

Bernard Hartman

Senior Software Engineer

Scranton, PA 18508 | bernardh.engineer@outlook.com | +1 (423) 471-8362 | [linkedin.com/in/bernard-h-engineer](https://www.linkedin.com/in/bernard-h-engineer)

PROFESSIONAL SUMMARY

Senior Software Engineer with 11 years of experience designing scalable web applications, backend services, API platforms, distributed systems, and cloud-native software across entertainment technology, healthcare technology, global eCommerce, and retail. Strong background in Python, Node.js, TypeScript, JavaScript, React, RESTful APIs, GraphQL, Kafka, PostgreSQL, Redis, Docker, Kubernetes, GCP, Azure, CI/CD, testing, observability, and secure production system development. Experienced in AI-assisted engineering workflows, LLM output review, prompt evaluation, structured rubric design, synthetic task creation, data quality validation, and gold-standard technical response development for model improvement. Known for combining senior-level software engineering judgment with clear written analysis, objective evaluation criteria, production debugging experience, and practical system design reasoning to improve software quality, developer productivity, and AI model reliability.

CORE SKILLS

- **Programming Languages:** Python, Node.js, TypeScript, JavaScript, SQL, HTML5, CSS3
- **Frontend Engineering:** React, Redux, React Hooks, Next.js, Vite, Webpack, Tailwind CSS, Material UI, Responsive UI Development
- **Backend Engineering:** Express.js, Fastify, NestJS, Django, FastAPI, Flask, RESTful APIs, GraphQL, Microservices, API Design, Backend Architecture
- **AI and LLM Evaluation:** LLM Output Evaluation, AI Trainer Workflows, Prompt Analysis, Model Reasoning Review, Rubric Design, Gold-Standard Response Writing, Synthetic Software Engineering Tasks, Data Annotation, Human Feedback, Response Ranking
- **Software Engineering Evaluation:** Code Review, System Design Review, Debugging Assessment, Test Case Design, Edge Case Analysis, Technical Writing, Criteria-Based Evaluation, Engineering Best Practices
- **Data and Messaging:** Kafka, Event-Driven Architecture, Real-Time Streaming, Message Queues, WebSockets, Webhooks, Background Jobs, Asynchronous Processing
- **Databases and ORM:** PostgreSQL, MySQL, MongoDB, Redis, Prisma, TypeORM, Sequelize, Data Modeling, SQL Optimization, Indexing, Query Tuning
- **Cloud and DevOps:** GCP, Azure, AWS, Docker, Kubernetes, GitHub Actions, Cloud Build, Jenkins, CI/CD, Nginx, Linux
- **Testing and Quality:** Jest, Pytest, React Testing Library, Cypress, Playwright, Unit Testing, Integration Testing, API Testing, Contract Testing, Test Automation
- **Monitoring and Reliability:** Logging, Monitoring, Observability, Alerting, Health Checks, Error Handling, Production Debugging, Performance Optimization
- **Security:** OAuth2, OIDC, JWT, RBAC, Secure APIs, Audit Logging, Encrypted Data Handling, Backend Security Best Practices, HIPAA-Aware Development

WORK EXPERIENCE

Senior Software Engineer

Oct 2024 – Present

Cast & Crew | New York, NY

- Designed and developed full stack production workflow applications using Node.js, TypeScript, Python, React, Express.js, Fastify, RESTful APIs, PostgreSQL, Redis, Docker, Kubernetes, GCP, and Azure to support payroll, workforce management, document processing, and entertainment finance operations.
- Built AI-assisted internal engineering and operations workflows using Python services, structured validation logic, and LLM-style review patterns to analyze payroll records, crew documents, approval states, accounting exceptions, and workflow data for accuracy and consistency.
- Created realistic senior-level software engineering evaluation scenarios for AI-assisted tooling, including backend debugging tasks, API design reviews, data validation cases, edge-case reasoning prompts, and production incident analysis exercises.
- Designed criteria-based evaluation rubrics to assess generated technical explanations, code suggestions, debugging steps, API recommendations, and system design responses against correctness, completeness, maintainability, security, and production-readiness standards.
- Reviewed AI-generated technical outputs for backend services, React components, SQL logic, event-driven workflows, and cloud deployment recommendations, identifying hallucinated assumptions, weak reasoning, missing edge cases, and unsafe implementation details.
- Authored gold-standard engineering responses that demonstrated senior-level reasoning across API design, distributed system behavior, database optimization, asynchronous processing, observability, error handling, and production troubleshooting.
- Evaluated model-style responses for software engineering correctness by checking whether proposed implementations matched requirements, handled edge cases, avoided unsafe assumptions, included appropriate tests, and reflected practical production constraints.

