



Pair Programming at Codesmith

...

What is pair programming?

- Coding with a partner using a single text pad for writing your code
 - **Remotely:** using Zoom and VSCode Live Share
 - **In person:** using a single monitor and computer

Basics of pair-programming:

- Work *together* to research the problem you're facing.
- One person guides (navigator) and the other writes the code (driver)
- **Swap every ~20 mins**

Research the problem

- Talk together about your ideas for possible solutions
- What would be good things to search for online (possible documentation to look at)?
- When you (or one of you) has an idea for a path, switch to the driver-navigator roles and start coding!

Driver and Navigator Roles

- Similar to driving and navigating a car
- Navigator talks through route - e.g. "Let's iterate through the array and add 3 to the 6th element"
- Driver converts navigators guidance into *actual code* and types it

How to be a great navigator

- Explain your reasoning - your partner can't read your thoughts.
- Be precise in instructions "Let's iterate over the array after line 2"
 - Avoid telling the driver what exact code to write, don't tell them syntax or even exact methods if it's not necessary.
- Be patient and don't be tempted to drive as well
- **Don't simultaneously drive and navigate**

How to be a great driver

- Listen to the navigator!
- Reiterate instructions back to navigator
- Remember there's more than one way to the destination
 - be patient and ask questions if you are unsure of their approach
- **Don't simultaneously drive and navigate**

Benefits of pair-programming

Helps you:

- articulate and clarify your ideas
- get started, not just research the problem
- overcome inevitable blocks
- practice for interviews and jobs
- encounter new ways of solving problems

Benefits of pair-programming

If you're *more* experienced than your partner you get to:

- practice the role of a senior engineer - guiding less experienced engineers but without ever just giving them the answer
- secure your knowledge through the highest form of learning - teaching

Benefits of pair-programming

If you're *less* experienced than your partner you get to:

- rapidly learn new concepts
- improve your ability to ask the right questions - this is a core programming skill
- practice being strong and asking your partner to better explain when you don't understand

Common Pitfalls

Mistake: Not switching roles

Solution: Switch roles every 15-30 mins

Mistake: “Can you move this to there”

Solution: “Can you move the return statement to the bottom of the function. We don’t want to exit our function too early.”

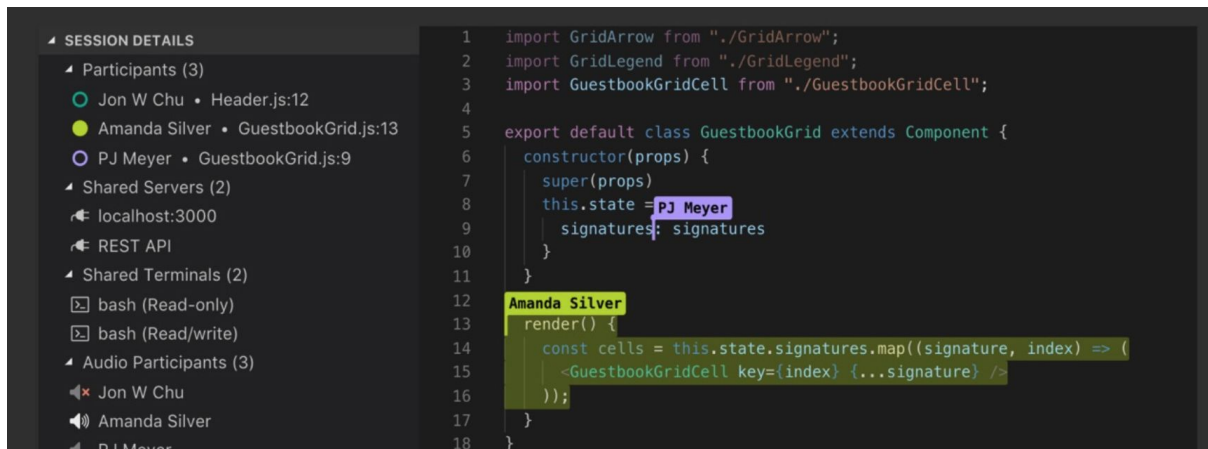
Mistake: Driving and navigating at the same time

Solution: Don't do it!!

VS Code LiveShare

A tool for pairing remotely!

Pair programming remotely with LiveShare



- Real-time collaborative coding
- Code lives on the host machine
- Access to servers and terminals on the host machine
- Up to 30 collaborators per host

Benefits of Using LiveShare

- Allows you to easily switch between driver and navigator roles as you work through complex challenges
- Eliminates the process of pushing up non-working code to your challenge/project repos
- Ensures that you are both working in the same development environment
- Oh and it's free!

How to Use LiveShare

- Install the VS Code Live Share extension
- Use the command palette (Ctrl/Cmd + Shift + P) to:
 - Sign in to Live Share via GitHub
 - Start up a collaboration session
- The link will be automatically copied to your clipboard - you're now free to share the link on Slack!

Helpful Tips when Using LiveShare

- You may notice that your screen will move when other folks scroll through around the file. This means you are currently have 'focus' on the other collaborator.
- You can unfocus a collaborator in the in the new 'live share viewlette' (official term!)
- You can access shared servers, terminals, and more from the live share viewlette



Pair programming is one of the secret tricks to accelerate your programming career