Project Milestones 1-6

**Due dates:** see course schedule for dates

Every week the project milestone is a check-in with the professor, where you will also be ensuring your documentation and schedule is updated, and everyone knows what they are working on.

Except for the Milestone 4: beta release, these milestones are not graded but designed to help you stay on track with your project.

**Tasks:**

**For the weekly milestones, you will want to create/update the following things:**

- In your README file, create a set of instructions that explain how to setup the project for any new developer. Explain how to build and test the system. This should include at minimum:
  - Instructions for checking out the source code for the latest stable or released version.
  - How to build the software.
  - How to test the software: describe the steps needed for a developer to get a test development environment working, such as running the website on their local machine (you could alternatively make a test server that everyone can use). Be sure to include information on setting up any databases or other necessary services.
  - For all of these instructions, state the operating system(s) that the instructions are intended to work on. You should have instructions for all the operating systems that your team will be using.

- Update your project’s schedule for the remainder of the semester. Clearly label in the schedule items that are completed, as well as any adjustments to future items that you decide to make.
  - If you have modified the features or tasks you are working on, make sure this is clearly noted in the schedule and describe the changes. Any substantial changes to features should be discussed with the instructor.

- Make sure your API Design documentation is updated if you made any changes – I expect this document to be a complete representation of your backend API at the final project release.

- For your project you will need a system to track bugs (I recommend GitHub Issues, though you can use some alternative method). In the README, provide instructions for where to check out the list of outstanding bugs and how a user can report a bug.
  - For all of the features that are marked as completed on the schedule, have someone on the team who did **not** implement that feature go through and verify
the feature is working as expected. Report any bugs or issues you find in your bug tracking system.
- For all serious bugs, make sure someone is assigned to fix the bug and that bug fixing time is accounted for in the schedule.

Notes:
- If you are using a bug tracking software outside of GitHub, I will need access to the system. You can setup access to my email address, or if that is not possible, provide me an account to use and tell me the username and password.
- If your project’s problem statement has changed over time, be sure to update the GitHub README with the most current problem statement and description of what your project will do. This can be taken directly from your Design Document, or modified as you see appropriate.

In-Class Meeting:

Each week in class, the instructor will meet with your group. Be prepared to give a brief demo in-class of your progress to the instructor.

During the instructor meeting, each member will also state briefly:
- What did you do in the last week?
- What do you plan to do this week?
- Anything keeping you from working / blocking you?

Milestone 4: Beta Release Instructions:

The fourth milestone is a milestone to show off your progress so far with a beta release and a 5-10 minute in-class presentation to show off your project so far. In the presentation you should show a demo of your project, describe the biggest difficulties you’ve faced so far, and your plans for the final release.

For this milestone, your project should be deployed online and accessible via a link. As stated before, given we are more than halfway through the development time I expect most of the core use cases to be completed, though not necessarily in their final polished form.

Along with the release and presentation, your team should also create a “Milestone 4 Progress Update” document. It should contain:
1. **Individual Progress Update**: each team member should write a brief progress update for themselves and add them together into this 1 document. This update should include:
   - A list of all the items that person was individually scheduled to complete by this milestone (from the schedule in the design document) and scheduled to be in-progress on by this milestone (label the in-progress tasks clearly)
   - A list of what they actually completed. For partially completed tasks, label these separately and briefly describe what is done and estimate what percent complete the task is. Note: if a feature is marked as complete, the code should be committed into GitHub and integrated into your main development branch (e.g., on “master”).

2. **Group Progress Update**: as a team, review the original schedule from the design document and compare all the use cases and features you expected to be built by this milestone against what is currently finished (everything should be accounted for in everyone’s individual progress update – if not go back and revise that). For this section:
   - Describe briefly whether you feel your progress is on track, ahead of schedule, or behind schedule.
   - **Give your team a grade for your progress so far and briefly state why you assigned this grade (“We feel our overall progress so far is a B because…”).**
   - If you are behind schedule, discuss with the responsible individuals for each component what caused the schedule delay and create a plan for any adjustments you’ll make to your work strategy and schedule going forward. In this document, briefly write up any issues you are facing and how you’ll adjust your work process and schedule going forward.

**How to Submit Milestone 4:**

By the due date, have all the required documents created / updated and committed into your GitHub repository. Have one member email the instructor the link to the “Milestone 4 Progress Update” document, which should be committed into your GitHub repository.

During the December 1 class period, you will give a 5-10 minute presentation showing off your current progress, the biggest difficulties so far, and your plans for the final release.
Grading

Note that the milestones are not graded (except for Milestone 4). However, if the requirements above are not updated you can lose points on your Milestone 4 beta release, and your final project release.

Milestone 4 Grading (60 possible points)

For this assignment, you will receive individual grades based on the progress you have made so far.

50 points for the in-class presentation and the progress each individual member and the team has made up to this point.
- You can lose up to 20 points if your website is not fully deployed online by this assignment.

10 points for completing the following:
1. Individual Progress Update – note that each group member must have their individual progress update in this document in order for that member to receive any points for the assignment.
2. Group Progress Update
3. Keeping your README, schedule, and other documentation up to date.