

Anjali Kumar

Santa Clara, CA | anjaliswat@gmail.com | (510) 248-9755
anjaliswat.github.io | github.com/anjaliswat | linkedin.com/in/anjali-kumar-44163b139

EDUCATION

Santa Clara University

Master of Science, Computer Science and Engineering

Santa Clara, CA

Sep 2016 - Jun 2018

Amity University

Bachelor of Technology, Biotechnology

Gurgaon, India

Aug 2012 - Jun 2016

TECHNICAL SKILLS

Languages: HTML, CSS, JavaScript, Python, C, Ruby

Frameworks: Angular, React, Sinatra, Ruby on Rails

Databases: MongoDB, PostgreSQL, SQLite

Other: Amazon Web Services, Heroku, Git

EXPERIENCE

Frugal Innovation Hub (Santa Clara University)

Full Stack Developer

Santa Clara, CA

Jul 2017 - Apr 2018

Created a website for an NGO called Collaborate For Africa with a team of 3 people. The website was created using the MEAN stack. The application allows users to add stories, research papers and events they are hosting. The rich text editor TinyMCE has been integrated in the website to allow users to style their content. Authorization has been provided using Auth0. A whitelist has been created that currently gives only members of the organization access to add and delete content on the website.

Website : collaborate4africa.org

ACADEMIC PROJECTS

Weather App : Created a weather app using Google's and Dark Sky's API. In this app users can enter a city or state in the search box. Once they have entered a location, the latitude and longitude for this location are retrieved through geolocation. These values are then passed to Dark Sky Net to obtain the weather forecast for that place.

Online Book Store: Created using Ruby on Rails and PostgreSQL. It allows users to add books to their shopping cart and check them out. With admin credentials, you can log in and add or remove books from the book store.

Inference Engine: Implemented a propositional logic resolution-based inference engine in Python. This inference engine accepts statements in CNF form or otherwise converts the statements to CNF form and answers queries based on the resolution algorithm. It keeps resolving until it is determined whether the query can be satisfied or contradicted.

FreeRTOS Synchronization Mechanisms: In this project, I conducted an evaluation of the performance of Semaphores, Reference Coloring Algorithm and FreeRTOS task notifications. FreeRTOS was run in an Xilinx Zynq SOC. The development was done within Xilinx SDK Development environment. Implemented in C.

Pathfinding Application: The application uses an Artificial Intelligence Search Algorithm (A* Search) to find an optimal path by calculating costs of all possible paths to the end point in a given environment and picking the one with the least cost.

Diagnosing Type of Breast Cancer: Uses a machine learning technique (ID3) to analyze data from the public archive - archive.ics.uci.edu and determines whether the patient has a malignant or benign tumor. Implemented in Python.