Gerald Lloyd

Senior Software Engineer

+1 979-308-2030
glloyd1121@gmail.com
linkedin.com/in/glloyd1121
Oakland, CA 94601, United States

SUMMARY

- Experienced Senior Software Engineer adept in full-stack development, possessing excellent communication, problem-solving, leadership, collaboration, adaptability, time management, attention to detail, analytical thinking, continuous learning, and mentoring abilities, dedicated to delivering high-quality solutions and fostering team success.

- 9+ years of full-stack experience, specializing in React.js, Redux, Next.js, TypeScript, Node.js, Express.js, Nest.js, Electron, Python, Flask, Django, Celery, Kafka, GraphQL, REST API, Docker, Kubernetes, AWS(Lambda, Load Balancer, RDS, DynamoDB, SNS, SQS, SES, API Gateway, ECS, EKS, and more).

WORK EXPERIENCE

SENIOR SOFTWARE ENGINEER

Spot, September 2021 – Current

- Crafted a highly responsive and intuitive web application utilizing React.js, Redux, Next.js, Node.js, Nest.js, Typescript, and Babylon.js, culminating in a remarkable 30% surge in user engagement and satisfaction.

- Developed Electron based desktop application, and mobile application to provide more seamless user experience, and published on Google Play, as well as App Store.

- Implemented advanced state management techniques to optimize the application's performance and decrease load times, resulting in a staggering 50% decrease in page loading times.

- Implemented A/B testing and user tracking using Google Analytics, resulting in a 10% improvement in conversion rates and a 5% increase in click-through rates.

- Effectively implemented agile development methodologies and industry-leading best practices, achieving a noteworthy 20% improvement in team productivity, along with a commendable 10% decrease in bug reports.

SENIOR SOFTWARE ENGINEER

Surreal Events, July 2018 - June 2021

- Collaborated closely with cross-functional teams and stakeholders to identify and prioritize features, leading to the timely delivery of a fully functional metaverse using Agora SDK, Twilio SDK, WebRTC, and WebSocket.

- Developed responsive and highly intuitive user interfaces using React.js, which resulted in an increase in user engagement by 35%.

- Implemented robust API and database architecture using Python, Flask, and Typescript, leading to a seamless and uninterrupted video conferencing experience for users.

- Conducted regular code reviews and implemented automated testing to maintain high code quality standards and a bug-free user experience.

- Designed and implemented real-time chat features, enabling users to communicate effortlessly during video calls and enhancing the overall user experience.

- Proactively monitored the app's infrastructure and implemented optimization strategies, resulting in an increase in the app's scalability by 50%.

FULLSTACK DEVELOPER

SAP, December 2014 - May 2018

- Designed and developed multiple high-traffic, mobile-first websites using React, Redux, and Python, resulting in a 30% increase in average session duration and a 25% increase in page views.

- Developed RESTful API utilizing Python, Django, and PostgreSQL, following microservices architecture.

- Collaborated with UI designers to create intuitive, accessible, and responsive user interfaces that improved usability and reduced bounce rates by 15%.

- Optimized website performance by implementing server-side rendering, code splitting, and lazy loading, resulting in a 20% decrease in page load time and a 15% increase in search engine visibility.

- Implemented sophisticated data visualization tools to provide comprehensive insights into user behavior and system performance, leading to a significant 30% increase in user engagement.

- Created and maintained a scalable and modular design system using React components and styled-components, resulting in a 30% reduction in development time and a 20% improvement in code maintainability.

