

The Team Collaboration Rules

Care about each other, share what you know, and this will help you dare to try more

1. Always be collaborating, instead of working on your own.

You go further if you have people supporting you than if you work on your own. Students should always be pairing or mobbing on tasks. By collaborating on tasks feedback happens sooner, the team levels up each other's skills, and they have more fun.

2. Aim for diversity to improve your work.

Homogenous groups miss out on other perspectives. Rotate pairs regularly for the work on tasks too, and include everyone in team discussions. Have a chat with the quiet one, to see they are ok, and to co-create a way to ensure they contribute more regularly to discussions.

3. Work in the open so that the team can see your work sooner.

Be visible and available to your team. Let them know which task you're working on. Push your work frequently, so that others can use your work sooner rather than later. Keeping your work secret means others worry whether you're doing the work, and the quality of your work.

4. Be humble and ask for help and feedback so that you learn more.

Everyone gets stuck sometimes, so ask for help. The feedback here applies to both the work being done, as well as asking for feedback from your team members too about how you are as a team member. Be kind and helpful to each other.

5. Meet regularly with a suitable cadence for your context.

Find the right cadence for your team meetings, and for doing the work. Sometimes this means weekly meetings, other times more frequently depending upon deadlines. This is something that you adjust as you learn more about your collaboration.

6. Accept that it's all guesswork and, start small to learn more.

You won't know how much work there is until you start building your product. Just start prototyping and exploring the application context and options, in order to gain perspective on what's possible in the time you have available.

7. Build deployable, small vertical slices to learn more quickly.

Building the product in small, thin slices will help you gain perspective on what's possible in the available time. Small slices will help you build a testable app, and provide frequent feedback from your client too, while also reducing risk with each slice as you add more work.

8. Keep doing the next riskiest item in order to reduce risks quickly.

By doing risky things early you remove them, or find ways to mitigate them sooner. This reduces stress in the team, and for your clients, who then know that risk has been resolved.

9. Shorten your feedback loops, so that you learn faster.

Smaller sprints or time boxes mean the client sees the work more frequently. Thin deployable slices also provide more feedback too, as does working in the open within the team and the client. By pairing and mobbing you are also gaining feedback from each other quickly too.

10. Always be pausing to review your collaboration.

This is also known as always be learning, always be pausing to reflect on how you do your work. Pause to reflect on how you might improve your work. Always aim to see learning opportunities when they arise.