

## Full Stack Development Capstone Project Description & Requirements

The capstone project is intended to showcase your competencies by building a full stack web application using software development skills, languages, and tools learned through the Full Stack Development (FSD) program. The FSD instructional team will be there to help you through coding challenges, but this design-build project will be your original work.

Your **Capstone Project Proposal** must be approved by an FSD instructor or teaching associate (TA) prior to starting work on the project, to help ensure it aligns with project requirements. TAs assess proposals for project scope and estimated development time to deliver a minimum viable product (MVP) reflective of an entry-level software developer.

### Capstone Project Requirements

- **You must build a full stack web application that incorporates microservices architecture and utilizes at least 60% original code from tools learned within the course.**
  - **Front End** - The front end must be built using React
  - **API** - The front end must communicate with the back end via an API call which enables CRUD (create, read, update, delete) functionality
  - **Back End / Database** - The back end must be built using Python or another back end language in conjunction with a database (e.g., SQLite, MySQL, MongoDB, Redis) learned in the course. Integrating other languages or tools is acceptable with TA permission.
- **You must demonstrate proficiency with topics covered in the course.**

Languages and Technologies (at least six from this list):

HTML	CSS	JavaScript	React
Python	Flask	SQL / NoSQL	Axios / Fetch / Ajax
JSON	UML	UI / UX	Git / GitHub

Methodologies / Best Practices (at least six from this list):

Control Structures	Functional Programming	Object-Oriented Programming	Algorithms
RESTful APIs	Project Management	Agile	Behavior-Driven Development
Test-Driven Development	Code Quality	Software Engineering	Problem Solving

- **You must incorporate one tool or feature that you did not learn in the course, but acquired through your continuous and expanding study of software development.** (Items such as new languages or frameworks are allowed but need to be approved by a TA or instructor). Examples include:
  - Search bar
  - Image carousel
  - Payment processing
  - UI component library
  - Third party API (weather, music, information, etc.)
- **Your application must be mobile responsive.**
  - If this is not feasible, explain your situation to a TA or instructor before submission (e.g., browser-based game, detailed data visualizations).
- **Your front end and back end must be hosted and operational.** You may use any hosting service (e.g., Heroku) and must provide working links in your documents.
- **Your project must include documentation.** Auto-generated readme files will not count, your code repositories must have essential information about your project (tools used, languages use, key functionality, etc.).
- **Use of tutorials is permitted.** However, tutorial-derived material **may not constitute more than 40%** of your capstone project.

## Examples of Approved Projects

- **eCommerce** – Build an online shop. User Story: A user can select from a set of products and walk through the full purchase process.
- **Content Management System** – Build a blog platform. User Story: A user can create a blog account and add entries that include various media with customization.
- **Chat App** – Build the key features utilized for chat functionality. User Story: A user can create a group chat room and post messages in the group.
- **Learning Management System** – Build an educational site (like devCamp). User Story: A user can create a set of lessons, complete exercises, take quizzes and see scores.
- **Small Business Application** – Build an application to help manage a small business. User Story: A business owner can create and update inventory, manage customers and employees, manage advertising and promotion, etc.

## Preparing Your Capstone Proposal

Your capstone proposal is a one-page document that answers five questions. Email your proposal to [fullstack@bottega.edu](mailto:fullstack@bottega.edu).

1. What is your project's title?
2. What is the purpose of your application or the problem you are trying to solve with your application?
3. Who is your intended end-user for your web application?
4. What applications and languages do you expect to employ in your design?
5. What is your predicted timeline and milestones to complete your build?

## Project Submission Directions

### Your project document will include:

- Summary / description of your application.
  - Keep in mind that only Bottega staff will see your submission document. Further detail can go into your README.md associated with your code.
- Live URLs for your front end and back end.
- URLs for your front-end and back-end GitHub repositories.
- Any usernames / passwords / hidden routes needed to test your application.
- Description of your self-taught tool / feature.
- Any additional information needed to fully assess your application.
- Please do not submit your code files.

### Uploading Your Project Submission Document

You will need a Bottega Canvas account in order to upload your final capstone document. The steps in Canvas are straightforward:

- Go to [Canvas](#) > [Courses](#) > [Modules](#) > [Final Project Submission](#) to upload your capstone documents.
- Your Project Submission Document file format must be a PDF. We will not accept or open ZIP files.

If you do not have a Bottega Canvas account or need a refresher on how to access Canvas, email [fullstack@bottega.edu](mailto:fullstack@bottega.edu). Please do not send your final capstone documents directly to a TA or peer mentor.

**NOTE ON SUBMISSIONS (VASS STUDENTS ONLY):** The 2023-2024 VASS Group will not be using Canvas for project submission.

- Please email your project submission document to [fullstack@bottega.edu](mailto:fullstack@bottega.edu).
- Your project submission document must be in PDF format.
- [PROJECT SUBMISSION DOCUMENT TEMPLATE](#)

**Your final capstone documents will not be returned to you, so keep a copy for your records.** It takes approximately three to four business days to review your code and determine you have a functioning application. Do not be discouraged if we do not approve your project right away. When necessary, we will provide a description of what we need to see via email.

We recommend that you submit your application prior to your deadline. Errors can and often do happen, so don't leave this until the last minute.

Learn something new and do something well. This is your creative work, so have fun with it.