Umesh Indla

SUMMARY:

- With over 8+ years of experience as a Senior Java Full Stack Developer, I bring a wealth of expertise in developing robust and high-performing web applications.
- My proficiency spans both front-end and back-end technologies, with a strong command of Java and associated frameworks.
- Developed and maintained Java-based applications using object-oriented programming (OOP) and design patterns.
- Wrote clean, efficient, and well-documented code, collaborating cross-functional teams for high-quality software solutions.
- Experience in developing web applications using technologies like HTML, Java Script, JSON, CSS, JSP Tag libraries, JAVA11.
- Utilized Java EE technologies, Servlets, JSP, and JSTL for seamless server-side logic and integration with back-end systems.
- Experienced in Restful Services and distributed Web Services with Spring MVC, Spring Core, Spring AOP, Spring Actuator, Spring Cloud, Spring Boot, Spring WebFlux, Spring Security, Jersey, JAXB, JSON, SOAP, AXIS, JAX-WS, and WSDL.
- Skilled in RESTful APIs, SDLC, Agile Methodology (Scrum) / Kanban, RUP, Waterfall, UML, and Test-Driven Methodologies
- Strong technical skills encompassing a wide range of programming languages, frameworks, and tools, including Java17,11,8, Spring, Hibernate, React.js, Angular.js, Docker, Kubernetes, AWS, ELK Stack.
- Proficient in advanced JavaScript, including React JS, for developing high-quality and maintainable applications.
- Proficient in frameworks like Spring MVC and Hibernate, delivering scalable and maintainable solutions.
- Experience in API development using Spring, Spring IOC, Spring MVC, Spring AOP, and other related frameworks.
- Developed microservices using Spring MVC and deployed on pivotal cloud foundry.
- Strong communication and collaboration skills, actively engaging with stakeholders throughout the development process.
- Provided leadership and mentorship to junior developers, fostering a collaborative and productive team environment.
- Expertise in creating web applications using HTML5, CSS3, Bootstrap, JavaScript, React, Vue JS, TypeScript, jQuery, and node JS.
- Developed interactive and dynamic user interfaces using jQuery, enhancing the overall user experience.
- Implemented CI/CD pipelines using tools like Jenkins, automating build, testing, and deployment processes.
- Experience in mobile application development for iOS using Objective C.
- Proficient in C# for backend development, implementing efficient algorithms and data structures.
- Familiarity with Ruby on Rails, JQuery, Ajax, Git, and software version control tools.
- Strong understanding of Object-Oriented Programming (OOP) and its application in JavaScript.
- Implemented unit test cases in a Test-Driven Development (TDD) approach using tools such as Jasmine and Karma.
- Thorough understanding of authentication methods, including OpenID Connect (OIDC) OAUTH & OAUTH 2.
- Hands-on experience in deploying and managing applications on AWS infrastructure.
- Proficient in utilizing AWS services such as EC2, S3, Lambda, and RDS for seamless integration and scalability. Implementation of cloud-based solutions to enhance performance and reliability.
- Familiarity with Azure cloud services, including Azure Virtual Machines, Azure Blob Storage, and Azure Functions.

- Experience in deploying applications and managing resources within the Azure environment. Knowledge of Azure DevOps for streamlined continuous integration and delivery.
- Experience using Splunk, Azure Monitor, Cloud watch for application monitoring and troubleshooting.
- Exposure to GCP services like Compute Engine, Cloud Storage, and Cloud Functions.
- Implementation of cloud-native applications on the GCP infrastructure.
- Extensively involved in various phases of Software Development Life Cycle (SDLC) in an Agile environment.
- In summary, I am a versatile and experienced Java Fullstack Developer with a proven track record of delivering high-quality solutions and providing leadership in collaborative team environments.

