**Rajesh Bandi**

**248-255-9670**

**rajeshreddysixb@gmail.com**

**rajeshb8345@gmail.com**

* **Over 7 years** of professional experience in design, analysis, development, deployment and testing of web and distributed n-tier applications using Java in different domains.
* Experience with advanced front-end technologies like Bootstrap UI, Bootstrap modal, HTML 4.0, XHTML, DHTML, JSON, jQuery, CSS, XML 1.1. Proficient in building multi-browser compatible Web User Interface (UI) following W3C Web Standards.
* Expertise in building high performance core java applications **utilizing multi-threaded programming (thread pools, semaphores, mutexes, thread locals, re-entrant locks, wait/notify, latches, deadlocks, preventing read-write interference amongst threads/processes, etc.**).
* Experience with various J2EE design patterns like Singleton, Command, Factory, Observer, Front Controller, DTO, DAO, MVC, Session Facade, Business Delegate, Service Locator, Transfer Object, and View Object in the development of enterprise applications.
* Experience in building API’s as micro services using Spring4, Spring Boot ORM frameworks like Hibernate, JPA.
* Coded JDBC calls in the servlets to access the IBM DB2 database tables.
* Designed and developed Micro services using Java 8, Spring Boot and Angular2. Hosted on **AWS** and used **Docker, Chef, Puppet or similar orchestrators/containers**. Used Jenkins as continuous integration server.
* Strong hands on experience in developing the Enterprise and Web applications in n-tier architecture using various open source frameworks: **Java/JAVA EE, Servlets, JSP**, **Spring Framework, Spring (MVC), Spring (AOP), Spring Batch, Hibernate, web services (REST/SOAP), JMS, JPA, CXF, and SLF4J.**
* Experienced in designing and developing asynchronous event-based framework based on messaging infrastructure to process huge volume of messages using Solace queues, Apache Kafka, Storm and Zookeeper
* Extensively worked on implementing SOA (Service Oriented Architecture) using SOAP based Web services using XML, WSDL, UDDI and XML Parsers. Experience in distributed architectures, strong OO background, SOA, web services and messaging paradigms.
* Experience in implementing/following TDD methodology, developed unit/integration test suites using testing frameworks like JUnit, TestNg Mockito, JMock etc.
* Proven track record delivering distributed scalable, secure, reliable software systems.
* Working knowledge of software development methodologies and more than one of: Perl, Python, PHP, Ruby on Rails, C++, Java.
* Proficient in **TCP/IP and UDP based network programming (sockets, datagrams and multicasting) in Java.**
* Experience working with **Java NIO (**vectored I/O, memory mapped I/O, non-blocking sockets, etc.).
* Experience in identifying **Java garbage collection issues and tuning JVM parameters for high throughput or low latency**.
* Experience with scripting languages like Shell scripts and Python to be able to write orchestration and automation scripts
* Experienced in database design, data analysis and development in **MYSQL**, **PostgreSQL**, **DB2, Oracle, and writing SQL, PL/SQL, Stored Procedures, Functions, Packages and Triggers.**
* Good knowledge on Hadoop ecosystem (HDFS, Map Reduce, AWS) and Big Data.
* Hands on experience working with NoSQL systems like **MongoDB, Pig, Hive, Dynamo DB and knowledge on Cassandra, Storm, Kafka, Cloudera.**
* Experience in Enterprise Integration Development using Apache Camel Framework.
* Experience in implementing OAUTH2 and SAML federation token-based authentication/authorization models using **Spring security**.
* **Experience in using AWS Java SDK, AWS Compute, Network, Storage, Database and Application Services.**
* **Experience in using AWS storage options like EBS, Provisioned IOPS, S3 and Glacier.**
* Experience in integrating different applications using JMS, WebSphere MQ connector, ApacheCamel and Mule ESB. E
* Experience in using AWS management and deployment facilities like **CloudWatch**, **Elastic** **Beanstalk**, Cloud Formation, Virtual Private Cloud and OpsWorks.

**Education:**

* Bachelor’s in computer science Jawaharlal Nehru Technological University.

