Thuytien “Thuy” Nguyen

(650) 376-0431

thuynguyen88112@gmail.com

|  |
| --- |
| Summary |
| * Android developer with 6 years’ experience in standard mobile industry design patterns * 5+ years’ experience in Java, and Object-Oriented Programming * Strong Android development experience with an appreciation of design, testing and backend Web Services * Excellent knowledge of building scalable code using dependency injection * Experience in optimizing performance and memory usage of Android Apps * Experience in writing custom UI components and libraries * Experience with multi-threading, integration with REST and SOAP APIs, custom view animations and transitions, and producing reusable libraries * Passion for Test Driven Development and Behavior Driven Development * Familiar with Wi-Fi, Bluetooth, NFC and other short distance wireless communications protocols * Used Android NDK to call native C/C++ code in legacy code |
| Skillset |

## 

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Framework | | Efficiency Tools | Testing, Analytics, CI/CD | | Project Management |
| Kotlin  Java  REST  RxJava | GSON  Android P  MVP  MVVM | Android Studio  Eclipse  Android NDK  Mockito  Sketch | JUnit  Google Analytics  Crashlytics  Espresso  Jenkins | Hugo  Circle-CI  Mixpanel  Robotium | Git/GitHub/GitLab  Scrum/Agile  Jira/Confluence |

## APIs, Libraries, and classes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BrainTree SDK  ExoPlayer  Parceler  Dagger2 | OkHTTP  Retrofit  IntentServices  Doze Mode | VideoView  Loaders  LeakCanary  JetPack | Volley  JobScheduler  Google Cloud Messaging  EventBus | Timber  SpongyCastle  Calligraphy  AndroidAnnotations | MPAndroidChart  Otto  Urban Airship |

## Concepts and implementation patterns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Singleton  Services  Adapter | Material Design  IoT  Data Class | Handlers  Push Notifications  Bluetooth/BLE | Extension Function  AES/Cipher | Factory  Builder  Dependency Injection | Chromecast  Material Dialogs |