TECHNICAL SKILLS

Programming Languages: Java 8, Java 11, Java 17 (JDK 1.4/1.5/1.6/1.7/1.8), Python
Methodologies/ Design patterns: OOAD, OOP, UML, MVC, DAO, Factory pattern, Session Façade, Singleton, Agile, Scrum, Kanban, Test Driven Development
J2EETechnologies: JSP (2.x), JSTL, Servlets (3.1, 3.0), JDBC, JAX-RS, JAX-WS, SOAP, WSDL, Java Security, Hibernate, JPA.
Web Technologies: JSP, Servlets, Angular, jQuery, React.JS, Node.JS, EJB, Hibernate, Struts, Spring, JDBC, HTML5, DHTML and XML, PHP, CSS3, Web Services, WSDL, SOAP, MQ Series, Apache Camel, Kafka, Spring Boot, Express.js
Web service Frameworks: Spring (4.x/3.x), Hibernate (5.x,4.x,3.x), Struts (2.5/2.3), Spring MVC, Spring IOC, Spring Boot, Spring AOP, Spring Web Flow, Spring Web Flux, JSF, Log 4J, Junit
Application/Web Servers: JBoss, Glassfish 2.1, WebLogic, Web Sphere, Apache Tomcat Server
Databases: Oracle 11g, MySQL, SQL, MongoDB, CockroachDB
OR Mapping: Hibernate, IBATIS
Web Services: SOAP, Microservices, WSDL, JAX-RS JAX-WS, Restful.
Build Scripts: Gradle, Maven
Testing Tools: JUnit, Log4J, Mock Framework, Cucumber, Selenium, JUnit, SoapUI, Mockito
Version Control/ IDE Tools: GIT, SVN, Confluence, Eclipse, IntelliJ, Spring Tool Suite (STS), RAD, NetBeans
Operating System: Windows, Linux, Unix, Mac, Solaris
Cloud Technologies, DevOps Tools & Other: AWS – (ECS, EKS, EC2, S3 Bucket, SNS, AWS cloud watch, AWS Lambda, CLI, CFT), Azure (Pivotal Cloud Foundry - PCF), Azure, GCP, Docker, AppSync Kubernetes, Connect, JIRA

EXPERIENCE

Client: Fifth Third Bank, New Jersey Jan 2023 – Till Date
Role: Full Stack Java Developer

Responsibilities:

- Involved in design and development on an Agile platform to develop processes on the Software Development life cycle (SDLC).
- Implemented Azure Identity and Access Management (IAM) to manage user roles, permissions, and access policies, ensuring secure and granular control over the customer portal's Azure resources.
- Experience in developing front end using HTML5, JavaScript, CSS3/4, TypeScript, jQuery, Bootstrap, Ajax, JSON, XML, Angular 14, React.js, and Node.js.
- Developed Python scripts to parse the Flat Files, CSV, XML, JSON files and extract the data from various sources, and load the data into a data warehouse.
- Used Angular to parse the data fetched from APIs and used backbone.js to develop an application to perform CRUD operations on the database server.

- Developed HTTP requests using Rxjs observables to communicate between server and client with Angular 14.
- Developed various screens for the front end using Angular 14 and used various predefined components from NPM (Node Package Manager).
- Integrated Azure DevOps and other CI/CD tools to automate the deployment of containerized applications on Azure Cloud.
- Implemented Spring framework for Dependency Injection, support for the Data Access Object (DAO) pattern and integration with Hibernate.
- Used Hibernate as ORM to map Java classes to database tables. Involved in setting up configuration properties, ORM mapping and CRUD operations with database.
- Used ReactJS to create views to hook up models to the DOM and synchronize data with the server as a Single Page Application (SPA) and, to create Controllers to handle events triggered by clients and send a request to the server.
- Experience in building isomorphic applications using ReactJS and Redux with GraphQL on the server-side. (Give everyone an easy and efficient way of accessing data (it uses fewer resources than the REST API, especially with mobile applications).
- Worked on reading and writing to/from the files by using newly introduced methods in Java17.
- Used Java 1.8 Method References feature to point to methods by their names and used functional Interfaces.
- Implemented all the functionality using Spring IO / Spring Boot, Thymeleaf, and Hibernate ORM.
- Executed JavaEE components using Spring MVC, Spring IOC, Spring transactions, and Spring security modules.
- Used node.js for the management of interchange of the data between the servers and Used NodeJS, Express JS to create server-side MVC framework with MongoDB.
- Used Spring Boot to build the Microservices for the system integrations.
- Monitored and optimized **GraphQL API performance** using GraphQL, Apollo Tracing, and performance profiling tools.
- Established testing frameworks covering both Spark Scala processes and Java service layers, ensuring that full-stack data flows met quality standards while maintaining efficiency in production environments.
- Designed and implemented **event-driven architectures** using **Java** and **Apache Kafka** for scalable real-time data processing.
- Created as well as consumed the SOAP as well as RESTful web services using WSDL, SOAP, JAX-WS, JAX-RS, CXF, and JAXB.
- Configured and managed **Kafka clusters** including **brokers**, **ZooKeeper nodes**, and **schema registry** for schema validation and backward compatibility Used Microservices architecture with Spring Boot, Apache Kafka message broker, deployed to AWS.
- Designed and developed the REST-based Microservices using the Spring Boot, Spring Data with JPA (Java Persistence API), and Swagger to interact with API endpoints of the Microservices.
- Designed efficient pagination, filtering, and indexing strategies in CockroachDB to enhance REST API performance.
- Conducted benchmarking and load testing to assess CockroachDB's scalability under high-traffic conditions.
- Used CockroachDB's geo-partitioning to optimize API performance for globally distributed applications.
- Used Hibernate as ORM to map Java classes to database tables. Involved in setting up configuration properties, ORM mapping, and CRUD operations with the database.
- Wrote unit and integration tests for REST APIs using tools like Postman, Jest, or PyTest with CockroachDB test environments.
- Implemented the authentication of the application using Spring Security and OAuth2.0.
- Integrate **Amazon Connect with CRM platforms** (Salesforce, ServiceNow, Zendesk).
- Design and develop RESTful APIs and microservices using Java and Spring Boot, adhering to best practices and design patterns.
- Implement Spring Security for role-based access control and authentication mechanisms in web applications.
- Utilize Spring Data JPA for efficient data access and manipulation, reducing boilerplate code and improving productivity.

