**CAPSTONE Project – code name – Prefect-print**

3D Printing Plastic Guide –

3D printing plastic guide page –

* Page to access and show stored database information about printing profiles for plastics
* Allows modifications to stored plastic information or new additions to be submitted
* Table formatting for display including averages and sections
* Buttons and drop downs to select desired action (add, delete, modify)

Features:

It will pull a database and display it

Allow modifications to current database

Allow new submits to database

Hidden buttons and boxes with rollover functions

Database will need –

Brand name

Color

Temperature low

Temperature high

Suggested Temp

Type

(added others and a second table for temps)

Animated gradient effect:

<https://alvarotrigo.com/blog/10-cool-css-animations-to-add-to-your-website/>

Rotate 3D object based on mouse position:

<https://armandocanals.com/posts/CSS-transform-rotating-a-3D-object-perspective-based-on-mouse-position.html>

<https://greensock.com/forums/topic/22796-3d-cube-animation-based-on-mouse-cursor-position/>

<https://3dtransforms.desandro.com/cube>

<https://speckyboy.com/3d-css-javascript/>

**Planning**

* project includes a wireframe for each view
* project includes a list of MVP features \*
* project includes a data model

**MVP**

* app has at least 3 main features \*
* front end makes a request to the server and handles the response \*
* front end is interactive \*
* app has custom styling \*

**Front End**

* app has at least 5 semantic tags \*
* app includes 1 view \*
* app includes 1+ additional view(s)
* styling includes flexbox
* at least 1 view is responsive
* styling includes animations

**Server**

* app includes a GET endpoint and handler function \*
* app includes a POST endpoint and handler function \*
* app includes a PUT endpoint and handler function
* app includes a DELETE endpoint and handler function
* app utilizes Sequelize
* project includes at least 1 controller file

**Database**

* project includes a seed file or function
* app uses 1 table
* app uses 1+ additional tables
* app uses a foreign key and join

**Presentation**

* discusses project purpose and demonstrates MVP \*
* does not discuss broken/unimplemented features
* recording is between 2-3 minutes

## Instructions

### Planning

It’s difficult to plan a software project the first few times. Your plans will not be perfect, and that’s okay. Try to make your plan as comprehensive as you can – it will help you throughout the process. And don’t be afraid to update your plan as you go.

1. Start out brainstorming - look through projects we’ve done and read through the ideas at the bottom of this page.
2. List out your 3+ MVP (minimum viable product) features. Name your project.
3. Wireframe your page(s).
4. Make a to do list for coding. It’s okay if this isn’t comprehensive, just try to give yourself a little structure.
5. When you have your features, app name, wireframe(s), and to do list, pass off your plan with a staff member.
6. Make adjustments if needed.
7. Set up a folder on your computer and a remote repo on GitHub. Connect them. Upload your completed planning documentation to your repo.

### Coding

You can choose how you want to do this, but here’s a suggested work flow if you’re feeling stuck:

1. Start by creating HTML, CSS, and JS files and linking them together for the front end.
2. Set up a basic server for your back end.
3. Build the HTML for your first feature.
4. Write the JavaScript for your first feature (front and back).
5. Add in CSS for that section of the page.
6. Repeat the process for your other 2 features.
7. Polish the layout and styling of the page.
8. Add any additional features.

## Ideas & Examples

### Ideas

* Choose a previous lab exercise from Foundations to build on.
* Use a previous lab as inspiration and create a similar project.
* Come up with your own idea!

### Examples

Magic 8 Ball. Your 3 features might be:

* Users can enter questions
* Users get randomized answers to their questions
* Users can save responses (until you refresh the page)

Trip List. Your 3 features might be:

* User can add a location to a “Want to Visit” or a “Visited” list
* User can delete items from their lists
* User can view one or both lists

Calculator. Your 3 features might be:

* User can add, subtract, multiply, or divide numbers
* User can continue calculation on a given answer (press 2 + 3 = and then can do \* 10 to the 5 that’s returned)
* User can save amounts (for budgeting or figuring out complex problems)