

Kalyan Puramsetti



Education

B.Tech - Electronics and Communication Engineering(2017-2021).

Technical Skills

- **Languages:** Core Java (Java 8), J2EE, AWS (EC2, S3, RDS), Linux, Python, C, C++.
- **UI Technologies:** Html, CSS, JavaScript, Reactjs(Basic)
- **Framework:** Spring boot and Microservices, REST APIs.
- **Databases:** MySQL
- **Tools:** STS IDE, MySql workbench, Postman, MobaXterm, Putty, Jira, Swagger, Jenkins, SonarQube, GitHub, Gerrit, Apache kafka.

Contact Details

Ph no: +916305628286

Email: puramsettikalyan99@gmail.com

Highlights

- Almost 2yrs (**Sep 2021 – Present**) of IT experience specialized in core java, Spring boot and microservices.
- Worked for US based health insurance client that provides health and insurance benefits to the citizens.
- Worked for the development of above health insurance intranet web application.
- 4 months (**Mar 2021 – July 2021**) of internship at cognizant trained in Core java, J2EE, Spring Boot and Microservices, AWS (EC2, S3, RDS).
- Developed an E-commerce web application as a mini project.

Experience

- Developed almost six APIs (Login management, Admin API, Database Manager, Data collection, Reports generation, EligibilityDetermination (ED)) in our application.
- Maintained one central MySQL RDS database that stored data for all APIs.
- Swagger and Postman are used for the REST APIs testing.
- Used SonarQube for the code review which detects bugs, vulnerabilities etc.
- For the source code management used Gerrit-code-review and GitHub.
- Used Jenkins tool to get code from github, to build using maven and to run in tomcat server.
- Deployed and run APIs using AWS EC2 linux instance through MobaXterm.
- Follow the Agile methodology for the project management.
- Participated in regular meetings, code reviews, and submitted progress on time.
- In internship developed PurchaseManagement Microservices which handles order request and cart details, Use MySQL cloud RDS database for storing of order and cart details.
- For every order or cancellation request stored one kafka message in separate table.
- Deployed using AWS EC2 instance and S3 bucket and made publicly accessible.
- For source code management used Github.