**Technical Mentor Session #1 | 24 Jan 2025**

Brian McNeilly

* Knows someone who was a developer at Blizzard

Kevin Ray, Senior Software Engineer MSFT

* Prior Coast Guard
* Worked in a Startup—RiskIQ—as a “back end developer”

Dakota Matthews

-Focus on fundamentals

-How to make yourself unique on resumes to separate yourself from others

-Algorithms

-Hiring managers best way to get your feet in the door

-MSSA is ultimately nothing more than a certification program

-Use the MSSA network

-It’s a pipeline

-It’s a resume builder

-Veteran’s status can be looked at as a diversifying experience

-Resume stuff

-Github

-3-tiered web app

-Write/change code to break your code to help you learn how to debug it, prove that you broke it and showed how to fix it

**Technical Mentor Session #2 | 7 Feb 2025**

Startup companies

* Everyone has to do a little bit of everything
* A lot of unknowns and instability
* Younger the startup is, the worse a work-life balance is
* Do your due diligence
  + When do they plan on being profitable
  + What’s their business plan
  + …

What to focus on next

* Learning to take the time and do it on your own
  + Your projects done in school/bootcamp will be the same projects everyone else does in school/bootcamp
* Go beyond your coursework
* Find out what holes the MSSA leaves for what you need
  + Personal projects
* Build up your Github

**Technical Mentor Session #3 | 21 Feb 2025**

* Certs can be helpful, but make sure you really know them. Think about how far you can go if asked “tell me everything you know about [cert]”
* Certs and education up higher on the resume until your experience can speak more than them
* If you’re working towards a cert, you can put “expected test date: “ on resume to let them know you know the knowledge but just don’t have the cert
* Certs:
  + What would be good additional certs for us learning C#?
  + AZ 304—Developing Azure Solutions
  + AZ 700—networking type cert
* “What algorithms should we know and have in place”
  + Know how each data structure works, general idea of how it’s structured
  + Use cases for why it would be good at certain scenarios
  + It’s more important to voice and talk through your thought-process
* Project Euler for mathematics problems

**Technical Mentor Session #4 | 7 March 2025—Technical Interview Prep**

* Probably random people from the team you’re applying for
* Questions:
  + Problem-solving process and logical thinking over specific code
* Get your foot in the door
* Dakota Matthews—can help find jobs that require clearance
  + CTJ is the keyword for searching cleared MSFT jobs
    - Cleared Technical Job
* VM Ware, Oracle, School districts
* Tell me about a time when you couldn’t get a customer a product that they wanted and how did you manage their emotional response
  + I told them they couldn’t have this and that was disappointing for me, but…
* Troubleshooting story in very specific terms
  + Define the technical problem, define the tools used to identify
  + Bullet points
  + Granularity, knowledge of technical tools
* Bring blank notepad and pen/pencil, show the interviewer and make sure they’re okay if you use it for notes

**Technical Mentor Session #5 | 21 March 2025—**

* Co-programming
  + Conflicting push/merging
  + IDEs might have co-coding features
  + Dividing labor/task management
* Prepping for certs
  + AZ-900
  + AZ-204
  + Find a project that uses what that cert requires
* Triaging technical skills
  + Leetcode—Good for specific interviews or portions of interviews
* Technical Interview demo from Kevin
  + Implement IsWinningMove for a 3x3 game of Tic-Tac-Toe (Noughts and Crosses)
    - You only need to implement enough of the game to support this IsWinningMove method. This method would eventually be used after each player has taken a turn to determine if their play won the game. It should evaluate the game state to determine this.
  + First: ask questions
    - Does it matter how values are stored in this data structures
    - Do I need to handle edge cases, such as a draw
    - What’s the return type of the method/what should it return?
  + Create a 2D thing of list to represent a 3x3 grid
  + Public class Game {
  + Private List<List<string>> gameboard;
  + Public Game()
  + {
  + gameboard = new List<List<string>>();
  + for I =0; i<3; i++
  + ( gameboard.Add(new List<string>());) }
  + }
  + Bool IsWinningMove() {
  + // Check horizontal rows
  + Foreach (List<string> row in gameboard)
  + { foreach(string token in row)
  + { //does it match
  + // Check vertical columns
  + // Check diagonals
  + Return true; }

Code-bility? Code-dility? Browser-based coding program for MSFT interviews

Pick a thread early and learn about it—find an additional specialization? Go deep into it

Stackoverflow