|  |  |  |
| --- | --- | --- |
| A person in an orange shirt  Description automatically generated | **Subhashis** |  |
| **Nath** |
| **Senior Staff Engineer** | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Professional Summary** |  | Subhashis is a technically proficient and highly insightful Software Engineer, with more than 8 years of experience in developing secure, robust, scalable and highly available web applications. He offers 3 years of efficiency in working on  full stack development using Java, ReactJS, Microservices, Postgres, Cosmos DB, Redis, TypeScript and deploying to Azure Functions and Azure App Services using CI/CD pipeline. He is competent in integrating 3rd-party solutions using REST APIs and message queues such as RabbitMQ. He has a strong know-how in OOP, design patterns, low level system design, coding best practices, and code reviews. He offers hands-on experience in writing optimized SQL queries, DB triggers for RDBMS and NoSQL databases.  He has latest worked on In-App survey platform integration project for a Swedish multinational networking and telecommunications company, involved in creating a service and related route and APIs in Express.js and TypeScript which could invoke company’s REST APIs and crud data in Azure Cosmos DB. |

|  |  |  |
| --- | --- | --- |
| **Education** |  | * Bachelors in Technology, Computer Science and Engineering, Netaji Subhash Engineering College, Kolkata, West Bengal, India |

|  |  |  |
| --- | --- | --- |
| **Training and Certifications** |  | * Certification on React – The Complete Guide 2023, Udemy |

|  |  |  |
| --- | --- | --- |
| **Skill Set** |  | * Areas of competencies include robust web architectures, domain driven design in Microservices architecture, cloud application architectures * Skilled in implementing database caching to backend to improve data fetch latency * Leveraged success in RCA of a major Java Spring Boot Azure App service in production environment which was underperforming and thus devalued the quality of client offering and client brand image * Experience in: * Upgrading the tech stack as it introduced breaking SDK interfaces which required refactoring entire backend service but ultimately resolved this critical blocker * Playing the key role in RCA, tech upgrade and application refactoring; it was an intermittent issue and performance test result confirmed and it was not dependent on load * Analysing TCP dump capture of network traffic and discovering that Cosmos DB connection was automatically getting disconnected and reconnected in different times thus increasing HTTP response latency * Adept in: * Connecting with MS Azure support team in number of calls if they could shed some light and suggest some workaround; there is no stack overflow thread or ChatGPT suggestion available which directly addresses this issue * Fixing this issue by upgrading Azure SDK client version to 4, Spring Boot version to 2.7 and Java version from 8 to 17 |

|  |  |
| --- | --- |
| **Programming languages** | * Java/Java 17, ES6, TypeScript, Python |
| **Web technologies** | * ReactJS, React Functional Components, React Hooks, React Router |
| **Server side technologies** | * Spring Boot/Spring Boot 2.7, Spring Security, Spring JPA and JDBC, Express.js, ,Azure Web App, Azure Function App, Azure Event hub, Azure Key vault, Microsoft Identity Platform Oauth2, Django, Quarkus |
| **APIs and middleware** | * Web Services, REST, JSON, Apache Kafka, RabbitMQ |
| **Database** | * SQL, PostgreSQL, Cosmos DB, MongoDB, Redis, Elasticsearch, database triggers, stored procedure, data schema design |
| **Tools and platforms** | * IntelliJ IDEA, VS Code, Git, TFS, Draw.io |
| **Additional** | * CI/CD pipeline, Docker, RabbitMQ, JUnit, Mockito, Jest, React Testing Library, Maven, Gradle, Git, JaCoCo, Fortify scan |
| **Others** | * Requirements analysis and effort estimation, defect RCA, CI/CD pipeline creation, technical documentation, code review, taking part in daily standup, sprint planning, sprint retrospective, demo, etc. |

|  |  |  |
| --- | --- | --- |
| **Honors and Awards** |  | * Got appreciation E-mail from client stakeholder for showing exceptional ownership and commitment and going extra miles when team required |