## 

|  |
| --- |
| Realtor.com Santa Clara, CASr. Android Developer Sept 2019 - present [🔗](https://play.google.com/store/apps/details?id=com.move.realtor) <https://play.google.com/store/apps/details?id=com.move.realtor>   * Migrated Java code and also wrote new classes in Kotlin. * Used Atlassian suite, Bitbucket for source control, Confluence for documentation, and Jira for team collaboration. * Collaborated with three other Mobile Software Engineers. * Reviewed other developers’ code by the use of pull request in Bitbucket. * Implemented Dependency Injection, Factory Pattern, Interfaces to make testing easier, and the (MVVM) Model-View-ViewModel architecture Pattern. * Removed deprecated activities and XML layouts from the User Onboarding Flow. * Supported backup location using Google Play Services Location API as an alternative to native Android Location Manager. * Fixed bugs as assigned in Jira that were raised by the QA team. * Researched and proposed using Dagger2 for creating a dependency graph. * Prototyped new simple Android Studio projects to test out different types of proposed UI designs. * Replaced findViewById’s with Data Binding to reduce boilerplate code. * Wrote Java mode JSON Parsers for the API URL call data. * Customized Settings View UI using a Preference Activity by adding a custom layout for the provided List View. * Researched and proposed new Android Architecture Components including Live Data, View Model and Room Persistence Libraries. * Used Alert Dialogs, Loopers and Handlers to flag error cases. * Used RxJava and RxKotlin to make asynchronous network calls. * Updated Android Mobile App UI for the Terms and Conditions, Fall Detection Acknowledgement screens, and Device Card View. * Replaced Circular Progress Bar in Firmware screen with Horizontal Progress Bar. * Unit tested Android Java code using JUnit and Mockito. * Used fragments for modular User Interface and Navigation Drawer to create a consistent user experience throughout the application. * Used Firebase Ads SDK to present assets using the RecyclerView providing a seamless user experience.  Expensify San Francisco, CASr. Android Developer  Sept 2018 - Sept 2019 [🔗](https://play.google.com/store/apps/details?id=org.me.mobiexpensifyg) <https://play.google.com/store/apps/details?id=org.me.mobiexpensifyg>   * Worked with the JDK (Collections, Generics, Networking, Core Java, Concurrency, memory management). * Used Shared Preferences to store local user account information. * Experienced working on a significant consumer app that is currently available in the marketplace. * Migrated code from Java to Kotlin. * Thrived working in a fast-paced, agile environment while constantly learning. * Refactored code to change a previous MVC implementation to an MVP design pattern. * Worked in an Agile development environment with peer engineers on a daily basis to scope, develop and deliver a software experience that realizes the product strategy. * Added selectors and animation effects on UI/UX elements to enhance the User Experience. * Created an entirely reusable and customizable Dialogs for various purposes. * Work on Adjust SDK for mobile app marketers, combining attribution for advertising sources with advanced analytics and store statistics * Retrofitted four legacy view combinations to a custom view. * Partner with cross-functional teams to define new product requirements, features, and functionality. * Implemented and maintained the Dagger 1 application’s Object Graph. * Setup proper interactions of the front-end design and implementation with backend servers. * Used Volley for RESTful service calls to create asynchronous communication and multi-threading in the background to take the load off UI thread. * Used RxJava to perform asynchronous tasks. * Used SQLite and Shared Preferences for data persistence. * Added value encryption in SharedPreferences using AES 128 with the Secure-preferences Library. * Applied Crashlytics for bug and app crashes reporting. * Interfaced with multiple web services SOAP and RESTful from data providers. * Cashed last seen content in the database and SD card for offline access. * Implement EventBus to manage most of the app event process interfacing. * Prototyped two candidate activities to facilitate decision making regarding features and flow to be implemented in the new module. * Worked in the new design, gave a new look and feel in the app, applying styles and themes. * Used the JIRA bug tracking tool for quality control. * Utilized a Tabs and Navigation Drawers to switch quickly between application Activities and Fragments. * Introduced Push Notifications to keep the user engaged with Parse. * Fixed Bugs to improve navigation speed.  Pregnancy App & Baby Bump Track San Francisco, CAMobile App Developer Aug 2017 - Sept 2018 [🔗](https://play.google.com/store/apps/details?id=com.glow.android.nurture) <https://play.google.com/store/apps/details?id=com.glow.android.nurture>   * Send up Reminders for due date, appointments, medication and send notifications using MixPanel library. * Use a PDFReader library to email PDF reports of due date, pregnancy symptoms, weight, medication. * Support troubleshooting and debug applications, working with the testing team to optimize application performance. * Engage with all relevant team members to ensure the required quality is delivered from the BDD processes, meaning the acceptance criteria being developed against are also those being automated in the functional test packs. * Implement custom native user interfaces using the latest Android programming techniques. * Replace bitmaps implementation code with Glide image loading library for animated GIF support and handle image loading/caching. * Implement Loaders and AsyncTaskLoaders to perform asynchronous data loading in the custom list view for grocery lists and recipe scaling. * Work on memory management, push notifications, multiple screen support, data offline usage, content providers and sync. * Call native C/C++ code used in legacy app using Ndk. * Worked on algorithm to create pregnancy tracker and display in custom view. * Worked with Google GData client library to access Google APIs using JSON and OAuth 2.0. * Worked with our Solution Architects to ensure the solution delivers the required quality and adheres to best practice. * Coordinate with testers to ensure they are engaged and ready to test the software at the right time in the development cycle.  