SWECC Meeting 01/26/23

Meeting Notes – Internship Panelists

Panelists: Hrik, Hawa, Jape, Audrey, Edwin

Hrik: Developer for web3 at Origin, blockchain related, full stack developer, smart-contracts

Hawa: Blue Origin SWE intern, full stack, back end, front end, amazon, google, “another insurance company”, Microsoft this summer

Jape: google intern past 2 summers

Audrey: Microsoft intern, UC Berkeley research, more research, product management internship at F5 this summer

Edwin: meta, google, stripe interns, mobile dev (iOS), wants to work in back end

**What has been some of the best advice you’ve gotten that’s helped you get where you are ?**

* Hrik: go to networking events, meet people, find opportunities. I got a job at a networking event. It’s about who you know and the importance of referrals. (Btw, Hrik doesn’t even go to UW; he goes to NYU and just sits in random classes at UW, does the HW, goes to club meetings and takes any opportunity he can get)
* Hawa: Take risks and don’t think about it. Don’t be discouraged by being underqualified. Don’t reject yourself, try applying to jobs anyways. Take risks with a growth mindset.
* Jape: Do projects on the side, not just school. A lot of understanding comes from first-hand experience. Apply to \*\*a lot\*\*of applications. Ask people questions !!
* Audrey: Don’t be afraid to take risks and don’t be afraid to pursue opportunities. Did research at a biology lab, every experience you have will be useful for your next job/experience. That bio research helped her get to UC Berkeley research, then that led to another research, etc. Everything works out in the end, don’t be afraid to pursue things outside of CS as well.
* Edwin: Don’t memorize leetcode, try to understand it. Leetcode is a stepping stone, try to really understand it. It’s like math where you don’t memorize every math problem, but you use it to understand new topics

**How important would you say Leetcode is compared to personal projects ?**

* Hrik: for big tech, leetcode matters a lot. Depends on where you want to work. I’ve only worked at two startups, and they looked at my coding skills and open-source contributions. Startups may care more depending on what kind of culture they want to create.
* Edwin: Hackerranks, online assessments (code signal, etc.) are pure leetcode and are considered as a part of the application.
* Audrey: I don’t like Leetcode, but some interviews have OAs that require coding knowledge from leetcode. Leetcode is good practice, but it doesn’t guarantee a position. F5 interview was a face-to-face interview, and while I didn’t solve it, I was able to talk to people through the problem and did well on the interview. Interns are there to learn, they don’t expect you to know everything.
* Jape: At first I thought I had to be able to go through all Leetcode problems but it’s important to find that balance between personal projects and leetcode. Understanding the problem can be shown through code that isn’t 100 % correct. Try to do stuff outside of school too and balance your knowledge. You don’t have to be a leetcode god, an interviewer can tell if you just memorize a question.
* Audrey: I was asked to whiteboard part of my code from my resume, which shows it’s important to actually understand what you’re coding. Don’t just memorize.
* Hawa: Have a good mindset when approaching a problem. They’re looking to see if you can approach a question with the knowledge you have.

**What classes do you recommend taking ?**

* Jape: CSE 331 for understanding industry and CS. Creating a data structure, creating the API, then creating the backend to the API. CSE 344 (414 is the nonmajor equivalent) for databases. CSE 452 for distributed systems.
* Audrey: CSE 451 (OS) and the prerequisite. I was a math minor at my old university where I took complex variables (UW equivalent Amath 401)
* Hawa: CSE 331, and the INFO equivalent (server side and client side). 451 would’ve been helpful to take before my first internship. Take hard classes, learn to debug, and learn to solve difficult problems.
* Edwin: CSE 311 for boolean algebra and DeMorgan’s law. CSE 331 is useful for being able to think about code logically and mathematically. CSE 351 was fun but a lot of work. Hated 312
* Jape: Take classes that challenge you

<< Classes listed by panelists with links to MyPlan: >>

[CSE 331 - Software Design and Implementation](https://myplan.uw.edu/course/%23/courses/CSE331?states=N4Ig7gDgziBcLADrgJYDsAmB7MAJApigOYAWALsrAIwCsA7ACwA0yY62YACllCmSljSUADAF8QooA)

[CSE 344 - Introduction to Data Management](https://myplan.uw.edu/course/#/courses/CSE%20344?id=02a5a416-3b3d-4ae6-92c9-4ef0b9075c0d)

[CSE 452 - Introduction to Distributed Systems](https://myplan.uw.edu/course/#/courses/CSE%20452?id=d809a7b5-0711-4c5d-8b59-4a41545a4e6c)

[INFO 314 - Computer Networks & Distributed Applications](https://myplan.uw.edu/course/#/courses/INFO%20314?id=4febe701-12ce-42e5-8778-e7c663a601e4&states=N4Ig7gDgziBcLADrgJYDsAmB7MAJApigOYAWALsrAJwAcNANMmOtmAApZQpkpZqUBWAGwBfECKA)

[CSE 451 - Introduction to Operating Systems](https://myplan.uw.edu/course/#/courses/CSE%20451?id=a4c8e945-ae96-4909-b3d6-322d9fd2785d)