**Technical Skills**

|  |  |
| --- | --- |
| Languages | Core Java, JavaScript, C  |
| Java EE Technologies | **Java 11, JSP, Servlets**, **Spring (MVC, Core, Data, JDBC, Security), Web Services (SOAP & REST), Log4j 1.4, Xml, Linux, script, JAVA EE 1.5 and 1.6,** EJB, Axis, BPEL, WSDL, JAX-RPC, JAX-RS, JAX-WS, JAXP, JAXB, XML-RPC, Python, Camel. |
| Web Technologies | HTML, DHTML, XSLT, Java Script, ICEFaces, Ajax, JQuery, **AngularJS**, Bootstrap, Alfresco 3. |
| Databases | ORACLE 9i, 10g, 11g, Postgres, MYSQL |
| BigData/NoSQL Technologies | **Cassandra, MongoDB, Kafka, Elastic Search, Redis Cache, Spark and Storm,** **postgre sql.** Oracle 9i/10g, PostgreSQL 9.0, MySQL, DB2. |
| Cloud Technologies | **Amazon (EC2, S3, lambda, CloudWatch, API Gateway, Elastic Cache, Elastic Search, SNS, SQS, Cloud Formation, Elastic Beanstalk, IAM, Cloundfoundry** |
| Application/Web Servers | Apache Tomcat, Jetty, JBoss, Web Logic, Web sphere |
| Frameworks | Spring2.5, Struts,Casper(J2EE)**,**XFactor(J2EE),Hibernate3.0,Sencha |
| Development Tools/Version Control | **EclipseIDE**Toad, SQL Developer, XML Spy, Rational Suite (Rose, Clear Case, Clear Quest), SOUP-UI, MS Visio, SVN, **Maven, Git and Jenkins** |

**Client: Ford, Dearborn Sr. Java Developer**

**Project: Vehicle Based Payment June 20 – Till Date**

**Description:** Authorizing payments from vehicle. Connected vehicle can provide sufficient information data to create trusted vehicle identification number there by enabling transactions accepted by participating banks.

**Responsibilities:**

* Proficient in writing build scripts using **Gradle** and deploy application code to **application**/**Web** **Servers.**
* Created Multiple screen **web** **Application** with loading multiple views using route services and adding more user experience to make it more dynamic by using **JavaFx.**
* Involved in the designing and development activities.
* Developed **REST/SOAP** web services as API using **Spring** **boot** and deployed on **Tomcat** containers.
* Developed the REST web services functionality in the application to allow external applications to access the data.
* Used **Dependency Injection** feature of Spring framework and leveraged the AOP features of Spring framework to develop reusable aspects to handle exceptions, logging and compute metrics.
* Managed Object Persistence, Data retrieval using **Hibernate**, **Spring Framework** with **SQL Server** as backend.
* Utilized **Azure** to deploy **Docker** containers with both monolith and **Microservices** architecture.
* Consumed **REST** based **Micro** **services** with **Rest** template based on **Restful** **APIs.**
* Worked on deploying windows **Kubernetes (K8s)** cluster with **Azure Container Service (ACS)** from **Azure CLI** and Utilized **Kubernetes** and **Docker** for the runtime environment of the **CI/CD** system to **build**, **test** and **Octopus Deploy**.
* Developed security module for this application using **Spring** **security** and implemented **Openid/Oauth2** token-based authentication and authorization model.
* Worked on **Swagger API** and auto-generated documentation for all **REST** calls
* Worked on **Jenkins**, which is an open-source tool to perform Continuous Integration (**building**, **testing** and **deploying**) of java projects.
* Designed and implemented fully automated server **build**, **management**, **monitoring** and **deployment** solutions spanning multiple platforms, tools and technologies including **Jenkins** Nodes/Agent, **VMWare**, **Azure VM**.
* Implemented **CSRF** and **OAUTH2** security models using spring security and filters.
* Integrated Embedded **Mongo** **DB** to store vehicle details for Authorizing payment.
* Used **MongoDB** document-oriented **database** to **store** **data** and created **Object** to **Mongo** **mappings** for various domain objects
* Implemented error and **exception** **handling** using custom exception throughout the application.
* Used **JUnit framework** for Unit testing of application
* Worked on exchanging of data in **JSON** format between multiple services.
* Used **Git** version control system for check in and check out code.
* Worked under **Agile**, **Scrum** and **Extreme Programming (XP)** **SDLC** methodologies with a good knowledge of full life cycle system **design** and **development**.

