

# Albert "AJ" Snow

(479) 774 - 1828 | [ajsnow2012@gmail.com](mailto:ajsnow2012@gmail.com) | [github.com/AlbertSnows](https://github.com/AlbertSnows) | [savyb.fly.dev](https://savyb.fly.dev)

## Education

**Kansas State University (2019)**

*Bachelor of Science in Computer Science (3.7 GPA)*

## Work Experience

### Senior Software Engineer

Feb 2024 - Present

*University of Chicago*

- TODO

### Backend Engineer

2022 - 2023

*Peerspace*

- Operated in several Clojure and JS (Node.js + Express) microservices which utilized MongoDB storage and GCP hosting
- Led team initiatives to implement requested features for top clients
  - \* For example, calendar sync, resulting in 30% initial adoption and positive post-release feedback via CX team
- Facilitated communication between services via two primary communication protocols
  - \* Direct calls were handled through REST APIs
  - \* Async calls leveraged Google's Pub/Sub architecture; typically to persist data in time-insensitive case
- Interfaced regularly with Kubernetes and Docker systems to deploy software images for testing and software releases
- Collaborated with teammates to create and scope design documents for new and legacy features
- Enhanced and standardized development environments via various initiatives (e.g. documentation, tests, dev configs)
- Optimized, refined, and standardized MongoDB schemas to accommodate new feature development
- Debugged complex issues involving multiple over-the-wire processes across the system (e.g. when inconsistent data fired faulty events)

### Software Engineer 2.5

2019 - 2022

*Paycom*

- Perform full stack (JS/PHP) operations, including creating a backend API and writing user-facing scripts
- Utilized MVC style framework, OOP style design, and software was regulated (ISO 9001:2015, SOX, PII)
- Lead initiative to design a new core feature on our team that added ability to customize *all* fields on company position; widely adopted by clients after release
- Expand various module features by adding data customization, new workflows, and new api routes
- Pushed performance improvements via code optimization, query aggregation, and ORM optimization (e.g. 100x speeds)
- Create DB designs based on feature requests and employ various data modeling and software design techniques
- Redesign the code structure in various parts of my code domain (e.g. refactoring code from 1k lines to 200)

### Assistant Instructor - Intro to Computer Science

2017 - 2018

*KSU*

- Provided mentoring to students, offering guidance on assignments, debugging, and best coding practices.
- Graded student projects, offering constructive feedback to enhance code quality and problem-solving skills.
- Held regular office hours to address students' questions, clarify concepts, and discuss software design.
- Shared insights to instructor about student progress, feedback, and contributed to curriculum improvement.

### Software Developer Internship

Fall 2018

*Xpansion*

- Completed comprehensive Java projects utilizing core MVC and OOP concepts such as polymorphism and inheritance
- Worked with company mentor on writing software automated testing frameworks
- Observed mentor's team sprint meeting and discussed project focus, goals, and implementation

### Instructor - Computer Science - K12

Summer 2018

*codeConnects*

- Offered guidance and counseling to students learning basic programming fundamentals in a summer program

### Apartment Custodian

Summer 2018

*KSU*

- Coordinated with a team members to prepare apartments for new residents quickly and thoroughly

- Coordinated with maintenance team to complete various aspects of building upkeep

## IT Technician

2015 - 2017

KSU

- Diagnosed and resolved computer-related issues, ensuring seamless functionality and optimal user experience
- Coordinated IT tasks with team members, streamlining operations and timely resolution of technical challenges
- Cultivated positive rapport with management and clients, achieved high satisfaction reports from clients

## Line Cook

2015

Cookies / Mema's

- Delivered food service in a timely manner under various levels of rush
- Performed regular maintenance on workspace including cleaning, inventory, and prepping