- Write unit tests using Mockito to ensure code quality, test coverage, and maintainability of applications.
- Document APIs using Swagger to facilitate communication between development teams and stakeholders.
- Implemented authentication and authorization mechanisms within GraphQL resolvers using tokens and role-based access controls.
- Designed and deployed Amazon Connect **cloud contact center** solutions to improve operational efficiency.
- Experience in working with GraphQL queries and use Apollo Graph QL library.
- Utilize Lombok annotations to minimize boilerplate code and enhance readability and maintainability of codebase.
- Containerize applications using Docker, creating Dockerfiles and composing services with Docker Compose for easy deployment and scaling.
- Implemented secure communication between ZooKeeper nodes and clients using SSL/TLS and Kerberos authentication.
- Built fault-tolerant, high-throughput messaging systems using Apache Kafka with custom serialization/deserialization using Avro and Protobuf.
- Integrated GraphQL with microservices architecture, enabling seamless communication between distributed systems.
- Manage source code using Git, collaborating with team members and managing feature branches and pull requests.
- Utilize Flyway for managing database migrations, ensuring smooth deployment and versioning of database changes.
- Work with PostgreSQL database, including managing views and schemas for efficient data organization and access.
- Setting up databases in AWS using RDS, Storage using S3 buckets, and configuring instance backups to S3 buckets.
- Build the services utilizing the Pivotal Cloud Foundry (PCF) as a platform as a service for Restful services.
- Building search feature based on open-source tool Elastic Search. Implemented system logging solution using Elastic Search, Kibana, and Logstash (ELK).
- Designed and deployed GraphQL APIs using AWS AppSync to provide real-time data queries and subscriptions.
- Utilized AWS Services platform to upload data into AWS S3 buckets and create EC2 instances.
- Implemented a Continuous Delivery pipeline with Docker, Jenkins, GitHub, and AWS AMI.
- Developed BDD tests using Cucumber by writing behaviors and step definitions. Developed required Selenium support code in JAVA for Cucumber.
- Implemented monitoring and logging solutions, such as Azure Monitor and Azure Log Analytics, to gain insights into the performance, availability, and behavior of applications running on Azure Cloud.
- Created and maintained GraphQL schemas, resolvers, and data sources (e.g., DynamoDB, Lambda) in AppSync for modular, scalable backends.
- Used Maven for project dependency management and Log4j for logging Involved in activities like code review, and performance improvement and used Jira for bug reporting.
- Automated deployment and configuration of AppSync APIs using AWS CloudFormation, AWS SAM, or the AWS CDK.
- Fine-tuned query performance through caching and used CloudWatch and AWS X-Ray for logging and tracing AppSync resolvers.
- Implemented robust OIDC authentication and authorization (e.g., AWS IAM, Cognito user pools) to secure GraphQL operations.
- Employed Kubernetes and Docker within the runtime environment of the CI/CD system, facilitating application build, testing, and deployment through Jenkins.
- Design relational database schemas using PostgreSQL's data types, constraints, and indexing features, ensuring efficient data storage and retrieval.
- Developed Automated scripts to migrate using Unix shell scripting, Python, Oracle/TD SQL, TD Macros, and Procedures.

