Luke Cook

lukecook.e0280@passmail.net	Ithaca, NY | United States of America
https://www.linkedin.com/in/invocatis	https://www.github.com/invocatis
Bachelor's[Computer Science] - 2017	Rochester Institute of Technology

Senior Software Engineer (2022-2024) **Software Engineer (2020-2022)**	December 2020 - July 2024
Datasembly	**Remote**

Responsibilities

- Build & maintain web scrapers for grocery domain data
- Implement architectural advancements to web scraping framework
- Reverse engineer site APIs; adapt to scraping framework
- React to bot detection; develop and apply obfuscation techniques
- Work with Product and Customer teams to convert data to actionable insights
- Lead domain-specific dictionary initiative

Projects

- Developed headless browsing framework in Scala using Playwright
  - Proposed, by me, to replace the existing framework written in Puppeteer / Javascript
  - Playwright on the JVM greatly reduced development and deployment complexity; no longer needed to maintain a cluster of headless workers, or a remote execution DSL
  - Enabled scraping of websites that required Javascript execution, had sufficiently complex login APIs, or employed encryption of API request information
  - Framework was developed to easily allow different headless browser libraries to be used
- Primary Developer who created & maintained scrapers for Amazon, Instacart, and Shipt
  - Some of the most complex data sources we collected from
  - Instacart was an easy avenue for collecting new banners, creating fast turn around for new customers
  - Instituted new code paradigms that others could leverage to make scrapers more legible, and stable

Scala | Scala Cats | Puppeteer / Playwright | SQL | Snowflake | GCP | HTTP Protocol | Airflow | Web Scraping

Software Engineer Contractor	March 2020 - September 2020
Hyperfiddle	**Remote**

Responsibilities

- Development on Full Stack Clojure (script), Datomic CRUD application framework
- Setup CI/CD pipeline using CircleCI and AWS ECR and ECS, configured in Terraform

Projects

- Formalized extensible algebra for Datomic statements
  - Algebra supported simplification of negating statments (eg. 1 + -1 = 0)
  - Frontend logic would submit new statements on change, and the state would self simplify
  - Datomic statements were proved to form an Algebraic Group
- Research and Development
  - Reactive streams for rendering in Clojurescript
  - Category theory-based programming in Clojure
  - Pattern Matching algorithms

Clojure(script) | Datomic | Reactive Streams | Haxe | AWS | Terraform | Docker

Application Programmer II	June 2019 - June 2020
Cornell University	**Ithaca New York**

Responsibilities

- Improved backend of genomic data ETL application
- Created CSV file digest framework
- Developed end-to-end testing framework

Projects

- Worked on backend of genomic data ETL application

- Created CSV file digest framework
  - Provided a data file, and a descriptive aspect file, the application produces a stream tuples that would be committed to a database.
  - Aspects would be used to generate streams of cells; these streams would be aligned based on properties of each aspect
  - Written in Java; strong focus on an object-oriented design
- Developed framework for automated end-to-end testing; focused on composability, ease of use, and self-cleanup
  - Developed domain-specific scripting language to simplify semantic layer of test suit in 3 layers: Arrange/Act, Assert, and Cleanup
  - Was built in 3 separate components: A language backend (written in Clojure), a business logic middle layer (written in Java), and a testing logic script suite (written in custom syntax)
  - Cleanup ensured automatic test independence and idempotency; tests were often run on deployed production environments]

Java | Clojure | Docker | PostgreSQL | Object-Oriented Design

Projects

eldritch *Algebraic Data Types & Pattern Matching* https://www.github.com/invocatis/eldritch	**motif** *Recursive Pattern Matching in Clojure* https://www.github.com/invocatis/motif
ernie *Testing Framework for Java written in Clojure* https://www.github.com/invocatis/ernie	**Scala 3 Algebraic Type Exploration** *Exploration of algebraic concepts defined in Scala 3* https://github.com/Invocatis/scala3-algebraic-type-exploration