Capstone Framework Guide

Introduction

Congratulations on making it to the Capstone project! In our experience, about 75% of the learning obtained during this course happens while doing this project. It is your crowning achievement as a new software developer and the primary way you will show your skills to potential clients and employers.

The Capstone Approval Process

To get from where you are now to having a clearly defined and properly scoped project, there is a process we will need to go through. The steps of that process are listed below:

1. Read this document and all supporting documents.
2. **Do some initial research on 3rd party APIs** (Application Program Interface) you believe you will need for the project.
3. **Submit a Capstone Pitch.** The format is on Page 4 of this document. Hopefully, by now, you have thought of a handful of ideas that you are excited about and will show your passion as a developer!
4. While you are awaiting feedback on your pitch, **continue research on your 3rd party APIs**. Be prepared to answer the following questions about each of your API selections:
    a. Is it accessible? Are you able to obtain an API key for it if needed?
    b. Can you make requests to it in Postman and obtain the data you think you need?

c. Have you read the documentation for the API and understand how it is laid out?

5. Once the pitch is approved, **draft a set of user stories for the project.** Please refer to the **User Stories Criteria** section for expectations.
6. **Submit your draft of user stories**. While awaiting feedback from Instruction, start working on the following:
    a. **Entity Relationship Diagram** – while this will change, it is a good idea to get started and sketch out what you think the basic database design for your capstone will need to be.
    b. **Wireframe/Prototype** - your designs here do not need to be extensive, but should give you a visual reference for what you intend to make when you get to the front-end portion of your project.
    c. **Set up starter code** - Note that this will be the *same* starter code utilized in the BookNook Project. A handy reference guide with all setup steps, as well as the repository link, will be included in this week's chapters.
    d. Utilize a whiteboard or digital tool to start **mapping out the tasks** you will need to complete for the project and the timeline along which they will need to happen.

Bear in mind that your user stories may need some revisions based on instruction feedback, so be prepared for some back-and-forth communication with Instruction to clarify specific features in your user stories.

Once Instruction has approved your user stories, they will assign point values to each one based on importance to the overall project and how difficult they will be to achieve. Make sure to review these user stories, ask questions, raise concerns, and then let us know if you agree with how they look! Once you have done so, you are officially approved for your project and ready to dive in.

Required Tech Stack

React/ASP.NET/MySQL (using provided starter code)

Unless strictly approved by Head of Instruction

Additional Requirements

As this project will be your crowning technical achievement representing the culmination of your previous few months of learning to program, the majority of coding & problem solving must be your own. Attempting to follow a tutorial to build the bulk of the capstone project will result in a zero. If you have questions on what is/is not allowed in this respect, please ask before attempting to integrate a tutorial.

Capstone Pitch

Answer the questions below for your capstone concept pitch. **Copy and paste the question and answers** into a Slack message to the instructors in your group instructor channel. The instructors will either approve your concept in Slack and/or meet with you to talk further about your concept.

For a listing of free-to-use, public APIs, check out this link!
https://github.com/public-apis/public-apis

Give a brief overview of what your app is:

Describe the business problem your app solves:

What third-party APIs, IF ANY, do you plan to use? Please list the API and what feature it ties to:

What is your project's TECHNICAL wow factor? You need your project to stand out in some way. If you imagine presenting your project in an interview, what is the

feature of your project you will be the most excited to talk about? That is your wow factor:

User Stories Criteria

Each user story should be from the perspective of a specific kind of user. Remember that a user story defines a feature that your user will be able to use and represents a shippable product increment.

🔴 NOTE: There should be no “as a user” or “as a developer” user stories. You will need to define whom your user is using a specific noun. 🔴

For example, a travel application that allows users to sign up for accounts and matches those users with a random destination for their vacations could have a user called a traveler. This noun describes the kind of user that will use this application. A sample user story would look like this:

“As a traveler, I want to be able to add a destination to a list of favorite destinations, so that I can easily reference that list when I am deciding on where to go on my next trip.”

