

# Bir programcının yol xəritəsi





**Agil**  
AgilAghamirzayev

I am a passionate Java Developer with more than 4 years of work experience in product-based companies and the banking industries.

Edit profile

48 followers · 41 following

- Kapital Bank
- Azeribaijan
- @AqilZeka
- in/agilaghamirzayev

Achievements



AgilAghamirzayev / README.md



# Welcome, This is Aqil

A passionate Java developer from Azerbaijan

Profile views 2,136

- I'm currently working [Kapital Bank](#)
- I'm currently learning ADVANCE PROGRAMMING CONCEPTS
- I'm currently studying Master of System Programming [Azerbaijan Technology University](#)
- All of my projects are available at [Here](#)
- I regularly write articles on [Medium](#)
- Ask me about Everything in Java
- How to reach me [aqilaghamirzayev@gmail.com](mailto:aqilaghamirzayev@gmail.com)
- Know about my experiences [LinkedIn](#)
- WhatsApp +994708144314

## Blogs posts

### Connect with me:

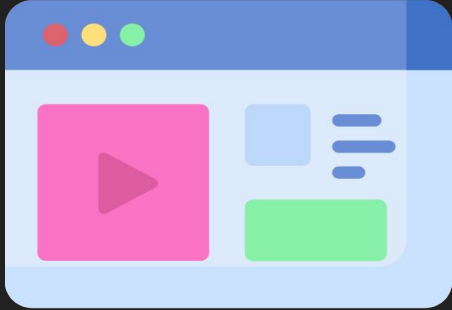


### Languages and Tools:

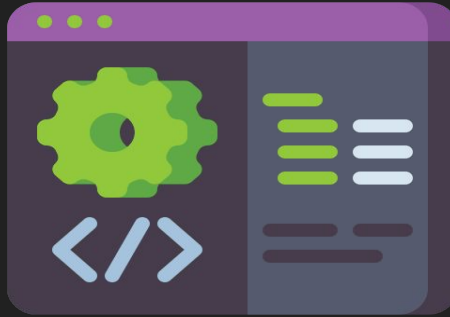


# Əsas sahələr

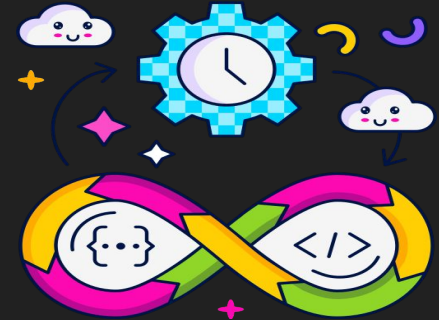
## Ful Stack



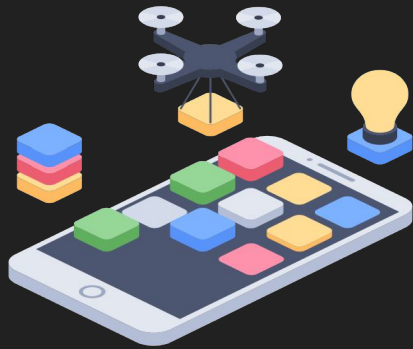
Frontend



Backend



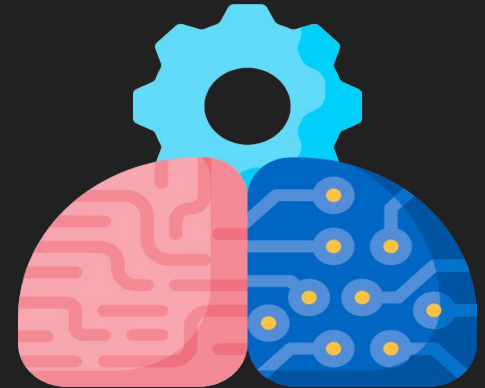
DevOps



Mobile



Data Science



Machine Learning



**Backend**



**Frontend**



**APIs**



**Full-stack**

# Backend yoxsa Frontend?

**X** YANLIŞ

Rahat

Pul

Çoxluq

**✓** DOĞRU

Cəhd

Həvəs

Bacarıq



# Backend

```
graph TD; Backend[Backend] --> Proqramlashdirma[Proqramlaşdırma dili]; Backend --> Versiyalama[Versiyalama]; Backend --> Verilənlər_bazası[Verilənlər bazası]; Proqramlashdirma --> Python[Python]; Proqramlashdirma --> Csharp[C#]; Proqramlashdirma --> Java[Java]; Proqramlashdirma --> Cplusplus[C++]; Proqramlashdirma --> Go[Go]; Proqramlashdirma --> Ruby[Ruby]; Proqramlashdirma --> JavaScript[JavaScript]; Proqramlashdirma --> PHP[PHP]; Versiyalama --> GIT[GIT]; Versiyalama --> GitHub[GitHub]; Versiyalama --> GitLab[GitLab]; Versiyalama --> BitBucket[BitBucket]; Verilənlər_bazası --> PostgreSQL[PostgreSql]; Verilənlər_bazası --> Oracle[Oracle]; Verilənlər_bazası --> MySQL[MySql]; Verilənlər_bazası --> MongoDB[Mongodb]; Verilənlər_bazası --> MariaDB[MariaDB]; Verilənlər_bazası --> MSSQL[MS SQL];
```

## Proqramlaşdırma dili

Python

C#

Java

C++

Go

Ruby

JavaScript

PHP

## Versiyalama

GIT

GitHub

GitLab

BitBucket

## Verilənlər bazası

PostgreSql

