Benjamin Berman

Location: Pittsburgh, PA,
Willing to do remote work

bnjmnbrmn@gmail.com

http://bnjmnbrmn.com
http://github.com/bnjmnbrmn

Overview

I am a software engineer with a **PhD** and **9+ years of experience in industry**. I continue to love writing new high-quality code, debugging and improving existing code, learning new skills and systems, helping my colleagues and our customers, and finding the right solution for the situation.

General problems I've enjoyed solving relate to microservices, REST APIs, streaming data , data storage (sharding, transactions, high availability, SQL and NoSQL), reliable and repeatable deployment with zero downtime, cloud infrastructure (IaC, provisioning, AWS, cost tradeoffs), performance (latency, throughput/bandwidth, resource consumption, parallization), reliability (logging, monitoring, testing, configuration minimization, code readability, redundancy, debugging of concurrency and memory issues), security (authorization, authentication, encryption, availability, integrity), DevOps (automation, continuous integration, containerization, orchestration), communication (technical and semi-technical presentations, documentation, multi-team coordination) and developer productivity (build speed, build tools, local environment setup, mentoring, knowledge sharing).

Some specific technologies I've used professionally include Java, Spring Boot, Kafka and Kafka Streams (please see experience with CAPD Data Services at 3M/M*Modal), Scala, Bash, Docker, Kubernetes, AWS, Git, Gitlab CI/CD, Maven, Gradle, Elasticsearch, PostgreSQL, MySQL, OAuth, JavaScript, TypeScript, Ruby, Ansible, Terraform, MongoDB, and Redis.

Work Experience

UPMC Enterprises, Pittsburgh, PA

***Senior Software Engineer**, February 2022 to April 2024*
Engineer on the Alexandria Charts medical records platform team. The main challenge in this position was the **wide range** of applications/services to be upgraded, improved, and debugged. I was also heavily involved in improving the security, reliability, and performance of our build and deployment processes. Applications/microservices included: "Document Viewer", a web app supporting searching patients' medical records across multiple EMR sources, viewing the highlighted results in multiple formats, and downloading the records in bulk; "Identity Reconciliation Tool", a web app allowing medical documents from multiple EMRs to be linked to the same patient; "Bulk Extraction", a tool for downloading zip files of medical records, based on searches (I improved performance by more than 3x); "Governor", a tool for appropriately transferring medical records between tenants; and "Indexing Service", a frontend for Elasticsearch, providing a domain specific language for searches (work included a major upgrade from Elasticsearch 7 to 8 in Kubernetes/EKS)

Technologies used in these projects included: Java, Spring Boot, Docker, Kubernetes, AWS (Secrets Manager, EKS, Parameter Store, S3, Cognito, RDS, SQS), Bash, Elasticsearch, PostgreSQL, MySQL, OAuth, JavaScript, AngularJS, TypeScript, Angular, Ruby, Ruby on Rails, OAuth, Azure AD/Entra, Git, Gitlab, Gitlab CI/CD, REST, RabbitMQ, Postman, newman, treetop parsing DSL.

M*Modal / 3M, Pittsburgh, PA

(3M acquired M*Modal in February, 2019, integrating it into its Healthcare Information Systems Division but keeping the brand)

***Senior Software Engineer**, July 2017 to February 2022*
Primarily Java/Spring Boot, with heavy usage of supporting technologies, plus leadership (included time as **tech lead**), architectural, and mentoring activities. The umbrella project/product I was responsible for was Computer Assisted Provider Documentation (CAPD) Services, a web service which connected a desktop client interacting with an Electronic Medical Record (EMR) system, such as Epic or Cerner, with a real-time natural language processor, providing layers for security and feedback customization. Activities related to this included:

- Designing and developing CAPD "Data Services"
    - Primary designer and implementer of a greenfield set of microservices, the core of which used Apache Kafka and the Kafka Streams library to gather and process ~1000 requests/second of data from our clients in real time, primarily for reporting purposes
    - Worked across teams (the desktop application developers generating data, the reporting application developers displaying the processed data, the maintainers of the legacy batch ETL system, and product/sales) in a high pressure situation to explain and resolve critical ambiguities:
        - Sessions -- do they resume when a provider comes back from lunch?
        - User facing alert resolutions -- what state transitions count? How should they be debounced?
        - Document drafts vs final documents
    - Aggregated, as well as mapped/filtered, incoming messages
    - Dealt with out-of-order messages, client restarts, and network interruptions
    - Included an HTTP API, backed by PostgreSQL, to allow paginated access to both historical and current results
    - Wrote clear, concise, functional code to help guarantee correctness
    - Successfully deployed to multiple AWS environments, including production
- Adding support for Encounter-Based Reasoning
    - Connected the real-time natural language processor with the results of processing completed documents from a customer EHR system
    - Required new systems to manage user and identifier mappings, configuration and credential caching, integration of JSON Web Tokens, fallback mechanisms, API call for extended NLP evidence, and improved metrics, monitoring, and logging
- Implementing the Notereader CDI interface
    - An intercompany project, connected 3M/M*Modal's NLP with Epic to provide near real-time encounter-based reasoning directly in the EMR
    - Modeled and stored provider feedback, and used that feedback in determining subsequent responses (Spring Data JPA/Hibernate, transactions)
- Migration from MongoDB to PostgreSQL
    - Took a low-risk, zero-downtime approach that wrote to both databases for a period of months before making a final cutover
    - Carefully refactored existing code to ensure equivalent data would be sent to the new database
- Migration to AWS
    - Created a library of Terraform and Ansible scripts that allowed our services to be deployed, consistently, across multiple environments
    - Implemented new security requirements (automatic generation and maintenance of per-customer Postgres schemas, TLS between microservices)
- Improvements to build scripts
    - Incorporated Ansible configuration generation into a Gradle plugin to maintain consistency between local development and other environments, and to reduce risk of password exposure.
    - Dockerized application as part of the build process
    - Incorporated local dockerized Mongo and Postgres databases, speeding up application startup during development by about 4x
- Leadership and mentoring

