Baba Adam Issah

issah.adam.baba@gmail.com linkedin.com/in/adam-issah github.com/Rhomboic

Education

Cornell University **August 2021 – May 2025**

Bachelor of Science in Computer Science in the College of Engineering *Ithaca, NY*

Experience

Meta **July 2025 – May 2026**

Software Engineer · Instagram Media Creation Platform *Menlo Park, CA*

- Architected and optimized the core Android media infrastructure for Instagram, Threads, and Edits, managing high-performance networking stacks and data pipelines for global-scale image and video uploads.
- Engineered cross-app frameworks and libraries in Java and Kotlin, streamlining media composition and transcoding processes used by product teams to build new creative features.
- Optimized client-side resource utilization, focusing on minimizing memory overhead and latency for complex media workflows to ensure reliable performance for millions of concurrent users.
- Redesigned the "Self Threads" publishing architecture, transitioning from sequential to concurrent dispatching; improved upload latency (-16%) and ensured atomicity (all-or-nothing publishing) to eliminate partial-chain failures (-39% publish failure rate).
- Developed a Network Retry Simulator tool in collaboration with backend teams to stress-test upload/publish APIs, improving platform resilience by simulating request failures under adverse network conditions.
- Diagnosed a bug in Instagram Stories where reshares of expired or deleted stories were still attempted client-side, artificially deflating the publish success rate metric; implemented a fix blocking invalid reshares upstream and added UX informing users why a reshare was prevented (+6% Stories publish success rate).
- Onboarded Feed Carousel and the new Instants feature onto async publish infrastructure, improving publish success rate (+2%) and reducing publish failures for Instants videos (-27%).

Meta **May 2024 – August 2024**

Software Engineering Intern · Instagram Media Platform (Systems) *New York, NY*

- Rebuilt an internal video encoding testing tool to improve efficiency and reliability for encoding validation workflows.
- Added support for running multiple encoding tests simultaneously, enhancing testing throughput and scalability.
- Developed a robust recursive algorithm to compare video transcode JSON outputs between production and test groups.
- Revamped the tool's UI for better usability and clarity, improving developer productivity.
- Built a CLI to extend accessibility of the testing framework across the company, fostering widespread adoption.

Meta **May 2023 – August 2023**

Software Engineering Intern · Instagram *Menlo Park, CA*

- Introduced 3rd party app attribution feature to Reels on Instagram Web, leveraging GraphQL and React.
- Wrote End-to-End (E2E) tests using Jest framework - ensured stability and compatibility across browsers and devices.
- Implemented comprehensive performance logging and alerts, capturing key metrics such as impressions, click-through rate (CTR), render latency and more.

Meta **May 2022 – August 2022**

Meta University Engineering Intern · iOS *Seattle, WA*

- Built a maps-centered iOS fitness application with reliance on the Apple Maps API for route queries and user location data - implemented using Objective C.
- Designed an algorithm to efficiently calculate interception paths and route users to a suitable rendezvous point.
- Collaborated closely with cross-functional teams to ensure UI and functionality was up to par with Meta's production level offerings.

Projects

FireDrill: RL-Compatible Incident-Response Agent Gym | firedrill.adamissah.com **June 2026**

- Built a Gymnasium-style agent gym (reset/step/verify, snapshot/restore) fully decoupled from any policy: an LLM agent, a human, or an RL training loop can be dropped into any of 10 sandboxed broken software projects and stepped through a 5-tool action space with reward queryable at every step.
- Designed 10 polyglot scenarios (Python, Node, React/Playwright, SQL, Docker) each targeting a distinct failure mode with an objective verifier that fails before the fix and passes after; made verifiers un-gameable by restoring protected grader files from a golden copy and planting regression traps that pass the naive fix but fail a held-out check.

- Scored a 4-dimension reward (resolution, blast radius, and LLM-as-judge diagnosis quality) with cost as a separate first-class axis: cache-aware token usage priced to dollars plus a saturating $k/(k + \text{cost})$ score, so capability and price are compared independently rather than blended.
- Provisioned the system as code with Terraform: ECS-on-EC2 with an Auto Scaling Group and managed-scaling capacity provider that scales the fleet from zero on task demand, plus ECR, S3, Secrets Manager, CloudFront/Route53, and GitHub OIDC CI (plan-on-PR, apply-on-merge); shipped a React results dashboard reading live from S3.
- Surfaced a cost-vs-capability story: small baselines matched reasoning flagships at 97% resolution for up to 24x less ($0.026/job for Claude Haiku vs $0.63 for Opus), while regression traps and the hardest scenario (which capped even Opus and GPT-5.5 at 0.76) isolated the real capability gap in fix precision.

PuzzleChess: LLM Chess-Puzzle Eval Benchmark | chess.adamissah.com **May 2026**

- Built an end-to-end eval gym benchmarking 7 frontier LLMs on 300 Lichess mate-in-N puzzles; designed a composite reward signal (correctness, partial move credit, format compliance, latency) that captures where in the reasoning chain each model breaks down, not just pass/fail.
- Iterated the verifier through three designs: exact match, then over-corrected to any mating line (which let the model play both sides), then landed on forced-solution matching with alternate final mate credit, distinguishing genuine model failure from environment artifacts.
- Provisioned all infrastructure as code with Terraform (ECR, ECS, S3, IAM/OIDC, CloudFront, Route53) and automated deploys via GitHub Actions; API keys kept in AWS Secrets Manager, never in the image.
- Surfaced that reasoning effort, not model size or provider, determines chess capability: o3 scored 76% vs single digits for all non-reasoning models; extended thinking lifted Claude Opus 4.8 from 11% to 54%, confirming the finding cross-provider.

Technical Skills

Languages: Python, C++, Java, Objective-C, JavaScript, HTML/CSS, OCaml, HCL
Cloud/Infra: AWS (EC2, ECS Fargate, S3, IAM/OIDC, CloudFront, CloudWatch, Route53, ECR, Secrets Manager, Bedrock, SageMaker), Terraform, Docker, GitHub Actions (CD)
Web/Backend: Node.js, React, Next.js, Flask, MongoDB, GraphQL, Git