|  |  |  |
| --- | --- | --- |
| **Recent Projects** |  |  |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Knowledge Management POC** |
| Duration |  | Oct 2023 – Dec 2023 |
| Description |  | POC involved sending notification to Nagarro subscribers when a HR creates or updates a document in Onedrive or updates the content in a MS Sharepoint page. |
| Technology Stack |  | Django, Azure Cosmos DB for MongoDB, Azure Event hub, Oauth2 OBO Auth flow, MS Graph Api, Azure Key vault |
| Responsibilities |  | * Active participation in high level design discussions for the app POC. * Bootstrap a new python Django app and integrated Azure Event Hub with MS Onedrive using MS Graph apis. Thus the subscriber could receive change notification event payloads when a new file is uploaded to Onedrive or file content is updated in an existing file in Onedrive. * The payload of above events contains subscription id of the file change event subscribers. Using that subscription id we implemented OAuth2 On-Behalf-Of or delegated authorization token flow as follows. * Write logic to fetch refresh token on behalf of the subscriber first. On successful retrieval we could fetch access token further which is used to interact with MS Graph Apis on behalf of the subscriber till the access token gets expired. Then we had to renew the access token using the refresh token retrieved earlier. Used python MSAL lib. * Download files for which change notification received using relevant MS graph api invocation. * Implemented a parser module to parse these pdf/docx/doc using python libraries. * implemented rest apis to CRUD parsed paragraph texts into Azure CosmosDB for MongoDB api. * Secured the sensitive credentials in Azure Key vault. Kept other config or setup parameters in dotenv files. * *Initially this work was decided to be implemented using java quarkus framework so did some implementation in quarkus also.* * *Initially Kafka was thought to be suitable message bus for our use case as the system must handle high throughput traffic so upskilling done on Kafka. Kafka is the backbone of Azure Event Hub that we used in our application*. |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **In-App Survey Platform Integration** |
| Client |  | A Swedish multinational networking and telecommunications company. |
| Duration |  | May 2023 – July 2023 |
| Description |  | Project involved integrating client’s In-App survey with client’s real estate portal. |
| Technology Stack |  | ReactJS, Express.js, Azure Cosmos DB, Azure CI/CD |
| Responsibilities |  | * Created: * Component level design of the interaction between all the modules and data flow within and across the system * A dialog in ReactJS to prompt user once they logged in for an In-App survey if available; user could take the survey by clicking on “take me to the survey” button which would open external company survey page in a new tab in the browser * A service and related route and APIs in Express.js and TypeScript which could invoke company’s REST APIs and crud data in Azure Cosmos DB * Implemented an API to check if an active survey was available at user home location at user login time and return a user specific survey URL in company platform; if user clicked on this link and user was not already part of company send out list for that survey, then added user to that list at runtime. A set of company API invocation happened in the background, creating recipient ID for that user for the on-going survey without any manual intervention * Responsible for generating the build for all environment and doing admin tasks such as merging the code and reviewing the code of peers * Troubleshot issues captured in QA * Wrote unit test cases for the components and services using Jest and React Testing Library |
| **Project name** |  | **Real Estate Fault Reporting Platform** |
| Client |  | A Swedish multinational networking and telecommunications company. |
| Duration |  | June 2022 – April 2023 |
| Description |  | Project involved providing employees a platform for reporting real estate (such as building, floor, desk, laptop, monitor, lights, Wi-Fi, coffee vending machine, sensors, desks, etc.) related faults and getting resolutions hassle-free. |
| Technology Stack |  | ReactJS, Express.js, PostgreSQL, Azure Function, Azure App Service, CI/CD |
| Responsibilities |  | * Developed a web application to facilitate fault/issue reporting experience by employees in the customer system * Contributed to both backend and frontend development * Wrote: * Few screens using ReactJS Functional Components, Hooks and Router using Vanilla.js; used customer’s design system to style all the components * One custom accordion component using React as such component was not available in the existing design system; extracted styles from the existing design system to create similar look and feel to the custom component * SQL queries to query the database having multiple tables containing relational fault report data using foreign key references * Azure Web App in Node.js using TypeScript; data was stored in Postgres DB * E-mailer using Nodemailer and Handlebars npm * Unit test cases for the components and services * Maintained daily interaction with client and updated them regarding progress * Implemented: * Fault reporting backend using Node.js, Express.js, TypeScript and database was Postgres * Azure Web Job to above database every 5 minutes to fetch newly reported fault and process them; based on the fault reported location and type of fault reported, a workorder request was dispatched to one of the Azure Functions. Each such function was responsible for integrating with 3rd party vendor REST APIs used to actually create a new workorder in the corresponding vendor. In response workorder ID or error was returned to the Azure Function app and then to Web Job. Web Job sent this status to the fault reporter via E-mail notification |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Workplace Seat Booking Application Upgradation** |