[AMATH 401 - Vector Calculus & Complex Variables](https://myplan.uw.edu/course/#/courses/AMATH%20401) (Currently not offered)

[INFO 340 - Client-Side Development](https://myplan.uw.edu/course/#/courses/INFO%20340?id=be4b5983-804b-4f57-8d7a-7713a0f2f556&states=N4Ig7gDgziBcLADrgJYDsAmB7MAJApigOYAWALsrAIwBMAHAOwA0yY62YACllCmSljSUAbLQB0DAJzTGUmpOF06AXxDKgA)

[CSE 311 - Foundations of Computing I](https://myplan.uw.edu/course/#/courses/CSE311)

[CSE 351 - The Hardware/Software Interface](https://myplan.uw.edu/course/#/courses/CSE351)

**What were the least useful or least favorite classes you’ve taken ?**

* Hawa: Didn’t like 331 but it was useful. Every class I learned something even if I didn’t like it. Try to take away something from everything you do :)
* Jape: Programming languages (341) a lot harder than I expected. Learn languages that aren’t usually used so it’s not the most important class.
* Audrey: CSE 312 and 331. There are important aspects but the front end part of 331 was hard (build the front end of the back end you designed)
* Edwin: 312

**What’s your favorite project you’ve worked on from your internships ?**

* Hrik: Not for a company, but my open-source project. There’s a blockchain theorem that has a problem with scaling. StarkNet is a scaling solution, so I worked on creating a byte code interpreter for StarkNet.
* Hawa: At Blue Origin, an operations monitoring tool for the rocket. Building a monitoring tool to ingest the data created from a rocket launch. Good mix of hardware and software like embedded and distributed systems.
* Jape: Making an API at Google… the concept wasn’t too hard. But the project was fun because I made a super in depth design for the back end. API took the tracking information from a shopping site and DHL. Got to learn about different features, testing, how users interact with different features.
* Audrey: Microsoft MakeCode, making educational software and hardware and designing a new project for kids. I designed a milk carton robot that opens its mouth with hand motion. Also a research project in MATLAB that created a software to model heart systems and blood flows in certain parts of the heart. Rewarding to see the outcome and the impact it had on other people.
* Edwin: This was for iOS and Google Meet. Using the dynamic island on iOS, integrated it with google meet. Found out it wasn’t possible and pivoted to making a widget for google meet instead.

**Why CS ?**

* Hrik: I didn’t choose CS at first, but blockchain was really interesting to me and I wanted to be involved in the community in a significant way. Decided to get into development to contribute to the blockchain community.
* Hawa: I didn’t know about software engineering until about Junior year. I like creating new technology and engineering. Want to get into business and entrepreneurship.
* Jape: I was seen as the techy guy since middle school, jail breaking iPhones and being good with technology. That label from other people carried me through. I wanted to know more about how everything works. My older brother is in software engineering as well. I was going to choose biomedical engineering and prosthetics, but my brother changed my mind last minute.
* Audrey: I’m going into product management full time next year, not software engineering. At my software engineering internships, I enjoyed thinking of solutions to problems. Learned during swe internships that I wanted to do more with design and research. PM doesn’t necessarily need technical skills but it’s good to know it to understand both engineer and designer opinions.
* Edwin: I didn’t know about CS in high school. Learned about it during COVID, was aiming for dental school before that. During covid started a free Codecademy subscription to learn how to code. Took math 207 as a chem engineering major at UW and took CSE 142, 143 then applied to CS and made it in. Also my handwriting is really bad, and I can’t make it to dental school like that lol

**How to learn about new technology and industry trends**

* Hrik: Tech is interesting because there’s no end, and there’s always new technology. I learned about new tech from social media and the internet. The algorithms on my social media pages are just geared toward new tech I guess.
* Hawa: I don’t keep up to date with new tech. I don’t like using my phone or social media. It’s okay if you’re not up to date with the newest tech, don’t jump on a bunch of random trends.
* Jape: CS is impossible to learn everything. As we speak, there’s like 10 different languages being made. It’s impossible to know everything.
* Audrey: When I saw people having a conversation about tech, I used to have a fear that I don’t enough about the topic to talk about it. There’s people who don’t keep up with everything, and don't worry about not knowing about the latest technology. Also make friends with PhD students.
* Edwin: I used to watch the YouTube channel Fireship which introduces new technology in like a 5 minute video. Keeping up with this stuff isn’t all part of your job. At Google, when there were new updates to a programming language for iOS, my manager would look through it and relay all the important information to the team. You don’t need to know everything.
* Hawa: Also just talk to people. Go up to random PhD students, they’ll talk about their research passionately, and you’ll learn a lot.

**What are your future plans ?**

* Hrik: Become a full time smart-contract contractor.
* Hawa: We’ll see. Maybe big tech
* Jape: Entrepreneur route but start in industry. Don’t know if the clock in, clock out lifestyle is meant for me.
* Edwin: Entrepreneur. Want to try interning at a smaller company, not big tech or join a startup.