# JavaScript Project Requirements

## Project Deadline: Last Friday of the class, July 31st by 11:00am

There will be a simpleform provided approximately 2 weeks prior to the due date that you will complete to “turn-in” your project.

**Late projects will not be accepted**

The purpose of the capstone project is to reinforce what you’ve learned and show off your skills. Your projects will be your go-to tool to show off to potential employers and demonstrate you know what you’re talking about. Pick an idea that is fun, challenging, and hopefully useful, too!

Create a website that demonstrates your knowledge of what you’ve learned in the JavaScript class. This application can be anything of your choosing **as long as it demonstrates the below requirements.**

### **Requirements**

The below items are the requirements for your personal project for the JavaScript class. Projects are graded on a pass/fail system, thus every required bullet must be met in order for your project to pass. You should discuss your project/plan with mentors in advance to verify that it will meet the requirements, but also to make sure you don't tackle too large of a project for this class.

**Failure to meet all requirements will result in you not completing the class, and you will be unable to continue with the program.** Program staff and mentors will review projects after they are turned in and will determine if your project meets these requirements.

**Requirements:**

##### Choose a **minimum of 3** of the below features and incorporate into your site

* Read and parse an external file (such as JSON or CSV) into your application and display some data from that in your app
* Retrieve data from an external API and display data in your app (such as with fetch() or with AJAX)
* Create a form and save the response (on click of Submit button) to an external file or API
* Create a dictionary or list, populate it with several values, retrieve at least one value, and use it in your program
* Create and use a function that accepts two or more values, calculates or determines a new value based on those inputs, and returns a value
* Implement a log that records errors, invalid inputs, or other important events and writes them to a text file
* Implement a regular expression (regex) to ensure a field either a phone number or an email address is always stored and displayed in the same format
* Create 3 or more unit tests for your application
* Build a conversion tool that converts user input to another type and displays it (ex: converts cups to grams)
* Calculate and display data based on an external factor (ex: get the current date, and display how many days remaining until some event)
* Analyze text and display information about it (ex: how many words in a paragraph)
* Visualize data in a graph, chart, or other visual representation of data
* Other features can be added to this list - just ask if your project needs something specific and as long as it’s a good demonstration of your programming skills, it almost certainly will count! Basically, we just want to see you do something interesting and challenging!

##### **Additional Requirements**

ALL of the below requirements must be met.

##### Your code have comments that document major sections of your code to make it easier to read

* + You don’t need to go crazy on this - a few code comments are perfectly fine

##### Your project code is uploaded to your GitHub account, in its own repository, with **at least 5 commits**

* + We need to see that you’ve used Git to update your GitHub profile at least 5 times

##### It must include a README file located at the top level directory of your project that includes:

* + A description of your project
  + What features you chose to included (so we know what to look for)
  + Any special instructions we might need to run your project

### Clarifications and Commonly Asked Questions

* How projects are reviewed
  + The project reviewer will use git to clone your project to their local machine and follow the directions listed in your readme file
  + They will look for the features you documented in your readme, verify they work properly and display an understanding of the code
  + If all requirements are met, your project passes
  + We are looking to ensure you understand what you wrote, so a poorly implemented project may still not pass if we believe you do not understand the code
* Naming your project
  + Choose a name for your GitHub repo that is relevant to the subject of your project. Do not name your repo "CodeLouisvilleProject" or similar. Choose a name based on what your project is about. For example “Brian’s House of Pancakes”, “Recipe conversion calculator”, etc
* GitHub
  + Yes, GitHub is a requirement. It’s not just how we find and view your project, it’s a critical skill you need to understand. Not having your project checked in on GitHub will result in not completing the class.
  + 5 commits is a minimum to show you’ve made multiple updates. Hopefully you have dozens of commits!

### Testing Your Project

You should test your project on another computer by having someone else obtain and run your project. Several projects in the past have failed to meet requirements because it was written in a way that only worked on that person's computer and not the reviewer's. Your project reviewer will not be responsible for tracking down why your project does not work and this may result in a failed project.

### Mentors

You should talk about your project early and often with your mentors. Explain your idea and the features you are thinking of implementing. They will help you understand if it will meet the requirements or if you've possibly decided to tackle too large of a project for the 12-week time period. Towards the end of the session, you should again show your project to your mentors and get confirmation that it meets the requirements of the project. You are perfectly fine to ask them to confirm so there should be no surprises about whether your project will meet the requirements when it comes time to submit.