**Environment:** **Java 11, Spring Boot, Micro services, JavaFX, GRPC, Azure, MongoDB, Log4j, Gradle, H2**, **XML, Jenkins, Docker, Rest Client, Rally, Jira, Git.**

**Client: Flagstar Bank, Troy Sr. Java Consultant**

**Project: Capsilon Aug 19 – May 20**

**Description:** Syncing loan creation documents between Flagstar(P8) and capsilon.

**Responsibilities:**

* Proficient in writing build scripts using **Maven** and deploy application code to **application**/**Web** **Servers.**
* Involved in the designing and development activities.
* Developed application specific **Restful** **API’s** leveraging distributed storage technologies like **Amazon SQS** and **Amazon** **S3.**
* Developed **REST/SOAP** web services as API using **Spring** **boot** and deployed on **Tomcat** containers.
* Utilized **Amazon** **Webservices** (**AWS**) **EC2** to deploy **Docker** containers with both monolith and **Microservices** architecture.
* Created **AWS** **lambda** **function**(serverless) to pull messages from **Simple** **Service** **Queue** (SQS) and route to a **REST** Endpoint in **Python**.
* Implemented **AWS API** **Gateway** to **create, publish, maintain, monitor,** and **secure APIs** at **any scale** asthe **"front door"** for **applications** to **access data, business logic, or functionality** fromthebackendservices.
* Developed security module for this application using **Spring** **security** and implemented **Openid/Oauth2** token-based authentication and authorization model.
* Developed micro services using Spring Boot and published as Restful API’S.
* Implemented **CSRF** and **OAUTH2** security models using spring security and filters.
* Experienced on **CloudWatch** to collect **monitoring** and **operational** **data** in the form of **logs**, **metrics**, and **events**, providing you with a unified view of **AWS** **resources**, **applications**, and **services** that run on **AWS** and **on**-**premises** **servers**
* Created a service to Implement **AWS** **S3** **service** to **upload** the **file** to **S3** **bucket** which would handle failures incase target system (P8) is Down.
* Integrated Embedded **Mongo** **DB** to store meta-data about the **Document** **Sync** **Process**.
* Integrated **AWS** **SQS** **API** to publish an event when the Document upload process to the target system i.e., P8 is completed.
* Implemented end to end integration of Endpoints of target system i.e., P8 with Flagstar application.
* Implemented a **Redis** **queue** **initializer** which creates a pool of tickets on application startup. This enables faster connectivity with the source system avoiding boiler plate functionality.
* Added end to end test coverage of the source code using **JUnit**.
* Implemented **AWS** **Lambda** functions for exchange of **event** **messages** with the source system i.e., **Capsilon**. This enabled me to create an event driven and a serverless architecture.
* Maintained source code quality standards using code quality scan tools such as **Check** **style** and **Sonar**.
* Worked with DevOps/Continuous integration tools like **GITHUB**, **Jenkins**, **Nexus**, **Team** **City**, **Urban** **Code** and **Jira**.
* Involved in complete development of **Agile** **Development** **Methodology**/**SCRUM** to meet customer expectations, timeline with quality deliverables and tested the application in each iteration

**Environment:** **Java 1.7/1.8, Python, Spring Boot, AWS, S3, AWS Lambda, SQS, CloudWatch, EC2, API Gateway, MongoDB, Log4j, REDIS Elastic Cache, Maven, PostgreSQL**, **XML, Bamboo, Docker, Rest Client, Micro services, Jira, Git.**

**Client: Target, Minneapolis Sr. Java Developer**

**Project: TMN Cerebro BlackBird April 18 - July 19**

**Description:**

Our promise is to provide an intuitive marketing platform that inspires guests to shop by connecting brands with the richness of our guest data & marketing reach with measurable results. We will fulfill this vision through an easy-to-use platform that breaks down the complexities of digital marketing into an accessible 4-step process (objective, plan, execution, measure) for vendors and media agencies.

**This fulfills core TMN objectives by:**

* Increasing customer investment by providing a tool that is adopted and retained.
* Reduce the time it takes to execute a campaign from planning to launch.
* Expand our customer base with advertisers small to big.

We aspire to provide a toolset to consistently deliver favorable marketing outcomes. We will be a scalable bridge for brands to meet our Guests throughout our marketing channels and inspire them to shop and engage.

Vendor representatives (sales or marketing or both) are expected to become knowledgeable in advertising, promotions, finance, and merchandising, as well as develop expertise in areas such as product knowledge and technical support.

