Ryan Scott

Phone: (469)-585-2996 US / 080-6076-3580 Japan e-mail: ryansgot@gmail.com

Described as a “legit software engineer” by my boss at Bypass Mobile, I leverage a background in mathematics and computer science to create interesting, innovative, and valuable software. Never content with my current level of education, I continue to improve by pursuing education inside and outside the university setting. With software as my main creative outlet, I also pursue personal projects such as a custom analytics logger project (see my github) as well as a few useful gradle plugins.

Work Experience

Swiftly Systems – Android Senior Technical Lead (10/2019 - 03/2023)

- Architect and develop Swiftly’s Client platform (Android, iOS, web)
- Transition the following at a platform and application level
    - Android Views -> Jetpack Compose
    - RxJava -> kotlin coroutines
    - Retrofit -> ktor client
- Produce multiplatform GraphQL framework for use in platform code on web, iOS, and Android
- Lead developer for android applications produced by Swiftly: Zion Market Rewards, PrestoQ, Family Dollar Store App, Save Mart App, Lucky Supermarkets App, FoodMaxx, and Ninety Nine Cents Only App
- Oversee growth from 20-25K monthly users to ~2M monthly users.
- Implement Continuous Integration/Deployment automation via Azure Devops Pipelines (to the Google Play store) for the various white label applications as well as internal deployment using Microsoft App Center
- Develop and execute strategy for white-label applications which specialize a core application and assembly of feature modules
- Develop pleasing UI/UX using Android tooling (androidx.navigation and other jetpack libraries) following material design guidelines
- Follow WCAG 2.1 Accessibility
- Combination of kotlin test, JUnit5 JVM tests and Android instrumentation tests (JUnit4) which consolidate to a single test coverage report
- Work closely with product management to develop strategy and task breakdown for achieving deliverables
- Manage Android development team including task organization, continuous improvement and internal, biweekly Android enrichment presentations

- Create internal library structure (azure dev ops feed) and breakdown for sharing code between many different application and library modules.
- Create custom gradle plugin to do automated versioning and deployment of library and application modules to both internal and external destinations that is applied across products.
- Introduce Kotlin Multiplatform for cross platform development and code sharing

Ookla – Senior Android Developer (10/2017 - 9/2019)

- Develop the Speedtest Android app--more than 35 Million active installs and 21-23 Million unique users per month
- Accelerated efforts to release full UI redesign of the Android Speedtest app before Mobile World Congress in February, 2018 and was successful.
- As Speedtest is an old app with a legacy of success, I am gradually transitioning it from legacy, mostly procedurally-oriented software design and old tooling to more modern and relevant application programming designs and tools, such as
    - MVP design
    - RxJava2
    - Retrofit2
    - AutoValue + AutoValueGson
    - Kotlin
    - kotlinx.serialization
    - JUnit5 (from JUnit4 and JUnit3)
    - Espresso
- Gradually update the Speedtest app with features like
    - Background data collection sensor data and reports
    - Graphs
    - Single Connection speed tests
    - Background reporting improvements
- Monitor and maintain release using telemetry tools like
    - Google Analytics (Firebase Analytics)
    - Fabric (Firebase Crashlytics)
    - Google Play Console
- Ship app to Google Play, Samsung, Amazon, and Yandex app stores (20+ million monthly users)
- UI Architecture for Speedtest Live Feature
- 5G identification on Android P (not officially supported on Android P) -- Samsung/Verizon, Samsung/Sprint, Samsung/T-Mobile, Samsung/AT&T, LG/Sprint
- Lead transition from Java to Kotlin including leading team learning sessions

Amazon – Android Developer (11/2016 – 10/2017)

- Amazon Flex developer—specifically regarding performance, stability, testing, and architectural improvements

Bypass Mobile – Android Developer (11/2014 – 10/2016)

- Enterprise Android application development in a multi-apk environment
- Lead development efforts for creating integration libraries and application packages for external point of sale hardware integration using network, bluetooth, and USB for communications
- Developed Mobile Device Management system
- Lead transition to service-oriented architecture on Android from monolithic application design

Barclays – Java Developer (09/2013 – 11/2014)

- Java application on Linux platform for client connectivity to futures and options exchanges
- Incorporated TeamCity into existing proprietary systems for continuous integration support
- Decommissioned legacy system connecting order management system to post-trade monitoring system and replaced it with a new solution.
- Revived undocumented, legacy Perl testing framework by importing into new Java framework leveraging TestNG.

Argo Data Resource Corporation – Dev Team Lead (11/2012 – 06/2013)

- Design software based upon requirements, working with product management to ensure accuracy
- Create test automation framework in proprietary language in order to promote test-driven development within my team
- Oversee and implement corporate software development life cycle and best practices
- Create team collaboration environment using tools such as Confluence and Jira
- Manage source using Subversion

Argo Data Resource Corporation – Programmer I (06/2012 – 11/2012)

- Used horseshoe model for re-engineering to recover design and requirements from legacy analytics software and perform corrective, preventative, perfective, and adaptive maintenance.
- Recovered knowledge of two C DLLs that had been lost at the company and updated them.
- Understand product requirements and business value and create test harness to prove their satisfaction

Cisco Systems – Test Engineering Intern (09/2009 – 05/2012)

- Built test automation and network configuration automation for scalable, shared test lab support on Linux platform using and Bash scripting, Tcl Expect, and Java

