

Shaojun Bao

Phone:(+1)515-218-2313 | Email:shaojunbao92@gmail.com

GitHub:<https://github.com/ShaojunBao> | Website:

SUMMARY

- **5+ years of experience** building user-centric, responsive, cross-browser web and mobile applications using modern front-end technologies. Proficient in React, Redux, and React Router; fluent in JavaScript (ES6+) and TypeScript with strong HTML5/CSS3 skills.
- Experienced with UI libraries and tools like Bootstrap, jQuery (for DOM manipulation and AJAX), and Webpack, while maintaining a constant focus on performance optimization and accessibility (a11y).
- Built **interactive, data-driven dashboards and visualizations** using Highcharts and custom D3.js integrations for real-time analytics.
- Developed and maintained **full-stack JavaScript applications** using the MERN stack (MongoDB, Express.js, React, Node.js) with seamless integration between front-end and back-end.
- Extensive experience managing **React component lifecycle** (including modern Hooks) and application state.
- Leveraged Redux (with middleware like Redux Thunk) for complex state management and used React Router for robust client-side routing and smooth asynchronous data handling.
- **Hardened web applications** against SQL injection, XSS, CSRF, and other OWASP Top 10 vulnerabilities by adopting secure coding practices and thorough input validation.
- Implemented **RESTful and SOAP APIs** using Node.js and Express, integrating Passport-based authentication (JWT/OAuth2 flows) and event-driven middleware for a modular, scalable back-end architecture.
- Managed the full **Software Development Life Cycle (SDLC)** in Agile/Scrum teams, from sprint planning and backlog grooming to CI/CD pipeline implementation and stakeholder demos.
- Fostered a collaborative engineering culture through peer code reviews, pair programming, and effective Git version control (GitHub/GitLab) workflows.
- Deployed and scaled applications on **AWS** (EC2, Lambda, S3) using Docker containers and Kubernetes for container orchestration. Ensured high availability and optimized performance of deployments in cloud environments.
- Wrote comprehensive **unit and integration tests** using Jest, React Testing Library, and Enzyme, as well as end-to-end tests with Cypress, ensuring over 95% code coverage.
- Performed API testing with Postman to verify the functionality and reliability of back-end services.

Technical Skills

- **Programming/ Markup:** JavaScript (ES6+), TypeScript, Python, Java, SQL, Bash/Shell, JSON, XML, YAML/Markdown
- **Front-End & UI:** React, Next.js, Redux, Angular, Vue.js, jQuery, React Native, Tailwind CSS, Material-UI, Styled-Components, SCSS/SASS, Bootstrap, D3.js/Highcharts, HTML5, CSS3
- **Back-End & Runtime:** Node.js, Express, NestJS, GraphQL (Apollo Server/Client), RESTful APIs, WebSockets/Socket.io
- **Databases & Caching:** MongoDB, PostgreSQL, MySQL, DynamoDB, Redis
- **Cloud & DevOps:** AWS (EC2, Lambda, SQS, API Gateway, CloudFront, RDS), EKS, Fargate, Kafka Heroku;; Docker, Kubernetes, Terraform, CloudFormation, GitHub Actions
- **API & Testing:** RESTful APIs, GraphQL, Axios, Jest, Enzyme, Mocha, Chai, Cypress, Selenium, Postman
- **Build Tools & Version Control:** webpack, npm, yarn, ESLint, Prettier, Git, Gitlab
- **Monitoring & Analytics:** Splunk, Datadog, Elastic Search

Experience

Client: **Obligo**, New York, NY, USA

Jan 2024 - present

Project: Lease Administration Management Software

Centralizes and simplifies the management of commercial lease portfolios, empowering businesses to stay on top of critical dates, rent obligations, and lease terms. With automated reminders, real-time data access, and intuitive workflows, the software ensures transparency and control over all lease-related activities. By eliminating manual tracking and reducing administrative burden, it helps companies mitigate risk, improve accuracy, and make informed real estate decisions. Designed for collaboration, Occupier's solution keeps stakeholders aligned while optimizing lease operations across the entire portfolio.

Responsibilities:

- **SPA Architecture:** Designed, developed, and tested a full-stack Single-Page Application using React (functional components and Hooks), React Router, Express.js, and MongoDB—providing seamless property browsing and user interaction without full page reloads.
- **Property Search & Listings:** Implemented advanced search filters, dynamic property listings, and modular sorting (by price, location, amenities) to deliver an intuitive browsing experience for prospective investors and buyers.
- **Interactive Map & Inquiry System:** Integrated Google Maps for geolocation-based property exploration and developed secure inquiry forms with server-side validation, improving lead generation and investor engagement.

- **UI/UX Design:** Created intuitive, mobile-first prototypes in Figma to optimize navigation and property exploration; conducted stakeholder demos and incorporated iterative feedback to refine usability and user engagement.
- **Front-End Modernization:** Refactored legacy gallery and navigation structures by replacing jQuery dependencies with React component-based architecture, improving modularity, code cleanliness, and parallel development efficiency.
- **Backend Analytics:** Built a browsing history tracking system using Express and Mongoose (MongoDB), capturing user engagement data with a 30-day retention policy and optimized retrieval for business insights.
- **Database & Performance:** Established a NoSQL MongoDB schema with key-based indexing to enable rapid lookup of user activity and functional analytics while maintaining efficient storage management.
- **Agile Collaboration:** Documented product requirements and specifications in Confluence, organized project milestones and Epics in Jira, and drove an iterative delivery process across four structured sprints.
- **Testing & Quality:** Conducted unit and integration testing for key components using Jest and React Testing Library, ensuring robustness across core user flows in property search, gallery, and inquiry submission modules.

