

To the Noobs: A Better Plan to Learn By Sara Brooks

To all the newcomers. The Flying Toasters, named because of Battlestar Galactica, and formed out of a love of engineering, has continuously gained members over the years. As a team, we take great pride in that. We are expanding as a team, and showing our community all that we can do, ranging from introducing elementary schoolers to STEAM¹ to building robots for Bosch². We have to integrate these members into our team, make them a part of the robotics family, show them the ropes, and allow them to grow and prosper in the world of robots.

No experience necessary all the sports³ posters say. That's true, you can come into robotics not knowing how to build a robot, but how do you leave knowing how to build a robot? How do we learn? Our team is mostly student run. We have a coach, and various mentors to lead us along the way, but as far as who-tells-who what to do, it's on the students. We're self governing, we take care of ourselves, like a cat. Leave us alone and we'll most likely be okay. Mentors step in when necessary, but if you "don't have anything to do" it's on you. Generally, this system works. The mentors help give the projects that need to be done to students that they feel are responsible and have the

¹ Science Technology Engineering Art and Math

² An elite engineering company

³ Yes robotics is a sport, we compete, get medals, and lifting robots is not as easy as one might think.

knowledge to accomplish the task, and that student passes smaller tasks down the food chain to other students with the skill set they need. How do students gain this skill set that makes them valuable? This is wherein the problem lies.

We have fall training⁴, that is, when you sign up to be in various groups⁵ during meetings so you can get a general idea of how to do the things you're interested in. Fall training groups included programming, electrical, CAD, and machining. These groups consisted of approximately a classroom⁶ or more of students and a teaching mentor⁷ along with various other mentors for moral support as well as helping to answer what questions they could⁸. These topics were taught in a very classroom like setting. Slideshows, lectures, with small amounts of hands-on seeing-things-happen-action.

Sounds like a great plan right? Not exactly. In these large groups you can only learn so much. If you know lots about the topic being taught and simply want to get a little more information, these large groups are fine. But if you know nothing about programming, how to use machines, or how to use CAD⁹, a large group just isn't going to cut it. You can't ask all the questions you have, and you don't have that one-on-one help to learn.

For those of us who were just starting to learn anything about programming or CAD, the "classes" went as such a fast pace, I was left sitting there looking between my computer screen and the "teachers" wondering what had happened. What went wrong? How far behind am I compared to the teacher and other students? How did they even

⁴ Aka: officially unofficial meetings

⁵ More will be explained momentarily

⁶ At least 20

⁷ The guy up front with the projector

⁸ Not very many

⁹ Computer Aided Design

do that? All the while the programming kids were making cartoon characters with the programs as the teacher talked, only switching screens over to what we were “actually working on” when the teacher walked around to check progress.

I can't even imagine how it felt when we were being taught how to use machines and not knowing what was happening. I was one of the lucky ones. I was the kid making cartoon characters in that class, but for the kids that didn't know how to use a drill press, or a band saw that must have been awful. Someone shows you how to use a bandsaw once with a giant group and you're supposed to be able to easily do this now? Now you know what a bandsaw is, and what you can do with it, not necessarily how to use one properly with all the tips and tricks learned from years of experience of using one.

We¹⁰ learn the most from doing things hands-on, from being put into a small group and following someone and learning directly from another team member, not a mentor with a projector in front of the classroom. We learn from integration. If you don't jump in with two feet into the freezing water that is robotics, you might just find yourself on the sidelines.

In our team, if you don't find a way to get yourself involved, no one is going to involve you. This is both genius, and terrible. Genius because we only have the most dedicated people that want to be involved working on our robot. Terrible because some of the people not working on the robot want to, they just don't know how.

We can't have all 60 people working on the robot at once. We all know that. But you also can't have it only be 6 people actually doing anything, while the rest are told that they don't need any help. That's a lie and we all know it. There is always something

¹⁰ Team members

to do, but if you keep getting told to clean the floor every time you ask to help, you're going to stop asking.

Our team members are nice, kind, and polite, but they have developed a fatal flaw. The flaw of the "if I want it done right, I'll just do it myself" attitude. This perspective doesn't allow for learning, or integration, or teamwork. This attitude gives you a team where 6 people do the work for 60. This attitude excludes people that are willing to learn simply because they don't already know something.

We don't mean to. We were there once too. One of the people being told to sweep the floor. But we've been trained, and this thinking gets passed down. We don't really mean to push them aside. Sometimes we do use their help, but most of the time we simply can't be bothered. Why tell them how to do it if it will be faster and easier for me to just do it myself? We need to create teachers, people that will do things for the good of the team not just themselves. Will showing them how to do something really trouble you that much? Will those 2 minutes of explaining now not make up for it when anytime we need new treads put on that wheel we have another person that can, or anytime we need something cut down on the mill we have someone else that can?

