Microsoft

~~One~~Two Year In: Insights from a Novice Cloud Native

Jose Blanquicet

Senior Software Engineer @ Microsoft (Kinvolk team)

😁 About me

- I’m Jose, nice to meet you! 🤝🏽
- Colombian ⚽ 🚵🏽 🏞️
- 10 years in Italy (Torino): *Mi sono innamorato della cucina italiana* 🍝
- Educational background 🎓:
    - ➢ Engineer’s degree, Electronics Engineering.
    - ➢ MSc., Computer and Communication Networks Engineering.
- Work experience:
    - ➢ 🚙 Automotive ✈️ Cloud Native ☁️

🧾 Agenda

Share my experience joining the Cloud Native world:

- 🚗 ✈️ ☁️ Why Automotive to Cloud Native?
- 🔎 Looking for a job in Cloud Native
- 🛹 Onboarding: Initial thoughts and approach
- 💡 Current situation

🚗 ✈️ ☁️ Why Automotive to Cloud Native?

- Automotive:
  - Pros:
    - **Comfort zone**: C, Linux, Networking, and electronics devices.
  - Cons:
    - Innovation: Only from one product to another.
    - Needs: Stability because the requirements are frozen during most of the product’s life cycle.
    - **Disclaimer**: Not a criticism. I just had bad luck.

- Stay up-to-date with new technologies.
  - 60% of job profiles were in Cloud Native (3 years ago).

🔎 Looking for a job in Cloud Native

- Requirements:
  - **Kubernetes**: I knew nothing 📦 . A few about Docker.
  - **Go**: Companies that recruit only Go programmers.
  - 100% in **English** with a distributed/multi-cultural team.
  - In other words, outside of my **comfort zone**.

- What kind of jobs did I candidate for?
  - Companies that were looking to create a **new team** (Senior -> Junior).

- Note about interviews:
  - It could take 6 months, and it's **not guaranteed** you will receive an offer.

🛹 Onboarding: Initial thoughts

- What does the team do?
  - ➢ Develop and maintain a tool for debugging and observing Kubernetes applications.
    - **Really?** For my first job as a cloud native I joined this team?

- I had to start **from scratch again**:
  - ➢ Opening that Kubernetes **Box**:
    - **User point of view:** API server, services, pods, containers …
    - **Under the hood**: containerd, CRI-O, runc, crun …
  - ➢ **Intimidated** by very expert colleagues.

🛹 Onboarding: Initial approach

- Learning:
  - ➢ User point of view: An initial rough overview.
  - ➢ For underneath components:
    - Avoid trying to understand **everything from the beginning**.
    - I started studying only what I needed for each task I was assigned.
  - ➢ Start from what **I already have confidence** in, e.g., networking.
  - ➢ **Community**: Conferences, Meetups, etc..

- Confidence:
  - ➢ The problem was me: I was not asking enough questions.
  - ➢ Just listened during the first meetings … Then I started **~~participating~~ making mistakes** and letting people explain me why I was wrong.

💡 Current situation

- Combining team members’ expertise:
  - ➢ The project has grown and thus the team (4 to 10 people).
  - ➢ I’m still the one with **less experience in Kubernetes/Go, but it is fine**.
  - ➢ **Combine** their Kubernetes/Go expertise with my knowledge of the project and how to coordinate the development (methodologies):
    - Junior -> Senior.

- English:
  - ➢ I’m **not the only one** struggling with the English:
    - Most of the people are **not native speakers** (Even in conferences).
    - Good practice: Virtual coffee with random **no work-related topics**.

Thanks!

Jose Blanquicet

GitHub: blanquicet
Twitter: @jose_blanquicet
Email: josebl@microsoft.com

Let’s connect!

Microsoft

Questions/Suggestions/Comments ❓