

Native Android App Tech Stack

Programming Languages:

- Kotlin
- Java

Development Environment:

- Android Studio

Build System:

- Gradle

Libraries and Frameworks:

- Android Jetpack (LiveData, ViewModel, Room, Navigation, WorkManager)
- Dagger/Hilt (Dependency Injection)
- Retrofit (Networking)
- Glide/Picasso (Image Loading)
- Coroutines/RxJava (Asynchronous Programming)
- Material Components for Android

Database:

- Room
- SQLite

Networking:

- Retrofit
- OkHttp
- Volley

UI/UX Design:

- XML
- Jetpack Compose
- Material Design

Version Control:

- Git
- GitHub/GitLab/Bitbucket

CI/CD:

- GitHub Actions

Analytics and Crash Reporting:

- Firebase Analytics
- Crashlytics
- Google Analytics

Push Notifications:

- Firebase Cloud Messaging (FCM)

App Distribution:

- Google Play Console

APIs and Services:

- Google Play Services
- Firebase

Other Tools:

- ProGuard/R8 (Code Shrinking and Obfuscation)
- Lint (Static Code Analysis)
- LeakCanary (Memory Leak Detection)

Example Stack:

- Language: Kotlin
- IDE: Android Studio
- Build System: Gradle
- UI Framework: Jetpack Compose
- Architecture Components: Android Jetpack
- Networking: Retrofit with OkHttp
- Image Loading: Glide
- Dependency Injection: Hilt
- Async Programming: Kotlin Coroutines
- Testing: JUnit, Espresso, Mockito
- Version Control: Git with GitHub
- Analytics: Firebase Analytics
- Crash Reporting: Crashlytics
- Push Notifications: Firebase Cloud Messaging
- App Distribution: Google Play Console

Web Development Tech Stack

Frontend:

1. **HTML5/CSS3**: For structuring and styling web pages.
2. **JavaScript (ES6+)**: Core language for client-side scripting.
3. **React.js**: A JavaScript library for building user interfaces.
4. **Redux**: For state management in React applications.
5. **Webpack**: Module bundler for JavaScript applications.

Backend:

1. **JavaScript (Node.js)**:
 - **Nodejs**: Nodejs is Javascript runtime environment.
 - **Express.js**: A minimal and flexible Node.js web application framework.
 - **NestJS**: A progressive Node.js framework for building efficient, reliable, and scalable server-side applications.
2. **Java**:
 - **Spring Boot**: Framework for creating stand-alone, production-grade Spring-based applications.
 - **Spring Security**: For securing applications.
 - **Spring Data JPA**: For data persistence and access.
 - **Hibernate**: ORM framework for mapping Java objects to database tables.
3. **TypeScript**: Strongly typed programming language that builds on JavaScript.
4. **RESTful API**: Designing RESTful web services.
5. **GraphQL**: Query language for your API, providing a more efficient and powerful alternative to REST.
6. **gRPC**: For efficient RPCs (Remote Procedure Calls).

Database:

1. **Relational Databases**:
 - **MySQL**: Open-source relational database management system.
 - **PostgreSQL**: Advanced, enterprise-class, open-source relational database.
2. **NoSQL Databases**:
 - **MongoDB**: Document-oriented NoSQL database.
 - **Redis**: In-memory data structure store, used as a database, cache, and message broker.

DevOps and Deployment:

1. **Docker:** Containerization platform for packaging applications and their dependencies.
2. **Kubernetes:** For container orchestration.
3. **Jenkins:** Continuous integration and delivery server.
4. **Git:** Version control system.
5. **GitHub/GitLab/Bitbucket:** For repository hosting and collaboration.
6. **AWS:** Cloud services for computing, storage, and networking.
 - **EC2:** Scalable virtual servers.
 - **S3:** Object storage service.
 - **RDS:** Managed relational database service.
 - **Lambda:** Serverless computing.
 - **Elastic Beanstalk:** For deploying and scaling web applications.
7. **Nginx:** Web server, reverse proxy, and load balancer.
8. **Terraform:** Infrastructure as code software tool.

Testing:

1. **JUnit:** Testing framework for Java.
2. **Jest:** JavaScript testing framework.
3. **Mocha/Chai:** JavaScript test framework and assertion library.
4. **Postman:** API testing tool.
5. **Selenium:** Automated testing of web applications.

Monitoring and Logging:

1. **ELK Stack (Elasticsearch, Logstash, Kibana):** For log management and analysis.
2. **Prometheus:** Monitoring and alerting toolkit.
3. **Grafana:** Open-source platform for monitoring and observability.
4. **New Relic:** Application performance monitoring.

Security:

1. **JWT:** For securely transmitting information between parties as a JSON object.
2. **OAuth2:** For authorization.
3. **OpenID Connect:** For authentication.