William Wagner

wcw.wagner@gmail.com

EDUCATION

Florida State University **Tallahassee, Florida**
Bachelor of Science in Computer Science Expected May 2018

- **GPA:** 3.9 / 4.0

WORK EXPERIENCE

Computer Architecture and Systems Lab **Tallahassee, Florida**
Undergraduate Research Assistant September 2016 - Present

- Built machine learning models capable of monitoring and predicting global disease trends based off internet traffic logs and official CDC disease reports
    - Forecasted US influenza levels up to two weeks in advance of the CDC, with a maximum average difference of 0.3%
- Utilized Apache Spark to aggregate and explore various big data sets to test their feasibility in detecting epidemiological trends

Bloomberg L.P. **New York, New York**
Software Engineer Intern, Trading & Analytics May 2017 – July 2017

- Developed a Bloomberg Terminal Function and an API back-end that replaced existing product installation toolchains for clients
    - Removed dependency on engineering department for new client installations
    - Achieved 85% decrease in new client install times
- Created a Bloomberg Terminal function that demos a new reporting hub, which provides prospective firms with transaction histories and regulatory audit trails
    - Necessary for new wave of regulations (MiFID II) affecting the EU in 2018

TECHNICAL SKILLS

Programming Languages: Python, JavaScript, C++, SQL
Technologies: HTML/CSS, React, MySQL, Apache Spark, Git, Linux, Flask, AWS, Node, Scikit-Learn

PROJECTS & OPEN SOURCE CONTRIBUTIONS

Vestview **GitHub: wcwagner/vestview**

- Developed a Flask web application that facilitates access to real-time and historical stock market prices, custom news summaries, and graphs
- Implemented a back-end with Python which aggregates and inserts daily stock market prices and news from various APIs into a MySQL database

Crudechain **GitHub: wcwagner/crudechain**

- Created an implementation of a block chain, which continuously maintains a growing list of transactions in a distributed data store
- Utilized a HTTP interface to control block nodes and web sockets to allow for communication between nodes, emulating a peer-to-peer network

Pandas (large Python data analysis library) **GitHub: pandas-dev/pandas**

- Wrote various bug fixes, along with unit tests to ensure correctness and compatibility in the library
- Added documentation to further explain Panda's handling of mixed data types