# Deepchand Damegunta

deepchanddamegunta@gmail.com | +1 (984)-226-1526 | LinkedIn

## Professional Summary

Results-driven Java Developer with 7 years of experience in designing, developing, and testing scalable web-based applications. Adept in the full Software Development Life Cycle (SDLC) with hands-on expertise in Core Java, Collections, Spring Framework (including Spring Boot, Spring Security, Spring Batch, Spring REST, AOP), Hibernate, JPA, and JDBC. Experienced in building RESTful web services using Spring Boot and tools like Postman, and skilled in Test-Driven Development (TDD), Agile, and Scrum methodologies. Proficient in using Java 1.7/1.8 with strong OOPs concepts including inheritance, polymorphism, and abstraction, and a solid understanding of design patterns such as MVC, Singleton, Factory, and Front Controller. Demonstrated expertise in microservices architecture, DAO components, and working with databases like MySQL using tools such as SQL Developer. Familiar with version control systems like Git and GitHub, build tools like Maven, and messaging systems including Apache Kafka. Experienced with development environments like Eclipse, IntelliJ, and Spring Tool Suite (STS), with a strong focus on clean code, reusability, and performance optimization in both front-end and back-end components. Additionally, knowledgeable in deploying and managing applications on Amazon Web Services (AWS), enhancing scalability, reliability, and cloud-native integration of enterprise systems.

## Technical Skills

| **Category** | **Skills** |
|---|---|
| Languages | Java, JavaScript, TypeScript, SQL |
| Backend Frameworks | Spring Boot, Spring MVC, Hibernate, JPA, RESTful APIs, Microservices, Maven. |
| Frontend Technologies | React.js, HTML5, CSS3, AJAX |
| Databases | MySQL, PostgreSQL, MongoDB |
| Version Control | Git, GitHub |
| DevOps & CI/CD | Jenkins, Docker, Kubernetes, GitHub Actions |
| Cloud Platforms | AWS |
| Messaging & Caching | Kafka, Redis |
| Testing Tools | JUnit, Mockito, Postman |
| API Documentation | Swagger, API Inspector |
| IDEs & Tools | IntelliJ IDEA, Visual Studio Code, Eclipse |

## Professional Experience

**Java Developer** **Feb 2024-Present**
Truist Bank, United States

**Project Description:** Collateral 360 is a centralized, vendor-hosted platform for managing property due diligence processes used by financial institutions. It automates workflows such as appraisal procurement, environmental assessments, site inspections, and valuation approvals. The system enhances compliance, collaboration, and decision-making across real estate loan operations.

The application was engineered with a Java Spring Boot microservices architecture, integrated with **React.js frontend**, deployed on **AWS Cloud**, and utilized **CI/CD pipelines**, **containerization**, and **distributed messaging** to ensure high availability and performance.

**Frontend:**

- Developed modular and responsive UI components using React.js and HTML5/CSS3, enabling users to manage property data, appraisal requests, and valuation reports.
- Implemented Redux for managing global state across the application, including user sessions, form data, and notifications.
- Integrated frontend with backend RESTful services using Axios, supporting real-time status updates and document transactions.
- Used React Router for SPA-style navigation between modules such as Dashboard, Appraisal Requests, and Vendor Management.
- Enabled dynamic field validation and conditional rendering using AJAX techniques for smoother UX.

**Backend:**

- Designed and developed stateless RESTful APIs using Spring Boot and Spring MVC to manage domain entities like Appraisals, Properties, Users, and Vendors.
- Utilized Hibernate ORM with JPA annotations to map Java objects to relational PostgreSQL tables, enabling transactional CRUD operations and lazy/eager fetch strategies.
- Employed DTOs, Service Layer, and Repository Pattern to maintain clean code structure and abstraction.
- Exposed Swagger-based API documentation using SpringFox, enabling QA and frontend teams to self-test APIs via API Inspector.
- Implemented role-based access control (RBAC) and exception handling using custom @ControllerAdvice and @ExceptionHandler classes.

**Cloud/DevOps:**

- Containerized all microservices using Docker and deployed to AWS EC2 instances with environment-specific configuration using application.yml.
- Managed structured data on AWS RDS (PostgreSQL) and stored files/documents securely on AWS S3 buckets with access control policies.
- Automated builds, tests, and deployment pipelines using Jenkins and GitHub Actions, facilitating CI/CD integration for dev, staging, and prod en vironments.
- Orchestrated microservices using Kubernetes, enabling scalable deployment and rolling updates with zero downtime.

**Environment:** Java 17, Spring Boot, Spring MVC, Hibernate, JPA, RESTful APIs, React.js, Redux, PostgreSQL, AWS (EC2, RDS, S3), Docker, Jenkins, Swagger, Kafka, Redis, JUnit, Git, IntelliJ IDEA

**Java Developer** **Oct 2022-Aug 2023**
**HerdX, USA**

**Project Description:** AgriVetTrack is a digital livestock management system focused on monitoring buffaloes' daily feeding behavior, health history, and disease symptoms. It helps veterinarians, dairy farm operators, and researchers collect and analyze data to identify illness trends, ensure timely vaccination, and optimize nutrition plans.

