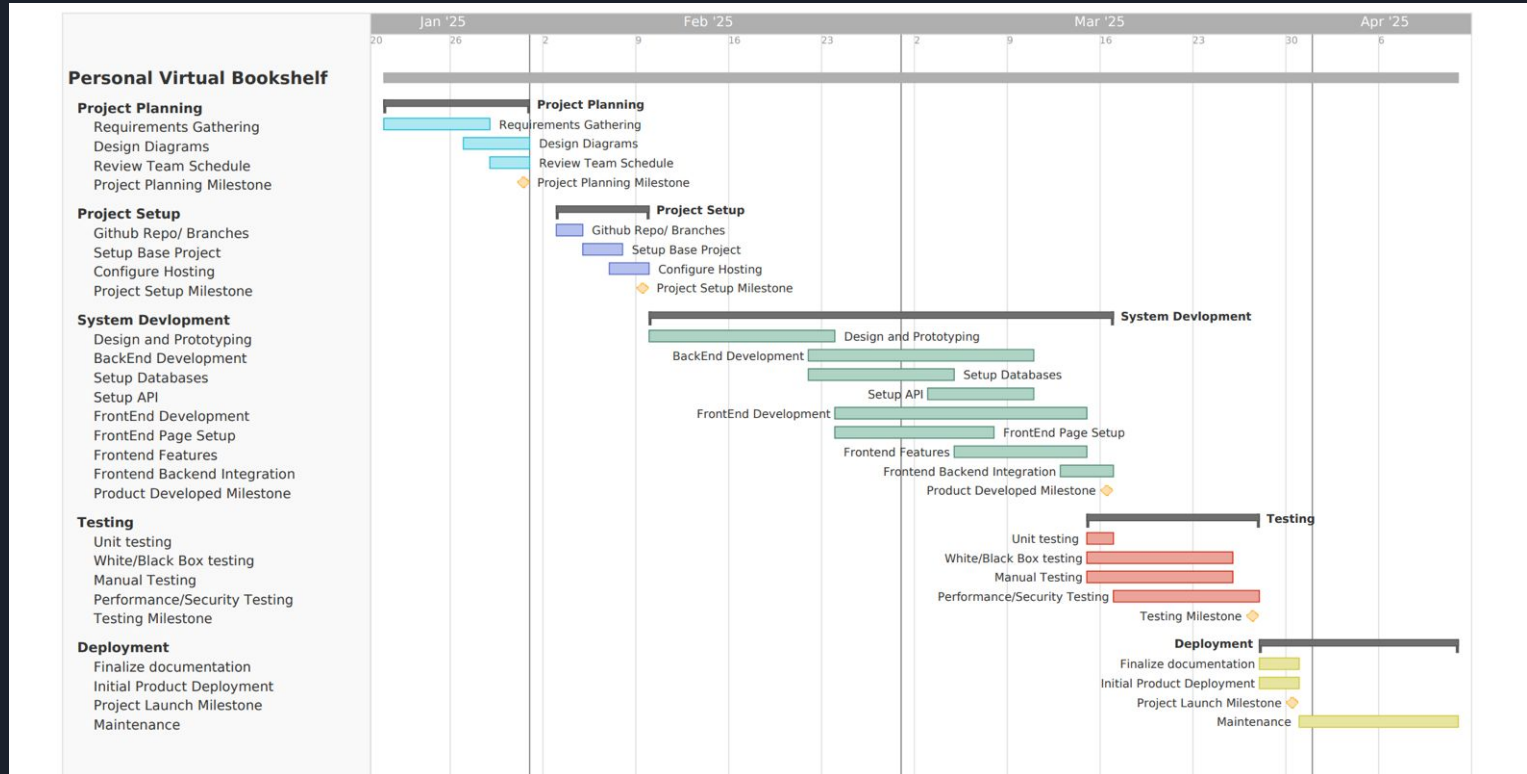
Estimated Cost of Software Products: \$40/yr

Estimated Cost of Personnel: \$10,000

- Project Duration: 4 weeks
- Staffing: 4



# Gantt Chart





# Requirement-Use Case Traceability Matrix

Req ID	Req Description	Priority Weight
R1	System must allow user authentication	3
R2	User must be able to set and track reading goals	2
R3	User must be able to manage (add/delete) books	3
R4	User must be able to manage (add/delete) book categories	2
R5	System must allow user to logout securely	1
R6	System must manage book records with database integration	2
R7	User must be able to add personal book reviews	2

UC ID	Use Case Name
UC1	Login
UC2	Set Annual Goals for Reading
UC3	Add Book for Reading
UC4	Delete Book for Reading
UC5	Add Book Category for Reading
UC6	Delete Book Category for Reading
UC7	Validate User
UC8	Main BookShelf Manager
UC9	Book Database/Details
UC10	Add Personal Book Review
UC 11	Logout

# Requirement-Use Case Traceability Matrix

Requirement-Use Case Traceability Matrix												
Req ID	Priority Weight	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11
R1	3	X						X				
R2	2		X						X			
R3	3			X	X				X			
R4	2					X	X		X			
R5	1											X
R6	2									X		
R7	2										X	
Score	15	3	2	3	3	2	2	3	7	2	2	1