- Developing customized reports and Unit Testing using JUnit and performing the integration testing manually checking the flow between microservices.

Environment: Java 1.8/11/17, Hibernate, BitBucket, Terraform, DynamoDB, Angular 14, GCP, Cassandra, Azure, AWS, Spring Boot, Spring MVC, Microservices, HTML5, JavaScript, JQuery, JSON, XML, Bootstrap, Ajax, SOAP/REST, CSS3, Python, Jenkins, PCF, GIT, Docker, Mockito, JUnit, Jasmine, Karma, Jmeter, Schema Registery, Eclipse, Apache CXF, Oracle, MySQL, NoSQL, MongoDB, Zoo keper, PL/SQL, Maven, Log4j, Kubernetes, Agile.

Client: Infosys, Hyderabad, India Nov 2021 – Aug 2022
Role: Full Stack Java Developer

Responsibilities:

- Work with external teams in TRP to check the requirements and help them in creating the scenario and job jSON files for data reconciliation process.
- Gather the data from different resources and store in database and change data for data analysis of system requirements and performance.
- Enhanced performance of jackpot application by 80% by designing new architecture to use s3 files to save data instead of Amazon Aurora MySQL.
- Used GIT as source code collaborator and GITHub for maintaining code and documentation.
- Used Java8 Method References feature to point to methods by their names.
- Using lambda expressions, Stream API, optional and JODA time features provided by Java 8.
- Used React JS for tinplating for faster compilation and developing reusable components.
- Design and developed the application using React with internal frameworks (Remix and GSSP framework).
- Analyse design and deterministic coding, programming and integration activities based on general objective or product of solution using java11.
- Worked on React JS Virtual Dom and React views, rendering using components which contains additional components called custom HTML tags.
- Developed reusable modules in Scala for data aggregation, integrated with Java controllers in a Spring Boot app.
- Optimized Spark jobs written in Scala and ensured efficient integration with Spring Boot for responsive full-stack performance.
- Using lambda expressions, Stream API, optional and JODA time features provided by Java 8.
- Used Java 8 Method References to point to methods by their names and used functional Interfaces.
- Experience in working with GraphQL queries and use Apollo Graph QL library.
- Collaborate with development team on MySQL data Sharding, query optimize, Memcached setup.
- Worked with Microservices related frameworks Spring, Eureka for client-side load balancer.
- Implement RESTful APIs using Spring MVC framework, integrating with Spring Data JPA for seamless data access and manipulation.
- Customize Gradle tasks for specific build requirements, including code compilation, unit testing, code coverage analysis, and artifact generation.
- Configure Spring Boot applications with external properties files and environment variables for flexibility and maintainability.
- Utilize Spring Boot Actuator for monitoring and managing applications, including health checks, metrics, and logging.
- Implement Spring Boot Security for role-based access control, authentication, and authorization management in web applications, securing endpoints with JWT tokens or OAuth 2.0.
- Implemented RESTful microservices using Spring boot, Spring Integration, Hibernate, Spring Data, MVC.
- Used microservice with spring boot interacting combination of REST and Apache Kafka message brokers.

