

CS496 Software Project: Karson Institute Digital Library

Vilnis Jatnieks, Aidan Marshall

2026-01-26

1 Client Information

By sharing this client information and the rest of this document, you are stating that this client has provided this project as something they want (not something you created and asked if they wanted), and that they are interested in having you complete this project for your capstone.

- Client name:
- Client title:
- Client email address:
- Client employer:
- How you know the client:

2 Project Description

2.1 Overview

[Add a few paragraphs describing your project succinctly. What problem are you trying to solve, what is the purpose of your project? Why does your client want this project?]

2.2 Key Features

[At this point you should have a basic understanding of your client's needs. List out the key features of the software system the client wants you to build.]

2.3 Why this Project is Interesting

[Why did you decide this project was interesting enough to you to be a capstone project? What about this project is enticing? Why should anyone care?]

2.4 Areas of CS required

[What subfields of computer science seem most likely to be relevant to your project? A capstone must involve multiple.]

2.5 Potential Concerns and Questions

[Is there any aspect of this project that makes you unsure if it will work, either due to your own interests/background, or that you aren't sure if it fits the requirements? Are there questions you have about this project that you want instructor feedback about?]

2.6 Summary of Efforts to Find a Project

(Not necessary for 482) [Briefly list out when/how you've discussed with this client, and if you've discussed with other clients who either didn't work out or didn't respond. If you considered a different project and it didn't work out, why didn't it work out?]

[Most CS495 projects end here. The sections below are for CS482 and CS496 software projects].

2.7 Comparison to Draft

[For CS496 only, focus on highlighting the major differences between the draft proposal in CS495 and this one here. If there are no major differences, you can remove this subsection.]

3 Requirements

3.1 Non-Functional Requirements

[Non-functional requirements are just as important as functional requirements. Dont forget to specify them.]

Table 1 presents the NFRs for this software project.

ID	NFR Title	Category	Description
NFR0	Restrict checkout	System	Users with overdue books cannot check out additional books
NFR1	Password strength	Security	Passwords must have 8+ characters with 1 capital letter, 1 number, and 1 special character
NFR2	2 Factor Authentication	Security	Allow users to add additional security to their account
NFR3	SSO	Security	Allow users to use Loyola SSO
NFR4	Notifications	UI	Bell icon + number of notifications
NFR5	Limit Review Posters	System	Only users who have checked out a book can leave a review for that book
NFR6	Comment filter	System	Filter out profanities and unwanted language
NFR7	Log In / Log Out	System	Ability to log in and log out to your personal account

Table 1: Non-Functional requirements

3.2 Functional Requirements (User Stories)

[In CS482, all functional requirements are written as User Stories. In CS496, some projects may use a different template to write the requirements. The table below is an example of writing the Stories. Adapt accordingly to different templates or if you want to display more info.]

Table 2 presents the functional requirements for this software project.

ID	Story Title	Points	Description
A0	Users CRUD	8	As a/an Admin, I want to CRUD other users, so that I can assign users to have staff permissions and delete bot accounts
S1	Book CRUD	8	As a/an Staff, I want to CRUD books, so that I can maintain the collection of books
S2	Fill book info via ISBN	2	As a/an Staff, I want an alternative way to fill book information with an ISBN number, so that make logging a book easier
S3	Scan barcode	8	As a/an Staff, I want to add book by scanning barcode, so that make logging even easier

