

# Namrata Bhaumik

☎ (+1) 782-882-5874    ✉ [namratabhaumik16@gmail.com](mailto:namratabhaumik16@gmail.com)    [in linkedin.com/in/namratabhaumik](https://www.linkedin.com/in/namratabhaumik)    [namrata-bhaumik.medium.com](https://namrata-bhaumik.medium.com)    [github.com/namratabhaumik](https://github.com/namratabhaumik)

## Technical Skills

**Languages:** Python, Javascript, Java, C#, Bash, HTML, CSS

**Technologies:** Django, FastAPI, Flask, Pydantic, Node.js, React, jQuery, OAuth, JWT, Firebase, PyTest (unit + integration), Android Studio, Unity, Oculus, Git, Agile, Jira, Kanban, REST, Netlify, PowerBI, Tableau, Figma, D3.js, Data Visualization

**Databases:** BigQuery, Firestore, MySQL, Postgresql, MongoDB, Elasticsearch

**Cloud & DevOps:** GCP, Docker, Terraform (IaC), GitLab CI, AWS, Azure, Github Actions, Jenkins, Linux, Shell Scripting

## Experience

### Publicus

April' 25 - June' 25

*Software Engineer (Backend)*

*Toronto, ON, Canada*

- Collaborated directly with the founders post-MVP to build, test, and scale backend features of the software platform, working across fast-moving sprints and shifting priorities.
- Led backend migration from old **monorepo** architecture to new modular **polyrepo** architecture leveraging AI chatbots (**OpenAI, Grok, Claude, Gemini**) to increase platform stability, optimize user workflows, and experiment with **LLM and LRM** - augmented development.
- Accelerated feature delivery by leveraging **LLM-assisted tools** like **Cursor IDE** and **NotebookLM** for code generation, documentation, and rapid prototyping in a high-pressure environment.
- Built RESTful APIs and microservices using **Python, FastAPI, Firestore, BigQuery, Cloud Run, Cloud Function, Pub/Sub, Cloud Build, GAR, WIF** ensuring smooth data ingestion, transformation, and secure file exports.
- Designed and executed robust test strategies with **Pytest** (unit + integration), mocking external services (GCP components) to enable end-to-end smoke testing and ensure deployment readiness.
- Owned full-stack developer workflows, **GitLab CI/CD, Docker**, and infrastructure planning with **Terraform**.
- Designed and tweaked **system architectural decisions**, authored **testing/feature plans in Markdown**, and contributed to **agile sprint planning** and daily team operations via **Slack** and **ClickUp**

### Dalhousie Transportation Collaboratory (DalTrac)

September' 24 - December' 24

*Software Developer*

*Halifax, NS, Canada*

- Developed a **Virtual Reality (VR)** evacuation simulation game to experiment with and analyze pedestrian behavior during fire emergencies in Downtown Halifax, focusing on real-world emergency response strategies
- Designed a 3D model of Halifax in collaboration with an Urban Design student using **SketchUp**, featuring detailed cityscapes with buildings, streets, buses, trees, and evacuation points, later integrated into **Unity** for an immersive environment.
- Implemented VR interactivity using **Oculus Quest 2** and **Unity's XR Toolkit**, enabling players to experience evacuation scenarios firsthand with realistic elements such as fire sounds, crowd voices, and dynamic environmental cues.
- Engineered 12 distinct evacuation scenarios with varying levels of guidance—ranging from structured evacuation signs to pedestrian voice cues—challenging players to adapt to unpredictable, real-life-like emergency situations.
- Programmed pedestrian AI behavior using **Unity's CharacterControllers, Rigidbodies, and Colliders** to simulate realistic crowd dynamics, including collision avoidance, varying movement speeds, and interactive voice prompts guiding the player.
- Developed dual player modes: a standard player with free movement using VR thumbsticks/buttons and an accessibility-focused wheelchair mode with controlled directional movement, enhancing inclusivity in gameplay.
- Integrated real-time data tracking and analytics, capturing metrics such as game duration, distance traveled, proximity to evacuation points, and critical decision-making moments, with session data exported in **CSV** format for behavioral analysis.
- Overcame technical challenges related to Oculus Quest integration, performance optimization for complex pedestrian interactions, and real-time data logging, enhancing both the research validity and user experience.