Oracle

MySql

Mongodb

MariaDB

MS SQL

# Java

```
graph TD; Java[Java] --> Fundamental[Fundamental]; Java --> Advanced[Advanced]; Java --> SpringBoot[Spring Boot]; Fundamental --> F1[Data tiplər və dəyərlər]; Fundamental --> F2[Şərtlər və döngülər]; Fundamental --> F3[Metodlar]; Fundamental --> F4[OOB, interfeys, class]; Fundamental --> F5[Alqoritm və data strukturlar]; Fundamental --> F6[Səhvlərin Tutulması]; Fundamental --> F7[Fayllar ilə işləmək]; Advanced --> A1[Yaddaşın idarə edilməsi]; Advanced --> A2[Collections (List, Set, Map)]; Advanced --> A3[Şəbəkələşmə, Sokerlər]; Advanced --> A4[Generiklər və stream-lar]; Advanced --> A5[Annotasiyalar]; Advanced --> A6[Garbage Collection, JVM]; Advanced --> A7[Thread-ların əsasları]; SpringBoot --> SB1[Spring Core (baxış)]; SpringBoot --> SB2[Rest API, HTTP Metodlar]; SpringBoot --> SB3[JDBC, Spring Data JPA]; SpringBoot --> SB4[ORM, Hibernate]; SpringBoot --> SB5[Maven, Gradle]; SpringBoot --> SB6[Unit testing, Mockito, JUnit]; SpringBoot --> SB7[Müşahibəyə hazırlıq];
```

## Fundamental

Data tiplər və dəyərlər

Şərtlər və döngülər

Metodlar

OOB, interfeys, class

Alqoritm və data strukturlar

Səhvlərin Tutulması

Fayllar ilə işləmək

## Advanced

Yaddaşın idarə edilməsi

Collections (List, Set, Map)

Şəbəkələşmə, Sokerlər

Generiklər və stream-lar

Annotasiyalar

Garbage Collection, JVM

Thread-ların əsasları

## Spring Boot

Spring Core (baxış)

Rest API, HTTP Metodlar

JDBC, Spring Data JPA

ORM, Hibernate

Maven, Gradle

Unit testing, Mockito, JUnit

Müşahibəyə hazırlıq



# Spring Boot

```
graph TD; SB[Spring Boot] --> F[Fundamental]; SB --> M[Medium]; SB --> A[Advanced]; F --- FTopics[Dependency Injection, IOC, MVC, AOP, Annotasiyalar, Spring Bean skopları, HTTP, REST, SOAP, gRPC, API-lər ilə işləmək, Spring Data JPA]; M --- MTopics[Spring Security, Authentication, Authorization, Oauth 2.0, JWT, Entity Lifecycle, Entity Relationships (1M, MM), Transactions, Async, Scheduler]; A --- ATopics[Spring Cloud Gateway, Cloud Config, Eureka, Sleuth, How Spring Boot works, Rabbit Mq, Kafka, Spring WebFlux, Müsahibəyə hazırlıq];
```

## Fundamental

Dependency Injection

IOC, MVC, AOP

Annotasiyalar

Spring Bean skopları

HTTP, REST, SOAP, gRPC

API-lər ilə işləmək

Spring Data JPA

## Medium

Spring Security

Authentication, Authorization

Oauth 2.0, JWT

Entity Lifecycle

Entity Relationships (1M, MM)

