

# JORDEN LOWE

## SKILLS & EXPERTISE

- o C#
- o SQL Server
- o OAuth2
- o REST APIs
- o GraphQL
- o JavaScript/NodeJS
- o TypescriptDatabricks
- o Docker
- o Linux/Mac/Windows
- o Azure
- o Azure DevOps
- o Terraform
- o Git
- o Visual Studio

## WORK EXPERIENCE

### Senior Software Engineer - Health Catalyst

05/23 - Present

Working on the Pop Analyzer and Insights team where we develop features to take data from a data warehouse and create populations and visualizations. Also responsible for all the platform applications of our databricks system. Tech: SQL (Postgress and SQL Server), Databricks, .Net, Azure, Azure DevOps

- o Built solution to import schema information from Databricks for configuring the query engine to generate a population.
- o Redesigned documentation wiki to organize popular information, like Runbooks, design, How-Tos, and more.
- o Lift and shift application from on prem solution to Azure Native solution.
- o Developed infrastructure in Bicep and Terraform.
- o Created Pipelines for deploying code and infrastructure.
- o Give Clients support and guidance on how to use the product.
- o Researched Pivot Table alternatives for showing Populations to allow users to drill down to details for finding trends and making decisions.
- o Implemented Custom Columns allowing users to create functions using NCalc to allow new columns to be created within Populations.

### Senior Software Engineer - Linus Health

11/22 - 05/23

Integrating with APIs of EHR's to allow our Brain Health Platform to send information from a Brain Health Assessment. These integrations use Node, typescript, AWS, Terraform, Github

- o Build infrastructure using terraform in AWS.
- o Participate in code reviews
- o Log errors to S3, DynamoDB and Cloudwatch
- o Manage and maintain the full end to end integrations with Athena Health APIS and Redox (FHIR)
- o Build Design Documents of the Lambda Orchestrations in Confluence

### Senior Software Engineer - Cardlytics

10/21 - 11/22

Part of the Data Team, managing APIs (REST and GraphQL) to expose processed data in dashboards and reporting.

- o Organized the team's documentation and links to help optimize onboarding and share information with the team easily.
- o Worked with developers from India, Ukraine and all over the US to build the applications under our charter.
- o Developed GraphQL and REST endpoints to help optimize API client's access to data.
- o Deploy and Maintain the applications' CI/CD pipeline using Octodeploy and TeamCity.
- o Shifted applications from on prem into the cloud with the help of Terraform, Kubernetes and Docker.

- o Transform our applications to utilize less code but more of AWS's platform as a service.
  - o Use API gateway to front APIs
  - o Use lambdas instead of timed applications
  - o SQS instead of standing up Kafka/Message Queues
- o Diagnose and fix issues related to oncall activities.
- o Write Unit/Performance/UAT tests to verify the code works, is fast and will deploy with no impact to downstream users.
- o Mentor junior devs to start their career and how to navigate a large company with enterprise level software.
- o Offer help to anyone in need and answer as many questions in slack as possible.

## **Senior Software Engineer - Great Lakes Psychology Group**

**11/20 - 10/21**

Working directly for COO to lead team with experience on Cloud technologies and Healthcare to build an EHR for the company.

- o Mentored team on development process and designing user stories.
- o Designed and reviewed initial technologies and architecture to use.
- o Managed MongoDB and Realm applications in hosted solution.
- o Created authorization system used for the application using MongoDB Realm to authenticate the user.
- o Built initial GraphQL API from MongoDB Realm.

## **Senior Software Engineer - Health Catalyst**

**04/17 - 11/20**

Patient Safety Team - Angular application that reads from the In-Home data warehouse to display patients with safety issues and alert doctors and nurses to help keep the patient safe and healthy.

- o Angular js application with .net core backend and SQL Server database.
- o Developed the automatic testing in our CI pipeline in Azure dev ops and gave weekly updates of our progress in .net and Angular
- o Designed the client side implementation of Authorization for the Patient safety Roles/Groups/Users and Permissions
- o Created Filter system on the backend using a strategy pattern to commonize Filtering approaches.
- o Used: .net core 2.0 and Angular 8.0.

Security Team - Merged with the data team, the security team managed Authentication via Azure AD, AD and OAuth2 (using Identity Server), as well as Authorization

- o Managed the whole build process using docker images in Azure DevOps using builds and releases.
- o Implemented Azure AD into our Identity Server integration to help clients who don't like AD trusts in their network.
- o Built features into an Authorization site to manage permissions/groups/roles in our Authorization service.
- o Supported after hours support for permissions.
- o Used: .net 4.6, .net core 2.0 and Angular 6.0

Data Team- Responsible for development and release of Metadata of the data warehouse for the Health Catalyst platform. The platform is a data warehousing and analytics platform that accesses EHR/patients/Billing work for health systems. These insights help save patients to hospitals' overall budget.