Present and sell products at store level according to national and regional promotions while maintaining focus on marketing objectives. Execute promotions in a manner that results in increased sales or space.

Merchandise products as indicated by national bulletins or other internal correspondence. Advise customers of upcoming ads, promotions, new products, etc. Rotate stock on all product lines, ensure POP material is available and displayed, and solicit additional space during promotions and peak sales periods.

**Responsibilities:**

* Proficient in writing build scripts using Gradle and deploy application code to **application**/**Web** **Servers**.
* Designed and developed micro services business components using Spring Boot.
* Developed microservices using **Spring** **MVC**, **Spring** **Boot**, **Spring** **Cloud**, **Spring** **AOP**, **Dependency** **Injection**, **Spring** **Security**, **Spring** **Batch**, **Spring** **Transaction**, and **JPA**.
* Implemented **Salesforce** **OAuth** 2.0 **JWT** **Bearer** **Token** **Flow**.
* Integrated **Salesforce REST** **API’s** with **Spring** **Boot, Feign Client** and **MongoDB** as data store.
* Integrated **Stripe** as a **Payment** **Gateway** to generate an invoice and charges the card through **Stripe** **API**.
* Worked on **MongoDB** database concepts such as **locking**, **transactions**, **indexes**, **replication** and **schema** **design**.
* Used **Kafka** **streams** as a **message**-**bus** and developed data pipelines to ingest data to **NoSQL** **data**-**stores** like **MongoDB**.
* Expert in creating **Docker** images for providing environment support at **production** and **testing**.

**Environment:** **Java 1.7/1.8, Spring Boot, WebSphere, MongoDB, JSP, Kafka, Slf4J, Gradle, XML, Javascript, Drone, Docker, Kubernetes, Feign, Micro services, Camel, Salesforce, Stripe, Git.**

**Client: Universal Service Administrative Co (USAC), Washington Dc Sr. Technology Consultant**

**Project: Schools and Libraries Oct 17- March 18**

**Description:**

The universal service Schools and Libraries Program, commonly known as the E-rate Program, helps ensure that schools and libraries can obtain high-speed Internet access and telecommunications at affordable rates.

**Responsibilities:**

* Created Single Page Application with loading multiple views using route services and adding more user experience to make it more dynamic by using **Angular** **JS** **framework**
* Involved in analyzing the requirements, Design, Code development and unit testing.
* Worked on monitoring and performance management tools/protocols including **SNMP**, **Rest** **API**, **jVisualVM**
* Worked on building caching layer using **AWS** elastic cache (**Redis**)and exposing data with **Restful** endpoints in addition to other application specific functionality Developed micro services using **Spring** **Boot**.
* Worked on **DevOps** tool chains and with cloud technologies like **Docker**, **AWS**, and **Cloud** **foundry**.
* Migrated existing application to **Cloud** **platform** using **Java 1.7/1.8**, **Spring Boot**, **Spring Cloud**, **Micro Services**, **REST**.
* Developed application specific **Restful** **API’s** leveraging distributed storage technologies like **Apache** **Kafka**, **Amazon** **S3**.
* Developed REST API’s using **Spring** (**MVC**, **Boot**, **security**, and **data**) and hosted them on **AWS**, used **Elastic** **Beanstalk** to setup application and configured **Jetty**/**Tomcat** environments.
* Published REST API’s to fetch data from **Elastic** **Search** **clusters** for client based applications by using multi-field indexes.
* Written **maven** **scripts** to **build**, **package**, and deploy application code to **application**/**Web** **Servers**.
* Developed persistence API’s that can store data synchronously and asynchronously to **Cassandra** **data**-**store** with batching.
* Worked on integrating Spring with the **Elastic** **Search** and developed API’s to read and publish data from **elastic** **search** **cluster** using native Java transport client as well as REST client.
* Designed, developed, and deployed a multitude application utilizing **AWS** stack (**EC2**, **RDS**, **Elastic** **Beanstalk**, **S3**, **SNS**, **SES** and **SQS**).

