

# ADWAIT KULKARNI

(604) 720-4861 | [adwait.kul.2018@gmail.com](mailto:adwait.kul.2018@gmail.com) | <https://www.linkedin.com/in/adwaitkulkarni58/> | [GitHub](#) | [Website](#)

## EDUCATION

**University of British Columbia**  
**BS in Computer Science and Minor in Data Science**

**Vancouver, BC**  
**September 2020 – May 2025 (Expected)**

- Developer in UBC Subbots, GDSC UBC, and UBC DSCI clubs. Past participant of Google's invite-only Software Product Sprint.

## TECHNICAL SKILLS

**Languages:** Java, Python, JavaScript, HTML/CSS, R, C/C++

**Frameworks/Libraries:** Spring, React.js, Node.js, Express.js, pandas, Scikit-Learn, Material-UI, Mongoose, JUnit, Vitest

**Tools/Technologies:** Apache Kafka, Redux, Git, GitHub Actions, Docker, SageMaker, Jenkins, AWS, OpenShift, Cloud Foundry

**Databases:** MongoDB, MySQL, RDS, Redis, H2, Elasticsearch

## EXPERIENCE

**Full-Stack Developer Intern**  
**Royal Bank of Canada (RBC)**

**Toronto, ON**  
**April 2023 – August 2023**

- Reduced development time by 70% for over 200 developers by launching a Java Spring and Thymeleaf-based secret retrieval tool on RBC's Innersource platform, using Hashicorp Vault.
- Devised 5 REST services post-HSBC acquisition, and built a Kafka app for real-time monitoring of 20+ production topics.
- Developed over 10 features for RBC's open-source projects using Java and Web technologies, composed technical documentation resulting in a 50% increase in project adoption, and resolved 20+ bugs.
- Spearheaded dev onboarding by creating development setup and environment access guides and mentored 3 new developers.
- Chosen as 'Top Talent' from 1400 interns; discussed key projects with executive VPs and Directors in a round-table conference.

**Full-Stack Developer Intern**  
**Royal Bank of Canada (RBC)**

**Toronto, ON**  
**August 2022 – April 2023**

- Architected microservices with Java Spring to extract deposit accounts from mainframe databases, process, and store them in Kafka topics, and using Redis cache, brought down data recovery time by 25%.
- Improved critical storage infrastructure for 20+ apps, covering 100,000+ profiles and 500,000+ client records.
- Reduced tech debt by 40% in JIRA by updating Maven service dependencies, transitioning secret storage from local YAML environments to online platforms, and releasing hot-fix patches for outdated third-party libraries.
- Authored cloud migration guides for OpenShift and Cloud Foundry and independently migrated 10 apps across cloud providers.

## PROJECTS

**TL;DReviews** | *React.js, AWS Lambda, Amazon SageMaker, AWS RDS, AWS S3, Docker*

**August 2024 – December 2024**

- Architected a scalable, end-to-end pipeline that processed and analyzed 6,000 UBC course reviews using AWS services (Comprehend, Lambda, SageMaker, S3, RDS), providing student-focused feedback and enabling real-time insights visualization.
- Built a text summarization ML model with Google's PEGASUS XSUM on AWS SageMaker, enabling automated batch processing of student reviews and generating course-level insights in 0.26 seconds per review, improving scalability for large datasets.
- Secured sensitive data by implementing advanced AWS security protocols, including VPC deployment, IAM role policies, Security Groups, and server-side AWS KMS encryption at rest and in transit, ensuring compliance with CIS/CISA and NIST standards.
- Developed a visualization dashboard with Recharts and D3.js, presenting word clouds, sentiment, and confidence level graphs.
- Cut costs by \$20/mo on SageMaker endpoints and idle assets by replacing VM-based EC2 solutions with Lambda auto-scaling.

**MovieHub** | *React.js, Node.js, Material-UI, Express.js, MongoDB, TMDb API, Docker*

**April 2024 – September 2024**

- Led a team of 4 to engineer a full-stack movie recommendation app, with options to rate, review, share, and stream movies.
- Formulated end-to-end encryption with BCrypt hashing, increasing security and reducing unauthorized access by 60%.
- Modeled evolving watchlists for user preferences using MongoDB Schemas and leveraged caching to reduce latency by 50%.
- Performed CI/CD with GitHub Actions, enhanced automated tests using Vitest and Postman, and deployed with Docker images.

**Upbeat Updates** | *React.js, MongoDB, Node.js, Express.js, Sentiment*

**January 2024 – July 2024**

- Created a positive news website using News-API with React.js, Material UI and Express.js, MongoDB on Vercel, ensuring secure authentication with BCrypt and handling over 1000 daily requests.
- Implemented sentiment analysis using Sentiment and visualization with React Charts to analyze real-time sentiment of articles and sharing functionality, improving Customer Satisfaction Score (CSAT) by 30%.
- Carried out unsupervised learning with K-means in Scikit-Learn, used Elbow and Silhouette plots for hyperparameter tuning.