The platform supports real-time data capture from field workers using a responsive UI, while backend microservices manage animal profiles, feed logs, and medical histories. Advanced filtering and reporting features assist in early disease detection and health risk analysis.

**Frontend:**

- Designed responsive dashboards using React.js to display individual buffalo profiles, feeding trends, weight progress, and disease alerts.
- Used Redux to manage global states such as current animal selection, active alerts, and logged health symptoms across sessions.

- Integrated Axios for performing authenticated REST API requests related to feed logs, vet check-ins, and symptom submissions.
- Created intuitive navigation with React Router for modules like Animal Records, Feed Intake Logs, Vaccination Schedules, and Reports.

**Backend:**
- Built domain-driven Spring Boot microservices to manage animals, feed types, diseases, and medical treatments.
- Used Hibernate and JPA to persist structured data like species details, daily rations, and symptom checklists in PostgreSQL.
- Developed REST APIs for CRUD operations on animal entries, vet records, and dietary adjustments.
- Integrated Swagger for interactive API documentation and enabled field testers to validate endpoints using Postman.
- Employed Kafka for sending asynchronous disease alert messages between medical service and analytics service.
- Caching was implemented via Redis for lookups like common diseases by region and feed types by age group.

**Cloud/DevOps:**
- Deployed services on AWS EC2 and configured AWS RDS (PostgreSQL) for storing animal data and visit history.
- Used AWS S3 for storing buffalo medical certificates, vaccination cards, and uploaded images from health inspections.
- Dockerized backend services for reproducibility across dev/stage/prod environments and orchestrated deployments via Jenkins pipelines.
- Configured CloudWatch for monitoring service health and error logs, and implemented alerts for API failures or service timeouts.

**Environment:** Java 17, Spring Boot, Spring MVC, Hibernate, JPA, RESTful APIs, React.js, Redux, PostgreSQL, AWS (EC2, RDS, S3), Docker, Jenkins, Swagger, Kafka, Redis, JUnit, Git, IntelliJ IDEA

**Office Depot, USA** **Jul 2021-Sep 2022**
**Java Developer**

**Project Description:** Worked as a Java Developer on the development and enhancement of Office Depot's cashback rewards system aimed at incentivizing customer purchases. The system tracks eligible transactions, calculates rewards based on purchase history, and integrates with user accounts for redemption and notifications. Focused on building scalable backend services, implementing business logic, and integrating with third-party APIs and the Office Depot frontend application.

**Frontend:**
- Developed and maintained reusable Angular 10 components for dashboards, transaction histories, and user statements, improving UI consistency and performance across modules.
- Utilized RxJS and Angular Services to manage reactive data flows and business logic, ensuring seamless real-time updates and a dynamic user experience.
- Implemented robust form validation mechanisms with custom validators and error handling, providing real-time feedback and improving data quality.
- Ensured secure session management and route protection using Angular guards, enhancing the security of sensitive user data.
- Applied SASS/SCSS for responsive and adaptive designs, maintaining brand consistency across devices and browsers.

**Backend:**
- Developed RESTful APIs using Spring Boot to manage customer accounts, rewards calculation, fund transfers, and email notifications.

- Integrated third-party credit score APIs securely uses OAuth2, enabling real-time credit-based decisions for reward eligibility.
- Employed Hibernate and JPA for ORM and built custom queries to support reporting and detailed transaction histories in PostgreSQL.
- Designed and implemented audit logging and transaction tracking features, ensuring transparency and compliance with business policies.
- Followed Test-Driven Development (TDD) using JUnit and Mockito, improving code reliability and reducing production issues.

**Cloud:**

- Deployed microservices on AWS EC2 and utilized S3 for secure document storage, reducing downtime and improving scalability.
- Automated the CI/CD pipeline using Jenkins, streamlining deployments and minimizing manual intervention for staging and production releases.
- Monitored system logs and API performance using AWS CloudWatch, proactively resolving issues and improving system uptime.
- Used GitHub for version control and effective branch management, ensuring code quality through pull requests and peer reviews.
- Followed DevOps practices and collaborated with the QA and operations teams to ensure smooth integration and delivery cycles.

**Environment:** Java 11, Spring Boot, Angular 10, PostgreSQL, AWS EC2/S3, Jenkins, GitHub, OAuth2

**Discover Financial, USA** **Nov 2019-Jun 2021**
**Java Developer**

**Project Description:** Fraud Alert System application is the process of notifying clients on fraudulent transactions. If a transaction is found to be unauthorized, the client can upload all the relevant information to the form and declare the transaction fraudulent.

