

DHAVAL SONAVARIA

(315) 314-0983

dhavalsonavaria.github.io

dhsonava@syr.edu

Employment

- | | | |
|---------------------------|----------------|------------------------------|
| Software Engineer, Intern | SVP Global LLC | Dec 2019-Present, Summer 19' |
|---------------------------|----------------|------------------------------|
- Developed desktop applications for Accounting, trading and Ops teams using .NET Framework and core.
 - Automated the process of transferring positions between funds and posted to the Geneva Accounting
 - Generated wire transfer instructions fed into State Street and Geneva for operations, accounting teams.
 - Created CI/CD pipelines leveraging IaaS and PaaS for applications and microservices, migrating to cloud.

Education

- | | | |
|--------------|---------------------|---------------------|
| Syracuse, NY | Syracuse University | Aug 2018 - May 2020 |
|--------------|---------------------|---------------------|
- M.S. in Computer and Information Science
- Analysis of Algorithms, Computer Security, Software modelling and Analysis in C#.NET, Object Oriented Design in C++, Structural programming in Haskell, Natural Language Processing, Operating Systems, Internet Security.
- | | | |
|---------------|----------------------|---------------------|
| Mumbai, India | University of Mumbai | Aug 2014 – May 2018 |
|---------------|----------------------|---------------------|
- B.E in Computer Engineering

Projects

- | | | |
|-------------------------------|-----------|-------------------|
| CFS, Virtual Memory in NachOs | Fall 2019 | C++, Shell Script |
|-------------------------------|-----------|-------------------|
- Implemented asynchronous I/O system calls, exception handling using call-back objects and alarm interrupt.
 - Implemented the Linux Completely Fair Scheduler(CFS) using Red-Black trees sorted in order of virtual time.
 - Implemented Virtual Memory as part of a Memory manager that used a Linux file as DISK, a page table and a round-robin scheduler that implemented Demand paging to implement multiprogramming in OS.
- | | | |
|--------------------------------|-----------|-------------------------|
| SEcurity EDucation (SEED) labs | Fall 2019 | C, Python, Shell script |
|--------------------------------|-----------|-------------------------|
- Hands-on experience performing and understanding counter measures of software security attacks
 - Software Security: Buffer Overflow, Shellshock, Format String, TOCTTOU, Dirty COW, Reverse Shell.
 - Web Security: CSRF, XSS, SQL Injection Attacks. Hardware Security: Meltdown, Spectre attacks
 - Network Security: Packet Spoofing, SYN Flooding, TCP Reset, Session Hijacking, DNS Cache poisoning.
- | | | |
|---------------------------------|-------------|-----------------------------|
| Accounting/Operations Dashboard | Summer 2019 | C#.NET, WPF, MVVM, RabbitMQ |
|---------------------------------|-------------|-----------------------------|
- Created applications to select positions and transfer funds between them showing real time positions.
 - Developed application to generate wire transfer instructions that automated the money transfer process.
 - Applications used RabbitMQ for real-time updates, logging and REST API calls following a SOA design.
- | | | |
|---------------------------|-------------|-------------------------------------|
| Remote Source code viewer | Spring 2019 | C++, JavaScript, HTML, WPF, Winsock |
|---------------------------|-------------|-------------------------------------|
- An asynchronous client-server desktop application that converts source code to informative web pages
 - Used a C++ back-end to find dependencies among source code files and parse them to HTML files.
 - Implemented code and comment drop downs and buttons on web pages using JavaScript and CSS.
 - GUI allows remote directory browsing and uses sockets to communicate with the C++ factory interface
- | | | |
|----------------------|-----------|------------------|
| Remote Code Analyzer | Fall 2018 | C#.NET, WPF, WCF |
|----------------------|-----------|------------------|
- Asynchronous client server application that parses source code to find SCC's using Tarjan's algorithms
 - Detected classes, functions, signatures using a stack-based parser to create SCC's between files.
 - Created an engaging GUI in WPF, used WCF to connect client and server via asynchronous blocking queue.
- | | | |
|----------------------------------|-------------|----------------------------|
| Paraphrase Resource using Tweets | Spring 2019 | Python: NLTK, Spacy, PropS |
|----------------------------------|-------------|----------------------------|
- Resource created by extracting binary predicates from news tweets reporting same events.
 - Ranked paraphrase pairs in bins and applied supervised learning to the top bin to achieve 72% accuracy.
- | | | |
|-------------------------------|-------------|-------------------------------|
| Stock Prediction using Tweets | Spring 2020 | Python: NumPy, Pandas, SciKit |
|-------------------------------|-------------|-------------------------------|
- Compared Regressive learning techniques on feature engineered stock tweets to predict future stock value.

Skills and Interests

Languages: C++, C#.NET(Visual Studio), C, Python, Shell script, Winsock, JavaScript(Node.js, Socket.io, TypeScript)
Tools: Azure Cloud, DevOps, Git, SQL Server, Postman, WireShark, nmap, Docker, Heroku, TDD, .NET 5, Power BI