Capstone Requirements

- The app must solve a problem that requires user interaction

- Display an effective approach to the product planning process

- Including Research, SWOT, User Stories with Acceptance Criteria, Wireframes, UI/UX design, User Persona Board using Mural, Roadmaps and Scrum board, Sprints, Daily Standups, Backlog Grooming, Sprint Retrospectives and Sprint Planning

- Display an understanding of semantic HTML and modern CSS

- Demonstrate the use of Git, including a sensical, well-organized 'commit' history

- No one word commit messages

- Create a Single Page Application following the pattern of the class SPA

- Using 'state', a 'render' function and 'router' to display 'views'

- Create a minimum of 4 views with working navigation to each view

- Use a 3rd party API to send and/or receive data

- To request data and add JSON response to a view

- To send JSON data via a form to be stored

- Must use a proper formatted XHR request using the following:

- An API key for authentication

- An appropriate HTTP verb such as GET, POST, PUT or DELETE

- iFrames are NOT allowed as a substitute for using an REST API

- API calls consisting only of url reference paths are NOT sufficient to demonstrate professional API comprehension

- Using a library that is served similar to an API is NOT sufficient

- Examples: Mapquest.js Web SDK, FullCalendar.js, Leaflet.js

- Implement a RESTful API using Node, Express & MongoDB and integrate it into your SPA

- Deploy your SPA and API to Netlify and Render.com respectively

- Display your development skills and creativity in a professional manner

- The scope of the app should be able to be completed in the allotted time of the cohort, plan wisely. Think big, implement conservatively!

- NO authentication (Login page with username and password) to be implemented during the cohort

- NO payment processing to be implemented during the cohort

- The use of the HTML file (upload) input to enable user file upload is NOT to be implemented during the cohort

- Media files (images, logos) are allowed in the codebase within the assets/img folder but will require a code deployment to be updated

# Defining Your Product

**Group Ordering System**

What problem does it solve?

**Organized, accurate, efficient & repeatable for order placers. It will make the order placer’s job, as well as the restaurant’s job, easier.**

How will you incorporate the end user / customer into your development process?

**I will ask peers of mine, executive assistants, to review what I have created, seek feedback and update accordingly.**

When will you involve the user / customer into your development process?

**I will float ideas by peers as I begin to work on them.**

**I did this for my concept idea, too.**

Why is it important to incorporate your end user / customer?

**The end user will use or lose this app. It is meant to make their jobs easier.**

What is a technique for obtaining user feedback?

**Feedback technique will be talking, document and tweaking my plan.**

# Determine what YOUR product will be

Create a high level story you want to tell about your product

**Menu placing order that specializes in streamlining group orders.**

- This will be your elevator pitch for your capstone project and used during demo day.

\*\*"I want to create a web APPLICATION that allows users to ask questions via chat during a talk and add resources to videos."\*\*

Discuss with potential end users of your application as to what they would like to have or see in your application:

- Acceptance Criteria

# Define your Audience

\*\*Who is the target audience for your product?\*\* \*It's more than just you!\*

**Assistants, Clerical Staff, Support Staff, Coordinators, Event Planners, Parents.**

**Mainly females.**

- Mentor & Peers, Friends & Family

- End users, customers that will be using your app for a product or service

How is your idea different, better, define your audience and run your ideas by them to get feedback on what they would like to see in an application? Document their feedback and keep in mind / refer to as you are building out your application.

# Define your Content

\*\*Determine what pages you need and the content of each page\*\*

\*Don't worry about the look and feel or style of the content, that comes later.\*

1. Make a list of pages (minimum of 4-5)

1. Write a brief description of the content for each page

1. Navigation

1. Main Content

1. Footer

# User Story Mapping

![](media/user-story-mapping.png)

<!--

Use the User Story Map Template to create your own

Select Savvy FS User Story Map / New Mural from Template and name it something relative to your capstone

-->

# Basic UI/UX Design

\*Designing your Web Application Interactions\*

## User Experience (UX)

- Keep it INTUITIVE

- Navigation should be obvious

- Make it consistent

- Follow the "standard" for interactions a.k.a clicking a hamburger icon should open the navigation and chevrons should open drawers.