U4	Search for book	2	As a/an User, I want to search for a book, so that I can find what I want
S5	Check out/in book	1	As a/an Staff, I want to check out a book, so that I can update the current status of the collection
U6	Reserve book	1	As a/an User, I want to reserve book, so that I can let people know to not take it
U7	Reminder email	2	As a/an User, I want to receive reminder emails when due date is approaching, so that I can return my book on time
U8	View books	2	As a/an User, I want to view the books, so that I can find a book that I'm
U9	Filter the view of the books	1	As a/an User, I want to filter book, so that I can easily change view
U10	Recommended books	13	As a/an User, I want to see recommended books, so that I can find something that is interesting to me
S11	All checked out books	2	As a/an Staff, I want to see all books that have been checked out, so that I know which books are not currently in the library
U12	Sort the view of books	1	As a/an User, I want sort books by when they were published, so that I can see the newest published books
U13	Upvote/downvote comments	2	As a/an User, I want to react to comments, so that I can show my approval
S14	Tags CRUD	8	As a/an Staff, I want to CRUD tags for books and other works, so that I can organize my collection how I would like
S15	Late users	2	As a/an Staff, I want to view users who have overdue books, so that I have an easy way to follow up with students who need to return a book
U16	User view my history	2	As a/an User, I want to see previous books I checked out, so that I can remember what books I previously read
S17	Staff view book history	2	As a/an Staff, I want to see a book's history, so that I can see where a book has went
U18	Popular books	1	As a/an User, I want to see which books have been checked out the most, so that I can read what other books people find interesting
U19	Password reset	5	As a/an User, I want to reset password, so that I can log if I forget my password
U20	View book covers	3	As a/an User, I want to view book covers, so that I can see the collection as if I were in the library in person
U21	Rate book	4	As a/an User, I want to rate a book I read, so that I can let other people know my opinion on the book
U22	Contact staff	1	As a/an User, I want to send emails to the staff, so that I can ask questions about the collection
U23	Book preview	2	As a/an User, I want to preview the first pages of a book, so that I can preview before checking out
U24	Request extension	2	As a/an User, I want to request an extension of a book, so that I can keep the book for longer
S25	Approve extension	2	As a/an Staff, I want to approve an extension of a book, so that I can allow people to keep books for longer
U26	Create account	2	As a/an User, I want to create an account , so that I can log back in
U27	Notifications	4	As a/an User, I want to get notifications about overdue books, new books added, receive notis from admin, etc., so that I know what actions need to be taken
A28	Admin send notifications	2	As a/an Admin, I want to push notifications to staff, so that I can send information

U29	Comments CRUD	8	As a/an User, I want to comment reviews on books, so that I can share my opinion about the book
U30	Report comments	2	As a/an User, I want to report comments, so that I can ensure a safe space
Total:		106	

Table 2: Functional requirements as User Stories.

4 System Design

4.1 Architecture

[Which type of software architecture are you team following? Layered architecture, MVC, other? What are the main modules for your software?] We will use client server architecture. The modules in server are Model, Controller, Routes, Services, and Middleware. The modules in client are Comonents, [Main modules are not the same as Layers. If you adopted any form of layered architecture (MVC included), then your layers already group components based on responsibility. Therefore, for modules, think about semantically related components. For example, in a parking lot, I could have a User, Payment, Parking (Vehicles), Contact/Issues modules.]

4.2 Diagrams

[CS482, on sprints/iterations 2-3, you need to create and update a diagram (check Iteration 2-3 assignment for which type of diagram). On CS496, since before sprint/iteration 1, you should have a class diagram and keep it up-to-date. In CS496, if your class diagram changes at each sprint, then create a Class Diagram subsection for the sprints, and show the changes; while keeping the one here the most up-to-date version.]

4.3 Technology

Technology	Function	Justification
NextJs	Frontend framework	Easier routing
React	Frontend framework	Reusable components, hooks
Tailwind	CSS library	Easier CSS
ShadCN	Component library	Pre-made components
Node.js	JS Runtime	Use one programming language across frontend and backend
Express	Backend framework	Less boilerplate, easier authentication
TypeScript	Programming language	Strongly typed, self-documenting
ESLint	Linter	Enforce code style
Jest	Test framework	Easier unit testing
Postman	Backend testing	Easier manual testing, backend development
PostgreSQL	Database	Relational database, speed

4.4 Coding Standards

[Are your team going to follow any coding standards? For example, using a naming convention for Database tables (like only singular lowercase names). Another example, only allowing code with unit tests and above 60% coverage to be committed (good convention since testing is going to be evaluated). If you need inspiration to define your coding standards, the Extreme Programming approach has a set of coding, design, and test rules.] Collections and React components will be written in Pascal case. Set up github actions to automatically run test cases. Features need to be created on new branches. A0-users-crud will be the naming convention for feature branches. Pull requests must be reviewed by each other. We will use a dev branch during sprints and merge to main at the end of each sprint. Require 80% line coverage for all features.

