**Karan Chopra**
Software Engineer| Seattle, USA| karanc4@uw.edu | +1-646-392-5451 | LinkedIn| GitHub| Website

## EDUCATION

**University of Washington, Bothell, USA |** *Master of Science, Computer Science & Software Engineering* **Sep 2022 - Present**
Relevant Coursework: Algorithm Design and Analysis, Distributed Computing, Software Architecture, Evaluating Software Design, Software Management, and Product Development.

**Guru Gobind Singh Indraprastha University, New Delhi, India |** *Bachelor of Technology, Information Technology* **Aug 2014 - May 2018**
Relevant Coursework: Data Structures, Object Oriented Programming, Database Management Systems, Operating Systems, Web Engineering, Mobile Computing, Advance Computer Networks.

## SKILLS

**Programming Languages:** Java, Python, C++, C, JavaScript, HTML, CSS, SQL, C# and .Net
**Cloud:** Amazon Web Services (AWS): Elastic Beanstalk, ECS, Fargate, ELB, Route53, Cognito, SNS, SQS, Kinesis, DynamoDB, EC2, VPC, Auto Scaling Groups, NACLs, VPC, IAM, CloudFront, CDK, CodeComit, CodeBuild, CodeDeploy, SQS, RDS, SNS, S3 and SAM.
**Databases:** MongoDB, DynamoDB, MySQL, RDS, PostgreSQL, and MS SQL
**Operating System:** Windows and Linux.
**Technologies and Frameworks:** Spring-Boot, React, NPM, NodeJS, Flask, RESTful APIs, Kafka, Apache Storm, Zookeeper, Hazelcast Jenkins, Agile, Redux, and Maven.
**Tools:** Git, Jira, GitHub, Visual Studio Code, IntelliJ, and PyCharm.

## WORK EXPERIENCE

**Software Engineer II, Accenture, India** **Apr 2020 – Aug 2022**

- Integrated various third-party services with the client's web application using Java **Spring Boot** and **MS SQL** as backend stack.
- 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 concurrent users.
- Wrote **PowerShell** script analyzing network load on MS SQL database, notifying bottlenecks resulted in a 30% decrease in 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** effort for enhanced team collaboration and conducted **code reviews**, yielding consistent code, a 20% reduction in errors, and a 15% increase in code quality.

**Software Engineer I, Accenture, India** **Oct 2018 – Mar 2020**

- Re-designed Hospital management application with a 30% improvement in response time. [**Java, HTML, and MongoDB**]
- Designed and implemented seamless REST APIs using **Spring Boot** and **Hibernate**, enabling the creation of meeting events.
- Created dynamic, responsive **React JS** dashboards for enhanced user experience, boosting engagement and saving 15% time.
- Used **CSS3, HTML5** for design/animations and **Redux**, RESTful APIs for efficient state/data management.
- Ensured top-notch code quality and scalability through comprehensive unit testing with **Mockito.**

**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**.

## 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 & 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 - Senior caregiver services marketplace application** (Python, Flask, Bootstrap-HTML, CSS, and JavaScript)
  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 visually appealing design.
- **Mobile-Agent Execution Platform** (Java and JSCH)
  Implemented a mobile-agent platform using **RPC**, **dynamic linking**, and **object serialization** in Java, incorporating **RMI**, **class loading**, and **Java object streams**.
- **Local vs Remote Execution of Hazelcast-based inverted indexing.** (Hazelcast, Java and JSCH)
  This assignment involves two versions of an inverted indexing program using **Hazelcast's distributed map**. The first version, counts word occurrences in each file, displaying file names and counts. The second version uses **remote execution** to count word occurrences in local files on each cluster node, aiming to explore Hazelcast's remote execution mechanism and measure performance.

## CERTIFICATIONS AND AWARDS

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