



Linting and Projects

Web Development Boot Camp
Lesson 16.1



Class Objectives

In today's class, we will cover the following:



Creating a project repo and adding members as collaborators.



Setting up a project with linting, using Travis CI.



Submitting your project proposal (ideas, outlines, APIs) by end of class for feedback.

Project 2



As we enter Project 2, we're going to start thinking a little more about **code quality**. Today we'll cover using a JavaScript linter and Travis CI checks to achieve this.

Project 2 vs. Project 1

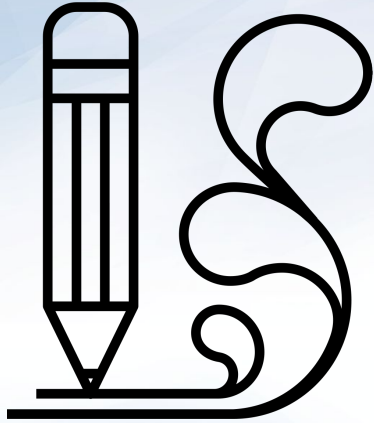
Unlike Project 1, in Project 2 you will be provided with some boilerplate code to jumpstart your applications. This will save some time and ensure that you can dive into building features for your apps.



Don't Worry!

You'll have plenty of time to brainstorm project ideas before submitting your proposals for review by the end of class.





Project 2:

To build a great project!

Project requirements sent via Slack.

Due Date:



Project 2: Coding Requirements



Must use a Node and Express web server



Must be backed by a MySQL database with a Sequelize ORM



Must have both GET and POST routes for retrieving and adding new data



Must be deployed using Heroku (with data)



Must utilize at least one new library, package, or technology that we haven't discussed



Must have a polished front end/UI



Must have a folder structure that meets the MVC paradigm



Must meet good-quality coding standards (indentation, scoping, naming, etc.)



Must protect API keys in Node with environment variables

Project 2: Not Required, But Nice to Have



Utilize Handlebars for server-side templating



Incorporate authentication (JSON Web Tokens, sessions, etc.)



Use an existing public dataset to power the database



Create a migration strategy for sharing data across team members, e.g., a seed/schema.sql file

Presentation Requirements

Your formal, 10-minute presentation must include:



Your application's overall concept



The motivation for your application's development



Your design process



Technologies used (and a brief description of how they work)



A demonstration of the application's functionality



Directions for future development



Sometimes, talking intelligently about tech > doing tech.

Metrics

Metrics

Project 2 will be assessed on the following:

01

Concept

02

Design

03

Functionality

04

Collaboration

05

Presentation

Linting



What is a JavaScript **linter**?
Why would we would use one?

JavaScript Linter

01

What is a JavaScript linter?

A **linter** is a tool that we can use to analyze our code for potential errors as we're writing it. Linters can also be used to enforce particular styles of coding based on rules we can turn on or off.

02

Why would we use a linter?

Linters help teams write more consistent and readable code according to what they consider best practices. Linters also help identify potential errors before code is even run, thus improving overall code quality.



There are several libraries available for linting JavaScript code, but ESLint, the one we'll be working with, is the most popular: eslint.org



Questions?



Partner Activity: No Lint Example

In this activity, you will examine a JavaScript file without ESLint and attempt to identify issues with the code.

Instructions sent via Slack.

Suggested Time:
5 minutes





Time's Up! Let's Review.

Project 2 Starter



Instructor Demonstration

Pre-built Boilerplate Code



Group Activity:

Project 2 Starter

In this activity, you will add the starter code to your project repo in GitHub.

Instructions sent via Slack.

Suggested Time:
10 minutes



Travis CI (Continuous Integration)

What Is Travis CI?

In the next activity, we will set up Travis CI.

01

Travis prevents any code from being merged into master that doesn't pass the linter.

02

Travis runs a check on any branches with pull requests to the master branch. You will only be allowed to merge the code if the lint passes.



Activity: Travis CI Setup

In this activity, you will add Travis CI to your project.

Instructions sent via Slack.

Suggested Time:
10 minutes



Take a Break!





Group Challenge:

Project Brainstorming

Work with your group to identify ideas, research APIs, and create project designs.

You will submit your initial project proposals by the end of class today.

Suggested Time:
69 minutes





Questions?