### Cerner Corporation (Oracle Health)

July, 2021-July, 2023

*Software Engineer I*

*Bangalore, India*

- **Optimized Wiki Searchability with Elasticsearch:** Spearheaded the integration of **Elasticsearch** into Cerner's internal wiki page search functionality. This enhancement significantly improved search speed and accuracy, ultimately reducing the onboarding time for new hires by 40%. Collaborated with cross-functional teams to understand user requirements, designed the solution architecture primarily using **Python's BeautifulSoup library**, and implemented indexing strategies to ensure seamless search experiences across large datasets.
- **End-to-End Ownership of Python-Django Automation Framework:** Took complete ownership of an automation framework built on **Python** and **Django**. Led the design, development, and implementation of the system, which scaled to support over 430 clinical data solutions. By optimizing backend processes and refining system architecture, the solution reduced clinical data turnaround time by 5 days. Applied best practices in software development and conducted rigorous testing to ensure stability and scalability.
- **Integrated RAPID APIs for Enhanced Search Performance:** Integrated the **RAPID API** platform into existing Django pipelines to automate data retrieval. This integration enhanced internal clients' ability to access real-time data, improving search performance by 20%. Managed API security keys, ensured proper data handling, and worked closely with the security team to adhere to best practices.
- **AJAX Integration in MySQL for Improved Data Processing:** Reduced data processing time by 30% by incorporating **AJAX** into existing data pipelines. This optimization enabled seamless handling of large datasets in **MySQL servers**, significantly improving system responsiveness. Additionally, optimized queries and data retrieval mechanisms in MySQL, enhancing database performance and reducing query execution time.
- **Efficient Data Fetching from PostgreSQL in Django:** Worked closely with the backend team to re-architect data fetching from **PostgreSQL** into **Django**, ensuring smooth interactions and minimizing delays during high-load operations. Improved query performance and database interactions, optimizing data retrieval for better system efficiency.
- **Designed Interactive Power BI Dashboards:** Developed and designed interactive **Power BI dashboards** using **PowerQuery** to track employee training schedules. The visually engaging dashboards enhanced usability and tracking efficiency by 50%, allowing associates and managers to easily visualize and update training data. The project involved close collaboration with business analysts and end users to ensure the reports addressed critical business needs and provided actionable insights.
- **Collaborative Problem-Solving and Cross-Functional Communication:** Collaborated closely with cross-functional teams, including data engineers, client teams, and product managers, to ensure alignment on project goals and timelines. Regularly participated in team (**scrum**) meetings and sprint planning sessions, contributing to the continuous improvement of processes and delivering high-impact solutions on time. Actively mentored junior engineers and shared knowledge about best practices in coding and system design.
- **Continuous Learning and Innovation:** Constantly sought opportunities to learn new technologies and improve existing systems. Contributed to internal technical discussions and code reviews, where I provided feedback and suggested optimizations to improve code efficiency and maintainability. Embraced an **Agile** development environment, completing multiple projects in **two-week sprints** while maintaining a high standard of quality.
- **Data Migration to Microsoft Azure:** Led data migration from **SQL** databases to **Microsoft Azure**, ensuring seamless integration and optimizing cloud storage for our Django-based website. Additionally, contributed to strategizing the website's migration to Azure hosting for improved scalability and performance.

## Cerner Corporation (Oracle Health)

Jan, 2021-July, 2021

Software Engineer Intern

Bangalore, India

- **Developed a RESTful API Workflow for GitHub Issue Management:** Took on the responsibility of developing a scalable backend API workflow to streamline **GitHub** issue management. Designed and implemented **RESTful** endpoints using **Java** and **Spring Framework**, enabling efficient issue tracking, assignment, and resolution. The system automatically fetched GitHub issues, categorized them based on parameters like labels and status, and facilitated faster management. This reduced issue resolution time by 40%, significantly improving team productivity and helping the engineering teams address and resolve issues more efficiently. Integrated authentication and security measures, ensuring safe access to the API.
- **Collaborated with Cross-Functional Teams:** Worked closely with product managers, senior engineers, and the QA team to ensure seamless integration of the API into existing workflows. This required a deep understanding of GitHub's API, **REST principles**, and the backend infrastructure. During development, I actively participated in sprint planning, reviewed code, and collaborated with my team to optimize performance and fix bugs.
- **Implemented Unit Testing and Continuous Integration:** In line with best practices, I implemented unit tests using Java's **JUnit** framework to ensure the reliability of the API endpoints. This helped catch issues early during development and maintained high code quality. I integrated the API into Cerner's **CI pipeline with Jenkins**, automating the build and deployment process. This ensured that each change was tested and deployed seamlessly.