    - Conducted rigorous code review and pair programming sessions
    - Presented architectural overviews and technical explanations for new hires and engineers on collaborating teams
    - Worked with Ops to debug production problems
    - Provided important product behavior details, verbally and in writing/diagrams, to non-technical stakeholders
    - Broke down, planned, prioritized, and provided updates for projects, in collaboration with product and QA managers

Interactive Brokers Group, Greenwich, CT

***Secure Application Developer**, February 2015 to July 2017.*
Primarily Java development, with some PL/SQL. Projects included (but were not limited to):

- File-based access to account management system
    - Processed XML files allowing $1,000,000,000+ clients to create batch requests for wire withdrawals, deposit notifications, position transfers, etc.; generated XML response files
    - Wrote XML schema specification files provided to clients
    - Created a web interface allowing users to
        - view the statuses of sorted and filtered instructions
        - upload and initiate processing of new files after providing 2nd factor credentials
    - Encrypted and signed files uploaded through the web interface
    - Refactored system to allow non-file instruction sources (e.g. SWIFT messages)
    - Designed and implemented fine-grained authorization system
- Second-factor authentication device mailing system
    - Generated pdfs with names and addresses for mailing envelope window
        - Gathered user information from Oracle database
        - Included barcodes for mapping devices (with their own barcodes) to users
    - Automatically activated devices after sending warning emails to device recipients
    - Generated system reports (e.g. number of "Bingo card" devices upgraded to PIN pad display cards)
- Client for OCR web service (SOAP) for processing passport images
    - Decrypted image files uploaded by users
    - Sent image files to web service over https
    - Compared extracted text with database records

The University of Iowa, Iowa City, IA

***Instructor**, spring semester, 2014.*
Introductory Java course for undergraduate and graduate students

- Developed new course material including programming projects, lesson plans, lectures, and exams
- Worked with students individually, in small groups, and as a class of 30

***Research Assistant**, September, 2012, to September, 2014.*
User interfaces for the Coq Proof Assistant

- Collaborated in the planning and development phases of NSF award CCF-1250306
- Developed a basic UI for Coq as a plugin for the jEdit editor
- Developed experimental extensions to the basic UI
- Developed and executed user testing protocol

***Research Assistant**, spring semester 2012*

- Developed a user interface for "KIND", a tool for automatically verifying properties of the programs written in the Lustre dataflow programming language
- Debugged connection with tool back end

***Teaching Assistant**, fall semester 2008 through fall semester 2013*

- Nine semesters in the University of Iowa Computer Science Department
- Guided discussion sections for three different courses
- Graded and answered student questions for the graduate level "Formal Methods in Software Engineering" course for three semesters

Education

- Ph.D. Computer Science, The University of Iowa, Iowa City, IA, December 2014
    - Dissertation Title: *Development And User Testing Of New User Interfaces For Mathematics And Programming Tools*
- B.S. Physics, Rice University, Houston, TX, May 2008

Skills

- Java (primary), Bash, SQL, Ruby, JavaScript, TypeScript, Scala, C, Groovy, Python (some)
- Spring, Spring Boot, Spring Data JPA, Spring Cloud Stream
- HTTP, REST, SOAP, node, npm
- MongoDB, PostgreSQL, MySQL, Redis, Kafka, Elasticsearch
- Java (Jakarta) EE (Servlets, JPA, JSPs, EJBs, etc.), Hibernate,
- Flyway, Jackson, Kafka Streams, Piccolo (Java library for zoom-able user interfaces), jEdit plugin development
- Unix/Linux, Vi/Vim, Emacs, SSH, nginx, gpg, openssl
- Docker, Kubernetes, Gitlab CI/CD
- Terraform, Ansible, AWS (CCP certified), Grafana, Graphite
- Gradle, Maven, Ant, Intellij IDEA
- JUnit, TestNG
- HTML, CSS, Angular, AngularJS
- LaTeX, JSON, YAML, XML, Markdown, Org Mode
- Git, Mercurial, Subversion, CVS
- Coq, Alloy, Lustre, ESC/Java2, Dafny

Publications

Berman, B. A. and Hourcade, J. P (2014). Keyboard Card Menus: A New Presentation of Non-Standard Shortcuts. *Journal of Universal Computer Science, Special Issue on Trending Breakthroughs in Human-Computer Interaction.*

Berman, B. A. and Hourcade, J. P (2014). Keyboard Card Menus: Faster Learning of Many Fast Commands. In *Proceedings of the XIV International Congress of Human-Computer Interaction (Interaction 2013), in the Spanish Congress of Informatics (CEDI)*, pages 105-112, Spain, 2013. ISBN 978-84-695-8352-4.

Presentations

New Coq User Interfaces: Survey and Ideas. Presentation co-authored with Aaron Stump and Juan Pablo Hourcade. Presented at *The Coq Workshop 2012*, Princeton, NJ, USA, August 12, 2012.