

# Chaitanya Deepthi Kaki

San Jose, CA, USA | [chaitanyadeepthikaki@gmail.com](mailto:chaitanyadeepthikaki@gmail.com) | [Linkedin](#) | [Github](#)

## EDUCATION

<b>San Jose State University</b> Master of science, Computer Engineering <b>CGPA:</b> 4.0/4.0 <b>Coursework:</b> C++, Computer Architecture, Enterprise software platforms, Artificial Intelligence and Data Engineering, Data Mining, Human Computer Interaction, Software security technologies.	San Jose, CA, USA August 2021 - May 2023
<b>Vellore Institute Of Technology</b> Bachelors of Technology, Electronics and Communication Engineering <b>CGPA:</b> 3.8/4.0 <b>Coursework:</b> Data Structures and Algorithms, Operating Systems, Object Oriented Programming, Cryptography and Network Security	Chennai, TN, India July 2016 - June 2020

## TECHNICAL SKILLS

- |   |   |   |
|---|---|---|
| <ul style="list-style-type: none"><li>• Python</li><li>• JAVA</li><li>• Javascript</li><li>• REST APIs</li><li>• Node JS</li><li>• React JS</li><li>• Kafka</li></ul> | <ul style="list-style-type: none"><li>• Flask</li><li>• REST API</li><li>• Django-Rest Framework</li><li>• Boto3</li><li>• MySQL</li><li>• MongoDB</li><li>• Docker</li></ul> | <ul style="list-style-type: none"><li>• Jenkins</li><li>• CI/CD</li><li>• Unit testing</li><li>• Amazon Web Services</li><li>• Grafana</li><li>• Git</li><li>• Kubernetes</li></ul> |
|---|---|---|

## WORK EXPERIENCE

<b>Software Engineering Intern</b> <b>Celona Inc</b>	May 2022 - August 2022 Campbell, CA, USA
<ul style="list-style-type: none"><li>• Analyzed Celona’s AWS and Splunk costs by developing an AWS cost-tracking dashboard</li><li>• Used Boto3, an AWS SDK for python to retrieve historical and forecasted cost data for all AWS services being used by the company's tech stack</li><li>• Architected an AWS pipeline to get cost data from AWS Billing to S3, and then trigger a lambda function to send data to API endpoint</li><li>• Designed Python API and Django Orchestrator to send data to time-series database, InfluxDB</li><li>• Created a Grafana dashboard to visualize the costs associated with each AWS service and usage type</li><li>• Built a rest API using Django-Rest Framework to serve cost data, in order to facilitate future applications.</li><li>• Took ownership in demonstrating the feature demos of the product to senior employees, resulting in constructive feedback and comments</li><li>• <b>Tech stack:</b> Grafana, InfluxDB, AWS Lambda, AWS S3, Boto3, Django, Rest API, Python, Pandas</li></ul>	

## PROJECTS

<b>Facebook clone using React JS</b> <ul style="list-style-type: none"><li>• Implemented frontend part of the Facebook Home page with React JS, Google-Firebase for Authentication, and Database Set-Up</li><li>• Containerized the application with Docker and Docker Compose and stored the image in AWS ECR</li><li>• Frabricated AWS codepipeline that automates deployment to AWS ECS (Fargate), provisioned with Elastic Load Balancer</li><li>• <b>Tech stack:</b> React JS, AWS, Google-Firebase, CI/CD, Jenkins, Docker</li></ul>
<b>Amazon Shopping Website UI clone using React JS</b> <ul style="list-style-type: none"><li>• Designed the navigation between pages, Sign-In, Home and Checkout, using React-Router-DOM component</li><li>• Authentication set-up for the sign-in page is framed using Google-Firebase. Products data on AWS RDS for SQL server.</li><li>• Fabricated “Add to Cart” functionality using Redux that transfers data of products from Home page to checkout page</li><li>• Automated deployment of application using AWS Amplify and GitHub</li><li>• <b>Tech stack:</b> React JS, Google-Firebase, AWS</li></ul>
<b>Rentify - A House Rental and Management System</b> <ul style="list-style-type: none"><li>• Built an end-to-end application with three of my teammates for both tenants and landlords to manage all their properties/rentals at one place.</li><li>• Used Stripe for processing the payments</li><li>• <b>Tech stack:</b> MERN stack, Redux</li></ul>
<b>Library API using Gin framework</b> <ul style="list-style-type: none"><li>• Architected an API to add books, checkout, return and view the books that are currently in the library</li><li>• Used Gin web framework to create the API</li><li>• <b>Tech stack:</b> Golang, Gin framework</li></ul>