**Frontend:**

- Developed dynamic web interfaces using Angular, including components for dashboards, user transactions, and account statements, ensuring interactive user experiences.
- Leveraged RxJS and Angular services to manage reactive data streams and shared logic across modules, enhancing performance and maintainability.
- Implemented form validations and real-time feedback using Angular's reactive forms, ensuring data integrity and user-friendly interactions.
- Applied SASS/SCSS for creating responsive and modular styling, maintaining design consistency across various screen sizes.
- Secured frontend routes and sessions using Angular guards, enhancing the application’s security posture and protecting user data

**Backend:**

- Architected a microservices-based backend using Spring Boot, improving application modularity, fault isolation, and scalability across services.
- Built and maintained RESTful APIs using Spring MVC and Spring Boot to handle account management, rewards calculation, and notifications.
- Employed Spring Data JPA with Oracle and MySQL for robust data persistence and efficient querying through repository patterns.
- Designed custom exception handling mechanisms to deliver meaningful error responses, improving debugging and end-user experience.

- Wrote unit tests using JUnit and Mockito to validate backend functionality, significantly reducing the risk of bugs and regressions.
- Implemented third-party API integrations via secure OAuth2 connections for enhanced business features.
- Maintained code quality through regular refactoring and participation in code reviews to align with design patterns and industry best practices.

.

**Cloud:**

- Deployed microservices and frontend modules on Pivotal Cloud Foundry (PCF), coordinating with administrators to optimize resource usage and ensure availability.
- Used Spring Config Server to manage application properties across environments, supporting consistent and scalable deployments.
- Stored user-uploaded documents securely on AWS S3, improving scalability and reducing on-premises storage needs.
- Automated CI/CD pipelines using Jenkins, reducing manual intervention and enabling reliable code delivery.
- Integrated SonarQube into the CI process for real-time code quality assessments, enforcing high standards across the development team.
- Enabled real-time application monitoring with Kibana, logging system health, user activity, and performance metrics for proactive issue resolution.
- Conducted load and performance testing, tuning services to handle high traffic scenarios with consistent reliability and uptime.

**Environment:** Java8, Spring Boot, Microservices, Spring MVC, Spring Security, Spring Config Server, GIT, Rest API, Eclipse, Intellj, Maven, Oracle Database, Tomcat Server, Gradle, Kibana, Junit, Mokito, PCF Server.

**Reliance, India** **May 2016-Oct 2019**
**Software Engineer**

**Project Description:** Jio Fiber is a tailor-made intranet application for Reliance which supports generation of Customer Order Forms, pricing, contract management and associated terms and conditions. It enables the creation of new sites and contacts using which Customer Order Forms (COFs) are usually created. In this project there are two modules Admin and User Module.

**Frontend:**

- Implemented reactive user interfaces using React and JavaScript, enabling dynamic client-side behavior and smooth user experience.
- Used form validations and dynamic rendering to provide real-time feedback and data integrity across UI components.
- Participated in MVC design alignment, ensuring frontend layers effectively communicated with backend APIs for seamless data flow.
- Applied responsive UI techniques to enhance accessibility and usability across different devices and screen resolutions.

**Backend:**

- Developed RESTful APIs using Spring MVC and Spring Boot, supporting business logic, data retrieval, and processing operations efficiently.
- Utilized Spring Core annotations for effective dependency injection and component management, ensuring loose coupling and scalability.
- Applied Hibernate ORM to map Java objects to relational database tables, streamlining data persistence and transaction handling. Leveraged Java 8 features such as Lambda expressions, Streams, and Optional class to write concise, readable, and functional code.
- Formulated comprehensive unit tests using JUnit and Mockito to validate backend logic and ensure stability across application modules.

- Conducted thorough code reviews to improve maintainability, catch bugs early, and enforce consistent coding standards.
- Contributed actively in sprint planning, retrospectives, and agile ceremonies, collaborating closely with cross-functional teams.
- Played a key role in the System Development Life Cycle (SDLC) by ensuring proper design, implementation, and delivery of backend services.

**Cloud**:

- Used Kibana for log analysis and proactive monitoring, helping identify issues, trends, and anomalies in real-time application behavior.
- Helped ensure secure communication and encryption between services and APIs, supporting data protection and compliance standards.
- Participated in the build and deployment process with tools like Maven, ensuring efficient delivery pipelines and environment readiness.
- Maintained high-quality service by analyzing logs and metrics, contributing to the reliability and stability of the deployed applications.

**Environment:** Java8, Spring Boot, Microservices, Spring MVC, Hibernate, GIT, Rest API, Eclipse, Maven, Oracle Database, XML, Junit, Mockito, Kibana, React, JavaScript, HTML, CSS, Tomcat Server.

## Education

**Central Michigan University**
Master of Science**:** Information Systems**,** USA

**Narayana Engineering College**
Bachelor of Technology**:** Electronics and Communication Engineering**,** India