

# **Lõputöö**

Tõhusate Andmestruktuuride ja Algoritmide Arendamine  
Veebirakendustele: Pythoni Jõudluse Optimeerimine WebAssembly  
(Wasm) abil

**Sergei Ivanov**

*IT Süsteemide arendus, Tartu Ülikool*

**Juhendaja: Andre Säask**

December 2, 2025

## Kokkuvõte

Akadeemiline artikkel räägib mikroteenustest, täpsemalt sellest, kuidas nad löid oma SpringBoot põhirakendusele Rust/Go lahenduse. See väidab, et Python on aeglane programmeerimiskeel isegi siis, kui rakenduse jõudluse kiirendamiseks saab kasutada JIT-kompilaatorit. Sisuliselt ei toeta kõik Pythoni teegid JIT-i. Go-d kasutati WASM-i jaoks plugeeriva ekstismi loomiseks. Rust oli API loomise algoritmi peamine keel, kasutades 'powerbot::api'. Lõpuks näitab artikkel mikroteenuste API loogika jaoks Go/Rust wasm-i kasutava rakenduse jõudluse kvantitatiivset analüüsi. See analüüs näitas, et Python on Rusti API-ga võrreldes väga aeglane. WASM-i eeliseks Kotlini SpringBoot rakenduse puhul on turvalisus ja kiirus, kuna artikli autor kasutas mikroteenuste vaheliste turvatud ühenduste jaoks TLS-sertifikaati.

**Võtmesõnad:** Wasm, Rust, Python, GO, API, WASI, monolith, microservices, lower level language, optimization, performance, scalability, agility, container, infrastructure, function, command, STOMP Protocol

## Sisukord

Transition from microservice architecture to a monolith to improve performance . . . . .	2
Allikad . . . . .	3

## **Transition from microservice architecture to a monolith to improve performance**

Palun, vaadake eelmist leht, kus asub sisukord ja kokkuvõtte koos võtmesõnadega.

## **Allikad**

Dogan, E. (2024). Transition from microservice architecture to a monolith to improve performance (Doctoral dissertation, University of Applied Sciences).