---

## Projects

<b>Personal Website</b> (CLJS, Fulcro)	A personal website using ClojureScript and the Fulcro framework for personal use
<b>Spring Boot State Machine</b> (Java)	Exploratory project to grok the Spring Boot framework; uses H2 and REST API
<b>Utility Functions Library</b>	Repo for experimental code concepts meant to be transferable between languages
<b>Order State Machine</b> (C++)	Example CLI state machine to experiment with core ETL concepts in C++
<b>Hotshot</b> (CLJS)	Basic API to solve a fun math case; utilized Calva for Interactive Programming
<b>Thermal Modeling</b> (Python)	College concurrency project; helped Graduated student parallelize modeling code
<b>Game Jam Unity Project</b> (C#)	Small game built by me in a weekend featuring lightbulb-themed puzzle platforming
<b>Modded Minesweeper</b> (Python)	Minesweeper but I added a feature to subtract numbers based on flagging
<b>Mobile App, Android</b> (Kotlin)	Simple app made from scratch to experiment with the Kotlin language on Android
<b>Snake</b> (Rust)	Built snake in Rust for exposure and to experiment with Rust's graphical libraries
<b>Flower Website</b> (CLJS/TS/JS)	Extra credit College assignment; wrote a website in three different JS languages
<b>2D Roguelike Tutorial</b> (Unity)	Involved tutorial demonstrating how to create a 2D rouge-style game

---

## Technical Experience

**Languages:** Clojure, JavaScript, PHP, Java, C#, Rust, Python, C++

**Tools/Frameworks/Libraries:** Node.js, Express, MySQL, MongoDB, Docker, GCP, React, Kubernetes, Spring Boot, JUnit, H2

**Skills:** Software Architecture, Database Design, API Design, Project Modeling, Algorithm Analysis, CI/CD, MVC, Functional-Programming, Microservices, ETL, OOP, TDD

---

## Accomplishments

A list of things to keep note of that I think are pretty cool.

- Independently developed new feature sets in 10+ Clojure and JS (Node.js + Express) microservice web applications
- Managed event-driven asynchronous communication protocols to process and interpret 100K-800k messages daily
- Reduced downtime in 3 separate services by ~10% by adding small optimizations in CI/CD pipelines
- 
- PAYCOM THING
- Introduced new data storage designs; employing data modelling techniques to store more than 30 new DB schemas
- 
- I IMPLMENTED PATTERN MATCHING IN JAVA. Utilized a assessment to flesh out what is essentially a kind of implementation of pattern matching in java; very happy with the results
- Made a demo state machine to \*really\* dig into spring boot framework
- Made a utility-functions repo as a primary placeholder for all the interesting software design concepts I come up with in my career
- Learned a bit about asp.net and C-sharp from their online documentation
- Learned a bit about angular

- Started exploring Spring Boot framework
- Learned about python APIs via flask
- Wrote a C++ state machine. Not optimal in terms of performance, but had some interesting data structures.
- Learned about calva; which makes interactive programming neat.
- Moved to Chicago; a hard decision since it would be moving away from friends and family
- Learned about .dotenv; super useful for mocking environments. Learned a little bit about microservices; they're pretty complicated. They do a lot of nuanced over the wire discussions that make tracking code flow difficult.
- Co-wrote calendar sync; learned about ics file types. Discovered that there's an interesting infinite loop problem with bi-directional syncing in calendar applications. Also learned about mocking design documents for systems.
- Re-made the website in Fulcro; probably one of the hardest things I've ever done
- Made a website in vanilla ClojureScript
- Adding mapping relationship to code to implement field customization for our team's core feature; the wizard
- Made the SPA which refactored a bunch of manually written code into a single loop. Brought 1k lines down to 200.
- Made that first page to add data at Paycom. It was a good first learning experience.
- Made an android app in kotlin
- Learned how to write snake in Rust
- Made a website for a college course in JS; typescript; and clojurescript
- Also passed Algorithm, Data Structures, and that logic class with Roby
- Passed 300, that class was hard
- Game Jam Project
- 3.7 GPA from college
- Graduated College
- High School Graduation
- Read philosophy for dummies (the most important accomplishment?)

---

## Open Source Contributions

- |                      |   |
|----------------------|---|
| <b><u>Calva</u></b>  | Collaborated with the lead dev to squash a bug preventing windows users from quick-booting the plugin   |
| <b><u>Fulcro</u></b> | Added a feature and documentation to the Fulcro template to build and deploy a production-ready uberjar |