

## New Hires

**What is something you learned about in the product department during your first month here?**

- The **organization of teams** and how I contribute to that team.
- **Cross-functional teams**, the various roles in product and how they work together
- How our codebase is structured, and how the **cross functional teams** are set up. • I learned about our **flow for working on tickets**, reviewing them, and getting them through to final review.
- I say the biggest thing I had to learn was LINQ to SQL
- **Team work flow.** The **process of a ticket** going from design, to the engineers, to Review, then to QA. I wish that I had gotten that lesson in the first week and not try to figure it out on my own.
- Linq to SQL
- The difference between award networks and award options and \*re\*ward sources
- I learned about **Confluence** and that a lot of my questions could be answered there, once I finally got access.
- **Software development processes** were very new to me, I'd been exposed to agile methods in the past but never participated in sprints or used a **ticketing system** much before. I learned a lot about the mechanisms for enabling quick development and ensuring product integrity. As far as tools go, I'd never used Visual Studio, my git skills definitely leveled up, learned about LINQ.
- How the **product works**, **how it is used by CSMs and implementation**, and the **code structure**

**What is something you wish you already knew before starting at Awardco? For example, this could be a skill you wish you knew how to do, the way the teams work, or anything not necessarily related to the Product department.**

- How to lead a meeting within a tech setting.
- I wish I knew a little bit more about razor and the mvc architecture.
- I wish I had had a little **more experience reviewing someone's code** instead of just writing my own. I feel like it took me a while to get to the point where I felt like my reviews were thorough and meaningful enough.

- It was tricky getting **familiar with the actual product**. As with any software company, when I first showed up I knew nothing about the code base and it took some time to figure where everything was and what everything did. • Visual Studio Code shortcuts
- I think it would have been nice to know a little more SQL. Also **giving feedback** in a constructive and uplifting manner for things like code reviews
- I wish I knew **how front end and back end worked** better
- I can't think of anything off the top of my head for this question.
- I wish I had I known more about the **structure of http requests and responses** before starting, I was lucky that I was taking a class at the same time that taught me a lot of these things but that's been a crucial skill for developing our product and finding solutions to issues quickly.
- I wish I knew about the **equity and the equity agreement**, about .NET Framework, and about **cross-functional teams**

### What is something you learned during your first month?

- **How to get set up** inside the department to have access to everything I needed to succeed
- All about our product, more about **dev team structure and processes** • C#, how companies function on a larger scale (**how departments work together**) • I learned a lot about C# and our codebase.
- You know something that was really helpful was the backend bootcamp they made all the engineers do. Just got me familiar with the code base and the **process of building an endpoint at awardco**
- How to create branches and pull requests on GitHub.
- How c# doesn't support multiple inheritance
- I learned a lot about how all our code fits together, **what our website flow is**, and **how our code and site are structured** in much the same way.
- JSON serialization, object locks, Lookup data structure, expression trees • I learned about **Awardco's culture**, about my team's processes, and about the code base

### Any additional information you'd like to share?

- I think this was implemented after I started but **getting introduced to other department teams and their roles in the company** would have been helpful • It has been great working here!
- We were the first group to go through **training on the platform** (the same training

the CSMs go through). I thought this was really helpful because it gave me a **basic understanding of the platform** that I realized some developers never got.

- I just felt like all three questions kinda asked the same thing. I tried to make sure to give different answers though
  - It's good to know **what everyone's roles are in Product** so people know what others are in charge of and who to go to for questions. I had to figure that out myself when I was hired in November and it slowed things down a bit for me
- I liked the backend bootcamp. I think demo projects like that are pretty useful
- None that I can think of.
  - All in all, it was a smooth first month, esp. starting part time while in school. I feel like I've come a long way, it was probably in the middle of month 3 when I felt like I was **solving more problems on my own** than those I needed help to solve.
- I think each discipline (Backend, Frontend, Test, UX, and PM) and team could produce **better documentation for onboarding** their team member

## Leadership

### What is the best thing about having a new hire?

- Getting to watch them **learn the system** and get excited about contributing to it
- **distributed workload**
- they provide a fresh view on our **processes and styles**, and bring experience from other places that is invaluable
- Opportunity to **improve the onboarding process**.
- Seeing them mesh with the team and getting up to speed quickly and wanting to do more.
- The **new perspective** they bring to the team! I think it's really important to listen to their ideas, especially at the very beginning because they come with fresh eyes and see things differently.

### What is something that frustrates you about the process of having new hires?

- I think new hires should have a **better understanding of the product departments vision and goals**
- not knowing how to keep them busy at first.
- **training, onboarding**, getting their environment **set up**

- Not frustrating, but I think the biggest challenge now is the balance between giving them a firehose of information vs not providing enough training upfront. We've added a lot of training to the engineers onboarding process, such that now they question its value until later.
- When the hire is a transfer from another department we have 0 order to having that person set up with what they need. It is almost like they are forgotten about. ● Laptop setup and all the licenses we need to get them. I think that's going to be a lot better after this last group though, since IT is in charge of IDE licenses. We need to remember to make sure we have enough Github licenses though.

**What do you wish the new hires already knew before joining your team? For example, this could be specific skills, more familiarity with Awardco's product, or the development process.**

- Maybe more familiarity with the development process and Jira? ● outside hires- awardco platform; internal hires- how we do work/ dev process. ● dev process, jira workflows, etc
- We've gone from new hires complaining that they don't understand the Awardco platform very well at all, to complaining that they get too much training on stuff they don't really need to know. We also have recently added training on specific parts of the development process - boot camp projects that teach key architectural elements of back end, front end, or test code bases along with the development work flow. I think the next task is to refine these different training aspects.
- I think it would be more skill in relation to frameworks we use here and our development process.

**Any additional information you'd like to share?**

- With the engineers we've put a lot of effort into this over the last 6 months. We have a formal mentorship program, each new hire goes through product training with CS, and we've recently added the boot camp projects. We have also encouraged new hires to improve onboarding documentation in Confluence, which has been a big help.