Karan Chopra

Software Engineer | karanc4@uw.edu | +1-646-392-5451 | LinkedIn | GitHub | https://karanchopra1996.github.io/

EDUCATION

University of Washington, Bothell, WA | *Master of Science, Computer Science & Software Engineering* Sep 2022 - Present
Coursework: Algorithm Design and Analysis, Distributed Computing, Software Architecture, and Evaluating Software Design.
Guru Gobind Singh Indraprastha University, Delhi, India | *Bachelor of Technology, Information Technology* May 2018
Coursework: Data Structures, Object Oriented Programming, DBMS, Operating Systems, and Web Engineering.

SKILLS

Programming Languages:	Java, Python, C++, C, JavaScript, HTML, CSS, SQL, C# and .Net.
Cloud:	Amazon Web Services (AWS): Amazon Elastic Beanstalk, Amazon Lambda, Amazon ELB, Amazon Cognito, Amazon Kinesis, Amazon DynamoDB, Amazon EC2, Amazon VPC, Amazon IAM, Amazon SQS, Amazon RDS, Amazon SNS, Amazon S3 and Amazon CloudWatch.
Databases:	MongoDB, DynamoDB, MySQL, RDS, PostgreSQL, and MS SQL.
Operating System:	Windows and Linux.
Technologies & Frameworks:	Spring-Boot, React, NPM, NodeJS, Flask, RESTful APIs, Apache Kafka, Apache Storm, Zookeeper, Hazelcast Jenkins, Agile, Redux, and Maven.
Tools:	Git, Jira, GitHub, Visual Studio Code, IntelliJ, and PyCharm.

WORK EXPERIENCE

Software Engineer, Accenture, India Apr 2020 – Aug 2022

- Elevated user engagement by 25% and improved operational efficiency by 20% over 6 months by seamlessly integrating 15+ third-party services into a client's web application using Java **Spring Boot** and **MS SQL.**
- Designed highly scalable **Apache Kafka** streaming app processing 200K-500K topics daily, reducing backend server workload.
- Developed **RESTful** APIs serving data in **JSON** to **front-end** based on dynamic user inputs handling over 30,000 users.
- Wrote **PowerShell** script analyzing network load on MS SQL database, notifying bottlenecks, resulting in 30% less downtime.
- Created **Jenkins** script to automate **CI/CD** pipelines to build, test, and deploy processes, saving 20% time.
- Developed **JUnit** test cases in **Java** to detect defects early in the development phase resulting in increased code coverage by 35%.
- Led **GitHub** Wiki documentation and **code reviews**, resulting in consistent code, 20% fewer errors, and a 15% code quality boost.

Associate Software Engineer, Accenture, India Oct 2018 – Mar 2020

- Re-designed Hospital management application with a 30% improvement in response time. [**Java, HTML, and MongoDB**]
- Implemented **REST APIs** with **Spring Boot** and **Hibernate**, enhancing event management and propelling a 40% improvement in coordination and a 30% user increase in the platform's event module over 6 months due to improved usability and functionality.
- Created dynamic, responsive **React JS** dashboards for enhanced user experience, boosting engagement and saving 15% time.
- Used **CSS3**, and **HTML5** for design/animations and **Redux**, RESTful APIs for efficient state/data management saving 9% time.
- Ensured top-notch code quality and scalability through comprehensive unit testing with **Mockito,** due to its straightforward API.

Software Engineering Intern, National Informatics Centre, India May 2017 – July 2017

- Created validation forms on **.NET** framework with **HTML, CSS** for UI, and stored procedures in **MS SQL server** reducing user input errors by 25% and boosting data retrieval speed by 15%.

PROJECTS

- **Canvas Learning Tool** (Python, Redux, React, NodeJS, Kanban, and MS Azure) [Ongoing]
  The tool used in our university by the faculty to interact with the students, currently working on this project to extend the features, make the website open source and live, create **CI/CD** pipelines, and perform acceptance testing. Practicing **Agile Methodologies**, **full stack development**, using **MS Azure** as the service base, and **Kanban** board for tracking progress.
- **P2P Online Tic Tac Toe** (Java, JFrame, and JSCH)
  This project exercises a **peer-to-peer** communicating program using non-blocking accept, multiple threads, (specifically saying, the main and the slave threads), and **JSCH** (Java secure shell). Involves two users in the same tic-tac-toe game or allows a single user to play with an automated remote user using **Java Socket** connections. **JFrame** is used for the GUI part.
- **Flight Data Analysis** (Java, JSCH, Apache Storm, HTML, and Zookeeper)
  Performed distributed data streaming and analysis of the flight data to measure the traffic in airports in the US using **Storm** framework along with **Zookeeper** coordination service for cluster management.
- **Carethroz** (Python, Flask, Bootstrap-HTML, CSS, and JavaScript)
  A Senior caregiver services marketplace application**.** Used **Python** as the backend language, **Flask** handles the server-side logic, dynamic content generation, and routing, while **Bootstrap** enhances the user interface by providing a consistent and appealing design.
- **Local vs. Remote Execution of Hazelcast-based inverted indexing.** (Hazelcast, Java and JSCH)
  This project creates two versions of an inverted indexing program with **Hazelcast's distributed map**. The first counts word occurrences in files and displays the result. The second uses **remote execution** to count word occurrences on cluster nodes, exploring Hazelcast's remote execution and measuring performance.

CERTIFICATIONS AND AWARDS

- AZ-900 (AZURE fundamentals), Microsoft & JAVA, C, and C++ training, IIT Bombay, India.
- Performance recognition awards “Star of the month”, April 2021, & “Act as a true partner-Stewardship”, July 2022 by Accenture.