- Keep interactions the same across the application

- Allow users to make errors by handlers errors gracefully aka: form validation that asks the user to correct inputs that are not correct

# UI/UX Resources

- [Fundamental Concepts UX](https://usabilla.com/blog/fundamental-concepts-ux/)

- [5 key principles for new UX designers](https://careerfoundry.com/en/blog/ux-design/5-key-principles-for-new-ux-designers/)

- [Interaction Design Principles](https://www.mockplus.com/blog/post/interaction-design-principles)

# Design your Web Application Interface

## User Interface (UI)

Responsive Design or Mobile First layout

- Rearrange (stack) elements on narrow screens

- Hide/Show different elements based on screen size

- Adjust font sizes, bigger on smaller screens

\*Resources\*

- [Mobile First versus Responsive Web Design](https://darwindigital.com/mobile-first-versus-responsive-web-design/)

- [Mobile First is just not good enough: Meet journey driven design](https://www.smashingmagazine.com/2017/02/mobile-first-is-just-not-good-enough-meet-journey-driven-design/)

# Define your User Flow

### What interactions will the user be required to do to use your web application

![h:320](media/user-flow.png)

- Focus on page flow not content of pages

- Make sure you analyze all user flows, creating multiple flow charts if necessary

# Create Wire Frames

## Now that you think you know what you want to create

The wireframes should be crude. We are not creating pixel perfect representations of your website.

- Start with pencil and paper or whiteboard, then if desired, move into a digital version.

- Create at least one per page.

- If a page has suffocated user interaction, multiple wireframes for a single page may be required.

- Create wireframes for both different screen sizes. I.E.: mobile and desktop.

- They should be proportionally correct.

- They should identify the content to be placed on each page but not display actual content.

- They should annotate any user interactions.

Start with pencil and paper or whiteboard, then if necessary create a digital version

---

![bg h:98%](media/rough-mockup.jpg)

![bg h:98%](media/mockup.png)

# User Flow and Wireframing Tools

Analog Tools:

- Pencil & Paper

- Whiteboard & Dry Erase Markers

Digital Tools:

- Draw.io ([draw.io](https://www.draw.io/)) <=- Recommend

- Mural: ([mural.co](www.mural.co))

- Excalidraw: ([excalidraw.com](https://excalidraw.com/))

- Inkscape ([inkscape.org](https://inkscape.org/))

- Mockflow ([mockflow.com](https://mockflow.com/))

# Fonts & Icons

- Find alternate fonts for logos and headers to make them stand out.

- Use icons only where appropriate.

### Resources

- [Google Fonts](https://fonts.google.com/)

- [Material Icons](https://material.io/tools/icons)

- [Font Awesome](https://fontawesome.com/)

# Images

Due to copyright law, images need to be your own originals, or licensed Creative Commons or Public Domain

### Resources

[Pexels](https://www.pexels.com/)

[Unsplash](https://unsplash.com/creative-commons-images)

[Pixabay](https://pixabay.com/images/search/profile/)

<style scoped>

.heading {

font-weight: var(--base-text-weight-semibold,600);

color: var(--h1-color);

font-size: 1.6em;

line-height: 1.25;

margin-bottom: 16px;

}

.red-text {

color: darkred;

}

.contrast-text {

background: darkslategray;

color: darkgray;

padding: .1em;

}

</style>

<div class="heading"><span class="red-text">Color</span> & <span class="contrast-text">Contrast</span></div>

- Pick a color palette and stick to it

- Select a palette that includes a max of 3 colors

- [Coolors](https://coolors.co/)

- [Paletton](http://paletton.com/)

- [Colorspire](https://www.colorspire.com/)

- Keep accessibility in mind

- Visually impaired & colorblind

- Review your website as if you are colorblind

- [Color Blind Filter](https://www.toptal.com/designers/colorfilter/)

- Review your accessibility score

- [WAVE Web Accessibility Evaluation Tool](https://wave.webaim.org/)

![bg contain](media/swot.png)

# Project Planning Homework

<!-- [Week 1 Homework](1-Homework.md) -->

- Create Definition Statement (I want to build an application that…)

- Define Your Audience / Who is going to be using your application

- Define your content (color scheme, fonts, icons, images, logo, what you want on each page of your application.

- [Mural Board - Complete User Persona](https://app.mural.co/template/c5cae3de-aeb4-487f-86da-35aa333d11f6/9ee833d5-2629-40ea-a239-6308d3a60980)

- Research: Look at other applications similar to what you want to do and document what you like / don’t like so you can make your application different

- Create wireframes

- One wireframe for each page of your application, min of 5 pages

- Create User Flow Chart

- SWOT Analysis

## This is due by the end of day (EOD) Sunday before week 4 begins.