4.5 Data

[What is the main structure of your data? In SQL-like databases, this would be the planning of the main tables, their attributes, and interactions with other tables (basically an ER diagram). In NoSQL databases, this would be the main collections and general attributes of the JSON you will store in each collection.]

[Tip to better find and write the data your system will need. Go back to your User Stories and for each one, think to yourself: which attributes/fields do I need to store for this to work?]

[Tip 2. When a system has many different roles for people, those are usually done in a single User table/collection. Especially when they share many common attributes/fields.]

4.6 UI Mocks

[Define and draw/sketch/code the main UIs your user will interact with in your software. Add your UI mocks here and a short caption about it. Do not forget about the main forms and CRUD UIs.]

5 Iterations

5.1 Iteration Planning

[In CS496, you plan all iterations beforehand. In CS482, you update the planning here at each iteration.]

Table 3 shows the iteration planning.

It.	Dates	Stories	Points	
			Planned	Done
1	01/20 - 02/10	A0 Users CRUD, S1 Book CRUD, U8 View books, U26 Create account	20	
2	02/11 - 02/24	U4 Search for book, U7 Reminder email, S11 All checked out books, S5 Check out/in book, S14 Tags CRUD, S2 Fill book info via ISBN, S15 Late users, U9 Filter the view of the books, U12 Sort the view of books	21	
3	02/25 - 03/17	U21 Rate book, U27 User get notifications, U20 View book covers, U6 Reserve book, U24 Request extension, S25 Approve extension, A28 Admin send notifications, U18 Popular books	20	
4	03/18 - 03/31	U29 Comments CRUD, S3 Scan barcode, U30 Report comments, U13 Upvote/downvote comments, U23 Book preview, U22 Contact staff	23	
5	04/01 - 04/14	U10 Recommended books, U19 Password reset, U16 User view my history, S17 Staff view book history	22	
			Total:	106 0

Table 3: Iteration Planning for Incremental Deliveries

5.2 Iteration/Sprint 1

5.2.1 Planning

[Which stories did you plan for this iteration/sprint. Add the total points for this plan. You can also explain the reason behind your planning, and what major feature(s) your team is focusing on delivering by

completing these stories. You may use a table for a summary display of the planning, but elaborate in text more detail in your focus and feature plan.]

5.2.2 Work Done

[Which stories did you complete in this iteration/sprint. Which ones did you partially complete? Who worked on which story? You may elaborate in paragraph(s) to add more detail about the work done.]

5.2.3 Testing Coverage

[Testing is very important. Show your coverage here. Is this coverage good enough? Explain why you think so. Is it not good enough? Explain a plan to increase the coverage. You may also elaborate on why some artifacts do not undergo much testing. If the testing changed from the last iteration, explain the reasons.]

Figure 1 shows the test coverage

5.2.4 Retroespective & Reflection

[What were the pitfalls, challenges, and issues you had in this iteration? How can you address them to improve the process in the next iteration? Did anything not go according to plan? Why so and how to avoid the same mistake? Write a personal reflection on what you learned in this iteration (even if a small technical thing like Database storage).]

5.3 Iteration/Sprint 2

5.3.1 Planning

[Which stories did you plan for this iteration/sprint. Add the total points for this plan. You can also explain the reason behind your planning, and what major feature(s) your team is focusing on delivering by completing these stories. You may use a table for a summary display of the planning, but elaborate in text more detail in your focus and feature plan.]

5.3.2 Work Done

[Which stories did you complete in this iteration/sprint. Which ones did you partially complete? Who worked on which story? You may elaborate in paragraph(s) to add more detail about the work done.]

5.3.3 Testing Coverage

[Testing is very important. Show your coverage here. Is this coverage good enough? Explain why you think so. Is it not good enough? Explain a plan to increase the coverage. You may also elaborate on why some artifacts do not undergo much testing. If the testing changed from the last iteration, explain the reasons.]

5.3.4 Retroespective & Reflection

[What were the pitfalls, challenges, and issues you had in this iteration? How can you address them to improve the process in the next iteration? Did anything not go according to plan? Why so and how to avoid the same mistake? Write a personal reflection on what you learned in this iteration (even if a small technical thing like Database storage).]