- Worked on reading and writing to/from the files by using newly introduced Java11 and migrating the IDEs from Java 1.8 to Java11.
- Implemented Amazon Aurora Serverless for cost-efficient database scaling in microservices architectures.
- Worked on Node.js Middleware framework like Express.js and Hapi.js and Yeoman, essentially developed microservices and serve them through middleware. Served any Client-side libraries through microservice.
- Used Node.JS, React.JS, and Backbone.JS MVC framework for developing the Single Page Application (SPA). Backend development done using Node.js, Cassandra, and Redis.
- Implemented the Restful API using Sequelize.js with Node.js (hapi.js framework.
- Used JSON Schema identifier to process the parameter files quicker to avoid any exceptions in next steps.
- Developed scripts to handle auto deploys to QA environment using Jenkins and git configurations on code pushes to various microservices.
- Implement CRUD operations using Jpa Repository interface provided by Spring Data JPA, abstracting away boilerplate code for data access.
- Configure Flyway migration tasks within the build and deployment pipelines using build automation tools like Gradle or Maven, enabling automatic execution of migration scripts during application deployment.
- Develop and implement database backup and recovery strategies using PostgreSQL's built-in backup tools such as pg_dump and pg_basebackup, ensuring data availability and disaster recovery readiness.
- Developed microservices architecture using Rest APIs, spring boot and Pivotal Cloud Foundry (PCF).
- Worked with Container service Docker with build port and other utilities to deploy Web Applications.
- Deployed Spring Boot based microservices Docker container using AWS EC2 container services and AWS admin console.
- Closely worked with Kafka Admin team to set up Kafka cluster setup on the QA and Production environments.
- Deploying docker images as a Fargate in AWS.
- Worked extensively with AWS services like EC2, S3, VPC, ELB, Autoscaling Groups, Route 53, IAM, CloudTrail, CloudWatch, CloudFormation, CloudFront, SNS, and RDS.
- Integrated Splunk with third-party tools such as ServiceNow, JIRA, and enterprise monitoring systems.
- Managed **Kafka administration** tasks such as topic creation, partition management, and consumer group monitoring. Created Docker containers and Docker consoles for managing the application life cycle.
- Setup full CI/CD pipelines so that each commit a developer makes will go through standard process of software lifecycle and gets tested well enough before it can make it to the production.
- Using Java8 developed API's as per the business cases analyses the current API with Spring Boot.
- Used Microservice architecture with Spring Boot based services interacting through a combination of REST and Apache Kafka message brokers.
- Used Log4j2 utility to generate run-time logs and NGNIX for reverse proxy server process to load balancing.

Environment: Java11, Java8, Eclipse, Visual Studio Code, Java 8, Postman, React, bootstrap, J2EE, Spring MVC, typescript, Hibernate JPA, Node.js, Arura MySQL, Microservices, Zuul api gateway, Restful Web services, Putty, AWS, Git, Spark Cluster, STS, Gradle

Client: Tata Consultancy Services, Hyderabad, India Jan 2017 – Oct 2021
Role: Full Stack Java Developer

Responsibilities:

- Maintained Interface compatibility and concurrency in the project using Java 8 new features like default, static methods and Concurrency API.
- Involved in creating a reusable component using React JS for DOM manipulation.
- Understanding functional specifications and documenting technical design documents for the Project.
- Developed server-side application to interact with database using Spring Boot and Hibernate.
- Built web-based applications using Spring MVC 3.0 Architecture