We need to pass on our skills. If everyone is trained in a way that will replace and even grow the skill set the team already has, it will allow this team to go further than it has ever been able to. We will no longer have to hunt down and call the one person that knows how to do something, we won't be struggling when the one person that is good at using that machine graduates and leaves the team, we won't be struggling to find someone that knows in a sea of people that were never taught. There are enough people on our team willing to learn, if only we would teach.

How? How do we teach? How do we stop the downward spiral? How do we stop the few people who are working on the robot from being the only people that ever work on the robot? Team. We are one team, one body, one voice as we are constantly reminded. A hand cannot do a foot's job, but both are necessary to function. We must work together. No more telling people to sweep the floor. No more saying there's nothing for you to do. The team always needs something. Chairman's needs people to watch their presentation, programmers need bug fixes, machinists need things deburred. But also, we need to train people to be in chairman's giving that presentation, we need people to be taught the code to be a programmer, we need people to be taught to use the machines to be a machinist.

People can't learn if we don't teach. Every student on our team deserves a chance. To be involved, to complete a project, to help this team achieve its goals, to push this team further than it has ever been, to improve. This is what STEAM is all about, this is what FIRST is all about. Learning and improving for the future. Making the students of today the teachers of tomorrow. What better place to start? We have the students, now we only need the teachers.

To teach the new students we need a system. A system that gives them a hands on, small group environment. A system that gives them the chance to ask questions and learn all the tips and tricks of years of experience. A mentor. Someone who is their person. The person they can go to, to help them earn their varsity letter, to help them machine a part, or teach them how to present a speech. An older student, a veteran member to show them the ropes, to teach them what they need to know to make it in the freezing water that is robotics. At the beginning of the year, before fall trainings, we

should assign all the new students a veteran mentor that currently does what they want to do. If the new student is interested in programming, assign them someone that can teach them programming. If the new student wants to machine, assign them someone that can teach them how to machine. Have the new students go through fall training with their mentor, that way they always have someone that can answer their questions, someone that has a general idea of what's going on.

By giving everyone a person, no one will be left behind. No one will be left on the sidelines, or constantly be told to sweep the floor when they desperately want to help. With everyone having a mentor it will also bring our team closer together. Everyone will be able to find their place, wherever on the team that may be.

Plans change, if someone needs a mentor that will be a better fit, so be it. Some people end up places they never thought that they would be. I have managed to do a little bit of everything on the team. I never thought I would be *that* person. The person that's involved with everything and somehow knows what's going on everywhere else with everyone else on the team. I thought I would just be a machinist, a builder. Hands on assembling everything physically on the robot. Turns out, I am that, but I also help to edit the chairman's essay, and debug programming.

By teaching every single person on our team, and creating a way to do so, we can help other teams that are having the same problems. Other teams that have 60 people on the team, but only 6 working on the robot. We can make robotics better for everyone involved. We can change¹¹ the lives of more students than we already do.

¹¹ improve

If everyone on the team is trained and has a valuable set of skills, we will be significantly more efficient in completing tasks and finishing projects on time. When we don't have enough people with the skills needed, we can't be an effective force at reaching our goals. When only one person has the skills to do something, and they have a million¹² other things to do as well, we are only hurting the team members that we need the most.

We need to be able to split the workload. Build season is rough. We have late nights, and weekends, and every day of the week meetings if you can be there. But sometimes school gets in the way. Too much math homework, a science lab you've been putting off, studying vocabulary words for English you won't know the meanings of in about a week. Our "main people" can't always be there to save us. We have to get backup. We need other people that will be able to pick up the slack¹³ for when other people aren't there. If no one else knows what's going on, or how to do something we will never be able to improve. We will always be held back by not knowing.

We can't have that. To not succeed because some people didn't know some easily available information because it would use up two minutes of someone else's time. We can change. We can improve. We can evolve. Our team is on a rollercoaster that only goes up.

Once we integrate a new and improved learning system, it will only get easier in years to come. Once everyone on the team acquires a good set of skills, it will be easier and easier for new generations of Toasters to learn what they need to know. We may still have the "involve yourself" attitude as a team, but it will be easier for members to

¹² A million could be a slight exaggeration, but you get the idea

¹³ Heh. Get it? The slack chat, slack. It's funny.

involve themselves when they know how to do things. They can't be excluded for not knowing how to do something¹⁴.

I hope that we can make this team a better place for everyone, that we can improve what we already have and soar to new heights, that we can include everyone in everything we do, that we stop pushing people to the sidelines. We can be better. I want us to be better. This team can do incredible things, if only we apply all the power we have to toast the competition. By making ourselves better as a team, we can improve robotics for everyone. We can share our training system with other teams to allow them to include everyone as well, to allow them to have everyone on their teams helping to improve themselves, and robotics as a whole.

¹⁴ And if they're still being excluded we know that we've got bigger problems.