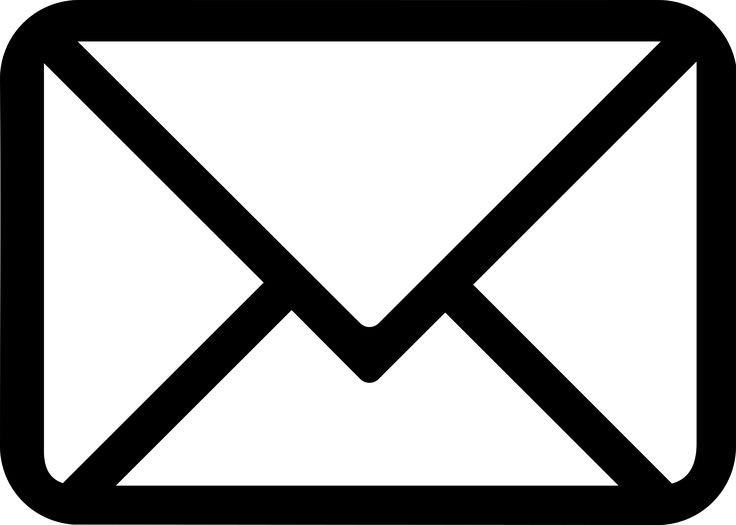
**PRAVEEN P**

: (913)-583-0890



: [pputapu.workmail@gmail.com](mailto:pputapu.workmail@gmail.com)



: <https://www.linkedin.com/in/praveen-p-040577176>



**PROFESSIONAL SUMMARY**

* Around 7+ years of experience in the IT industry with expertise in Software Analysis, Design, Development, Implementation and Testing of Object Oriented applications using **Java**, **J2EE, UI, APIs** and **cloud-based** applications.
* Experienced working with different Software methodologies like **Agile Scrum**, **Agile XP**, **Agile Kanban** and **Waterfall** model.
* Expertise working with major versions of Java (jdk 1.6, 1.7) and new features of Java (jdk 1.8).
* Expertise working with **Java 8 features** likes **Lambda Expressions**, **Streams API, Default Methods** and **Functional Interfaces**.
* Experienced with **J2SE** Technologies like **Multithreading**, **Collection API**, **Exception Handling** and **J2EE** Technologies like **Servlet**, **Listener**, **Filter**, **JSP, Java Security API** and **JSTL**.
* Expertise in maintaining the code quality with solid understanding of **JVM** Architecture.
* Practical knowledge and expertise developing MVC based applications with frameworks like **Spring Core (DI (IOC)), Spring MVC, Spring Boot** and **Struts MVC.**
* Implemented DAO layer using DAO frameworks like **Spring Data JPA**, **Spring Data Cassandra**, **Spring Data MongoDB**, **Spring JDBC**, **Spring ORM** and **Hibernate**.
* Excellent in developing and consuming the **Microservices** and **SOA** (Service Oriented Architecture) services based on **SOAP** (JAX-WS) and **REST** (JAX-RS) specifications and particularly with SOAP implementations like **Apache CXF** and REST implementations like **Spring**, **Jersey** and **Restlet**.
* Experience in developing and managing **Microservices** using **Spring Cloud** services like **Config server, Service Registry, Hystrix** and **Circuit Breaker Patterns** using **Spring Boot**.
* Experienced in securing the applications using **Spring Security** with different authentication/ authorization techniques include **Basic Authentication**, **OAuth 2.0** and **LDAP**.
* Implemented SSO (Single Sign On) functionality using SAML.
* Experience in deploying, monitoring and binding a specific service to the java web applications on **Pivotal Cloud Foundry** (**PCF**).
* Having experience with **Amazon Web Services (AWS) EC2, EBS, RDS** and **AWS Lambda**.
* Experience in implementing the fast search capabilities using open source frameworks **Apache Solr** and **Elastic Search** engine.
* Working knowledge of Front End development technologies like **HTML**, **CSS**, **JavaScript**, **JSP Taglib, JSTL**, **Ajax, jQuery, Bootstrap, Angular JS** and **Angular 2**.
* Having Knowledge on developing I/O intensive web-applications using **Node JS**.
* Working experience on Relational Databases like **Oracle SQL (PL/SQL), SQL Server (MS SQL), DB2** and **NoSQL Database** like **Apache Cassandra** and **MongoDB**.
* Experience in setting up and scheduling the batch jobs using **Spring Batch** and **Spring Integration**.
* Proficient in implementation of middleware layer using **Rabbit MQ**, **Apache Kafka** messaging brokers and implemented client layer using **Spring AMQP** and **Spring Kafka Producer, Consumer** framework.
* Working experience of application servers like **WebSphere**, **Apache Tomcat** and **Docker, Kubernetes**.
* Hands-on experience working with Version control tools like **GitHub, IBM Serena, Tortoise SVN**.
* Hands-on experience working with Continuous Integration (**CI**) build-automation tools such as **Maven, Gradle, Apache ANT, Apache IVY** and **Jenkins**.
* Expertise in tracking bugs and issues using **JIRA, Rally, Jazz** and **Pivotal Tracker.**
* Having knowledge and hands-on experience with **JUnit, Mockito, TestNG, Selenium, Cucumber and Gherkin** testing frameworks.
* Experienced with **Linux** based commands of **BASH** and **Shell Scripting**.
* Comfortable working with **Windows** and **Linux** based environments.
* Ability to adapt to new software applications and products, self-starter, with excellent communication skills and understanding of business workflow.
* Well experienced in collaborating with onsite as well as offshore teams and coordinating with business to convert business requirements into technical specifications.

