←

Project #2 Guidelines: Full-stack Web Application

PROJECT

Find below the general guidelines. Please check in with your instructor regarding any additional requirements. Good luck!

Introduction

Now that you are more familiar with the backend, you can make a complete Web application for your 2nd project. You'll be building an **Express** app, which means you'll learn/practice what it takes to make a functional application from the ground up yourself.

We suggest that you **do this project in pairs**, but please check in with your teacher.

We hope you'll exercise creativity on this project, sketch some wireframes before you start, and write user stories to define what your users will want to do with the app.

Make sure you have time to run these ideas by your teacher and TAS to get their feedback before you dive too deep into code!

Remember to keep things small and focus on mastering the fundamentals – scope creep/feature creep is the biggest pitfall for any project!

Examples

Here are some examples of past projects created by the students:

M2 Project | Something's Cooking

M2 project example - Something's Cooking

M2 Project | Horizons

M2 project example - Horizons

▶ **See more examples**

Schedule

Full-time:

- Stand-ups*
- Coding Kata
- Project development

Part-time:

- **Week Nights**
    - Stand-ups*
    - Project development
- **Saturdays**
    - Stand-ups*
    - Coding Kata
    - Project development

**Every day, during project weeks, you will start with this activity. Take a minute to update your colleagues with your progress: what did you do so far, do you feel on the track, behind or ahead, what's your plan for today and do you think you'll have any blockers.*

For both: two days before presentation day

Presentation Practice: About You & Your Project + Demo. Refer to the Presentation Section to see how to create your presentation.

Presentation day

Finally, it's time for your presentation. You will present your projects to the staff and other students. Remember to dress at least one degree nicer than normal. Check out the Demo Tips.

Technical Requirements

Your app must:

1. **Use Express** as a foundation.
2. **Use Mongoose** for models and database communication.
3. **Have 2 models *or more***. Having one for users is a no-brainer. The other one(s) should represent the main functionality of your app. Don't force it if having more than two models doesn't make sense.
4. **Have validation on the models** with feedback for users if their submission is invalid.

5. **Include sign up, log in & log out functionality** with encrypted passwords (and/or social logins) and authorization (logged in users can do extra things).
6. **Implement all CRUD actions** on models other than users. You should have the Create, Read, Update and Delete features even if they aren't all for the same model.
7. **Have a repo on GitHub**.
8. Have at least **1 commit per day** that you worked on.
9. **Be deployed online** using Heroku so that anybody could use your app.

Responsive design **is not a requirement, but it's nice to have**.

Deliverables

- A **working full-stack application, built by you** that runs on a server.
- A **working app deployed** on Heroku.
- The URL of the **GitHub repository** for your app.
- The URL of the **live app on the Internet**.
- The URL of the **slides for your app's presentation**.
- You must present your app during Project #2 final presentations (last day of Project #2 time).

Presentations

Project #2 presentation will focus mostly on your project. You **won't** have to talk much about yourselves or your background. You should still **introduce yourselves briefly**.

Format

- Talking with Slides: **3 minutes**
- Demo: **2 minutes**
- Total: **5 minutes**

Attire

- Dress nicely for this and all final project presentations (last day of each project's time).
- Dress at least **one degree more elegantly** than you usually dress for class.
- *Examples*:
  - If you wear t-shirts every day, wear a button-down shirt.
  - If you wear jeans every day, wear some slacks.

- If you wear sneakers every day, wear nicer shoes.

Slide Applications

- All presentations will be done **from a staff member's computer**, so **your slides need to be online**.
- PowerPoint files, Keynote files or files of any kind **will not be accepted**.
- Suggested online slide applications:
  - Slides
  - Prezi
  - Google Slides

Presentation Structure

1. **Title Slide** (1 slide): your project's name & your name
2. **Project Elevator Pitch** (1-2 slides):
   - What is your project?
   - How does it work?
   - Why did you choose it?
3. **Technical Challenge** (1-2 slides):
   - What was **the most important** technical challenge you faced?
   - How did you overcome that challenge?
4. **Big Mistake** (1-2 slides):
   - What was **the biggest** mistake you made during this project?
   - What did you learn from it?
5. **Demo Slide** (1 slide): literally says "DEMO" with **a link** to your project so you can open it easily
6. **Closing Slide** (1 slide): your project's name, your name & a *"Thank You"*
7. **Total**: 6-9 slides

Presentation Structure Notes

- **Don't** include a slide just for the technologies.
- **Don't** include any code in your slides. Nobody will read it.
- **Don't** include a slide for GitHub graphs.

- **Don't** go into detail about how the app works. Your demo is where you want to do that.
- If you think that deviating from the structure improves your presentation, feel free to do so. This suggested structure is mostly for people who don't know what to do.

Demo Tips

1. Plan what you are going to demo and **practice it on the live site**. That way you won't be surprised if something breaks on the live version.
2. **Deploy early** so you can squash bugs. There are *always* bugs on the live site at first.
3. Add **link to your live project** to your DEMO slide so you can start it smoothly.
4. Your app's colors and sizing **might look different on the projector**. If you think it might be a problem, ask to test it beforehand.
5. **Refresh it before you present**, otherwise, there will be a delay in the initial load.

Project Feedback and Evaluation

- **Technical Requirements**: Did you deliver a project that met all the technical requirements?
- **Creativity**: Did you add a personal spin or creative element to your project?
- **Code Quality**: Did you follow code style guidance and best practices?
- **Deployment**: Did you deploy your application to a public url using Heroku/some other platform?
- **Total**: Your instructors will give you a total score on your project between:

Score	Expectations
0	*Didn't meet expectations*
1	*Met expectations, good job!*
2	*Exceeded expectations, you magnificent creature, you!*

You should use the feedback to focus on what needs to be improved for the next project.

Good luck! ❤️

Mark as completed

←

PREVIOUS

Project planning session

NEXT

VCS | Collaboration with GitHub