**Environment:** **Java 1.7/1.8, Spring framework, WebSphere, Hibernate, JSP, Rabbit MQ, Log4j, Maven, XML, Javascript, Jenkins, Micro services, Camel, Elastic search Cassandra, Git.**

**Client: AT&T MPLSMon, New Jersey Technology Consultant**

**Project: SDN Monitor Feb 16- Sep 2017**

SDN-Monitor is a framework that would be used to monitor data plane connectivity. SDN-Monitorwill be deployed on selected Compute Nodes connected to the C-Leaf, and will allow us to validate connectivity to remote ends of the data plane across the Service Assurance VPN (SA-VPN).  For this to be supported, one SDN-Mon instance will validate connectivity to another SDN-Mon instance, if that is available, or to a remote L3 IP Address configured within the context of the SA-VPN.  A lack of response to the SDN-Mon validation test will be interpreted as a possible connectivity failure across the fabric (gray failure) that might in-turn impact customer traffic flow across the same two end-points.

**Responsibilities:**

* Created Single Page Application with loading multiple views using route services and adding more user experience to make it more dynamic by using Angular.js framework.
* Design and developed **Microservices** using **Java 8, Spring Boot and Angular2**.
* Perform system integration and Testing as part of an agile team Able to work on developing and designing software application for multiple environments.
* Utilizing the Hadoop framework to process historical OMS and Market Data
* Working on bug fixing and enhancements on change requests using Agile methodology.
* Involved in analyzing the requirements, Design, Code development and unit testing.
* Developed Server-side Linux based Java systems that interface with Enterprise Class Layer 2-3 SDN/NFV Network Devices, with a good application of IP Networking, network concepts, TCP stack, routers/switches.
* Applied partitioning strategies to support a quick responsive UI and meeting the scalability requirements.
* Used DB2 as the backend database using UNIX OS. Involved in design of the database schema and development of Stored Procedures.
* Worked on monitoring and performance management tools/protocols including SNMP**, Rest API, NetFlow, IPFix, JVision, Telemetry**.
* Experience in DevOps tool chains and with cloud technologies like **Docker, AWS, and Cloud foundry**.
* Developed application specific **Restful API’s** leveraging distributed **storage technologies like Apache Kafka, Amazon S3.**
* Developed **REST API’s** using **spring (MVC, boot**, security, and data) and hosted them on AWS, used Elastic Beanstalk to setup application and configured **Jetty/Tomcat** environments.
* Published **REST API’s** to fetch data from **Elastic Search clusters** for client based applications by using **multi-field indexes.**
* Designed, developed, and deployed a **multitude application** utilizing **AWS stack (EC2, RDS, Elastic Beanstalk, S3, SNS, SES and SQS).**
* Written **maven scripts** to build, package, and deploy application code to application/Web Servers.
* Worked on Jenkins, which is an open source tool to perform Continuous Integration (building, testing and deploying) of java projects.
* Used java 1.8 features like **filters and streams** in collections to retrieve and persist the data.
* Used **Kafka streams as a message-bus and developed data pipelines to ingest data to NoSQL data-stores like Cassandra.**
* Customize the display in the necessary UI framework, using PHP/JSP, jQuery, JavaScript, Angular JS, elastic search Vendor
* Developed **persistence API’s that can store data synchronously and asynchronously to Cassandra** data-store with batching.
* Worked on integrating **Spring with the Elastic Search** and developed API’s to read and publish data from **elastic search cluster** using **native Java transport** client as well as **REST client**.
* Worked on developing API’s and hosted them on AWS, created a data set & API’s (using S3 from Amazon as storage) with the ability to store and pull useful data, and do auto-scheduling & analysis.

**Environment: Java 1.7/1.8, JSP, Spring, Web Services (REST), RabbitMQ, Hibernate, Log4j 1.4, Maven, Xml, Postgres, JMS, Birt Reports, MyBatis, Jasper Reports**,**MySQLWeb Strom, Linux, JQuery, Java script, Jenkins, JMS, Elastic search, SOLR**

**Client: UnitedHealth Group, Southfield, MI Java Developer**

**Project: POLARIS JUN 14 – JAN 16**

UHC (United Health Care) is implementing the program "POLARIS" with its subsidiary OPTUM for the Healthcare Clients ACET, UPM and ICET. This is an integration project so that the Health Providers can have access to all of the patient data. All the patient data will be stored in a big data lake like Hadoop and it will be used by downstream systems. Downstream claim processing system will relate the claims data of a particular provider with the patient information and process the Medical and Dental Claims.