- Developed the application using Webworks/Xwork Framework that leverages classical Model View Controller MVC architecture using spring.
- Used Apache Kafka (Message Queues) for reliable and asynchronous exchange of important information between multiple business applications. Involved in developing an automation tool, in Java and Scala on Akka Framework.
- Develop various screens for the front end using React JS and used various predefined components from NPM (Node Package Manager) and Redux libraries.
- Consumed Web Services by implementing Spring REST support is to make the development of RESTful Web services
- Used Spring Boot at back end which helps to develop application with ease.
- Used AWS services like VPC, EC2, S3, ELB, Autoscaling Groups (ASG), EBS, RDS, IAM, CloudFormation, Route 53, CloudWatch, CloudFront, CloudTrail.
- Responsible for overall Docker & Kubernetes setup, configuration, and Architecture.
- Build microservices for the delivery of software products across the enterprise.
- Entire application is developed on Service Oriented Architecture SOA, UI User Interface being .NET and backend being JAVA, entire backend services were developed using Apache CXF Web Services, spring, Hibernate and EJB.
- Technical environment applies the latest software development principles with TDD, CI, Agile Software Development, jUnit and JIRA.
- Involved in implementing unified data platform to gather data from different sources using Kafka and Java Producers and consumers.
- Implemented Apache Kafka to provide a unified, high-throughput, low-latency platform for handling real-time data feeds, Used Java 8 Method References feature to point to methods by their names and used functional Interfaces.
- Designed and implemented Cassandra NoSQL based database and created REST enabled Web Services, involved in JUNIT testing using Mockito.
- Worked on Proof of Concept for developing application using Node Js and Mongo DB.
- Create RFP (Request for Proposal) microservice to provide RESTful API utilizing Spring Boot with Spring MVC, Integrating spring with Elastic search.
- Contributed greatly to increases in code quality and productivity through his leadership and guidance in areas such as Test/Behavior Driven Development, Refactoring, Pair Programming (Extreme Programming/XP), and Agile Estimation and Planning
- Automated Kafka topic management and configuration using **Kafka Admin API** and **Ansible scripts**.
- Worked on the Implementation of system as an independent component to be supported/integrated with Appian based BPM applications to deliver management capabilities using Appian 7.5 SAIL features
- Developed and deployed **Kafka Producers and Consumers** for asynchronous message communication across microservices.
- Create Data Model for and initial Java JPA entities for RFP project.
- Design and develop enterprise or application architectures and RFI/RFP delivery solutions using SOA Service Oriented Architecture web services OOA NoSQL OLAP cloud-based system operations ITIL XML OOD OOP XML Bamboo Jenkins IIS Chef Puppet MVC DODAF Department of Defense Architecture Framework and segment architecture.
- Involved in design and development of keyword-driven automation framework using Selenium Web driver.
- Moved the project to Java 8 to extensively use Lambda expressions and have better connectivity to Cloud in addition adding Spring Boot Libraries and UI Enhancement.
- Implemented **Kafka Streams** for real-time data transformation and enrichment pipelines.
- Prepared test cases and provided support to QA team in UAT.
- Used WebSphere Application Server 7/ WebSphere Portal server 7.0 to create, debug and deploy application modules.
- Collaborated with business analysts to convert business/functional requirements into technical specifications, employing Agile methodology and participating in Scrum meetings to tailor features to customer needs.

- Resolved application issues, conducted unit testing using JUnit, and assisted in defect mitigation.
- Utilized Spring MVC to design the web application following the MVC design pattern.
- Crafted SQL queries and customized stored procedures for TERADATA tables.
- Implemented Maven scripts for automated deployment of the application.
- Utilized Web Sockets to stay connected, receiving subscriptions and dispatching local notifications upon new requests.
- Developed applications on the Spring framework, leveraging features such as Dependency Injection, Spring Beans, Spring Security, Spring JPA, Spring AOP, Spring IOC, and Spring Web Flow using MVC.
- Scripted build, deployment, maintenance, and related tasks using Jenkins.
- Designed and developed the web application using JSP/Servlets and the Spring framework.
- Created Spring beans for handling business logic as part of maintenance and change request implementations.
- Conducted integration and deployment of the application, performed load testing, and fine-tuned performance.
- Configured Log4j for debugging purposes.
- Designed web screens for various modules using HTML, BOOTSTRAP, and CSS, incorporating client-side validations with JavaScript, jQuery, and Angular.
- Designed base architectural components based on Spring MVC pattern using Spring Boot.
- Implemented Spring IOC for core business application logic and utilized Hibernate as a persistence framework.
- Employed AJAX for implementing dynamic web pages, fetching content via API calls and updating the DOM (JSON Parsing).
- Developed stored procedures, triggers, and packages in MySQL. Utilized GIT for version control and JIRA for bug tracking.
- Actively participated in documentation, code reviews, release management, and post-implementation support.
- Used JIRA tool for Issue/bug tracking, monitoring of work assignment in the system and GIT for version control.

Environment: Java, HTML, CSS, BOOTSTRAP, JSP, AJAX, jQuery, Jenkins, Angular, JavaScript, Spring MVC, EJB, Apache Tomcat Application Server 8.0, Spring 4.2.3, GIT, Azure, Hibernate, Oracle 10g, Eclipse, Swagger, Postman, WinSCP, Teradata Studio, Gulp, Node Server, Junit, spring 3.0, Hibernate 3.0, REST, HTML, CSS, JavaScript, Nexus, Jfrog, SonarQube, React, Vue.js, Kafka, EXT Js, Oracle 12c, Zoo keeper, Mule ESB, Jasper soft, Scale, Oracle OSB, Tomcat, Web Sphere 7.0, Elastic Search, ANT, TFS, Joint, Mockito.

EDUCATION

- Masters, Management Information Systems, University of Utah (August 2022 - August 2023)
- Bachelors, Computer Science, Gitam University, Visakapatanam, India, (July 2014 - May 2018)