**TECHNICAL SKILLS SET**

|  |  |
| --- | --- |
| **Languages** | C, CPP, Java (1.6, 1.7, 1.8), SQL, PL/SQL, Shell Scripting |
| **Enterprise Java** | Servlet, JSP, JDBC, JSTL, Data structures. |
| **Mark-up/XML /UI Technologies** | HTML, HTML5, XSLT, JavaScript, Dojo, CSS, CSS3, XML, XSD, AJAX, jQuery, JSON, Angular JS, Angular 2, Node JS, React JS. |
| **Tools & Framework** | Struts, Spring Core (IOC), Hibernate, JPA, Spring MVC, Spring AOP, Spring Data (Spring JDBC, Spring JPA), Spring Boot, Spring Batch, Spring Security and Spring Cloud. |
| **Web services** | Microservices, SOA Architecture, SOAP, RESTful, JAX-RS JAX-WS, Apache AXIS II, Apache CFX, Jersey, RestEasy. |
| **Middleware Technologies** | JMS, RabbitMQ, Spring AMQP, Spring Integration, Apache Camel. |
| **Cloud Platforms** | PCF (Pivotal Cloud Foundry), AWS (EC2, EBS, RDS, Lambda). |
| **Web/App Servers/Containers** | WebSphere, Apache, Tomcat, JBoss EAP, WebLogic and Docker. |
| **Databases** | IBM DB2, Oracle, PostgreSQL, Apace Cassandra, NoSQL MongoDB |
| **Development Tools** | Eclipse, Spring Tool Suite (STS), IntelliJ, IBM RAD. |
| **Testing Tools/ Frameworks** | SoapUI, Rest Client, SonarQube, JProfiler.  JUnit, Mockito, Jasmine, Karma, Selenium, TestNG, Cucumber, Gherkin. |
| **Version Control Tools** | GitHub, BitBucket, TFS, IBM Serena. |
| **Build-Automation Tools (CI)/Issue Tracking Tools** | Maven, Gradle, Apache ANT, Apache IVY, Jenkins  JIRA, Rally, Jazz, Pivotal Tracker |
| **OS & Environment** | Windows, Linux, Mac OS. |
| **Design Patterns** | MVC, Factory, Front Controller, Singleton, Façade, Business Delegate, Decorator, Observer and DAO patterns |

**EDUCATION DETAILS**

* **Masters** in **Computer Science Electrical Engineering (CSEE)** from University of Missouri, Kansas City, Aug 2014-Aug 2015.
* **Bachelors** in **Electronics and Communication Engineering (ECE)** from Jawaharlal Nehru Technological University, Hyderabad, June-2009 to May 2013.

