Eliseo Nathaniel Ruiz Nowell

Computer Engineering | University of Waterloo

Contact: 778-883-5778 | enruizno@uwaterloo.com

LINKS

GITHUB:
https://github.com/NathanielRN

LINKEDIN:
https://www.linkedin.com/in/nathanielruiz

PORTFOLIO:
http://www.eliseoruiz.life

SKILLS

PROGRAMMING
Swift • JavaScript • Java • Python • SQL • C++ • Objective-C

TECHNOLOGIES
git • Xcode • node.js • vim • MySQL • YAML • Memcached • mongoDB • Android Studio • CocoaPods • HTML • CSS • Subversion • Splunk

FRAMEWORKS
iOS • AWS • Mocha • Android

LANGUAGES
Spanish (Fluent) • French (Fluent)

TECHNICAL
Level 9 RCM Piano Practical Exam with Honours • RCM Advanced/Intermediate Rudiments Theory

INTERESTS

Model United Nations • Half Marathon Runner • Duke of Edinburgh Silver Award

VOLUNTEER

President of Graduation Committee • Free The Children Group

EDUCATION

UNIVERSITY OF WATERLOO
B.A.Sc. in Computer Engineering
Expected June 2021

BURNSVIEW SECONDARY
Double Dogwood Diploma
Received June 2016

EXPERIENCE

YAHOO! | BACKEND DEVELOPER INTERN   September '17 – December '17
node.js, MySQL, express, AWS, Python, YAML, Mocha, Memcached
› Developed node.js CRUD handlers for MySQL database modification and retrieval
› Created Python file I/O methods to populate .tsv files with series and episode items
› Used AWS CloudSearch to facet and filterQuery buckets, used Elastic BeanStalk together with SQS to queue automatic async server requests on SSL enabled HTTPS
› Finished episode stream SQL fetch tool with retry, query parameters for genre/offset
› Introduced validation of assets with YAML swagger files and Mocha system/unit tests

RAVE INC. | iOS DEVELOPER INTERN   January '17 – April '17
Swift, git, Xcode, vim, Objective-C, CocoaPods
› Worked with Swift database and Objective-C libraries to fix bugs, fix crashes, revamp the UI and optimize performance by considering multi-threading/async requests
› Mastered Xcode IDE, iOS SDK, and 3rd party library integration (SDWebImage)
› Communicated with coworkers to localize the app into 4 different languages
› Polished git skills to solve conflicts with vim, ease rebasing, and cherry-pick commits
› Refactored user flow across all screens as part of collaboration with UX designer

PORTAL HACKATHON | 1st PLACE - WEB FEATURE   October '17
HTML, JavaScript, https://uwaterloo.ca/student-success/blog/post/hacking-portal-edition
› Collaborated on a web-based exam question database application in 9 hours for UW Portal website using JavaScript and HTML
› Implemented data retrieval, random question generator, and answer hiding

PROJECTS

COLOR CONQUEST | SELF-PUBLISHED iOS APP   October '17 - present
Swift, git, CocoaPods, AdMobs, https://itunes.apple.com/app/id1316312994
› Designed Bluetooth connected state-machine for communication across devices
› Researched docs to add tables/collections populated by dynamically changing data
› Respected Apple's MVC by implementing delegates and protocols as part of POP
› Considered game end algorithm performance to optimize slower devices response

PRAY4MEBACKEND | SERVER SIDE FOR OWN iOS APP   April '17 - June '17
node.js, express, mongoDB, git, https://github.com/NathanielRN/Pray4MeBackend
› Functional server client on heroku to handle CRUD requests using express module
› Included a driver to store by logged in FB id to the incorporated mongoDB database
› Integrated with authorization tokens to make JSON data request from bibles.org to allow verse lookup and with a separate driver to support image uploading

PONG | 2 – PLAYER PADDLE GAME   May '16
Python, http://www.codeskulptor.org/#user43_pBHSjwEezlFhJeF.py
› Implemented Coursera course game with score-keeping, ball physics, and custom UI

WAT 2048 | UW ANDROID APP   May '16 - August '16
Java, Android Studio
› Processed data received from accelerometer/rotational sensors and ensured appropriate event handlers were called using the Android Studio IDE
› Replicating the popular mobile game using OOP Java design techniques
› Implemented team design using Subversion

STRING LIBRARY | ASCII BASED PROJECT   September '16
C++
› Constructed from primitive data types using OOP, MinGW compiler, and GDB