- Developed Node.js microservices with Express.js and Fastify to process payroll events, approval transitions, document updates, reconciliation jobs, and production accounting workflows while keeping API contracts consistent and backend services scalable.
- Implemented Kafka-based event-driven workflows for payroll status changes, document processing events, accounting updates, production code changes, and exception queues, enabling reliable asynchronous communication across distributed backend services.
- Created RESTful and GraphQL APIs for React applications, documented endpoints with Swagger and OpenAPI specifications, and standardized request validation, response modeling, authentication behavior, and error handling across service domains.
- Optimized ORM-based backend logic using Prisma and TypeORM with PostgreSQL by improving schema design, transaction handling, query execution, indexing strategies, and data access patterns for payroll records, approval states, crew profiles, and accounting references.
- Strengthened backend reliability by adding structured logging, centralized error handling, health checks, monitoring dashboards, alerting, retry-safe processing, and production debugging workflows for distributed Node.js services in containerized cloud environments.

Senior Full Stack Engineer

Jul 2020 – Sep 2024

Parachute Health | Princeton, NJ

- Built healthcare full stack applications using Node.js, TypeScript, Python, React, Express.js, Fastify, RESTful APIs, GraphQL, Kafka, PostgreSQL, Redis, Docker, Kubernetes, GCP, and Azure to support clinical ordering, documentation, supplier coordination, and patient workflow operations.
- Developed React and TypeScript interfaces for clinicians, suppliers, and operations users to capture patient information, payer requirements, medical necessity details, documentation steps, signatures, supplier updates, and order statuses across regulated healthcare workflows.
- Designed Node.js microservices and distributed backend systems for healthcare ordering workflows using TypeScript, Kafka, REST APIs, GraphQL, PostgreSQL, and Redis, improving scalability and reliability across patient, provider, supplier, and payer-facing processes.
- Integrated Python-based AI and automation capabilities into healthcare services by connecting clinical documentation, order metadata, payer rules, and intelligent validation workflows to reduce repetitive review tasks while preserving HIPAA-aware auditability and traceability.
- Evaluated AI-assisted validation outputs for clinical documentation workflows by checking correctness, completeness, traceability, missing-context risks, and unsafe assumptions before recommendations were surfaced to internal operations users.
- Created structured review criteria for automation outputs, including whether generated explanations matched source data, followed healthcare workflow rules, handled edge cases, and produced actionable results for non-technical operations teams.
- Implemented Kafka-based event-driven processing for order status changes, document generation, eligibility updates, supplier notifications, audit events, workflow transitions, and background validation tasks that supported asynchronous healthcare operations.
- Developed secure RESTful APIs with Express.js and Fastify, maintained Swagger and OpenAPI documentation, and standardized endpoint contracts for order submission, clinical documentation validation, supplier routing, user authorization, and healthcare status tracking.
- Optimized ORM-driven data access using TypeORM and Prisma with PostgreSQL by improving schema relationships, repository patterns, transaction boundaries, indexing, loading behavior, and query performance for complex healthcare workflow data.
- Implemented HIPAA-conscious security patterns including role-based access control, JWT authorization, secure API validation, encrypted data handling, structured audit logging, traceable state transitions, and protected health information safeguards across full stack features.
- Enhanced production observability with structured logs, metrics, alerts, monitoring dashboards, traceable error handling, and distributed debugging practices for API failures, slow database operations, workflow exceptions, and message processing delays.
- Collaborated with frontend, DevOps, data engineering, product, compliance, and QA teams to deliver secure full stack healthcare features that combined React interfaces, Node.js APIs, Python services, Kafka streams, API documentation, and cloud-native deployments.