**PROFESSIONAL EXPERIENCE**

|  |
| --- |
| **T-Mobile Mar 2018 – Till Date**  **Sr. Java Full Stack Developer**  **Bellevue, WA** |

**Description**: **T-Mobile** is the brand name used by the mobile communications subsidiaries of the German. In addition to its cellular mobile network, T-Mobile US operates a nationwide Wi-Fi Internet-access network under the T-Mobile HotSpots brand.

**Projects: T-Mobile Device Promotions and Loans** (**Awards** and **Accounting**) & **Business Hub Connect**

This **Promo Award** service is to support credit management for mercury loans enrolled in promotions and to manage its promotions life cycle whose operations include enroll, drop, re-instate, credit injection and orphan credit injection. **Promo Accounting** service is to generate the promo accounting transactions based on loan and promo accounting events, and Keep promo data in accord. **Business Hub** Connect is the voice and messaging app built natively in a mobile network. It makes business number a talk and text line that delivers the HD call quality.

**Responsibilities:**

* Involved in Requirements gathering, analysis and design with **Agile** Environment based on **J2EE** specifications and **Micro services** architecture.
* Drive a team of 5 with design solutions and detailed implementation guidelines.
* Closely work with Product Management and higher management in designing the solutions and provide relevant feedback as necessary in having the Technical Specification Document.
* Extensively used **Java 8** features like Lambda expressions, Method references, Stream API, Parallel Streams, Producer, Consumer, Bi-Consumer etc. and implementation using core java concepts of Multi-Threading, Exception Handling, Collections and Concurrent APIs.
* Involved in end to end development using **Spring Boot**, **Spring Data**, **Spring Batch**, **Rabbit MQ**, **Oracle**, **Cassandra** and **Mongo DB**. Database Modelling, **Rest Services** Development (**Java** 1.8).
* Used **Spring Core** Annotations for Dependency Injection, Transaction management, **Spring REST** for **REST API’s**, **Spring Boot** for micro services and **Kafka** as message broker, **Oracle** as backend database, **Hazelcast** as in-memory database, **Apache Solr** for search and auto completion capabilities.
* Involved in designing the reusable UI components using **Angular 2** and made asynchronous REST API calls using angular services and bound the data to HTML directives.
* All the functionality is implemented using Spring IO **/ Spring Boot**, Thymeleaf and **Hibernate** ORM. Implemented Java EE components using Spring MVC, Spring IOC.
* Used **OAuth 2.0** and **Apigee** for authentication and authorization process for the backend APIs.
* Provided connections using **Spring Data JPA, Spring Data Mongo** and **Spring Data Cassandra** to the database and developed SQL/NoSQL queries to manipulate the data.
* Used **Spring Batch** and **Spring Integration** to run the nightly job and to send files/emails notifications to customers and supplier on events.
* Created **Docker** Containers leveraging existing Linux Containers and AMI's in addition to creating Docker Containers from scratch.
* Container management using Docker by writing Dockerfiles and set up the automated build on Docker HUB and installed and configured Kubernetes.
* Managed local deployments in **Kubernetes**, creating local cluster and deploying application containers.
* Building/Maintaining Docker container clusters managed by Kubernetes, Docker on AWS. Utilized Kubernetes and Docker for the runtime environment of the CI/CD system to build, test deploy.
* Defined **unit test** cases using **JUnit** and **Mockito** based on the technical design/functional design in a way that is comprehensive and verifies accuracy of developed features.
* Followed Agile Practices Grooming, planning, code, code reviews, retrospectives and CI/CD.
* Working on **spring cloud** for configuration management and maintained the apps on **Pivotal Cloud Foundry**.
* Develop and designed SQL Database scripts based on the application design for both DML and DDL.
* Used **Log4j** to print the logging, debugging, warning, info and **Splunk** to monitor the issue.
* Worked with **Spring Cloud Sleuth**, **App Dynamics** and **Zipkin** for monitoring the issue/defects at **micro service** level.
* Used **BitBucket** repository for version control and implemented a Continuous Delivery pipeline with **Jenkins**.
* Used **Maven** script for building and deploying the application.
* Involved in bug fixes, enhancements and code refactoring.