**Responsibilities**

* Involved in analyzing the requirements, Design, Code development and unit testing.
* Developed and consumed SOAP/Restful Web Services using XML and JSON message formats.
* Developed **REST/SOAP** based web Services as API using spring and deployed on **Jetty/Tomcat** containers.
* Worked on Camel-based integration middle-ware solution for Provisioning Services by design and Implementation of business logic and data processing routes using Apache Camel.
* Worked in Agile TDD environment using SDLC tools like Git, Jenkins, maven, Gradle for build.
* Build the application using TDD (Test Driven Development) approach.
* Worked on building application caching layer using **EHcache** and integrated with **Spring JDBC**.
* Improved the performance of the application by having a caching layer using **REDIS Elastic Cache.**
* Used service bus and message queues to integrate with EMR and clearing houses to submit claims, applied right integration design patterns.
* Used **AWS** as the hosting environment and followed a cloud-based design with NoSQL database (**Cassandra**)
* Implemented agent-server messaging dialog using Camel and JMS (Active MQ implementation).
* Published REST API’s on Elastic Search clusters to search patients and claims by multi-field indexes.
* Designed and developed **event based framework** using **Kafka distributed messaging** with custombinary message protocol.Leveraged a cascade of **producer-consumer processing chains to parallelize claim processing.**
* Integrated **elasticsearch** and developed powerful search engine by building complex queries using **elasticsearch** Java API.
* Working on bug fixing and enhancements on change requests using Agile methodology.
* Worked on writing Shell Scripts and used Git, Jenkins and maven for making builds and deployments of the application components.
* Involved in designing various Restful controllers using Spring Rest and other reusable components.
* Infrastructure setup on AWS/on demand cloud with app servers, web server, load balancing, monitoring.

**Environment:** Java 1.7/1.8, JSP, Spring, spring Boot, Web Services(REST), **Kafka**, Hibernate, Log4j 1.4, Maven, Xml, Postgres, MySQL Linux, JQuery, Java script, Angular.js, Ajax, Json, Jenkins, JMS, ESB Mule, **Cassandra, Elastic search, Elastic Cache(REDIS)**.

**Client : HOLCIM, Dundee MI Java/J2EEDeveloper**

**Project:ePOD NOV 13 – MAY 14**

EPOD (Electronic Proof of Delivery) which manages delivery management of company that allows drivers to use the mobile application for the latest deliveries. And customer support team can manage the deliveries on the backend.

**Responsibilities:**

* Involved in analyzing the requirements, Design, Code development and unit testing.
* Implemented and integrated Spring Framework.
* Worked on Spring framework for dependency injection, transaction management and AOP.
* Designed and coded Business Classes using Spring IOC and AOP features.
* Created connections to database using Hibernate session Factory, using Hibernate APIs to retrieve and store data to the database with Hibernate transaction control.
* Worked on designing and developing Core Services that are responsible to store, retrieve large data sets in scalable, performant and in secure fashion.
* Involved development of front-end components using JSP, JavaScript, and Java.
* Designed and developed all components JSP, controller, spring configuration, and DAO and Hibernate configuration for various modules.
* Interpreted and manipulated Spring and Hibernate configure files.
* Developed and Refracted the Action Servlets and JSPs using Spring following MVC architecture.
* Used the JNDI for Naming and directory services.
* Used Hibernate framework for back end development and Spring dependency injection for middle layer development.
* Design and developed Webservices client to send service requests to Webservices.
* Web UI development using HTML, CSS and JavaScript.
* Developed User- Interface screens involving Servlets, JSP, JavaScript and XML DOM parser.
* Used JSON as response type in REST services.
* Implementation of XSLT processing on server with JSP and WebSphere.
* Wrote Database objects like Triggers, Stored procedures in SQL.
* Used Spring MVC framework controllers for Controllers part of the MVC.
* Application was designed and developed using the Spring framework
* Implemented various Java design patterns like Front Controller, Service Locator, Business Delegate and Data access Object, Factory Pattern, Session Façade, etc.
* Created user-friendly GUI interface and Web pages using HTML and DHTML embedded in JSP.
* Developed various test cases and performed unit testing using JUnit.
* Implemented Maven as build and configuration tool
* Used SVN repository for version control

**Environment**: Java, J2EE, JSP, JNDI, JDBC, AJAX, HTML, Servlets, WebSphere Application server, JMS, UML, Rest Webservices, Hibernate, Spring, CVS, XML/XSL, SQL Server, Agile, Junit, Maven, SVN.