

# 34

## Work as a Team

*7. I will continuously ensure that others can cover for me and that I can cover for them.*

Imagine a team of soccer players moving the ball down the field toward the goal. Imagine that, during a play, one of the players trips and falls down. What do his teammates do? They step into the breach, cover the hole, and *keep the ball moving down the field*.

The sailors on a navy vessel each have a job to do, but they also know how to do someone else's job. Why? Because in the heat of a battle, a sailor might go down. Another sailor *must* be able to step into the breach and *keep the ship in the battle*.

It is no different on a software team. Indeed, that's why we call it a *team*. If one of the members of the team goes down for any reason, others on the team must be able to step into the breach, fill that hole, and keep the project on track.

## Collaborative Programming

Segregation of knowledge into silos is extremely detrimental to a team and an organization. The loss of an individual can mean the loss of an entire segment of knowledge. It can paralyze the team and the organization. It also means that the individuals on the team don't have sufficient context to understand each other. Often, they wind up talking past each other.

The cure for that is to spread knowledge through the team. Make sure each team member knows a lot about the work that other team members are performing.

And the best way to spread that knowledge is to *work together*—to *pair*, or *mob*.

The truth is that there's hardly any better way to improve the productivity of a team than collaborative programming. Because a team that knows the deep connections among the work that's being done can't help but be much more productive than a group of silos.

## Open/Virtual Office

It is also important that the members of the team see and interact with each other very frequently. The best way to achieve this is to put them into a room together.

In the early 2000s I owned a company that helped organizations adopt Agile development. We would send a group of instructors and coaches to those companies and guide them through the change. Before each engagement began, we told the managers to rearrange the office space so that the teams we would be coaching worked together in their own team rooms.

It happened more than once that, before we arrived to begin coaching, the managers told us that the teams were already much more productive, just because they were working together in the same room.

I wrote the first version of this chapter in the first quarter of 2021. The COVID-19 pandemic was beginning to wane and we were all looking forward to a return to normal life. But the pandemic has left, in its wake, a large number of software teams that work remotely.

Working remotely can never be quite as productive as working together in the same room. Even with the best electronic help, seeing each other on screens is just not as good as seeing each other in person. Still, the electronic systems for collaboration are very good nowadays. So, if you are working remotely, *use them*.

Create a virtual team room. Keep everyone's face in view. Keep everyone's audio channel as open as feasible. Your goal is to create the illusion of a team room, with everyone in it working together.

Pairing and mobbing enjoy a lot of electronic support nowadays. It is relatively easy to share screens and program together over distances. While you do that, keep the faces and the audio up and running. You want to be able to *see* each other while you are collaborating on code.

Remote teams should try, as hard as they can, to maintain the same working hours. This is very difficult when there is a huge East–West distribution of programmers. So try to keep the number of time zones in each team as small as possible, and try very hard to have at least six contiguous hours per day when everyone can be together in the virtual team room.

Have you ever noticed how easy it is to yell at another driver while driving your car? This is the windscreen effect. When you are sitting behind a windscreen, it's easy to see other people as fools, imbeciles, and even enemies. It is easy to dehumanize them. This effect happens, to a lesser degree, behind computer screens.

So teams should get together in the same physical room several times per year. I recommend one week each quarter. This will help the team congeal and maintain itself as a team. It is very hard to fall into the windscreen trap with someone you ate lunch with, and physically collaborated with, two weeks ago.