Additionally:

- **Do NOT write any user stories for styling or login/logout/registration features, as these will be added by instructors.**
- **Do NOT assign point values to your user stories, as this will be done by Instructors.**

Sample User Stories

These are sample user stories for a Trash Collector project. Notice there are no “As a user...” or “As a developer...” user stories. User stories must be from the perspective of the user who you have defined using a specific noun (traveler, sumo wrestler, photography enthusiast, etc) who is interacting with the application.

- As a customer, I want to be able to select or change my weekly pickup day.

- As a customer, I want to be able to request an extra, one-time pickup for a specific date of my choice.

- As a customer, I want an uncomplicated way to see my current balance so I can budget accordingly.

- As a customer, I want to be able to specify a START and END date to temporarily suspend my pickups.

- As an employee, I want to see a list of today’s customers who meet ALL the following criteria:
    - Customers in my zip code
    - Non-suspended accounts
    - Pickup day is today’s day of the week OR a one-time pickup date that falls on today

- As an employee, I want to be able to filter customers in my pickup area by a particular day of the week to see who gets a pickup for the day selected.

- As an employee, I want to be able to confirm that I have completed a pickup.

- As an employee, I want all confirmed pickups to have a charge applied to the customer.

- As an employee, I want to be able to select a customer profile and see their address with a pin on a map (Google Maps API, Google Geocoding API).

Capstone Presentation Expectations

Here are some expectations for your capstone in terms of preparing for presentations to instructors, companies, and potential employers.

Presentation Steps

1. Personal introduction, personal connection to project (show your passion!)
2. Who you are, what your background is, and why you are excited about this project
3. Introduce the Application: what it is and the business problem it solves.
    a. The business problem is the real-world challenge that this application addresses. Focus on your users and how their quality of life would be improved by using this application.
4. Explain Technologies Used
    a. This includes programming languages, libraries, frameworks, third-party APIs, etc.
5. Showcase/walkthrough of features (explain each feature as you walk through it!)
    a. Be sure to explain each feature as you walk through it - remember, your audience doesn't know this app as well as you do!
    b. If there is a particular piece of code you are excited about or proud of, make sure to point it out here! Remember that your audience could be a mix of technical and non-technical individuals, so stay focused on the working application itself!

6. Discuss a specific challenge you encountered in this project and how you were able to overcome it, and be prepared to discuss how you solved other specific problems
    a. Be willing to admit any items in the project which you couldn't complete or weren't able to solve.
    b. If you had to pivot for a specific reason, explain the reason to pivot & what the ultimate outcome was.
    c. Stay positive! Pretend that the audience does not know the scope of what you were aiming to accomplish for this project.
7. Wrap up, thank your audience, and open the floor to questions.
8. Be prepared to explain any challenges you faced while working on your application, and what you did to overcome those challenges.
9. Have fun & be yourself! 🙂

Presentation Key Points

- You need to show passion for the project you created. Be excited about your accomplishment!
- Have an introduction explaining your inspiration for the project
- Be sure to indicate which technologies and APIs you used for the project
- After presenting the features, be sure to open the floor for any questions or comments about the project
- Discuss a specific challenge and the solution you produced for a part of the project in every presentation

- If there is a part of the project where you were excited about the solution or the code you implemented, make sure to point it out. It is a way to show your passion for the project and Software Development.
- Share your plans for future sprints on the project (additional features you would like to add)

Database

⚠️ **NOTE: You do not need to formally overview your ERD during the presentation.**

- Be able to thoroughly explain the Entity Relationship Diagram for your application's database and why you designed the relationships in this manner
- Be able to explain the individual relationships between classes i.e. the use of foreign keys or junction tables.
- Be able to state whether you have normalized your database and why you chose to do so.
- Be able to explain any one-to-one, one-to-many, or many-to-many relationships your database implements

Challenges

- Be prepared to explain any challenges you faced while working on your application and what you did to overcome those challenges.
- Be prepared to explain what the most difficult part of the project was and why it was the most difficult.

- Be willing to admit any items in the project which you could not complete or were not able to solve. If you had to pivot for any specific reason, explain the reason to pivot and what the ultimate outcome was.
- When discussing these pivots, stay positive!

Solutions

- Be prepared to explain how exactly you solved specific problems within the project
- Explain exactly what real-world problem your project addresses and how your users will benefit from your software!