

# Bhagyesh Rathi

Bay Area, CA | 6698886214 | [bhageyesh2161@gmail.com](mailto:bhageyesh2161@gmail.com) | [LinkedIn](#) | [GitHub](#) | [bhagyesh.dev](http://bhagyesh.dev)

## EDUCATION

### *San Jose State University*

*San Jose, CA*

#### **Masters of Science in Artificial Intelligence | GPA: 3.9**

Expected: May 2027

**Relevant coursework:** Machine Learning, Deep Learning, AI Threat Intelligence, Natural Language Processing(NLP), AI and Data Engineering.

### *San Jose State University*

*San Jose, CA*

#### **Bachelors of Science in Computer Science, Magna Cum Laude**

August 2020 - May 2024

**Relevant coursework:** Java OOP, Data Structures and Algorithms, Software Engineering, Relational Databases, Data Visualization.

## SKILLS & CERTIFICATIONS

**Programming Languages:** Java, Python, JavaScript, Kotlin, SQL, HTML, CSS

**Frameworks:** Spring, SpringBoot, Flask, Scikit-Learn, Tensorflow, Pytorch, Pandas, Numpy, Seaborn

**Skills & Technologies:** REST APIs, Distributed Systems, Multi-tiered Systems, Microservices, CI/CD, Agile software development, Unit/Integration Testing, Monitoring and Alerting.

**Databases:** MongoDB, MySQL, PostgreSQL, Redis

**Cloud:** GCP, Docker, Kubernetes, AWS

**Tools:** Git, GitHub, Gitlab, Postman, Jira, Confluence.

**Certifications:** [Machine Learning Specialization\(Stanford\)](#)

## RELEVANT EXPERIENCE

### **Rakuten - Software Engineer Co-op | San Mateo, CA**

**August 2023 – December 2023**

- Developed **event listeners** in **Kotlin** to handle consent-revoked and delete-account events from Google, Facebook and Apple.
- Implemented features to validate and decode **JWTs** to extract required data, and deserialize **JSON** responses.
- Built an **SDK** to abstract low level service to service calls, published artifact in **Artifactory** through **GitLab CI/CD**.
- Achieved **85%** test coverage(**JaCoCo**) with **Unit**, **Integration testing**, and leveraged **SonarQube** to eliminate code smells.
- Deployed **microservices** on **Google Cloud Platform (GCP)** using **Gitlab CI/CD** automated build, test, and deploy pipelines.
- Created **Dockerfiles** and orchestrated containers using **Kubernetes**, deploying across non-prod and prod environments.
- Integrated **OpenTelemetry** and **Google cloud Trace** for real time **monitoring** and performance analysis.

### **Rakuten - Software Engineer Intern | San Mateo, CA**

**May 2023 – August 2023**

- Contributed to **Social Sign Up / Login** with Google, Apple, Facebook for Rakuten businesses in Spain, UK, and Germany.
- Achieved a **40%** adoption rate of Social Authentication and a **37%** increase in CVR over Username and Password Authentication.
- Developed **Microservice** in **Kotlin** for social link(with Google, Facebook, Apple) management (Display and Unlink) features.
- Implemented **REST APIs** leveraging **OpenID Connect & OAuth2.0** to implement token based **Authentication and Authorization**.
- Implemented structured logging using **Log4j**, analyzed the logs through **GCP Cloud Monitoring**.
- Participated in **Agile software development** such as daily stand-ups, sprint planning, and sprint review.

## PROJECTS

### **RAG-Powered Interactive Portfolio | Next.js, Pinecone, Vertex AI, Vercel AI SDK, Langchain | [GitHub](#), [Chat](#)**

**Feb 2026**

- Built a production-grade **RAG pipeline** — PDF ingestion, recursive chunking (LangChain), 768-dim embeddings, Pinecone upsert.
- Designed low-latency retrieval with **top-K vector search**, score thresholding, and metadata filtering to reduce hallucination.
- Built a streaming API endpoint using **Vercel AI SDK** with Gemini 2.5 Flash, achieving sub-second TTFT.
- Implemented multi-layer security: input validation, system prompt hardening against prompt injection, and topic-scoped guardrails
- Authored a **27-test integration suite** covering embedding quality, retrieval accuracy, pipeline correctness, and injection defense
- Deployed on **Vercel** with secure environment-based GCP auth, ensuring consistent local and production behavior.

### **Resource aware AutoML platform and code generator | Python, Scikit-Learn, Flask, React | [GitHub](#)**

**Nov 2025**

- Engineered an AutoML system using **Scikit-Learn & Flask** that automates preprocessing, task detection, and parallel model training/evaluation.
- Reduced training latency by 40% implementing **Stratified Sampling** and dynamic model switching to handle large datasets efficiently.
- Developed a context manager to **profile real-time CPU/RAM usage** & a leaderboard to sort models by **accuracy, time, or resource efficiency**.
- Built a transpiler engine to enable one-click downloads for both serialized models (.pkl) and their reproduction code.
- Designed a responsive React frontend with real-time interactive **ROC/Scatter plots** and **confusion matrix** for visualizations of top model.

### **Skin Cancer Detection using CNN | Python, Tensorflow, Scikit-Learn, Pandas, Seaborn | [Colab](#)**

**March 2025**

- Developed a **CNN** using **Scikit-Learn** to classify skin lesion images into 7 cancer categories, achieving **80% accuracy**.
- Preprocessed data with **resizing, normalization, one-hot encoding**, and **oversampling** to address class imbalance.
- Optimized model performance using the **Adam optimizer, learning rate annealing**, and **hyperparameter tuning**.