Transactions

Async, Scheduler

## Advanced

Spring Cloud Gateway

Cloud Config

Eureka, Sleuth

How Spring Boot works

Rabbit Mq, Kafka

Spring WebFlux

Müsahibəyə hazırlıq

# Arxitektura

```
graph TD; A[Arxitektura] --> B[Clean Code]; A --> C[Mikroservislər]; A --> D[Soft skills]; B --> B1[SOLID, DRY, YAGNI, KISS]; B --> B2[Design Patterns]; B --> B3[Design Principles]; B --> B4[Meaningful names]; B --> B5[Indentations and Code Style]; B --> B6[Avoid passing null, booleans]; B --> B7[Be consistent]; C --> C1[API Gateway]; C --> C2[Microservices pitfalls]; C --> C3[CQRS, SAGA]; C --> C4[Circuit Breaker]; C --> C5[Event sourcing]; C --> C6[Bulkhead pattern]; C --> C7[Backend For Frontend]; D --> D1[Communication]; D --> D2[Team working]; D --> D3[Decision Making]; D --> D4[Estimating and Evaluating]; D --> D5[Mentoring, Teaching]; D --> D6[Documenting]; D --> D7[Müsaibəyə hazırlıq];
```

## Clean Code

SOLID, DRY, YAGNI, KISS

Design Patterns

Design Principles

Meaningful names

Indentations and Code Style

Avoid passing null, booleans

Be consistent

## Mikroservislər

API Gateway

Microservices pitfalls

CQRS, SAGA

Circuit Breaker

Event sourcing

Bulkhead pattern

Backend For Frontend

## Soft skills

Communication

Team working

Decision Making

Estimating and Evaluating

Mentoring, Teaching

Documenting

Müsaibəyə hazırlıq

# Desing

```
graph TD; Desing --> LLD[Low Level Design (LLD)]; Desing --> HLD[High Level Design (HLD)]; LLD --> DP[Design Patterns]; LLD --> PL[Parking Lot]; LLD --> ES[Elavator System]; LLD --> CRS[Car Rental System]; LLD --> TTT[Tic Tac Tao]; LLD --> ATM[ATM]; LLD --> IMS[Inventory Management System]; HLD --> PRP[Proxy, Revers Proxy]; HLD --> RL[Rate Limiter]; HLD --> W[Whatsapp]; HLD --> US[URL Shortening]; HLD --> U[Uber]; HLD --> TW[Twitter (X)]; HLD --> IG[Instagram];
```

