

# Selina Li

University of Waterloo / Bachelor's of Computer Engineering

✉ [g339li@uwaterloo.ca](mailto:g339li@uwaterloo.ca)  
🐙 [github.com/Selinaqianli](https://github.com/Selinaqianli)  
in [linkedin.com/in/qian-selina-li/](https://linkedin.com/in/qian-selina-li/)  
🌐 [selinaqianli.github.io/](https://selinaqianli.github.io/)

## QUALIFICATIONS

**Languages** : C#, Java, C++, SQL, TypeScript, JavaScript, HTML, CSS, Python  
**Technologies** : ASP.NET, Spring Boot, Angular 8, Angular JS, Redis, MyBatis, Servlet-JSP  
**Tools** : AWS Lambda, AWS S3, Apache Maven, Apache JMeter, Apache Tomcat, Microsoft SQL Server, MySQL, Talend API Tester, Postman, Git, Ubuntu

## PROJECTS

### Login API

- xxx

### Portfolio

- xxx

### Data Management System

- xxx

## WORK EXPERIENCES

### Software Developer Co-op

Sep. - Dec. 2020

University of Waterloo, Waterloo, ON, Canada

- Worked full-stack on TA application system that ensures the for covid-19
- Implemented email sending to notify instructors and TAs about their matching status by using the **SMTP** (Simple Mail Transfer Protocol)
- Created **unit tests** in the AAA (Arrange-Act-Assert) pattern to verify the behaviour of controllers by breaking down the functionality into discrete testable units
- xxx

### Software Developer Co-op

Jan. - Apr. 2020

Dealer-FX, Markham, ON, Canada

- Developed an **AWS Lambda** function to process Amazon SQS (Simple Queue Service) message events and redirect them to VPC-specific (Virtual Private Cloud) lambdas
- Implemented **health checks** in .NET Core apps to confirm availability of the app's dependencies (such as databases and external service endpoints)
- Developed a webcam **Angular** component, which allows users to capture photos, and automatically update the corresponding image URLs of other components by two-way data bindings
- Created **RESTful API** endpoints in .Net Core apps to interact with the database

### Backend Developer Internship

Jun. - Aug. 2019

XYebank, Chengdu, Sichuan, China

- Developed a log-in API with **Redis**, that will automatically lock the account for 15 minutes after 5 attempts to access it with wrong passwords
- Built a CRUD (create, read, update, delete) **RESTful API** with **SpringBoot** in Java and configured dependencies with Apache Maven
- Enabled **single sign-on (SSO)** to allow users to log in to any several related but independent software systems using a single ID and password
- Load tested the web application by **Apache JMeter** and reduced duplicate database accesses by Redis to upgrade the website scale