Technical Skills

Languages/Frameworks	Kotlin, Kotlin Symbol Processors, TypeScript, Express.js, Node.js, Sequelize, UI Testing, Unit Testing, Jetpack Compose, Kotlin Coroutines, Android Jetpack, Java, Java8, Kotlin Test, JUnit, Gradle, Bash, Java Annotation Processing, Kotlin AnnotationProcessing, Dagger2, Koin, RxJava, Retrofit, OkHttp3, AutoValue, Gson, AutoValueGson, Moshi, Moshi Codegen, Android, Espresso, Mockito, MockK, State Machine Compiler, SQLDelight, kotlinx.serialization
Software/Hardware	Azure Devops, Azure Devops Pipelines, git, GitHub, Google Firebase, Airship, Microsoft AppCenter, CircleCI, Jenkins, TeamCity, Linux, Confluence, Jira, Buildkite
Related Coursework at UT Dallas	Undergraduate ● Operating Systems ● Data Structures and Algorithm Analysis Graduate ● Computer Networks (CS 5390) ● Software Engineering (CS 5354) ● Software Project Planning and Management (CS 6388) ● Software Maintenance, Evolution, and Re-engineering (CS 6356) ● Database Design (CS 6360)

Education

MS in Software Engineering (Unfinished)

University of Texas, Dallas, TX (2009 - 2012)

B.A. Mathematics (May 2008)

Texas A&M University, College Station, TX

Certifications

- CCENT (Cisco 2010-2013)
- Computing for Data Analysis (Johns Hopkins via Coursera)

- Algorithms: Design and Analysis (Stanford via Coursera)
- Creative Programming for Digital Media & Mobile Apps (University of London via Coursera)
- Introduction to Systematic Program Design – Part 1 (University of British Columbia via Coursera)
- An Introduction to Interactive Programming in Python (Rice University via Coursera)
- Creative, Serious and Playful Science of Android Apps (University of Illinois at Urbana-Champagne via Coursera)
- Programming Mobile Applications for Handheld Systems Part 1 and Part 2 (University of Maryland, via
- Coursera)
- Programming Mobile Services for Android Handheld Systems: Concurrency and Communication
- (Vanderbilt University, via Coursera)
- Programming Cloud Services for Android Handheld Systems: Part 1 and Security (Vanderbilt University via Coursera)
- An Introduction to Programming in Python Part 1 and Part 2 (Rice University via Coursera)
- Web Application Architectures (University of New Mexico via Coursera)
- Ruby on Rails, an Introduction (Johns Hopkins University via Coursera)
- Rails with Active Record and Action Pack (Johns Hopkins University via Coursera)
- JLPT N3 Japanese Language Certification (The Japan Foundation)

Projects

Logging (currently maintained)

- Logging abstraction and plugins leveraging Kotlin Multiplatform to work in web, JVM, Android, and iOS.
- On JVM and Android, uses Java Service Provider Interface to enable a plugin-style logging facade for logging analytics and metrics to multiple destinations without extra effort or platform dependencies
- See the source code here: https://github.com/ryansgot/logging
- Contains Plugins for Google Firebase Analytics, Microsoft App Center Analytics, Airship, and NewRelic
- See MavenCentral for a list of each artifact

ForSureDB (not currently maintained)

- Java Annotation Processor and code generator that generates java code for database access and migration, compatible on Android (Java7/Android API 16+), Java8, and Kotlin. See the http://forsuredb.org/ documentation website

- Available publicly on bintray/jcenter and used by both Bypass Mobile and one non-profit project
- Automated builds via CircleCI create jacoco test coverage reports and uses codecov and findbugs to surface potential issues
- Gradle plugin for assist the above annotation processor https://github.com/ryansgot/forsuredbplugin

CircleCI Viewer (not currently maintained)

- Android app for viewing and interacting with continuous integration service CircleCI available on Google Play
- An app I wrote in my spare time to keep current on the latest tools and trends such as
    - auto value (and auto value gson)
    - Dagger2
    - Rxjava2
    - Retrofit2
    - MVP design
    - Espresso testing
    - kotlin

Several Gradle Plugins (not currently maintained)

- https://github.com/ryansgot/smc-gradle-plugin – generate state machine code at build time given the SMC DSL (to avoid having to store generated state machine code in source control)
- https://github.com/ryansgot/android-java-coverage-merger – provides gradle tasks that combines local jvm JaCoCo report with the JaCoCo report generated by instrumented Android tests

Bypass Mobile

Main Engineer for

- Scanner library: USB-serial and Bluetooth barcode/QR code scanners such as Unitech and Symbol/Motorola
- Printers application: USB, Bluetooth, and network thermal restaurant printers such as StarIO, Epson, and Cognitive TPG
    - As far as I know, I'm the only person to write an Android/Java driver for CognitiveTPG A799.
- Cash drawer library: USB and printer-connected cash drawers
- Remote Display application for interacting with customer-facing displays and payment systems like Clover mini and Ingenico IPP 320 RBA and PosPad devices
- Library for separating hardware device configuration from its implementation
- Network library that leverages retrofit and unifies the different approaches previously used for network communications into a single, reliable interface

- Card Swiper application for interaction with USB, Bluetooth, and Audio credit card swipers
- Mobile Device Management (application for receiving/installing work packages to perform, such as installing/upgrading our software) leveraging Samsung Knox Standard Manager.

Argo Data Resource Corporation

- Cash Inventory Optimization
  - Lead developer responsible for maintenance and evolution of Forecast engine for forecasting cash usage for bank entities and recommendation engine for order amounts (in C, lua, and two proprietary languages)
- Workforce Management
  - Lead developer responsible for maintenance and evolution of Forecast engine for forecasting transaction volume at bank entities and recommendation engine for teller staffing (in C, lua, and two proprietary languages)