Tanya Balaraju

Software Developer, Student Leader

Education

Rutgers University > Fall 2014 - Spring 2018

B.S. Computer Science, interest in Cognitive Science

Honors Program > GPA: **3.75**

Phi Beta Kappa

Coursework

Undergrad: Algorithms, Databases, Systems Programming, Distributed Systems, Cryptography

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

Leadership/Experience

pedul.com > Summer 2017

PeduL > Backend Web Developer

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

> Implement backend flows supporting payments, user auth, OAuth, and email verification in **Python/Flask** using MongoDB

> Deploy on **Apache** HTTP server on **Ubuntu** Linux cloud host; integrate front end using Jinja2; configure unit and load testing

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

Department of Computer Science > iLab Assistant

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

> Tutor undergraduates in computer science coursework, particularly providing help with Data Structures & Algorithms

usacs.rutgers.edu > April 2015 - Present

Rutgers USACS > Community Director

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

> Established social media campaigns that increased online exposure by over **100%**, from **<500 unique views to 1000+**

> Launched initiative to make Rutgers CS more inclusive of underrepresented groups, particularly **women in tech**

hackru.org > Fall 2017

HackRU > Director, Volunteers / Hacker Experience

> Recruit, guide, and lead team of **~50 volunteers** for HackRU

> Ensure smooth experience for **several hundred developers**

> Contact experienced technologists to serve as mentors to attendees during the event, **encouraging project completion**

> Prior **Organizer** on Day-Of and Hacker Experience teams for three semesters of HackRU, planning and hosting mini-events

sashonors.rutgers.edu > Spring 2015 - Fall 2015

SAS Honors Program > Ambassador & CS Tutor

> Guided students through Data Structures and Intro to CS

> Represented CS major to hundreds of SAS honors freshmen

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

Skills

Python, Java, C, Swift/iOS, C#, HTML/CSS/JS

Bash, Git, Flask, MVC, Apache, MySQL, MongoDB

Linux (RHEL, DEB), Docker, Kubernetes, Nomad

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

Projects

github.com/tanya/Hawk > Fall 2016

Hawk: Crowdsourced Safety App

> Applied MVC to create end-to-end 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 2016 - Present

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