Milestone 1 Checklist

- ❑ Demo video
- ❑ Who-did-what document for the demo video
- ❑ Version of your code from the demo video tagged in GitHub
- ❑ Up-to-date versions of your user stories, design artifacts, and planning artifacts
- ❑ Individual assignments and outcomes
- ❑ Instructions to the instructors, submitted to eCourseware dropbox
- ❑ Teammate evaluations

Milestone 1 Instructions

There are three main deliverables for Milestone 1: a demo video, a collection of project artifacts, and a live in-class demo session.

1. Demo Video

Your team will be responsible for creating a demo video of your software. This video is mainly to assist the course instructors in grading your progress on the project. The video must also have an accompanying who-did-what document that lists who built each of the demoed features. The demo video and document must meet the following grading criteria:

- **Criterion: Demonstrate the progress that the team has made so far.**
    - **All the new features**. Include all the latest features in the demo. Don't leave any out. A big point of this exercise is to demonstrate all the wonderful progress that the team has made. Note that this criterion does not mean that you should skip re-demoing old features. It just means you shouldn't skip the new ones.
    - **Backend too**. Although UI features are a high priority, you may also demonstrate that backend functionality is working, even if it's not yet connected to the frontend. The key thing is to prove that the code runs and works! Along those lines, you may demo automated tests.
- **Criterion: Make clear who contributed what to the project in the *who-did-what document.***
    - **Who made it?** In the text document, give an entry for each feature demoed. Each entry must include the name of the feature, the time offset in the video where the feature was demoed, and the U of M username(s) of who built the feature. If multiple people contributed to a feature, say who did what. Don't forget to credit anyone. Be sure not to make any factual errors.
    - **New or old?** For each entry in the who-did-what document, list whether the feature is new (done in the most recent iteration) or old. If part of the feature is new and part is old, give clarification of this.
- **Criterion: Display the team's work in the best possible light.**
    - **Story form**. Any demonstration of UI must take place in the context of a cohesive story. That is, the presenter must describe one or more characters (with names, like Alice and/or Bob) and relate a story about the character using the software. The presenter must stick to this story format. The story and accompanying demo must be well thought out, and not leave the audience with the impression that the presenter is making it up as he/she goes along. Use realistic names for things and not made-up placeholders, like "foo" and "slafjsd".
    - **UI first**. Since the UI is generally most interesting, you should lead with that.

    - **General audience**. Don't forget that not everyone is as familiar with your project as you are. To be on the safe side, explain it as if you are talking to someone who has never seen it before.

    - **No special effects or fancy editing**. The video should clearly show a user (or users) interacting with your web app. Don't add special effects or sound effects, which distract or detract from the authenticity of the interaction.

- **Criterion: Length and format constraints.**

    - **Time limit**. The video must be no more than 10 minutes long.

    - **Fill the time**. Your video should be at least 8 minutes; otherwise, you're probably doing it wrong.

    - **Video format**. The video must either be shared via YouTube.

Note that the creators of the demo video and accompanying document are eligible for A&B points.

2. Project Artifacts

For Milestone 1, you will submit the following artifacts:

1. a copy of your code (tagged in GitHub), and
2. up-to-date versions of your Milestone 0 artifacts (i.e., USs, sitemap, UI sketches, and model-class diagram).

The artifacts should satisfy the following grading criteria:

- **Tagged submission in GitHub**. To grade your code, I will clone your team's GitHub repo, and checkout a tagged version of your code. You must tag your code as "milestone1".

- **Code builds and runs**. I should be able to build and run your code using the usual approach from the Boot Camp projects. If any special instructions are required to build/run your software, include them in the README file in your project's top-level directory.

- **Replicable demo**. I should be able to replicate your demo video. If seed data is required to do so, you must somehow make that data available to me (possibly giving instructions in the README).

- **Artifact quality**. All your artifacts must be of high quality. The criteria from the Initial Planning Milestone still apply, with the following addition.

- **Code quality**. Your code must follow common style guidelines and be well organized and readable. For example, all code must be properly indented, and class/variable/method names must be sensible. You should also do your utmost to avoid bugs and other sloppiness.

- **Customer satisfaction**. Your customer will provide feedback on how well your team has satisfied the requirements he/she gave you and how well aligned your team's prioritization of the work has been with the customer's priorities.

Note that there is an A&B eligible role (Quality Assurance Czar) with special responsibilities regarding milestone artifact quality.

3. Live In-Class Demo Session

For this session, each team will operate a demo booth. One member of your team (the "demo-booth operator") must run the booth, providing visitors with an interactive demo of your team's software. The remaining members of your team will circulate about the other booths, acting as visitors. The interactive demo must meet the following grading criteria:

- **Clearly explain your project to visitors**. Assume that visitors have never seen your project before. Thoroughly and clearly explain what problem your project solves and how it does so.
- **Display the team's work in the best possible light**. Use presentation techniques discussed this semester to present your team's software in an engaging and compelling way. Also, think about the best way to set up your booth. What equipment will you need? Extra monitors?
- **Allow visitors to use your project**. This is an interactive demo, which mean that visitors should be allowed to try out your project.
- **Time limits**.
    - **Don't go too long**. The demo must be no more than 10 minutes long. Be sure to allow enough time for users to "play" with your system.
    - **Fill the time**. Keep your visitors engaged throughout the 10 minutes.

Note that demo-booth operator is an A&B eligible role.

4. Individual Assignments

Full task planning and outcome reporting must be part of each iteration and milestone. See the *Individual Assignment Specification* document for detailed instructions.

5. Instructions to the Instructors

Since the way to access the various milestone artifacts may vary from team to team, each team must submit a document containing instructions to the instructors regarding how to access each artifact. This document must be submitted to an eCourseware dropbox by the milestone deadline.

6. Teammate Evaluations

At the end of each iteration, each team member must provide an evaluation of each other team member. Instructions and forms for performing these teammate evaluations will be communicated by email near the end of the iteration.