# Specification Table: Part A

(a) Test case specification				
#	Name	Description	Input	Expected Output
1	Login	Test for Login	Valid Username/Password	User is authenticated
2			Invalid Username/Password	User is not authenticated & Display Error Msg
3	Goal	Test for Set Goals	Input a number of books	Goal saved and shown in dashboard
4	Add book	Test for adding book into bookshelf	Valid Book Name	Book is added into bookshelf from database
5			Invalid Book Name	Display Error Msg
6	Delete Book	Test for deleting book from bookshelf	Valid Book Name	Book is removed from bookshelf
7			Invalid Book Name	Display Error Msg
8	Add book Category	Test for adding book category into bookshelf	Valid Book Category	Book Category is added into bookshelf
9			Invalid Book Category	Display Error Msg
10	Delete book Category	Test for deleting book category from bookshelf	Valid Book Category	Book Category is removed from bookshelf
11			Invalid Book Category	Display Error Msg
12	Logout	Test for logout	Select Logout	Session Ends & Redirect Login
13	Book Management	Test for search book in database	Valid Book Name	Book is searched in database, returned to User
14			Invalid Book Name	Display Error Msg
15		Test for rename book	Valid Book Name	Book name is renamed in database
16			Invalid Book Name	Display Error Msg
17		Test for rename book category	Valid Book Category	Book Category is renamed from database
18			Invalid Book Category	Display Error Msg
19	Review	Test for leaving a review from user	Valid review	Review is saved and linked to that book
20			Invalid review	Display Error Msg

# Specification Table: Part B

(b) Tests produced from the Test Case Specification							
#	Name	Description	Input	Expected Output (EO)	Actual Output (AO)	Passing Criteria	Test Result
1	Login	Test for Login	Username/Password	Login Successful and Proceed forward	TBD	EO = AO	TBD
2			Username/Password	"Invalid Username or Password" Msg displayed	TBD	EO = AO	TBD
3	Goal	Test for Set Goals	Set Goal: 10 books	User profile shows reading goal as 10	TBD	EO = AO	TBD
4	Add book	Test for adding book into bookshelf	Title: "Harry Potter"	"Harry Potter" appears in user's bookshelf	TBD	EO = AO	TBD
5			Title: "innf"	"Invalid title! No book found" Msg displayed	TBD	EO = AO	TBD
6	Delete Book	Test for deleting book from bookshelf	Delete: "Harry Potter"	"Harry Potter" removed from user's bookshelf	TBD	EO = AO	TBD
7			Delete: "innf"	"No book found in bookshelf" Msg displayed	TBD	EO = AO	TBD
8	Add book Category	Test for adding book category into bookshelf	Create Category: "Fantasy"	"Fantasy" appears in category list	TBD	EO = AO	TBD
9			Create Category: "Fan"	"Invalid Book Category" Msg displayed	TBD	EO = AO	TBD
10	Delete book Category	Test for deleting book category from bookshelf	Delete Category: "Fantasy"	"Fantasy" is removed from category list	TBD	EO = AO	TBD
11			Delete Category: "Fan"	"No such Book Category exists" Msg displayed	TBD	EO = AO	TBD
12	Logout	Test for logout	Selecting logout	Session Ends and Saving User Profile	TBD	EO = AO	TBD
13	Book Management	Test for search book in database	Search: "Harry Potter"	Returns search results include "Harry Potter"	TBD	EO = AO	TBD
14			Search: "innf"	"No such Book exists" Msg displayed	TBD	EO = AO	TBD
15		Test for rename book	Rename: "Harry Potter" to HP	All of the books named Harry Potter is renamed HP	TBD	EO = AO	TBD
16			Rename: "H_P" to HP	"No books with H_P were found!" Msg Displayed	TBD	EO = AO	TBD
17		Test for rename book category	RenameC: "Fantasy" to "Adventure"	"Fantasy" is renamed into "Adventure" in category list	TBD	EO = AO	TBD
18			RenameC: "Fan" to "Adventure"	"No fan Book Category exists" Msg displayed	TBD	EO = AO	TBD
19	Review	Test for leaving a review from user	Review: "Harry Potter": "Loved it"	The review is saved and displayed with book	TBD	EO = AO	TBD
20			Review: "H_P": "Loved it"	"No such title found" Msg displayed	TBD	EO = AO	TBD



# Conclusion

Throughout this project, our goal was to design the foundational elements of a virtual bookshelf interface, outlining the structure, user interactions, and key features. Each team member contributed to different aspects of the design. Abeel Khan led the documentation and requirements, ensuring that all functional and non-functional aspects were clearly defined. Kevin Sun worked on the sequence diagrams, mapping out how users would interact with the system. Marvin Farinas focused on the class diagram, organizing the components of the system. Syed Shaheer Ali contributed with the use case diagram. Chris Castro designed the architecture, providing a structural overview. Jay Trivedi chose the Software Process Model and created the slideshow. Albion Krasniqi helped organize the repository and the overall project. As for me, Yusuf Kuzey, I worked on the conclusion and references. Throughout the design process, we faced a few challenges. One of the main challenges was ensuring that our design was flexible and scalable, allowing for future additions and changes. We had to make sure the system could handle new features, such as user ranking and private notes, without compromising the performance. Another challenge was organizing user data in a way that was both efficient and easy to manage. If we had more time, we would have refined the design further, focusing on areas like data storage and user interface elements. We would also have liked to explore additional features and improvements, ensuring the design was as user friendly as possible. Overall, this project has allowed us to develop our skills in system design and collaborate as a team to create a solid foundation for a virtual bookshelf. We are proud of the work we have done, and we are excited to see how this concept could evolve with further developments.