Client: Element Chat, London, UK(Remote)

Dec 2022 – Nov 2023

Project: Chat App

Partnered with the product team to design and implement a cross-border social media platform built in partnership with the product team to enable users around the globe to share life updates and hold real-time conversations. It provides one-on-one messaging, group chats, video conferencing and media/file sharing—all enhanced by on-the-fly translation powered by ChatGPT Whisper—so that multinational users enjoy seamless, language-agnostic communication.

Responsibilities:

- **SPA Architecture:** Designed, developed & tested the Single-Page Application in React, Redux, React Router, TypeScript, HTML5, CSS3, and Material UI—delivering seamless navigation without full page reloads.
- **Real-Time Chat UI:** Built one-on-one and group chat interfaces using WebSocket (Socket.IO), with message threading, read receipts, typing indicators and infinite scroll.
- **Video Conferencing:** Implemented multi-party video calls in React/React Native via WebRTC, including dynamic grid layouts, screen sharing, and mute/unmute controls.
- **Media & File Sharing:** Developed drag-and-drop upload, chunked image/file transfers, preview thumbnails and progress bars—optimizing network usage and UX feedback.

- **AI-Powered Translation:** Integrated ChatGPT Whisper for live audio/text translation overlays, auto-detecting source language and rendering translated captions in real time.
- **Authentication & Security:** Implemented OAuth2/JWT and Google social login flows in React Native & Node.js, boosting conversion by 10% and ensuring secure token management.
- **Offline Support & Caching:** Added Service Worker and IndexedDB/localStorage caching to persist messages offline and synchronize on reconnect.
- **Push Notifications:** Integrated Firebase Cloud Messaging for web and mobile, handling background notifications and deep-links into specific conversations.
- **Responsive & Themed UI:** Crafted responsive layouts with Material UI's Grid and SASS mixins; built light/dark mode toggles using CSS variables and Context API.
- **Agile Collaboration:** Owned sprint planning, daily stand-ups and backlog refinement in Jira; wrote technical specs in Confluence to align cross-functional teams.
- **Testing & Quality:** Achieved 90 %+ coverage with Jest & React Testing Library; added Cypress and Detox for web / React-Native e2e flows.

Client: **Memorial Sloan Kettering Cancer Center**, New York, NY, USA

Jan 2021 - Dec 2022

Project: MSK MyChart – HIPAA-Compliant Patient Portal

Contributed to the design and implementation of MSK's next-generation patient portal, a HIPAA-compliant platform for managing healthcare workflows such as appointment scheduling, patient demographics, prescription tracking, and pharmacy integration. Built with React and integrated with secure RESTful APIs and HL7/FHIR data standards, the portal empowers patients and providers with real-time access to health information and seamless care coordination.

- **SPA Architecture:** Built modular, scalable Single-Page Application architecture using React (Hooks, Context API), React Router, and TypeScript—delivering responsive, dynamic healthcare interfaces without full page reloads.
- **Patient Demographics & Appointments:** Developed interactive components for profile updates, insurance entry, and calendar-based appointment scheduling; integrated React UI with HL7/FHIR-compliant APIs.
- **Pharmacy & Prescription Integration:** Implemented interfaces for medication views and refill requests, integrating with secure pharmacy APIs; handled real-time form validation and dosage input logic.
- **HIPAA-Secure UI Patterns:** Enforced strict PHI protection standards across the frontend by implementing OAuth2 token-based auth, HTTPS-only communications, automatic session expiry, and encryption-aware UI rendering.

- **Accessibility & Mobile Optimization:** Used Tailwind CSS and Material UI to ensure responsive, mobile-first layout across viewports.
- **Performance Tuning:** Optimized bundle size with lazy-loading and dynamic imports; used memoization and debounce techniques to reduce render overhead for high-traffic modules like the appointment dashboard.
- **Error Handling & Fault Tolerance:** Introduced React error boundaries and user-friendly fallback UIs to gracefully handle API failures and session timeouts in critical workflows.
- **API Integration & Observability:** Consumed RESTful and GraphQL APIs using Axios and React Query; collaborated with backend teams to refine request contracts and troubleshoot data consistency issues using Postman and Splunk logs.
- **Testing & Validation:** Wrote unit and integration tests with Jest and React Testing Library; built E2E coverage using Cypress to validate key patient flows such as login, record updates, and prescription requests.
- **Agile Collaboration:** Participated in cross-functional sprint planning and backlog grooming in Jira; documented frontend specs, API schemas, and accessibility audits in Confluence.

Education

Northeastern University, Boston, MA, USA

Master of Science in Information System, GPA: 3.38

Concentrations: Frontend Development | Web Design & Development | Requirement Analysis

Iowa State University, Ames, IA, USA

Bachelor of Science in Management Information Systems, GPA: 3.30