**Environment:** Java (jdk 1.8), J2EE, Spring Core, Spring MVC, Spring Security, Spring Boot, Spring Data, Spring Batch, JSP, Hibernate, Oracle, Mongo DB, Cassandra, Servlet, JDBC, Web services, SOA (Service-oriented) Architecture, RESTful, SOAP, Microservices, Docker, Kubernetes, Spring cloud, AWS, OAuth 2.0, XML, Stored Procedures, Triggers, JUnit, Mockito, TDD, Log4j, STS, Apache Tomcat, PCF, Agile/Scrum Methodology, Maven, JIRA, BitBucket, Jenkins, SonarQube.

|  |
| --- |
| **Blue Cross Blue Shield of Illinois Nov 2015 – April 2018 Sr. Java/J2EE Full Stack Developer**  **Chicago, IL** |

**Description: Health Care Service Corporation (HCSC)** is the largest customer-owned [health insurance](https://en.wikipedia.org/wiki/Health_insurance) company in the United States, and an independent licensee of the [Blue Cross and Blue Shield Association](https://en.wikipedia.org/wiki/Blue_Cross_and_Blue_Shield_Association). It also deals in [life insurance](https://en.wikipedia.org/wiki/Life_insurance) products and is structured as a mutual legal reserve insurer.

**Projects: Financial Suspense System (FSS)** and **Financial Distributed Batch System (FDBS)**

FSS project aim is to **simplify overpayment reconciliation and related processes** by providing Single sign-on convenience through Availity, RealMed and Secure Provider Portal, Electronic notification of overpayments, Online Inquiry/Dispute/Appeal functionality. FDBS deals with batch jobs that can be managed in a container from a web with dynamic launching of any new jobs and primarily it deals with automating the process of void checks.

**Responsibilities:**

* Involved in all phases of project from analysis and requirement phase to delivering the project.
* Created **System Requirement Specifications (SRS)** report of the project by collecting all the requirements from client.
* Developed the functionalities using **Agile Scrum Methodology**.
* Build the application based on new features of Java **jdk 1.8** features at service layer level.
* Implemented **RESTful** web services in SOA (Service-oriented) Architecture to retrieve data from client side and made **REST API** calls from **Angular JS** and parsed the data to project to client and handled the security using **Spring Security** and **OAuth 2.0** authentication protocol.
* Created reusable UI components using **Angular** and rendered them at several places in application.
* Consumed **SOAP** based services to store and retrieve attachments in ECM (Enterprise Content Management System).
* Designed front-end presentation logic using **JSP**, **CSS3, Bootstrap** and **HTML 5**.
* Created servers on **NodeJS** to deliver the static contents and made API calls to REST API accordingly.
* Used **Spring MVC Design Pattern** to hold the UI data and to serve the request and to send the response.
* Implemented **DAO pattern** to fetch data from database using **Hibernate** in order to carry out various database operations and also avoid redundant database access statements
* Designed and developed business components using **Spring Boot**, Spring Dependency Injection (Core), Spring AOP, EhCache Framework and **Spring Annotations**.
* Refactored existing **monolithic** application by identifying components that needs to be transformed to **Micro-services** architecture, with **Spring Boot** based services interacting through a combination of **REST**, **RabbitMQ** and **AWS** API-Gateway.
* Used AWS SDK for connection to **Amazon S3** buckets and **EC2** to deploy **Docker** container/images with **Microservices** architecture.
* Implemented **Kafka** producer and consumer applications on Kafka cluster setup with help of Zookeeper.
* Used **Spring Kafka** framework to publish and consumer messages into/from **Kafka** Cluster.
* Designed a sync tool for the feature, when the data refresh happens, data from sources has to be moved to the data source. For this, data is transformed and stored in **AWS S3**. The S3 path is shared via **kafka** message and the RFQ downloads them automatically.
* Exposed the REST API through the **Swagger** tool and Swagger annotations.
* Created and managed the messages asynchronously by creating queues on **Rabbit** **MQ** server and also involved in implementing the client layer using **Spring AMQP** framework to consume messages.
* Scheduled cron batch jobs using **Spring Batch** framework along with Quartz and Spring Task scheduler.
* Created new tables, **Stored Procedures**, functions, views, **indexes** and constraints, **triggers** and required SQL tuning to reduce the response time in the application in **DB2 DB**.
* Worked on **MongoDB** database concepts such as locking, transactions, indexes, shard, replication, schema design etc.
* Used **Log4j** for debugging the application at all layers of application.
* Used **SonarQube** code quality tool to maintain the code quality with solid understanding of JVM and monitored its performance.
* Focused on **Test Driven Development (TDD)** thereby creating detailed **JUnit** tests for every single piece of functionality before actually writing the functionality and decoupled it by mocking the dependencies using **Mockito** framework.
* Used **Maven** building tool for creating EAR file to be deployed in application servers.
* Developed and deployed the war/EAR files on to **WebSphere Application Server 7** and also worked with **AWS Elastic Bean Stalk.**
* Used **JIRA** ticketing system to keep track of issues and tasks on individuals.
* Used **GitHub** as Version Control tool to develop the enhancements for every release.
* Triggered the jobs in **Jenkins** for continuous integration and continuous deployments (CI/CD).
* Created **Shell Script** to invoke Java programs on UNIX batch platforms.
* Involved in production support, resolving the production job failures, interacting with the operations support group for resuming the failed jobs.

