6.170 Final Project Team Contract

Sherwin Wu, Danny Chiao, Rebecca Odim, Ibrahim Abdussabur

Team Member Achievement and Effort
We expect each team member to put their full feasible effort into this project, with an approximation of around 50 hours of work total for each team member. We expect each team member to code efficiently and design-minded, as well as being punctual and compromising with the whole group. The end result is that our project will be better than each team member's individual projects.

Personal Goals
All: Learn how to work with team dynamics for design and code decisions, learn new web technologies specific to this project (especially MIT specific technologies)
Sherwin Wu: Learn about user interface implementations and considerations.
Danny Chiao: To be able to make something that is useful and looks good.
Rebecca Odim: Learn about simpler ways to do things that may not be obvious (such as Ruby gems and existing libraries).
Ibrahim Abdussabur: Complete a larger scale group project in rails, and learn more Javascript.

Team Correspondence
We will contact each other through our moira mailing list: eventplus@mit.edu

Team Meetings
We will meet twice a week:

- Mondays from 1pm to 3pm
- Thursdays from 1pm to 2pm

We will meet in the Student Center 5th floor Athena cluster group conference rooms. If a team member is late to a meeting, he/she will bring snacks for everyone else at the next meeting. If a team member cannot make a meeting, he/she will notify the entire group at least 24 hours in advance.

Code Version Control
Each team member will pull from our main development branch and will push to the same development branch as well. There will be no branching off from the main branch. If there are conflicts when pushing, we will manually resolve these.

Work Quality Maintenance
The quality of work in our team will be maintained through a system of code reviews. After each major push to our main branch, a team member will email out to the group mailing list, and everyone will look over the new code. At one of our meetings, we will bring up any qualms we may have with someone else's code and work it out.

Task Assignment
At each of our physical meetings, we will divide up tasks for the next 2-4 days. We will suggest partitions and will only finalize the assignments if everyone agrees on it. This also means that if one team member has a lot of other work for a week, we will try to let the other team members take some of his/her work.

Deadline Enforcement
Deadlines will be strictly enforced. Our deadlines will always be at midnight the day before our physical meetings on Mondays and Thursdays. If a group member fails to meet a deadline, we will talk about why that happened in the meeting in front of everyone else, and that will affect how we divide up next tasks at the meeting.

If a group member repeatedly misses deadlines to the point that it is impeding our group's overall progress, we will talk to our staff member, Dalton Hubble, to discuss what we can do about the team member in fault.

Decision Making
We will strive to only make decisions at our physical meetings (or, if it is truly is an emergency, we will make a decision over email). Decisions will be made only if every team member in our group supports it (as in a 4 to 0 vote). We believe that this is the best approach, because if one team member doesn't support a decision and we go forward with it, the one team member would be displeased and would affect the overall team progress.

This suggests that disagreements will be resolved only when everyone is happy, meaning that we will have to compromise as a team to satisfy every single team member. We understand that this may be tough, but we believe it will be for the best of the overall team health.

Team Reflections

Danny Chiao
Our team had a variety of people who had different skills to offer to the group. Everyone was responsible and punctual, and we followed through with our team contract. As our TAs can attest, each individual always knew what other people were doing and what the next steps of the project were.

One lesson learned was that more time spent on object model creation and design analysis as a team is especially valuable, in particular thinking about user flow. For example, we might have immediately thought of implementing recurring events, which would have focused our efforts on our website's purpose, improved user experience, and saved us time.

Sherwin Wu
I think our team followed the ground rules that we set at the beginning pretty well. We were, generally speaking, very constructive together, and we worked well. We didn't have any arguments, and everyone was receptive to whatever work was assigned to them. Work was done on-time and any problems we had was worked out as a group by emailing out to everyone. Because of the nature of this project, a lot of work was modularize and that helped with organization. People were also very receptive to help from other team members.

One thing I learned from this project that I wasn't expecting was how easy collaboration and team-coding was with git and github. All merges and code conflicts were handled very gracefully and were worked out easily. It really got out of the way and allowed us to have maximum productivity as a team. One thing we could have done better as a team though, was work together as a group more often, as we had the highest productivity when we were all coding together.

Rebecca Odim
Our team followed our Team Work Guidelines that we set forth at the beginning of the project. We made decisions by consensus, made constructive comments on each other's work, and were very productive in meetings. Because our application was modular it was easy to assign tasks based on the different parts. It was interesting that we all came to this project with a similar perspective on how to solve the problem of running events at MIT, based on our various backgrounds in dealing with the current system.

I think we could have started off with coding in an organized fashion, rather than coding and organizing it at the end. Though we were able to bring it together at the end, better documentation could have kept our project working efficiently from the beginning.

Ibrahim Abdussabur

Our team followed our guidelines closely. We designed our project to be very modular from the very beginning so that we could split up features easily. This helped us because we were able to work independently at times in addition to working together. We regularly assigned action items in our team meetings. This helped us keep track of deadlines and help each other out if someone was lagging behind.

Our team worked very well together and we were able to complete the project ahead of time. This team really helped me achieve my goal of working on a big rails project. I learned by keeping a big project modularized, you're able to split up work easily between teammates and as a result the project goes faster because people are able to work on mostly the aspects they are good at.