EDUCATION

UNIVERSITY OF CALIFORNIA, BERKELEY - July 2009 – September 2014

Bachelor of Science, Computer Science

SKILLS

JavaScript, TypeScript, React.js, Redux, Next.js, HTML, CSS, D3.js, Babylon.js, Three.js, Babel, Storybook, Webpack, Node.js, Express.js, Nest.js, Python, Flask, Django, Celery, Kafka, Git, MySQL, PostgreSQL, MongoDB, Redis, RESTful APIs, GraphQL, Docker, Kubernetes, Mocha, Jest, React Testing Library, Cypress, WebRTC, WebSocket, AWS (API Gateway, ECS, CloudFront, Route53, DynamoDB, S3, SNS, SQS, RDS, Lambda), Agile, Scrum

PROJECTS

Spot

Spot is a virtual office for remote teams. It supports all the features of Slack and Zoom plus the things you used to do in the office: walk up to your colleagues' desks, high-five, gather around the water-cooler, celebrate big wins, decorate your office.
On the front end, we leverage the power of **React.js**, **TypeScript**, **Redux**, **Babylon.js**, and **GraphQL** to deliver a cutting-edge user interface that ensures a smooth and intuitive experience. This technology stack enables efficient state management, dynamic data retrieval, and interactive user interactions.
At the core, Spot utilizes the state of art **Node.js** framework **Nest.js**, as well as **TypeORM**, and **PostgreSQL**, which provides secure, and performative **GraphQL** API endpoints.
To ensure a reliable and scalable infrastructure, all the services are fully containerized using **Docker**, and are deployed on **AWS EKS(Elastic Kubernetes Service)** using **Terraform**, backed by **AWS S3**, and the industry-leading **CloudFront Content Delivery Network (CDN)**.
The entire code base is covered over 90% by **unit tests** using **React Testing Library**, **Jest**, and **Mocha,** as well as **e2e(end-to-end) tests** utilizing **Cypress**.

Surreal Events

Surreal Events is the next generation of video conferencing. It provides end-to-end metaverse solutions for platform publishing, art production and professional services.
AMD, Red Bull, Braves Atlantis, and more enterprises and online stores successfully hosted their webinars on Surreal Events platform.
Exciting game scenes are built inside SR Studio, utilizing Unreal Engine 4, and pixel streamed via **WebRTC** server, backed by another **Websocket** server used for server-client intercommunication.
Secure and reliable **RestAPI** is built with **Python, Flask** following microservice architecture.
User-facing web application built with **React**, **FluentUI**, **Agora SDK**, **Twilio SDK**, **WebRTC**, and **Websocket** client is fully customizable by branding, and feature flagging.
Reusable components with storytelling using **Storybook** is published to **NPM** as private package, and is reused everywhere within the codebase.

Shosho

Shosho is Medium-like Rich Text Online Editor with powerful **Grammar** and **Spell checking** abilities.
Integrated **LanguageToolPlus** and **BigHugeLabs API** for **real-time** grammar and spell checking. The client provided custom dictionary data for checking.

Editor.js was a great start but there were difficulties while integrating into **React.js** project, so decided to use **Draft.js**, Rich Text Editor framework for **React**. The front-end is built with **React.js** following **Pixel-perfect** design on **Figma**. Back-end is using **Python**, **Django**, **DRF(Django Rest Framework)** at its core, **SSO(Single sign-on)** using **Okta** for authentication, **Stripe** for billing and subscription.

EnsureDR Recovery Ready Dashboard

The idea of the project is to visualize recovery ready data of various system resources, such as CPU, RAM and Network. It's a **React.js** based project using **Redux** for global state management. Gauge Chart, Donut Chart, Time-series Chart, and more provided by **Chart.js** is used to display different types of system metrics. It is fully tested for **cross browser compatibility** and **mobile responsiveness**.
All the contents are delivered on **CDN(content delivery network)** for faster page loads, and deployment process is streamlined by **CircleCI** actions.

Pulse Tile

Ripple Foundation is a clinically led, not for profit company, supporting **open source**, **open standards** and an **open architecture** for healthcare that can be re-used worldwide.
Back-end is built with **Express.js, Typescript** and **Redis**, supported by **Node-Red**, an upcoming technology designed for **IoT**.
Front-end of PulseTile framework is using **React.js** with **Context**/**Hooks**, and the modern libraries including **React Query,** and **Material UI**.