Debbie Pao

Email: depao168@gmail.com
Website: www.debbiepao.me
Phone: (415) 713-7936
Location: San Francisco Bay Area, *US Citizen*

WORK EXPERIENCE

Amazon Web Services (AWS)

Elastic Block Store (EBS) Developer Experience Team: Software Engineer (L5) 9/2022-*present*

- Directed cross-functional collaboration with an 18-member engineering team to ensure smooth maintenance of mission-critical EBS core product services. Streamlined critical release testing of their services on dedicated qualification platform. Resolved all misconfigured testing resources that blocked high-priority production software delivery, automated 720+ hours of manual setup effort per year, and reduced operational load of manually maintaining testing resources by 98%.
- Led 4 teams of 33 engineers to design and implement an authorization system for managing critical testing resources in EC2 data centers. Reduced unauthorized access incidents by 91% and only allowed programmatic access to invoke critical operations in order to proactively prevent 60+ wasted engineering hours per incident to re-configure accidentally deleted testing resources.
- Designed and released scheduler cron job for installing persistent server configurations and RPM packages on EBS servers and physical EC2 compute hosts in multi-region data centers. Eliminated 1.9K+ engineering hours of manual setup per month and improved entirety of EBS production software release efficiency by 88%.
- Launched new feature to enforce custom timeouts on all testing instances in EBS software release lifecycle. Prior to this feature, 900+ tests hung indefinitely per month. This feature cut down the on-average 500+ hour testing duration to 48 hours max and reduced monthly testing resource costs by $17K. Usage rate grew to 70% a month post-launch and testing resource turnover rate increased by 86%.

Twilio

Messaging Services Sender Team: Senior Software Engineer 9/2021-9/2022

- Managed a team of 6 engineers to migrate a new microservice from MySQL to DynamoDB to address restrictions on scalability and efficiency problems and drive down monthly costs by $30K, building a more resilient Twilio Messaging product that processes over 300+ million messages/day.
- Led 5 engineers to redesign and modularize the Messaging Services product architecture to prepare for next generation v2 large-scale API features by separating complex business logic into 3 distinct microservices and 3 distinct database storage systems.
- Directed 8 engineers to redesign and conduct load testing on the complex number selection algorithm that minimized the queuing and delay time Twilio delivers messages to destination by 5 seconds, decreased database writes by 9K queries/second, and improved accuracy to 100% guaranteeing that no number is over-selected.

Messaging Core Team: Software Engineer 7/2019-9/2021

- Led 6 teams in cross-functional initiative to re-architect existing distributed system to strengthen resiliency of message delivery receipt processing by migrating processing of OTT delivery receipts over to a generic Messaging Channel-agnostic processor. Eliminated risks to allow for removal of 40K messages/second scalability restriction by moving away from MySQL to AWS. Focused on 10 functional requirements, such as: new queue data model, processing order and retries, database storage schema updates, status callbacks, error code mappings, NACK processing, billing, OTT Read receipts, and failover/fallback handling strategies.
- Non-linearly scaled and defined the capacity model to ensure high availability and resiliency during high traffic holiday season for most critical distributed system in Messaging by load testing the system expected to handle more than 40K+ requests/second and process a total of 960+ million message segments/day with different patterns based on peak production customer traffic.
- Presented solution for MySQL split brain incident at R&D company wide ops review that prevented losing 60K+ messages by decreasing the automatic failover time window from minutes to seconds and making sure the widely-used internal MHA tool cleanly shuts down connections to the old dead primary.

Amazon

Amazon Web Services (AWS): Software Development Engineer Intern 5/2018-8/2018

- Implemented customer-facing features to improve the developer workflow experience in continuous deployment pipelines for Amazon's cloud version control system. Designed algorithms to calculate quality of source code files compared to existing code stored in system.
- Engineered data models for AWS DynamoDB using operating systems and security principles to provide secure and scalable read-and-write operations for customers' data. Practiced test-driven development using Cucumber.io to validate.

NOTABLE PROJECTS

Esc Twilio Python 12/2020-8/2021

- Virtual escape room experience based in Twilio HQ that started as a company-wide internal hackathon project to showcase to new hires.
- Flask app that integrated Twilio's Programmable Messaging and Voice TwiML Call products with integrity checks and rate limiting.
- Launched successfully and was incorporated into the official onboarding process for all Twilio's Summer 2021 interns.

SKILLS

Coding Languages: Java, Python, MySQL, Scala, HTML, CSS, Javascript

Technical Skills: Dropwizard, MySQL MHA, AWS EC2, AWS DynamoDB, AWS S3, Apache Kafka, Datadog, Rollbar, Pagerduty, Nagios, HAProxy, Jenkins, Kibana, Mockito, UNIX, Apache JMeter, OpenAPI Swagger, Redis, Apache Maven, Zeppelin, AWS CloudWatch, AWS Step Functions, AWS Lambda

EDUCATION

University of California, Berkeley Graduated: 5/2019

B.S., Major in Bioengineering, Minor in Electrical Engineering and Computer Sciences

Relevant Coursework: Operating Systems and Systems Programming, Computer Security, Efficient Algorithms and Intractable Problems, Data Structures and Programming Methodology, Great Ideas in Computer Architecture (Machine Structures), Structure and Interpretation of Computer Programs, Discrete Mathematics and Probability Theory, Designing Information Devices and Systems