**Environment:** Java (jdk 1.8), J2EE, HTML, CSS, jQuery, JavaScript, CSS3, Angular, NodeJS, Dojo, AJAX, Spring Core, Spring MVC, Spring Security, Spring Boot, Spring Data, Spring Batch, JSP, Hibernate, Servlet, JDBC, Web services, SOA (Service-oriented) Architecture, RESTful, SOAP, Microservices, Spring cloud, Kafka, Spring Kafka, Kafka Producer, Kafka Consumer, AWS, OAuth 2.0, XML, DB2, Stored Procedures, Triggers, JUnit, Mockito, TDD, Log4j, STS, WebSphere 7, Agile/Scrum Methodology, Maven, JIRA, GitHub, Jenkins, NoSQL MongoDB, SonarQube.

|  |
| --- |
| **Wells Fargo Jan 2015 – June 2015**  **JAVA/J2EE Software Engineer**  **Des Moines, IA** |

**Description**: Wells Fargo & Company is an American multinational banking and financial services corporation headquartered in San Francisco, California, United States.

**Project: E\*Trade**

This application allows the users to get real time streaming quotes, place trades and manage the accounts. The application was developed for web platforms. The main functionality of the applications is to place trades for stocks, options, complex options and mutual funds and check the status of open orders. Addition to that, the user is also able to access all the brokerage accounts, receive real time account alerts and trade confirmations.

**Responsibilities:**

* Worked on high-level and low-level system design, planning, estimation and implementation.
* Developed presentation layer using JSP, **HTML5**, **CSS** and client validation using **JavaScript**, DOM.
* Used two-way data binding features of **AngularJS** to implement Data binding to bind various server controls to the data from database.
* Created **Node.js** Express Server combined with Socket.io to build MVC framework from front-end side **AngularJS** to back-end **MySQL**.
* Implemented **Multithreading**, **Concurrency**, Exception Handling and Collections whenever necessary.
* Developed **Microservices** and **RESTful** web services using **JAX-WS** and secured them using **Spring** **Security** **Basic Authentication** technique by validating against **LDAP** Active Directory.
* Documented and published **REST API** using **Apigee** tool and managed the API users.
* Worked with **Spring IOC** for implementing the future of DI across different layers and implemented **Spring AOP** for logging and auditing.
* Used **JSTL Tag library** to create standard dynamic web pages.
* Worked on **XML** and **JSON** for transferring and retrieving data between different applications.
* Worked on generating the web services classes by using **Service Oriented Architecture** SOA, WSDL, UDDI and **SOAP**.
* Designed and integrated the full-scale **Spring MVC**/**Hibernate** persistence solution with the application architecture.
* Used **JMS** Messaging system to send/receive messages from MQ and differentiated them by using **Apache Camel**.
* Deployed the **Spring Batch** application on **Pivotal Cloud Foundry (PCF**) instance and bind the services like Java and PostgreSQL services.
* Monitored and maintain the logs of PCF **Spring Batch** application from **PCF** Apps manager.
* Worked in wide-area network to maintain all the device related data onto **MySQL** database.
* Involved in writing complex **PL/SQL** queries using joins, **Stored Procedures**, Functions, **Triggers**, cursors, and indexes in Data Access Layer.
* Worked with **JUnit**, **Mockito** and build tools such as **Gradle**. Worked on **Log4j** for logging errors, messages and performance logs and **GitHub** for version control.
* Worked with **Jenkins** CI/CD tool for deploying the application war on to different environments.
* Deployed and tested the application in **Unix** on IBM **Web Logic Application Server**.
* Used **Eclipse** as IDE for developing and deploying in local environment.
* Collaborated with testers and developers and prepared test plans for producing high quality software applications.