- **API Documentation and User Guide:** To ensure the internal team could effectively use the new API workflow, I wrote comprehensive documentation and provided detailed instructions on how to interact with the API. This included API endpoint details, request/response formats, and examples for easy integration with existing systems. I also created a set of use case scenarios to highlight common workflows.
- **Optimized GitHub Issue Resolution Process:** By automating repetitive tasks like issue categorization and assignment, the workflow allowed team members to focus on higher-priority issues, reducing bottlenecks in the process. This automation improved team response time to issues and created a smoother, more efficient process for handling critical tasks.
- **Exposure to Agile Development and Mentorship:** Throughout my internship, I followed **Agile** methodologies, participating in daily standups, sprint retrospectives, and planning meetings. I received mentorship from senior engineers who helped me improve my coding practices and provided valuable feedback through code reviews. This experience not only refined my technical skills but also gave me an understanding of the full software development lifecycle, from design and implementation to testing and deployment.
- **Learning and Skill Development:** The internship helped me grow technically and professionally, as I learned to write efficient, maintainable code, and gained practical knowledge in API design, backend development, and automation. I also learned the importance of collaboration, clear communication, and problem-solving in a fast-paced development environment.

## VAP Technology Pvt. LTD.

Software Engineer Intern

May, 2019-June, 2019

Kolkata, India

- Engineered an Android application with CRUD functionality using **Java** and **Android Studio**.
- Worked under the supervision of a Senior Android Developer who created the project roadmap.

## Projects

**DalVacationHome** : A cross-cloud distributed system using AWS IAM, Lambda, Cognito, Dynamodb, S3, Lex, SNS/SQS, CloudFormation, GCP Pub/Sub, Functions, Cloud Run and IaC (Terraform, Docker).

**Tech Stack Used:** AWS, GCP, Terraform, Node.js, React.js

**Github:** <https://github.com/namratabhaumik/DalVacationHome>

**Medium:** <https://shorturl.at/RNrDc> , <https://shorturl.at/eoQxn>

- Developed a **cloud-native vacation rental platform** leveraging **AWS (SNS, SQS)** and **GCP (Pub/Sub)** for efficient, real-time messaging and event-driven architecture.
- Integrated **AWS Lex** to enable natural language interactions, enhancing the user experience with AI-powered conversational interfaces.
- Designed and deployed **serverless functions** for scalable backend processing, reducing operational overhead and optimizing resource usage.
- Automated deployment pipelines using **CI/CD** tools, ensuring seamless code integration and faster delivery cycles.
- Implemented **Infrastructure as Code (IaC)** with **Terraform** to manage cloud resources consistently across AWS and GCP environments.
- Containerized applications using **Docker** to ensure portability and maintainability across different deployment stages.

**BlogIt** : Developed a full-stack blogging platform with my team implementing secure authentication, RESTful APIs, and dynamic content management for a seamless user experience.

**Tech Stack Used:** React.js, Python-Flask, Mongoddb, Firebase, Netlify, Render

**Github:** <https://github.com/namratabhaumik/BlogIt>

**Medium:** <https://shorturl.at/F3uNh>

- **User Authentication:** Implemented secure user authentication with **Firebase Authentication**, allowing users to sign up, log in, and securely manage sessions.
- **Dynamic Content Management:** Enabled users to create, edit, and delete blog posts through a dynamic user interface. Each blog post includes a title, content, and tags for better organization.
- **RESTful APIs:** Designed a set of **RESTful APIs** using **Flask** to manage blog data. These APIs handle CRUD (Create, Read, Update, Delete) operations for blog posts, allowing for easy integration and scalability.
- **Real-time Updates:** Leveraged **Firebase Realtime Database** for live updates on blog comments and post interactions, ensuring users have an engaging, interactive experience.
- **Responsive Design:** Developed a responsive design using **CSS** and **Bootstrap**, ensuring compatibility with both desktop and mobile devices.

