

Case Study: Giga Safe's ICS 1.0

Carl had been put in charge of version 1 of Giga-Safe's inventory control system (ICS). He had a general idea of the capabilities desired when he attended the first meeting of the oversight committee for the project. Bill was the head of the oversight committee. "Carl, how long is ICS 1.0 going to take?" he asked. "I think it will take about 9 months, but that's just a rough estimate at this point," Carl said.

"That's not going to work," Bill said. "I was hoping you'd say 3 or 4 months. We absolutely need to bring that system in within 6 months. Can you do it in 6?"

"I'm not sure," Carl said honestly. "I'd have to look at the project more carefully, but I can try to find a way to get it done in 6."

"Treat 6 months as a goal then," Bill said. "That's what it's got to be, anyway." The rest of the committee agreed.

By week 5, additional work on the product concept had convinced Carl that the project would take closer to his original 9-month guess than to 6 months, but he thought that with some luck he still might be able to complete it in 6. He didn't want to be branded a troublemaker, so he decided to sit tight.

Carl's team made steady progress, but requirements analysis took longer than they had hoped. They were now almost 4 months into what was supposed to be a 6-month project. "There's no way we can do the rest of the work we have to do in 2 months," he told Bill. He told Bill he needed a 2-month schedule slip and rescheduled the project to take 8 months.

A few weeks later, Carl realized that design wasn't proceeding as quickly as he had hoped either. "Implement the parts you can do easily," he told the team. "We'll worry about the rest of the parts when we get to them."

Carl met with the oversight committee. "We're now 7 months into our 8-month project. Detailed design is almost complete, and we're making good progress. But we can't complete the project in 8 months." Carl announced his second schedule slip, this time to 10 months. Bill grumbled and asked Carl to look for ways to bring the schedule back to around 8 months.

At the 9-month mark, the team had completed detailed design, but coding still hadn't begun on some modules. It was clear that Carl couldn't make the 10-month schedule either. He announced the third schedule slip number-to 12 months. Bill's face turned red when Carl announced the slip, and the pressure from him became more intense. Carl began to feel that his job was on the line.

Coding proceeded fairly well, but a few areas needed redesign and reimplementation. The team hadn't coordinated design details in those areas well, and some of their implementations conflicted. At the 11-month oversight-committee meeting, Carl announced the fourth schedule slip-to 13 months. Bill became livid. "Do you have any idea what you're doing?" he yelled. "You obviously don't have any idea! You obviously don't have any idea when the project is going to be done! I'll tell you when it's going to be done! It's going to be done by the 13-month mark, or you're going to be out of a job! I'm tired of being jerked around by you software guys! You and your team are going to work 60 hours a week until you deliver!" Carl felt his blood pressure rise, especially since Bill had backed him into an unrealistic schedule in the first place. But he knew that with four schedule slips under his belt, he had no credibility left. He felt that he had to knuckle under to the mandatory overtime or he would lose his job.

Carl told his team about the meeting. They worked hard and managed to deliver the software in just over 13 months. Additional implementation uncovered additional design flaws, but with everyone working 60 hours a week, they delivered the product through sweat and sheer willpower.