**Failed Project Assignment**

**Example 1**

**Category:** Requirements Issue

**Failure:** Vague or open-ended requirements (such as requirements that end with “etc”)

**Example:** One of our big clients is a large restaurant chain (approx. 1200 locations). Once a year they roll-out all new menu designs for their 3 main product lines: Core menu, dessert menu, and drink menu. Our team is responsible for programming the templates that build the production files for each menu. For example, we build 1 template for the Core menu that pulls in artwork and data specific to a location, so that the Miami location will see the menu items and pricing related to their location and the Milwaukee location will see their info.

Each year when these updates come through the client never provides any instructions. There is no outline of what menu items are changing, which fonts/styling are changing, or what additional programming logic is needed for file versioning. It’s up to our team to plug in the new artwork and determine what changes need to be made.

**Resolution:** Because the client is not willing to take on this task, we have resorted to providing very vague timelines for project completion. We’ve provided timelines from 3 days to 3 weeks all based on what changes they might be making to the artwork. Because even when they say the changes are minimal, we’ve learned that is not always the case.

**Example 2**

**Category:** Decision Making Problems

**Failure:** Failure to establish clear ownership of decisions or the process by which key decisions will be made results in indecision and confusion.

**Example:** About a year back, my company rolled out a new client facing app that would act as a dashboard for all the client’s applications. One of my responsibilities was to assist with the development of the reports section of the site. The information flow for this task was as follows: Client > Production Team > Me/eCommerce Team > IT Business Analysts > Application Developer. With so many teams involved, this led to a lot of confusion when defining the report requirements. Each department would slightly tweak the requirements.

However, where the real problem came in was that the reports were not being tested by the Production Team and Client. Instead, they assumed all the testing was being done by my team and the IT teams. Which it was, and all our testing was successful, but because the requirements changed through the information flow, the final product was different than originally designed.

**Resolution:** The first step we took was to identify the report owners. We identified that the Production Team were the end users of the report, and my team along with the client did not need to be involved. This led to changing the information flow to as follows: Production Team > Business Analyst > Application Developer. We also clearly identified that the Production Team will be 100% responsible for testing and approving the final report.

**Example 3**

**Category:** Leadership and governance

**Failure:** Appointing a Sponsor who fails to take ownership of the project seriously or who feels that the Project Manager is the only person responsible for making the project a success

**Example:** Because of all the requirement issues we were having with our restaurant client in example 1, we held a 2-day Kaizen event to find ways on improving the relationship. My company is really big on continuous improvement and entire teams solely focus on these issues. The first step for this event is to identify all the people/teams involved with the production process. This includes sales, project management, production, scheduling, IT, and a couple managers from each of those departments. The second step is to find a Sponsor for the event who will assist with any road blocks and makes sure action is taken to resolve the issues. Sponsors are usually company Directors, or other people who are high up in the company. This makes choosing a sponsor a really big deal.

During the 2-day event, the sponsor stopped by for only 15 minutes. They gave a pretty good speech about how this event will help the client and our company resolve issues. Then they stepped away from the project entirely.

**Resolution:** There has been no resolution to this example. After identifying all the issues and ways to improve them, upper management has taken no action on any of them. All of us that were involved found it to be a big waste of time and resources. There has been talk about repeating the event, but all of us have turned down those ideas because nothing will happen once it is done.

**Example 4**

**Category:** Planning

**Failure:** Teamcommitments themselves to a schedule without first getting corresponding commitments from other groups and stakeholders who also have to commit to the schedule (aka schedule suicide)

**Example:** For one of our clients we created a beer menu that takes in a data feed from their other vendor. Last year the Federal Government mandated that all menus have to show calories. This posed a challenge both to my team who is responsible for the menu template, and the other vendor who is in control of the data. The problem came when the client assumed that both updates could be made at the same time. However, this update was a bigger challenge for the data vendor because they had to change their database structure. And we could not fully test our work until the new data was being passed over. We were able to do some preliminary work until the data was ready, but couldn’t push out a live product until it was fully vetted.

**Resolution:** The client was hoping to have this complete in 1 month and it ended up taking 3 months. The data vendor took 2.5 months to complete the work and our team took .25 months. During testing, we found that the vendor had mapped the data incorrectly on their end and to fix the issue. That update took the remaining .25 months of the schedule. So in the end, this project made our team look really good.

**Example 5**

**Category:** Team Issues

**Failure:** Failure to provide team with appropriate training in either the technology in use, the processes the team will be using or the business domain in which the system will function

**Example:** One of the biggest benefits of attending school while working, is that I can apply everything I learn immediately at work. However, this also becomes a big drawback because I start using techniques and tools that are foreign to my team. The best example of this is when I started using GIT versioning control. Not only did I start using this new method at work, but I also built it into a project I was working on. This meant that when the project was complete, I would be the only one that could provide updates to it because no one else knew how to access the files.

**Resolution:** I created training sessions for all the team members and shared online resources like GIT documentation and Codeademy tutorials. I also took the time to show them how they could apply this to the projects they were working on. Once they could see the power of the new tool on their live projects, the learning really took off.