- o Managed the whole build process using Azure DevOps using builds and releases.
- o Added OAuth integration into the application to lock down access to the metadata.
- o Mentored new team members and was team's scrum master
- o Maintained a strong unit and integration test suite for easier regression testing.
- o Worked remote with the teams from all over the country in other timezones.

## **Software Project Lead - Panasonic Automotive**

**04/16 - 04/17**

Aupeo - Responsible for development and release of a cloud music service running in AWS. This included using my experience with documenting process, APIs and creating developer VMs to make a developers life quicker and easier to maintain the software. The service was built in Ruby, MySQL, PostgresDB and CassandraDB. Duties Include:

- o Working with remote engineers to complete tasks from Ukraine.
- o Act as Product Owner to investigate and Create Requirement Documentation.
- o Worked on projects like Audio Level Management, hybrid recommender and integrating the project with TheFilter to manage personalized music playlists.
- o Creating Release Plans and Guiding a team of 5 developers on 3 Releases.
- o Developed API Documentation using swagger to describe the Cloud Music Service.
- o Developed Docker Images for database and service environment.
- o Developed Vagrant VM to make development setup experience faster.
- o Developed Confluence documentation of all our internal processes, including environment locations, procedures, setup guides, release notes, swagger documentation and more.
- o Helped troubleshoot logs which enabled engineers to fix bugs quickly.
- o Established a GitFlow like branching scheme.
- o Worked together with team of engineers and dev ops to smoothly deploy applications in under 30 minutes.
- o Used Jira to track bugs and collaborate with the team on what is left in the release.
- o Worked with outsource team to build a concept data warehouse.

## **Senior Software Engineer – Quicken Loans**

**01/12 to 04/16**

API Gateway Team – An API gateway is enterprise software to aggregate and secure all the API's within Quicken Loans. This solution helps manage cross-cutting concerns, like security, metrics and documentation. It is an integral part in allowing our business to integrate our cloud software with our on premise software. The system is java-based and used a RESTful modeling language called RAML. It runs on CentOS servers deployed in AWS and on premise.

- o Mentored, Consulted and Designed developer awareness of building RESTful services.
  - o Used modeling in a language called RAML
  - o Workshops and mentoring sessions
  - o Developing presentation materials
  - o Using Confluence to document all aspects of our API's
  - o Improved developer efficiency when generating REST clients.
  - o Building clients based off RAML specification
  - o Creating tools around managing client Id and Secrets for their API's to allow searching and migrate from environments easier
  - o Created RAMLsharp, a tool to automatically generate their API RAML
- o Ensured the most uptime by monitoring the servers of the API gateway with logs
- o Identified performance bottle necks in TLS 1.2, VPN tunneling, clustering and JVM using JMeter to ensure the architecture was optimally performant.
- o Brought new ideas to QA and load testing teams like service virtualization (mountebank) so that they could mock 3rd party services while testing End to End.

Document Repository Team - The mortgage process relies heavily on documents and we used OnBase to store ours. Not only does OnBase store and retrieve documents, it also has a rules engine to allow business rules to route documents in the organization. OnBase uses a combination of file system and SQL Server database to store documents. It has a .net API wrapper and now, a RESTful API.

- o Empowered developers in the company to access documents without our intervention by creating a RESTful API using Hypermedia (HAL media type) to access documents.
- o Since the .net API wrapper takes time to execute, we used a message queue to drop updates for another windows service to process.
- o Utilized cache (Redis) to make sure queries from the RESTful service responded quickly.
- o In my spare time I redesigned and built a document exporting system that dropped investor onboarding from one week to a couple of hours.
- o The document export system used a workflow-like architecture and plug-in model, using Razor Engine, to dynamically change investors on the fly, reducing wait time on new investors.
- o The document export system exported several million documents a week to 50+ different clients.

- o I mentored engineers through talks, one-on-ones and panels to drive home Service Oriented Architecture, Microservice design, API design, unit testing, continuous integration and design patterns. My goal was to give developers a good sense that design is within their reach.

## **EDUCATION**

**Master's Degree in Computer Science** - *University of Michigan, 2009*

**Bachelor's Degree in Computer Science** - *University of Michigan, 2006*

## **TRAINING / CONFERENCES**

- o 2024

Azure AZ 900 Certification