

Group 7: Team Organization and Evaluation Report

Robert Moe:

Our team organization was effective but a bit looser in structure than on the previous project. Everyone was bursting with ideas for the robot whether they were on software or on the hardware team. This unstructured organization leads to a lot of changes through the course of the robots design. Although they say the more minds the better, in the case of our robot it lead to many changes. In the future I propose that we must stick more strictly to the hardware team working on the hardware and the software working on the software. Although we did stick to this the hardware team had a lot of say in ideas for software and software had the same for hardware. Our team was not organized to deal with this many different opinions.

Although I am an advocate for creative and clever designs, they do require more planning and time, in which this case we did not have. We should stick with a plan from the start and deal with that plan.

John Zumwalt:

Overall, I think our team structure was still effective. There are, however, some things that could use adjustment.

It seems that we don't really have a formal design phase. It might do us some good to go through some mock-ups and try to model what we're to accomplish before we jump straight into it. In this project we never really had a well thought out plan, we were just improvising our way along. This wasn't so bad, but we ended up going down a path that wasn't really foreseen in the beginning.

I also think it's a good idea to practice better coding standards during implementation. It seems that we're rushed during the development phase, operating under the assumption that we'll go back later and 'fix' the code. However, we're always in this rush and the corrections never really happen. If the code was easier to read (comments and readable code) the final push to get everything perfected might be a little less hazardous.

Finally, I don't know how beneficial our "crazy" ideas have been. It's always good to be creative and find inventive solutions, but this time it seems to have been our downfall. Our initial ideas were to have a small robot, but one crazy idea later, we've got the biggest robot in the class. That beings said, I for one will probably not change when it comes to this. I'll almost always prefer the contrarians approach to the more normal.

Celi Sun:

Our team needs meetings and a leader who can coordinate the two sub-teams or the whole team's members and keep an eye on our timeline and feedback plan. The meetings

(especially for our design) are necessary and very important for us to express every member's design idea so that we can discuss them and find the best design for our project. It will encourage everyone to be engaged in it (if someone needs help, he can ask for help during the meetings or from the leader directly), it also can help us keep more even and avoid the risk of wasting time and redesigning .

In most cases, Hardware design and software design are dependent on each other. Hardware design should consider and meet the software design's requirement. So, it will be better to find a good combination point between the hardware design and software design.

Mark Woehrer:

We need to have assigned times for team meetings. At least once a week for about 1/2 hour so we can talk about our current progress and help other team members when they get stuck on a problem. Meetings will also help keep everyone involved in the project. We will need to get everyone's schedules so that we can plan a time.

We need a new design phase for experimentation. Experimentation is important but we need to set limits on who is doing the experiment, what resources are needed (legos, sensors, servos, etc.), how long it should last, how we know if the it is successful. In this way we can try out new ideas without using up all of our resources (time, parts, people), this will keep one idea from taking up all of the available time, and give us more time to evaluate alternate designs. This is one of the things that can be decided at team meetings.

We need to do a better job with the testing task. We need to get more feedback about project requirements from the instructor. We might need to make this a new group. We need this feedback to see if our design meets the project requirements. Without it our project loses direction. I think we did a much better job with Project 1 in this respect. We need to understand the problem better so that we can find specific test cases we can use in our testing.

We need to understand the sensors we are using better. I believe this should be a new task for the hardware team. We might even dedicate a team member to it. I believe it would have helped us figure out the range sensors. I believe sensors will be even more important on the next project.

We need to stick to the assigned tasks better. I think we are doing a pretty good job here but there is a lot room for improvement. Without assigned tasks we don't know who is responsible when things go wrong, or if someone needs help with their task.

We spent too much time on lego hardware and not enough time on software. We need to spend more time on sensors and algorithms described in class. We need more brains less bronze.