**Environment:** Java (Jdk 1.7), J2EE, Spring Framework, Spring MVC, Spring Security, Spring JDBC, Spring ORM, JSP, Hibernate, CSS3, Angular JS, Servlet, JDBC, AJAX, SOA (Service-oriented Architecture), SOAP, RESTful, JMS, XML, HTML, CSS, jQuery, JavaScript, MySQL, Stored Procedures, Triggers, Eclipse, WebLogic, Agile/Scrum Methodology, GitHub, Jenkins, Gradle, PCF (Pivotal Cloud Foundry), JUnit, Mockito, Shell Scripts.

|  |
| --- |
| **GGK Technologies Jul 2012 – Jul 2014**  **Java Developer**  **Hyderabad, India** |

**Description**: GGK Tech mission is to help global enterprises grow, adapt, reinvent and digitally transform themselves through its relentless pursuit of engineering excellence & its commitment to deliver with certainty.

**Project:** **Cash Application**

This application is used to move the payments that come across the company to its proper department and if some payments were not properly disposition then this application provides the ability to move them manually and it also provides ability to import payments that can be scheduled for processing.

**Responsibilities:**

* Designed & Developed Request, Response **XML schemas (XSD)** for the Module which are part of **SOAP** request and response in SOA (Service-oriented) Architecture**.**
* Developed the functionalities using **Agile Scrum Methodology**.
* Developed service oriented middleware to incorporate two legacy source systems and published them as web-services. Incorporated source system specific business logic in Java code and exposing them as Web Service as per Base profile specifications and **WSDL** contract.
* Built the application based on **Java jdk 1.7** enhancements of Exceptions and Collections at service layer.
* Consumed **REST** Web services from server side application using **JAX-RS 2.0** libraries for modifying the data and the data expecting from client has no modification and requested from **jQuery ajax** and sent to UI directly.
* Developed the web pages using HTML, CSS, **JavaScript**, **jQuery** and **Ajax**.
* Used **Bootstrap** JS framework to make the webpage responsive.
* Developed Stateless session beans to parse (parser frame work) and process the user request into domain object (**DAO** design pattern) to delegate the business request to the business delegate object.
* Developed necessary parsing and XML construction logic using**XML parser APIs JAX-B.**
* Developed user interface using **JSP** published validation framework to support client side and server side validation.
* Used Dependency Injection feature of **Spring Framework** to inject dependency into objects and to wire objects of business classes.
* Used **Spring MVC pattern** to map the client requests and to serve it to get finally the desired object with modified data.
* Used **Spring JDBC Template** (Spring Data) in connecting to Data source to get the database connections.
* Developed persistence layer Data access objects using**DAO pattern, JDBC, Hibernate, JPA** ORM mapping tool with **Oracle** database.
* Implemented different transaction isolations based upon the business needs using **Hibernate**.
* Developed costume queries using hibernate criteria, native **SQL** and **HQL.** Developed inheritance domain objects using table per class hierarchy strategy.
* Configured **EhCache** to load the static tables into secondary storage area.
* Used built in **Stored Procedures** to reach the requirements from the Client to improve the response times of the services.
* Used **IBM RAD** to develop the application.
* Used **IBM Rally** to keep of bugs and stories.
* Used **GitHub** as build-automation tool for the process of Continuous Integration the project.
* Used Eclipse as IDE for the development work.
* Worked with **IBM WAS** (WebSphere Application Server) for deployments.

**Environment:** Java (jdk 1.7), JEE, Web-Services, SOA (Service-oriented) Architecture, SOAP, JAX-RPC, WSDL, WebSphere Application Server, Spring (Dependency Injection, AOP), Hibernate, Spring MVC, Spring JDBC (Spring Data), JDBC, JPA, JAX-B, Oracle 10g Enterprise, HTML, CSS, JavaScript, AJAX, JQuery, Bootstrap, JUnit, GitHub, Rally, Agile Scrum.