- **Error Handling and Logging:** Incorporated error handling mechanisms and utilized **Flask's logging** functionality for debugging and improving the overall user experience.

**Mealmate** : An Android app for meal planning, recommending and sharing recipes.

**Tech Stack Used:** **Android Studio, Java, Firebase**

**Github:** <https://github.com/namratabhaumik/MealMate>

**Medium:** <https://shorturl.at/zz8qO>

- **Meal Planning & Recipe Recommendations:** Users can browse through a variety of recipes, each accompanied by ingredients, instructions, and estimated cooking times. The app intelligently recommends recipes based on user preferences, dietary restrictions, and available ingredients.
- **Interactive Ingredient Inventory Management:** Implemented an ingredient inventory system where users can track what ingredients they have at home, allowing the app to suggest recipes based on the ingredients they already possess.
- **Sharing and Community Interaction:** Enabled users to share their own recipes with others. Users can upload their recipes, photos, and tips, fostering a sense of community among cooking enthusiasts.
- **Firebase Integration:** Leveraged **Firebase Authentication** for secure login and user management, **Firestore** to store and retrieve user data such as recipes and preferences, and **Firebase Cloud Functions** to add custom backend functionality like sending notifications for new recipe suggestions.
- **User-friendly UI:** Focused on delivering an intuitive, sleek UI that made it easy for users to navigate through recipes, meal plans, and ingredient lists. The app's design was optimized for both performance and aesthetic appeal, ensuring a smooth experience for the user.
- **Agile Development:** Employed an Agile methodology with two-week sprints to iterate on features and prioritize user stories. Regular scrum meetings kept the team aligned, while feedback from the client ensured the app met real-world needs and expectations.

**ExpenseTrackerExtended** : A cloud-native expense tracking platform leveraging AWS (Cognito, DynamoDB, S3, SNS, IAM), Docker, Kubernetes (EKS), and Python-Django/React for a distributed, secure, and scalable architecture.

**Tech Stack Used:** **AWS, GCP, Kubernetes, Docker, React.js, Python-Django**

**Github:** <https://github.com/namratabhaumik/ExpenseTrackerExtended>

**Medium:** <https://rb.gy/9n47b5> , <https://rb.gy/g1ibvd> , <https://rb.gy/acto8p>

- **Cloud-Native & Serverless:** Architected a hybrid serverless and containerized backend—using **AWS Lambda** for event-driven logic and **Django REST APIs** on **GCP Cloud Run**—integrated with **DynamoDB, S3, and Cognito** for secure, scalable data and file management.
- **CI/CD & DevOps:** Automated build, test, and deployment pipelines with **GitHub Actions**, enabling seamless delivery to **Firebase** (frontend) and **GCP Cloud Run** (backend), with robust test coverage and environment parity via **Docker**.
- **AWS Integration:** Orchestrated secure inter-service communication using **IAM** roles, **S3** event notifications, and **SNS** for real-time alerts; implemented **CloudWatch** logging for monitoring and debugging.
- **Frontend & API Security:** Developed a responsive **React** frontend with **Cognito**-authenticated API requests, ensuring secure, smooth user experiences and robust **JWT**-based access control.
- **Kubernetes & Networking:** Deployed containerized services on **AWS EKS** with secure **VPC** networking, load balancing, and auto-scaling for high availability and cost efficiency.
- **Documentation & Testing:** Consolidated and enhanced project documentation (README), implemented comprehensive **unit/integration tests** with mocked AWS services, and maintained high code quality standards.
- **Cost Optimization & Scalability:** Designed the architecture to operate efficiently within the **AWS Free Tier**, optimizing resource allocation for cost-effectiveness, while enabling auto-scaling for both serverless functions and Kubernetes pods to handle varying traffic loads, ensuring high availability and performance.
- **Future Roadmap:** Planning **CloudFront CDN** integration, advanced analytics, and further UI/UX improvements.

## Certifications

**AWS Certified Cloud Practitioner**

**Microsoft Certified Azure Fundamentals**

## Education

**Dalhousie University**

*Master of Applied Computer Science (GPA: 3.95 / 4.30)*

September '23 - December '24

Halifax, NS, Canada

**SRM Institute of Science and Technology (SRMIST)**

*Bachelor of Technology - Computer Science and Engineering (GPA: 8.69 / 10.00)*

2017-2021

Chennai, India