5.4 Iteration/Sprint 3

5.4.1 Planning

[Which stories did you plan for this iteration/sprint. Add the total points for this plan. You can also explain the reason behind your planning, and what major feature(s) your team is focusing on delivering by completing these stories. You may use a table for a summary display of the planning, but elaborate in text more detail in your focus and feature plan.]



figures/placeholder.png

Figure 1: Iteration 1 test coverage report

5.4.2 Work Done

[Which stories did you complete in this iteration/sprint. Which ones did you partially complete? Who worked on which story? You may elaborate in paragraph(s) to add more detail about the work done.]

5.4.3 Testing Coverage

[Testing is very important. Show your coverage here. Is this coverage good enough? Explain why you think so. Is it not good enough? Explain a plan to increase the coverage. You may also elaborate on why some artifacts do not undergo much testing. If the testing changed from the last iteration, explain the reasons.]

5.4.4 Retroespective & Reflection

[What were the pitfalls, challenges, and issues you had in this iteration? How can you address them to improve the process in the next iteration? Did anything not go according to plan? Why so and how to avoid the same mistake? Write a personal reflection on what you learned in this iteration (even if a small technical thing like Database storage).]

5.5 Iteration/Sprint 4

[CS496 has 5 sprints. CS482 only has only 3 sprints (remove Iterations 4 and 5 from this doc if you are writing a doc for 482)]

5.5.1 Planning

[Which stories did you plan for this iteration/sprint. Add the total points for this plan. You can also explain the reason behind your planning, and what major feature(s) your team is focusing on delivering by completing these stories. You may use a table for a summary display of the planning, but elaborate in text more detail in your focus and feature plan.]

5.5.2 Work Done

[Which stories did you complete in this iteration/sprint. Which ones did you partially complete? Who worked on which story? You may elaborate in paragraph(s) to add more detail about the work done.]

5.5.3 Testing Coverage

[Testing is very important. Show your coverage here. Is this coverage good enough? Explain why you think so. Is it not good enough? Explain a plan to increase the coverage. You may also elaborate on why some artifacts do not undergo much testing. If the testing changed from the last iteration, explain the reasons.]

5.5.4 Retroespective & Reflection

[What were the pitfalls, challenges, and issues you had in this iteration? How can you address them to improve the process in the next iteration? Did anything not go according to plan? Why so and how to avoid the same mistake? Write a personal reflection on what you learned in this iteration (even if a small technical thing like Database storage).]

5.6 Iteration/Sprint 5

5.6.1 Planning

[Which stories did you plan for this iteration/sprint. Add the total points for this plan. You can also explain the reason behind your planning, and what major feature(s) your team is focusing on delivering by completing these stories. You may use a table for a summary display of the planning, but elaborate in text more detail in your focus and feature plan.]

5.6.2 Work Done

[Which stories did you complete in this iteration/sprint. Which ones did you partially complete? Who worked on which story? You may elaborate in paragraph(s) to add more detail about the work done.]

5.6.3 Testing Coverage

[Testing is very important. Show your coverage here. Is this coverage good enough? Explain why you think so. Is it not good enough? Explain a plan to increase the coverage. You may also elaborate on why some artifacts do not undergo much testing. If the testing changed from the last iteration, explain the reasons.]

5.6.4 Retroespective & Reflection

[What were the pitfalls, challenges, and issues you had in this iteration? How can you address them to improve the process in the next iteration? Did anything not go according to plan? Why so and how to avoid the same mistake? Write a personal reflection on what you learned in this iteration (even if a small technical thing like Database storage).]

6 Final Remarks

6.1 Overall Progress

[Have you completed everything? If so, present evidence on how you brought value to your client, and the overall client satisfaction. Otherwise, estimate how much progress you done and how long it would take to finish this project. Be concrete about your progress, you know how many story points your software is, how many points you completed (this shows your progress). You also know how many points your team delivers at each iteration, therefore you can estimate how many more iterations it would take to finish the leftover points (show the math).]

6.2 Project Reflection

[Your personal reflection on the project. What lessons did you learned. What would you have done differently? How can you do better work in future projects? You may write this as a team or per person (or both — if all your iterations were team reflections, then it would be better to write individual reflections here)]

Appendix

[Appendix section if needed]