https://github.com/ttran9 Work Authorization: U.S. Citizen
wptran58@gmail.com Todd Tran (510) 766-0650

Skills
Programming/Web languages: Java, C++, Objective-C, HTML, CSS, JavaScript.
Databases: PostgreSQL.
Platforms: Windows 7, Ubuntu 16.04.1 LTS, OS X Yosemite.
Other: React, Redux, iOS 8.4, Xcode 6, Android Studio, Visual Studio Professional 2008, Eclipse Mars, Spring, Thymeleaf, JUnit.

Experience

Software Engineer Intern Synactive, Inc. March 2015 - July 2015

- Collaborated with the Development team to implement iOS Bluetooth printing feature for customers (Objective-C).
  - Modified existing solution working for Zebra Bluetooth printers to create a generic solution applicable to all Bluetooth printers.
    - Modified existing code dependent on the Zebra SDK to use iOS SDKs.
    - Created code to handle memory leak issues.
    - Wrote code to display user menus to select Bluetooth printer.
    - Wrote code to save a default Bluetooth printer.
    - Wrote code to display informative error messages.
  - Assisted members of the Professional Services team to create documentation for customers.
- Worked independently to fix customer reported bugs (Objective-C and C++).
  - Used Visual Studio and Xcode to refactor code and add requested functionality.

Education
California State University, East Bay: Master's Degree, Computer Science, Fall 2014
University of California, Davis: Bachelor's Degree, Biological Sciences

---

Projects: (source code is uploaded to GitHub)
Personal Project Management Tool (https://github.com/ttran9/ppmtool-backend): A project that I followed along on an online Udemy course and which I have added a few new features to and a bug fix.

- Fixed a bug where the user cannot type a URL in and had to navigate via on-screen items.
- Adding in new features such as requiring a user to activate a account through e-mail and changing password through email both requiring a valid token generated from the back-end.
- Used React and Redux to create the front-end portion of application.
- Used Spring Boot and Spring Security to create the back-end API.
- Used JWT authentication to protect secure endpoints such as when making calls to the back-end API.

TT-WeatherForecast-Spring (https://github.com/ttran9/tt-forecast-spring): For more details please view the README which is provided on the repository.

- Used a PostgreSQL database with an ORM (Hibernate) to store searches and forecasts.
- Used JUnit to test services which made API requests to Google Geocoding and Darksky APIs.
- Used JUnit to test controllers which handle GET requests which display the forecasts.
- Used CanvasJS and Thymeleaf to create views to display the forecasts as column graphs.

ARImplementation (https://github.com/ttran9/AllRecipesJavaSpring): An independent project allowing me to develop with a variety of Java Technologies such as Spring MVC, Spring Security, and JUnit.

- Created a relational database in PostgreSQL to store User information, Recipes, Shopping, and Pantry Lists.
- Wrote JUnit tests to test functionality.
- Used Spring Security to prevent unauthorized access to protected web pages.
- Used Spring MVC annotations to provide URL mappings to allow users to make web requests and have HTML pages returned.