| Client |  | A Swedish multinational networking and telecommunications company. |
| Duration |  | December 2022 – January 2023 |
| Description |  | Project aimed at working on a platform for client employees and contractors to book workplace seats in advance. The project involved technical upgradation of this platform to fix few production blockers. |
| Technology Stack |  | Java 17, Spring Boot to v2.7, Azure Function, Azure App Service, Azure Cosmos DB, Java SDK v4, Azure Redis, Azure CI/CD |
| Responsibilities |  | * Upgraded desk booking backend Java Spring Boot application; it was suffering from Cosmos DB connection shutdown intermittently and Java memory leak issue in Azure production deployment * Migrated Java 8 to Java 17, Spring Boot to v3, Azure Cosmos DB Java SDK to v4 * Used: * Java SDK new feature such as record, sealed classes, etc. * Redis datatypes to store data instead of Java collections * Renamed Java EE namespace from Javax to Jakarta * Replaced MVN build agent by Gradle * Introduced Redis caching to data retrieval CRUD APIs; API data was cached on cache miss of the keys using Spring Data Redis. Data was saved and updated through API triggers Azure Functions to evict related key values from the Redis. Cache eviction logic was taken care of by Azure Function which was external to Spring Boot Azure Web App to decouple cache eviction from the app itself to maintain cache data integrity * Generated the build for all environment and did admin tasks such as merging the code and reviewing the code of peers * Troubleshot issues captured in QA * Wrote unit test cases for the components and services using JUnit and Mockito |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Office Management Platform for Real Estate** |
| Client |  | A Swedish multinational networking and telecommunications company. |
| Duration |  | January 2022 – June 2022 |
| Description |  | Project involved working on a real estate platform with several workplace related tools to make workplace experience much better for client employees and contractors. |
| Technology Stack |  | Java 11, Spring Boot, Cosmos DB, Azure Functions, Redis |
| Responsibilities |  | * Developed a web application which facilitated client employees and business partners in various workplace related activities * Wrote: * Backend app using Express.js and TypeScript and deployed to Azure Web App * Code using Java, Spring Boot, Cosmos DB, Azure Functions * Secured the application using Single Sign-On Microsoft Authentication Library (MSAL) * Made customer employee and contractor seat booking experience hassle free; on successful booking an Azure Function having Cosmos DB input trigger sent E-mail notification with desk booking details to the user * Introduced Redis caching to data retrieval CRUD APIs * Used: * Azure Function for cache eviction * Redis datatypes to store data instead of TypeScript data types |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Inventory Management with Autonomous Drones** |
| Client |  | A Swedish multinational networking and telecommunications company. |
| Duration |  | June 2021 – December 2021 |
| Description |  | Project involved providing customer a smart inventory management platform with autonomous drones. |
| Technology Stack |  | Java 11, Spring Boot, ReactJS, MQTT, Docker |
| Responsibilities |  | * Served as a Full Stack Developer * Developed indoor inventory management portal for autonomous drones * Wrote REST APIs as well as few of the screens using ReactJS * Participated in different phases of the delivery (development, unit test, CICD pipeline creation, defect fixes) to release this application on agreed time without compromising quality |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Payment Platform** |
| Client |  | A British multinational bank with operations in consumer, corporate, and institutional banking and financial services. |
| Duration |  | July 2020 – May 2021 |
| Description |  | Project involved providing a customer bank automated regression test framework with zero tolerance to failure for their quality assurance team. |
| Technology Stack |  | Spring Boot, JDBC, ELK, Docker, OpenShift cluster, Cucumber |
| Responsibilities |  | * Developed: * Components/modules in the Microservice based architecture * A simulator for couple of business-critical core-banking Microservices viz EBBS and T-SaaS in payment flow; the simulator mocked expected response payload and response status from these external core banking B2B endpoints, thus, making regular automated sprint tests and regression tests independent of the core banking system * Involved in developing automation test framework |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Secure Apps via SSO** |
