Tanya Balaraju

Github tanya
Linkedin /in/tanya-b
Website tanya.github.io
Email tanyabalaraju@gmail.com

Education

Rutgers University > September 2014 - May 2018
B.S. Computer Science
Honors Program and **Phi Beta Kappa >** GPA: **3.77**

Coursework

Undergrad: Algorithms, Databases, Systems Programming, Distributed Systems, Cryptography, Cognitive Science Concepts, Applying Cognitive Science

Grad: Artificial Intelligence, Brain-Inspired Computing, Natural Language Processing, Internet Services

Leadership/Experience

datadoghq.com > Summer 2018 (hackNY Fellowship)

Datadog > Software Engineering Intern

> Modified components in **React** frontend and endpoints in **Python** backend to display aggregate usage summaries

> Created **PostgreSQL** DB migrations and used **SQLAlchemy ORM** for new payment workflow affecting **all users**

> Wrote and ran Python unit tests and feature flags for all new features; implemented A/B testing for new workflow

pedul.com > Summer 2017 (Contract)

PeduL > Backend Web Developer

> Worked with team to design sitemap, UX flows, and ER diagrams for a higher education crowdfunding startup

> Implemented backend flows to support payments, user auth, and email verification in **Python/Flask** using MongoDB

> Deployed on **Apache** HTTP server on **Ubuntu** cloud host; integrated frontend using Jinja2; wrote detailed documentation

cs.rutgers.edu/resources/instructional-lab > Fall 2017

Department of Computer Science > iLab Assistant

> Administrated Rutgers computing resource cluster running CentOS 7 for **thousands** of undergrads, graduates, and faculty

> Tutored undergraduates in computer science coursework

usacs.rutgers.edu > April 2016 - April 2018

Rutgers USACS > Community Director

> Organized events to **reform** and **expand** CS student org

> Led online campaigns to increase exposure by over **100%**

> Launched initiative to increase community presence of minority groups by **200%-500%,** particularly **women in tech**

hackru.org > Fall 2017 - Spring 2018

HackRU > Director, Volunteers / Hacker Experience

> Organized and lead team of **~50 volunteers** for HackRU

> Ensured smooth experience for **hundreds of developers**

> Recruited experienced developers as mentors to attendees during the event, **encouraging project completion**

> Prior **Organizer** on Day-Of and Hacker Experience teams

Skills

Python, Java, C, Swift/iOS, C#, HTML/CSS, JS (React)
Bash, Git, Flask, Apache, MySQL, PostgreSQL, MongoDB
Linux (RHEL, DEB), Docker, Kubernetes, Nomad

Honors/Awards

hackNY Fellowship: Summer 2018
Best Hack Design: hackNY Spring 2016
Best Mobile App: CS336 Databases Fall 2016
Scholarships: Henry Rutgers, Rutgers Trustee
First-Year Leaders Fellow: Rutgers, Spring '15
Dean's List: Fall '14, '16, '17; Spring '15, '16, '17, '18

Projects

github.com/tanya/Hawk > Fall 2016

Hawk: Crowdsourced Safety App

> Applied **MVC** to create iOS app that included data persistence, networking, and location services in **Swift**

> Built **Python/Flask** backend that used SQLAlchemy to interact with **MySQL** database on Amazon Web Services

> Used Alamofire to connect HTTP requests to REST API with user interface to create a smooth user experience

github.com/tanya/YUGE > Spring 2016

Yuge: Tetris / Candy Crush Mobile Game

> Implemented design in Swift, won **Best Hack Design**

> Collaborated with two teammates at hackNY to blend external mechanics into politically-themed iOS game

> Designed an elegant 8-bit style UI using various tools

Spring 2017 - Spring 2018

Graduate-Level Coursework

> Implemented Simulated Annealing and **space-optimized A*** search algo in **Java** for Traveling Salesman Problem

> Compared production-grade container deployment frameworks: Kubernetes, Nomad, Swarm, Mesos/Marathon

> Integrated time-based Spiking Neural Network with Convolutional Neural Network to classify written digits

> Implemented **Natural Language Proc.** in **Python** using bigram-based feature augmentation on TFIDF vectors

github.com/tanya/Multithreaded-Bank > Fall 2015

Multithreaded Bank Server

> Implemented low-level solution to classic multithreaded producer-consumer problem in C on UNIX-based system

> Built client-side scripts equipped with dynamically-sized buffers and debugged server-side thread management

> Utilized mutual exclusion locks and thread coordination