# Aryaman P. Singh

San Diego, CA
aps010@ucsd.edu | *aryamanpsingh.github.io* | 213-420-9792

## EDUCATION

UC San Diego
**DEGREE**: B.S **MAJOR**: Applied Math

## RELEVANT COURSEWORK

- Data Structures & Object Oriented Design
- Math for Algorithms & Systems
- Combinatorics, Statistics, Data Analysis w/ R

## SKILLS

- **Programming –** Python, Java, C++
- **Math –** R, Matlab, Pandas
- **Web –** JavaScript, HTML, CSS
- **Cloud –** AWS, GCP, Kubernetes, Docker
- **Frontend –** ReactJS, Redux
- **Backend –** Django, Flask, PostgreSQL

## WORK EXPERIENCE

**Hadoolytics** **April 2021 - Present**
Software Engineer Intern

- Worked with a team that helped businesses migrate their applications from legacy servers to **Kubernetes** clusters installed on **AWS EC2** instances.
- Architected the backend of a policy management tool for software deployed on **Kubernetes** clusters on **GCP**.
  - Set up a **Flask API** through **Google App Engine** for the UI to consume.
  - Installed a server instance of the **Grafeas metadata API** on the cluster which is used to store vulnerability scan reports and attestations.
  - Installed a **Kritis binary authorization webhook** on the cluster, which is configured to query the **Grafeas API** to ensure vulnerability and attestation requirements are satisfied.
  - Set up a **Kubernetes CronJob** for continuous vulnerability scanning of the images pushed to pods which flags bad pods for attention.

## PROJECTS

**Stat90** (143.110.225.43) **2020**
*Visualization tool for soccer statistics.*

- Developed a **React** progressive web-app which uses **Chart.js** to provide an interactive visualization tool for soccer stats.
- Set up a weekly cron job which scrapes data using **Selenium,** stores it in a CSV and populates the **PostgreSQL** database.
- Set up a **Django REST API** to serve the data.

**IronLog** (143.110.146.187) **2020**
*Fitness tracker to create and modify routines.*

- Progressive Web App implemented in **React** using **Redux** for state management and **Knox** for token authentication.
- Created a **Django** backend which utilizes many-to-many model relationships allowing users multiple start-points to their log and reducing database redundancy.

**Reddit.Record** **2019**
*Explore a term's sentiment and popularity on Reddit by subreddit.*

- Built a **Flask**-based web-app which generates graphs using **PyPlot and Bokeh**.
- Implemented **PushShift API** to collect data from Reddit and used **NLTK** to analyze the sentiment of the vectorized data.

**Position Predictor** **2018**
*Predicts position based on skill values.*

- Developed a **Flask** web-app which predicts the user's soccer position based on entered attributes.
- Trained a Classifier model, applied **Principle Component Analysis** reducing a 51 feature dataset to 11 while maintaining 92% accuracy.