# Jing Yi (Jimmy) Li

1329 Birchcliff Drive
Oakville, ON, Canada
L6M 2A5

Github: jimmy1231
Cell: (647) 922-5012
dfjimmy.li@gmail.com

## EDUCATION

**University of Toronto** **Sept 2015 – May 2020**

**Bachelor of Applied Science (B.AS.), Computer Engineering major**

## EXPERIENCE

**FutureVault Inc.** **August 2019 – Present**

**Software Engineering Consultant (AWS, Serverless, Java)**

- Serverless SFTP file ingestion with OCR, file-name suggestion
- POC's involving AWS+SAML SSO

**FutureVault Inc.** **May 2018 – Aug 2019**

**Software Developer Intern - PEY (AWS, Serverless, Java, Node.js, React.js, Python)**

- Design and implementation of a Serverless Document Processing model
- Internal SDK: Frontend → Document Processing
- Internal SDK: Multi-[AWS]-region compatibility
- AutoFile: ML-based filing suggestion system (scikit-learn, SciPy)
- Performance profiling on Autofile-AI and OCR - ELK stack+Tableau
- Key user flows: "forgot-password", "change-password", "MFA", "sign-in", "onboarding" (full-stack)

**RBC** **May 2017 – Aug 2017**

**Developer Co-op (Angular.js, Java)**

- Several admin-side web-pages (full-stack)
- Database query performance tuning with JPA+Hibernate+Spring Boot

## TECHNICAL SKILLS

| Skills | # Yrs Exp. | Highlights |
|---|---|---|
| Java | 4 | Spring/Boot, JPA, Apache: [ Maven, Tika, Tomcat, OpenNLP ], Mockito, AWS SDK |
| C/C++ | 5 | make, cmake, OpenGL, libigl, gprof, gcov |
| JavaScript | 3 | Node, NPM, Express, Angular, React, Mocha, Chai, AWS: [ Amplify, Node SDK ] |
| Database | 3 | MongoDB, MySQL, PostGres, CSC343 - Relational Database |
| AWS | 2 | Lambda, CloudFormation, API Gateway, Cognito, S3, Step Functions, IAM, KMS, EC2 |
| Other | 2+ | Python: [ scikit-learn, SciPy, AWS SDK ], HTML, SVG, CSS, Agile, Git, REST, Atlassian tools, Tableau, ELK stack, UML, Docker |

## PROFESSIONAL CERTIFICATES & TRAINING

- Oracle Certified Associate (OCA) Java SE8, Oracle Corporation

## PROJECTS

**Function/Graph Visualizer with SVG** (1-person)

- Plots multiple functions on one x-axis and, optionally, on 2 y-axes via incremental sampling
- Fluid re-rendering (up to 60 FPS) during user interaction

**Implementation of *malloc* in C: Based on *dlmalloc*** (1-person, ECE454: Systems Programming)

- Dynamic segregated list to minimize internal fragmentation
- Dynamic preallocation, on-demand coalescing on free to maximize speed
- Rank 3 (out of 75) in class with 100% speed, 95% memory efficiency