| Client |  | An American multinational corporation specialized in selling data storage, information security, virtualization, analytics, and cloud computing. |
| Duration |  | February 2020 – June 2020 |
| Description |  | Project involved working on a solution which would eventually have very high impact on all the projects under ESRS modifying security requirement to SSO. |
| Technology Stack |  | Java, Spring Cloud Gateway, Spring Security, PCF SSO |
| Responsibilities |  | * Implemented: * An all-in-one SSO solution using Spring Cloud Gateway and Spring Security SSO securing access to several sensitive business APIs and web portals using PCF PingFederate SSO Service plan as Identity Provider (IdP) * An authentication filter to support both BASIC, SecureID mode of SSO authentication, differentiated by AUTH Mode header, and integrated the filter with above gateway |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **360-Degree Service with Actionable Intelligence** |
| Client |  | An American multinational corporation specialized in selling data storage, information security, virtualization, analytics, and cloud computing. |
| Duration |  | June 2019 – February 2020 |
| Description |  | Project involved working on a platform which could provide all round services with actionable intelligence. |
| Technology Stack |  | Spring Boot Scheduler, Spring JDBC, Oracle, PostgreSQL, MongoDB, RabbitMQ, PCF |
| Responsibilities |  | * Wrote: * PCF Spring Boot Scheduler service which monitored some table every minute, sent notification once there was an update in table entries; the notification message contained device details such as serial number, product type and API status as payload and routed to message queue * APIs to retrieve device, gateway, and cluster details respectively from multiple databases such as MongoDB API, Oracle, and PostgreSQL and sent response in a specific template that client API understood * Corresponding API Gateway Controllers to authenticate inbound requests using Spring Boot and HTTP basic authentication and deploy to PCF * SQL and DB for inserting, updating, and deleting triggers |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Remote Service Authentication** |
| Client |  | An American multinational corporation specialized in selling data storage, information security, virtualization, analytics, and cloud computing. |
| Duration |  | May 2018 – May 2019 |
| Description |  | Project involved establishing a secure communication channel between an end-user device which needed to be troubleshot remotely and customer gateway by generating OTP securely during each remote login attempt by the support personnel to the devices so that there was a minimum downtime of the device during the troubleshoot process. |
| Technology Stack |  | Java, Spring API Gateway, PostgreSQL, Cronjob, OpenSSL, X.509, IDP |
| Responsibilities |  | * Wrote: * Microservice endpoints and corresponding API Gateway Controllers to authenticate inbound requests using Spring, Jersey and HTTP basic authentication and API key concept * Python scripts for device specific activities and automated tests * Deployed the services in PCF * Involved in the development and maintenance for remote access to different IOT storage products using complex cryptography techniques viz Bouncy Castle and OpenSSL to generate X.509 Certificates and ensure secure HTTPs communication between customer storage device and end user gateway * Fixed SP unauthorized access issue by syncing time between remote Shibboleth IDP at backend and SP at end user gateway * Wrote and scheduled DB archive Cronjob |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Cloud Foundation Automation** |
| Client |  | An American cloud computing and virtualization technology company. |
| Duration |  | July 2017 – May 2018 |
| Description |  | Project involved working on automated imaging and bringing up Cloud Foundation Racks in VMware Cloud (VCF). |
| Technology Stack |  | Java, Spring Boot, Selenium |
| Responsibilities |  | * Developed automated imaging and brought up framework for VMware Cloud Foundation (VCF) Racks consisting of ESX, servers, switches, vSAN, vCPU, NSX, etc. |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Concept Search** |
| Client |  | A full–service compliance and trade analytics software company. |
| Duration |  | December 2015 – June 2017 |
| Description |  | Project involved working on an AI solution in the litigation context. |
| Technology Stack |  | Java, Open NLP library |
| Responsibilities |  | * Wrote Parser using Apache Open NLP 1.5 library which could parse Word Doc/Docx files based on pre-defined model templates * Used artificial intelligence to provide user the facility to search text phrases (concepts) within the document and provide meaningful textual values |

|  |  |  |
| --- | --- | --- |
| **Project name** |  | **Internet Content Delivery Acceleration** |
| Client |  | An Indian software company providing cyber security network and web development solutions. |
| Duration |  | April 2014 – November 2015 |
| Description |  | Projects involved working on a solution that could accelerate page load time on low bandwidth network by using CDN intelligently. |
| Technology Stack |  | Java, Amazon S3 |
| Responsibilities |  | * Used Amazon S3 to store media files * Managed Selenium WebDriver Java coding to automate measurement of web page load times * Wrote NSIS scripts to build exe installer of the application |