Software Engineer

May 2017 – Jun 2020

Global-e | New York, NY

- Developed cross-border eCommerce features using Node.js, TypeScript, React, JavaScript, Python scripts, Express.js, RESTful APIs, PostgreSQL, Redis, Sequelize, Swagger, and cloud services to support checkout, pricing, taxes, duties, shipping, and payments.
- Built React dashboards that helped internal teams review international orders, payment events, shipment statuses, localization settings, fraud exceptions, and settlement details across multiple countries, currencies, and retail partners.
- Created Node.js backend services for merchant integrations, checkout rules, pricing, order processing, and payment workflows, using REST APIs, relational databases, Redis caching, and asynchronous patterns to support global commerce operations.

- Implemented Kafka, webhooks, message consumers, retry logic, and background jobs to process payment callbacks, shipment events, merchant updates, fraud signals, localization changes, and order status synchronization.
- Built Python scripts for reconciliation, reporting, validation, and analysis, helping commerce teams compare transaction records, shipping events, tax outcomes, payment activity, merchant settlements, and localization data across global regions.
- Reviewed data validation results, reconciliation exceptions, and automation outputs against objective business rules, helping teams identify inconsistent records, false positives, missing fields, and edge cases in payment and shipping workflows.
- Documented RESTful APIs with Swagger and OpenAPI for merchant configuration, transaction lookup, checkout rules, pricing services, shipping events, tax calculations, payment reconciliation, and internal operations tools.
- Improved ORM-based data access with Sequelize and TypeORM by refining model relationships, transaction handling, indexing strategies, query patterns, and data retrieval for orders, merchants, payments, shipments, and regional pricing rules.
- Improved checkout and operations performance by reducing API latency, caching localization data in Redis, tuning PostgreSQL queries, improving Kafka consumer processing, and refining React rendering behavior during international shopping sessions.
- Added structured logging, centralized error handling, monitoring dashboards, alerting, and debugging support for failed payment callbacks, delayed shipping updates, API timeouts, and merchant integration issues across commerce workflows.

Full Stack Developer

Aug 2014 – Apr 2017

NIKE, Inc. | *Beaverton, OR*

- Developed retail web applications using JavaScript, React, Node.js, Express.js, Python scripts, RESTful APIs, SQL, Sequelize, Swagger, Docker, Git, Linux, and CI workflows to support digital commerce, product discovery, inventory visibility, campaigns, and merchandising systems.
- Built React interfaces for product browsing, campaign pages, merchandising workflows, promotional modules, search filters, navigation elements, and responsive retail experiences that connected with Node.js backend APIs.
- Created Node.js and Express.js services for product catalog data, store inventory workflows, merchandising tools, pricing information, and commerce API integrations used by retail and eCommerce applications.
- Documented RESTful APIs with Swagger for product details, pricing, availability, campaign content, merchandising data, inventory views, and order-related backend workflows.
- Implemented ORM-based data access using Sequelize with relational databases for product, inventory, campaign, merchandising, and commerce-related data.
- Built asynchronous backend workflows using Node.js, scheduled jobs, and event-oriented patterns to support inventory refreshes, campaign updates, reporting tasks, and commerce data validation.
- Implemented Python scripts for reporting, data transformation, inventory analysis, automated validation, and commerce data comparison across internal retail systems.
- Improved frontend and backend performance by optimizing JavaScript assets, improving API response structures, tuning SQL queries, refining CSS delivery, and streamlining product-focused pages.
- Added unit tests, integration checks, browser testing, API validation, release verification, logging practices, and defect tracking for product launch, campaign, inventory, and customer-facing commerce features.
- Supported deployment and troubleshooting workflows using Git, Docker, Linux environments, CI pipelines, build automation, environment-specific configuration, and production validation practices.

EDUCATION

Stevens Institute of Technology | *Hoboken, NJ*

Sep 2010 – Jun 2014

Bachelor of Science in Computer Science