NEXTDOOR.COM San Francisco, CAMid-Level Android Developer Feb 2016 - Aug 2017 [🔗](https://emojipedia.org/emoji/%F0%9F%94%97/) <https://play.google.com/store/apps/details?id=com.nextdoor>   * Wrote algorithms in MVP architecture for real-time updates, local data access for offline scenarios, and data synchronization with customizable conflict resolution when devices are back online. * Used GATT to send data to other devices over a BLE link. * Independently worked on ‘Find local events like cookouts and community activities’ feature using Google Maps SDK, Places API and location services. * Created custom and compound views for ‘Share neighborhood crime reports’ UI screen with material design principles. * Liase with external client based in London to implement Krux Mobile Data Activation Platform to capture people data from all mobile platforms. * Included Urban Airship sdk forIn-App Purchase and Subscriptions, configure push notifications to know about great deals and latest coupons. * Tested and verified Audience Studio mobile app SDK implementation using Charles Proxy, MockServer API and Espresso. * Wrote Network calls using Reactive Streams API to govern the exchange of stream data across an asynchronous boundary with transformation, splitting, merging. * Performed Branch SDK integration for deep linking, user routing to increase mobile conversion, retention, and engagement. * Made secure HTTP requests to GraphQL and REST endpoints to access, manipulate, and combine data from one or more data sources with Amazon DynamoDB. * Used RxJava to make network calls and run asynchronous background tasks. * Customized the background Executor used to run Workers, and the WorkerFactory used to construct Workers for multithreading implementation. * Included ImageViewTouch ImageView library, Jose Time library, Glide image loading and FaceBook Audience Network library to support multiple features. * Managed user content securely in the cloud or on the device RSA and Java cryptography API. * Implemented DIP using Dagger library by creating and managing the graph of decouple dependencies using custom scopes, custom components and modules. * Implemented In-App update feature using Android Play Core library with immediate flow option which requires users to download and install an update before continuing the app.  Sonic Inc Services LTD Oklahoma, OKNative Android Developer Jan 2015 - Feb 2016 [🔗](https://play.google.com/store/apps/details?id=com.sonic.sonicdrivein) <https://play.google.com/store/apps/details?id=com.sonic.sonicdrivein>   * Added maps-based data on Google Maps to find the closest SONIC Drive-In locations in user area and see their hours. * Used GATT to send data to other devices over a BLE link. * Included Splunk MINT to collect crash, track all HTTP and HTTPS calls, monitor fail rate trends and send it to Cloud server. * Coded network module using Volley library to mediate the stream of data between different API Components, supported request prioritization and multiple concurrent network connections. * Used okhttp, Retrofit, Realm database library to implement on-device data store with built-in synchronization to backend data store feature. * Worked with lead to integrate Kochava SDK for mobile install attribution and analytics for connected devices. * Generated a custom behavior in multiple screens included in the CoordinatorLayout to hide the Toolbar and the Floating Action Button on the user scroll. * Used Firebase Authentication for user logon and SQL Cipher to encrypt transactional areas. * Used Paging library to load information on demand from data source. * Created unit test cases and mock object to verify that the specified conditions are met and capture arguments of method calls using Mockito framework. * Implemented authentication support with the remote server using a Bound Service and an authenticator component, oAuth library. * Coded in Clean Code Architecture on domain and presentation layer in MVP and apply builder, factory, façade, design patterns to make code loosely coupled in layer communication (Dependency principle) * Integrated PayPal SDK and card.io to view billing history and upcoming payment schedule in custom view. * Included Google Guice dependency injection library for to inject presenters in views, make code easier to change, unit test and reuse in other contexts. * Used Bolts framework to perform branching, parallelism, and complex error handling, without the spaghetti code of having many named callbacks.  Square, Inc. San Francisco, CA *Software Engineering Intern May 2014 – Dec 2015*   * Built Ruby on Rails web application used to collect and store information on Square’s new hires * Automated on-boarding notifications via email, SMS, and other instant messaging platforms * Developed a Rails app to store and organize of all the NDA’s signed at Square by guests and employees * Utilized the Twitter API to maintain privacy on Twitter accounts owned by Square |
| Education |

## 

## Cal Poly, San Luis Obispo, California | Bachelor of Science

Computer Science