## Low Level Design (LLD)

Design Patterns

Parking Lot

Elavator System

Car Rental System

Tic Tac Tao

ATM

Inventory Management System

## High Level Design (HLD)

Proxy, Revers Proxy

Rate Limiter

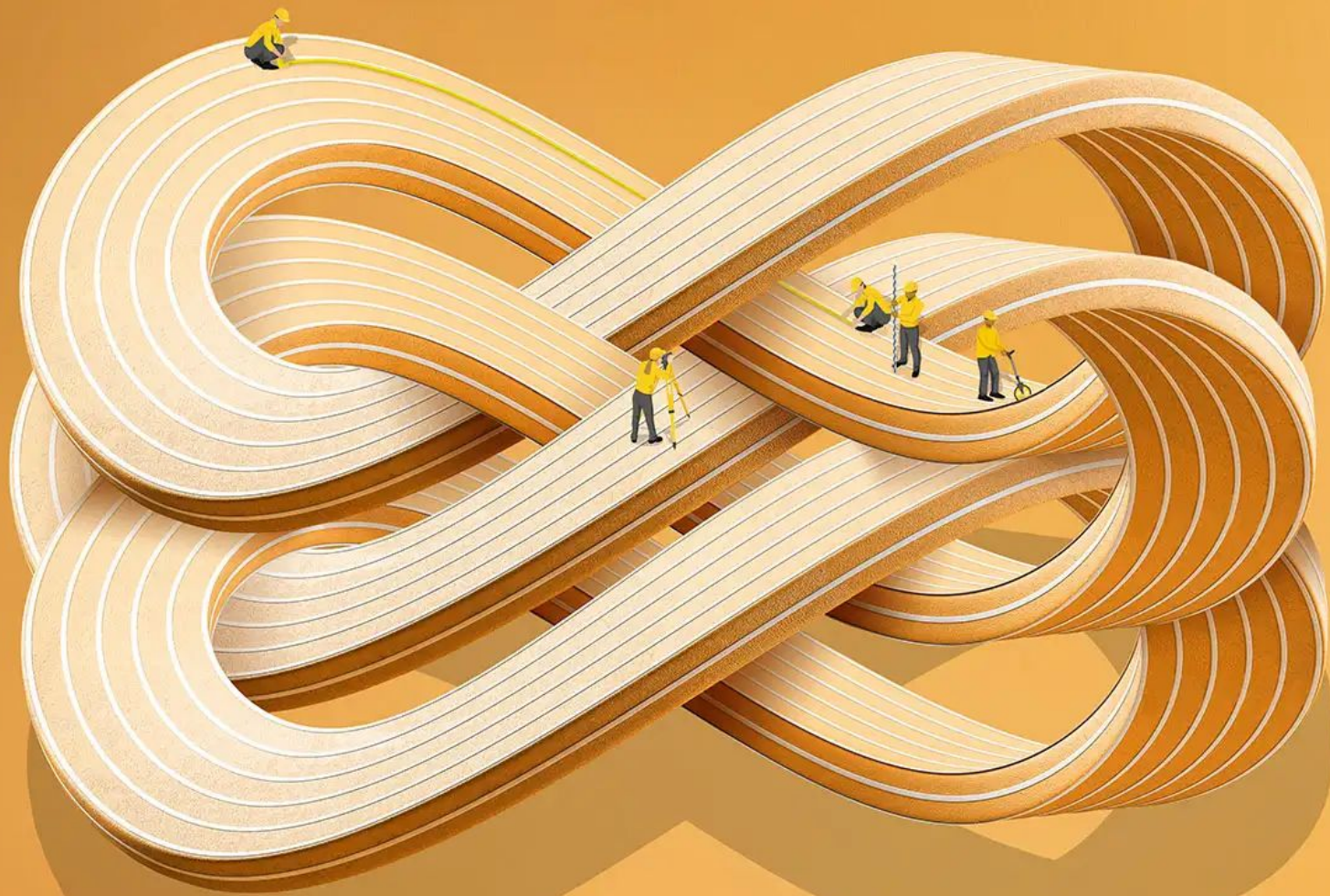
Whatsapp

URL Shortening

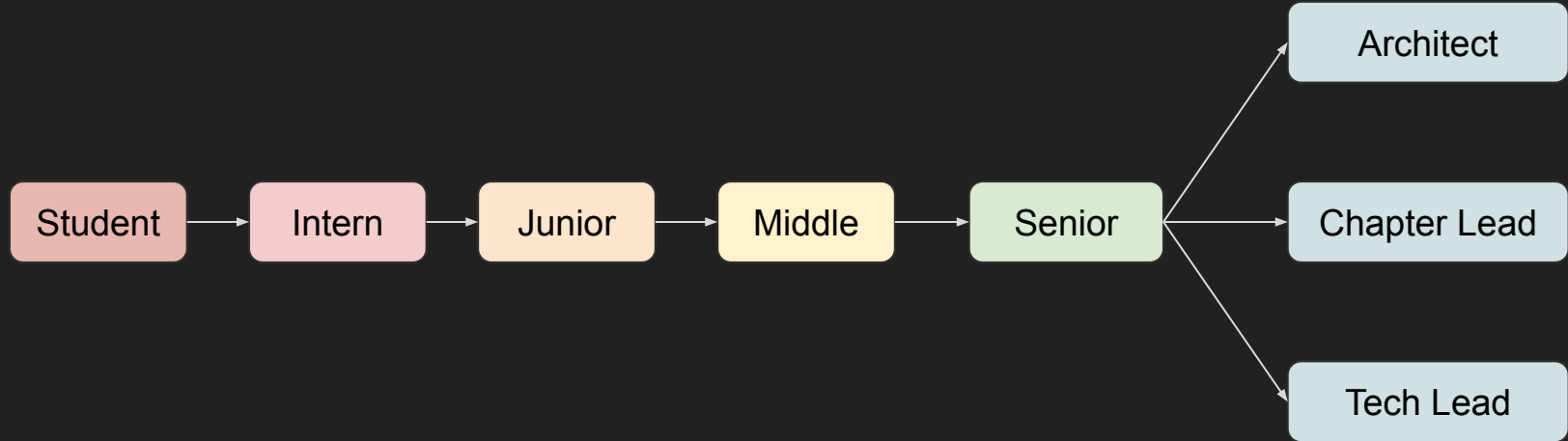
Uber

Twitter (X)

Instagram



# Level of Developers





# Hardan öyrənək?



## Materiallar

- 📌 <https://www.w3schools.com>
- 📌 <https://www.geeksforgeeks.org>
- 📌 <https://archive.org>
- 📌 <https://www.freecodecamp.org>
- 📌 <https://medium.com>
- 📌 <https://hashnode.com>

## Məsələ və misallar

- 📌 <https://leetcode.com>
- 📌 <https://www.hackerrank.com>
- 📌 <https://www.freecodecamp.org>

# Suallar

