# Tyler Adams

tjadams@uwaterloo.ca - http://tjadams.ca - https://github.com/tjadams

## WORK EXPERIENCE

**Microsoft** Aug 2018 - Present

*Software Engineer – Microsoft Advertising Campaign Platform (formerly Bing Ads)* Bellevue, WA

- Specialized in full-stack web development and distributed systems
- Task Engine – Distributed system
  - **Owned a distributed system** responsible for offline task scheduling, processing, and management. Supported 100s of types of tasks each with different parameters and schedules
  - Mitigated and resolved production and test environment issues towards maintaining system correctness, low latency, and high availability. On-call efforts led to improvements which resulted in maintaining and increasing the current scale. Task Engine supports a throughput of ~3000 SQL sproc executions per second, 99.9% availability, and an average API call latency of less than 1 second
  - **Owned a cluster of 25 Azure SQL DBs** partitioned over multiple servers. Developed T-SQL stored procedures and improved performance by modifying table schemas and indexes
- Designed the data migration approach for advertiser targeting data critical to the business. Implemented the ad-hoc load balancing logic in C# to migrate data between Azure SQL databases with high reliability and minimal downtime for users. Updated T-SQL scripts to allow the databases to support a locking mechanism for data in migration. **Successfully migrated gigabytes of production data** with no customer escalations or data loss
- **Implemented backend APIs, stored procedures, database schema changes, and user interfaces** for several new Advertising features such as: Cookie-based experiments, Feeds, Dynamic Search Ads, Website Exclusion Lists, Enhanced Manual Bidding, and Import Google Shopping Campaigns. Implementation was done via Knockout.js, React, Backbone.js, Pug, C#, .NET Core, and T-SQL
- Analyzed API call patterns for creating and updating Ads. Identified sub-optimal database calls. Implemented lightweight T-SQL stored procedures to get only the information required for the API calls. Called the new procedures in the C# backend which **reduced P90 latency by 30% (~250 ms)**
- During on-call rotations, **mitigated and root-caused production tickets** due to low availability and high latency. Analyzed query plans to find memory, CPU, and latency issues in Azure SQL databases and fixed the corresponding T-SQL stored procedures. Maintained uptime, throughput, low latency, and correctness for distributed data pipelines, user interfaces, databases, and backend APIs
- Removed faulty servers from rotation and took memory dumps. Also **assisted in analyzing low memory of production backend machines** as well as finding and fixing the root-cause

**Microsoft** Sept – Dec 2017

*Software Engineering Intern* Bellevue, WA

- Created a tool in C# that aggregates **millions** of rows of telemetry data from **Bing Ads Editor** into sequences of features by inputting the data into a state machine and priority queue
- Filtered **50 gigabytes** of telemetry data from production by developing database queries
- Determined common behavior of monthly active users that create 60% of Bing Ads

**Amazon** Jan – Apr 2017

*Software Development Engineering Intern* Santa Cruz, CA

- Designed and implemented the predictive downloading and caching feature of the Amazon Seller Android app which reduced latency by an average of **31% (902 ms)** on the homepage of the app
- Manual and functional testing resulted in the deployment of code to **~2,000,000 users** without bugs

**Amazon** May – Aug 2016

*Software Development Engineering Intern* Seattle, WA

- Created an internal website which enabled content on **Amazon Vendor Express** to be edited by product managers, and reduced deployment time from **5 days to 5 minutes or less**
- Coded APIs to read and write to an AWS DynamoDB database using Java and the Spring framework

**Misc. 4 month internships (Rogers, Thalmic Labs, Virtual Power Systems)** 2014 - 2015

*Software Engineering Intern* Toronto, Waterloo, and Santa Clara

- Developed a data center simulator in Scala which tested power distribution software using Akka
- Implemented the overlay used in Myo for Presentations on Windows and OS X using C++ and Qt
- Coded the Sportsnet Google Glass Android app which included NHL game scores, videos, and radio

## EDUCATION

**University of Waterloo** Graduated June 2018

*B.A.Sc. in Computer Engineering (Co-op program)* Waterloo, ON

- Gained 2 years of software engineering internship experience across 6 internships during co-op terms
- Attended 10+ collegiate hackathons hosted by Major League Hacking and others. Hackathons included: Hack the Planet (invite-only for top collegiate hackers), Hack the North, and more

**Continuous learning** 2018 - Present

- Books: Clean Code, Designing data-intensive applications, Introduction to Algorithms ("CLRS")

## SIDE PROJECTS

**MMORPG Server -** https://github.com/tjadams/Dedicated-Online-Server 2014

- Coded a server and database schema for MapleStory v62 using Node JS and MySQL. Implemented a subset of functionality such as login and character selection. Reverse engineered packets sent from the MapleStory client to localhost with the help of an online reference

**Produce –** Android app published to Google Play 2013 - 2014

- Built and published a productivity Android app which allowed users to lock usage of certain apps
- Created a prototype called SmartyPants which won 2nd place during 2013's Startup Weekend Toronto

**Personal blog –** http://tjadams.ca 2014 - Present

- Wrote about the tech industry and maintained an Ubuntu instance on a DigitalOcean droplet

## SKILLS

- Languages: C#, JavaScript, Java, Python, T-SQL, Powershell, HTML, CSS
- Libraries and frameworks: .NET Core, .NET Framework, React, Knockout.js, Pug, Backbone.js, selenium-webdriver, jQuery, Underscore.js
- Platforms: NodeJS, Android, RabbitMQ
- Tools: Git, SQL Server, Azure SQL Database, Azure Cloud Services, Azure Blob Storage, Azure Table Storage, Azure DevOps, SQL Server Management Studio, Visual Studio Code, VS 2019