Tamerlan Zharlykassym

Almaty, Qazaqstan
Email: tamerlan.zharlykassym@gmail.com
Github: @zharlykassym
LinkedIn: in/zharlykassym
Website: zharlykassym.github.io
Telegram: @zharlykassym
Mobile: +7 (708) 521 75 51

EDUCATION

- **Astana IT University** — Astana, Qazaqstan
  Bachelor of Science in ICT - Cybersecurity; GPA: 3.25 — *September 2020 - July 2023*
  Diploma project: *Development of Spyware and Detailed analysis method. Grade: A+*
  Courses: *Java Advanced, Algorithms and Data Structures, Databases Management Systems, CISCO Security, Reverse Engineering*

- **Nazarbayev Intellectual school** — Almaty, Qazaqstan
  Physical-Mathematical direction — *September 2013 - May 2018*

TECHNOLOGY STACK

Kotlin, Java, SQL, Android SDK, Dagger2/Hilt, Coroutines, Flow, Retrofit, RxJava, Glide, Jetpack (LiveData, ViewModel, Navigation), Git, GitHub, RESTful API, JSON, Room, PostgreSQL, SQLite, Firebase Realtime Database, Service, Broadcast Receiver, Content Provider, JUnit, Gradle.

PET-PROJECTS

- **EffectiveTravels (Kotlin)**: Technical assignment from Effective Mobile. Ticket search application. Designed a multi-module architecture following Clean Architecture principles. Configured Gradle dependencies based on the principle of centralizing dependencies in buildSrc. Set up Dependency Injection using Dagger2. Configured Retrofit, ApiService, defined DTOs, Mappers, UseCases, and Entities. Implemented ViewModelFactory. Currently working on ViewModel.

  `Multi-module app` `Dagger2` `Retrofit` `Clean Architecture`

- **Crypto App (Kotlin)**: Built a cryptocurrency tracking app featuring real-time price updates, historical data visualization, and an intuitive user interface. Followed Clean Architecture principles and implemented dependency injection with Dagger2. Utilized Room for local database management, Coroutines for asynchronous operations, and Retrofit for seamless API integration. Enhanced UI responsiveness with Fragment-based landscape mode, image loading via Picasso, and optimized list updates using RecyclerView and DiffUtil.

  `Clean Architecture` `Dagger2` `Coroutines` `Retrofit` `Room` `RecyclerView` `DiffUtil` `Picasso`

- **Shopping List (Kotlin)**: Developed a shopping list app with task addition, editing, and swipe-to-delete features. Implemented long-press functionality for task completion and empty field warnings. Utilized MVVM architecture, Fragments, Room, Coroutines, and Data Binding for efficient UI management. Enhanced user experience with animations and optimized list updates using RecyclerView and DiffUtil.

  `Dagger2` `Clean architect` `MVVM` `Fragments` `Room` `Coroutines` `Data Binding` `RecyclerView` `DiffUtil`

- **Number Composition Game (Kotlin)**: Developed an educational puzzle game focused on number addition. Implemented game logic using Fragments for a dynamic and responsive UI. Utilized Data Binding to streamline UI updates and Jetpack Navigation for seamless screen transitions.

  `Fragments` `Data Binding` `Jetpack Navigation`

- **RealTimeMessenger (Java)**: Developed a real-time messaging app with Firebase integration. Implemented user authentication using Firebase Authentication. Managed real-time data storage and synchronization via Firebase Realtime Database.

  `Firebase Authentication` `Firebase Realtime Database`

- **API Movie (Java)**: Developed a movie catalog app using Kinopoisk API with MVVM architecture. Implemented Retrofit, Glide, and RoomDB for data handling and offline storage. Applied SOLID principles, clean code, and design patterns (Singleton, Factory Method). Integrated RxJava for background processing and implicit intents for trailer playback.

  `MVVM` `Retrofit` `Glide` `RxJava` `Room` `Singleton` `Factory Method`

CERTIFICATES

- Android development 100+ hours (5) courses, Stepik - 2023
- Samsung Innovation Campus, Mobile Development, Java&Android - June, 2023
- Certificate of Completion CCNA Security course Cisco, CCNA, Security - May, 2022
- Stanford International Youth Program at Stanford University, Stanford University, Campus - August, 2015

SKILLS

- **Programming Languages**: Kotlin, Java, Python, SQL, Assembly
- **Mobile Development**: Android Studio, Android SDK, Retrofit, RxJava, Coroutines, Room, Glide, Jetpack (LiveData, ViewModel, Navigation), ConstraintLayout, Activity, Fragment, Service, Broadcast Receiver, Content Provider, RecyclerView
- **Software Engineering**: Git, GitHub, Gradle, Firebase Realtime Database, RESTful API, JSON, JUnit, PostgreSQL, SQLite
- **Additional**: Kali Linux, BurpSuite, SQL Injection, OllyDBG