

Mob / Ensemble Programming Quick Start Guide



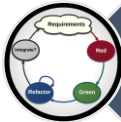
Use a mobbing tool -
(mobster, mob.sh, others)



Favor short switching
intervals



Take disciplined breaks via
the Pomodoro Technique



Pair Mob Programming with
Test-Driven Development



Rely on Llewellyn Falco's
Strong-Style Pairing Model



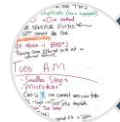
Invite the entire team to
participate



Find the Driver's level



Track waste and waits
with a Waste Snake



Document team learning
on a Learnings Board



Avoid runaway Drivers



Maintain shared team
settings



Debate in code



Invite your leadership



Invite the experts

Additional Resources:

Industrial Logic's Modern Agile [Mob Programming Cheat Sheet](#)

Jay Bazuzi's [Mob Programming Pattern Language](#)

Dillon Kearn's [Mobster \(mob programming tool\)](#)

[Mob.sh](#) (remote mob programming tool)

Tim Ottinger's [A Few Tips For Mob Programming](#)

Anthony Sciamanna's [Practices for Effective Mob Programming](#)

Credits:

This guide distills the advice I've given in talks and a blog post about starting teams with mob / ensemble programming. The ideas have come from coaching with colleagues from [Industrial Logic](#) Tim Ottinger, Michael Rieser, Bill Wake, Steve Ropa, and long-time colleague Nick Goede.

Llewellyn Falco's Strong-Style Pairing Model image by Derek Graham ([@deejaygraham](#))

Reminder:

These are recipes to get a team started. As you gain experience with mobbing adopt and adapt the practices that work best for your team. There is no single "right way" to collaborate.