Atamert Ölçgen

İzmir, Turkey
muhuk@muhuk.com
http://www.muhuk.com
https://github.com/muhuk

- *I ♥ functional programming & Lisp. I also secretly appreciate well executed OOP.*
- *I am good at both application design & systems design.*
- *I can write concise technical documentation.*
- *I can explain complex concepts using examples, analogies and diagrams.*
- *Languages I would like to work with: Rust, Haskell, Scala, Clojure.*
- *We should always pick good and trade off between fast and cheap.*
- *TDD is the way to go for business rules. There are more efficient testing strategies for I/O code.*

Technical Skills

Likes:	clojure lisp scheme scala frege haskell love2d android c java game-programming game-engine game-ai django rust kotlin purescript idris
Dislikes:	php

Experience

Independent Consultant – Cogitative OÜ *May 2020 → Current*

Senior Software Engineer – FlyBot Pte. Ltd. *Mar 2019 → Oct 2019*
clojure, apache-kafka, apache-kafka-streams, manifold

- Designed and helped build kafka-streams based user engagement tool.
- Designed and helped build a high-concurrency message routing library to be used for distributing user notifications. (Kafka topic to client SSE connection)

Solution Architect for Digibank – DBS Bank *Feb 2017 → Feb 2019*
architecture, microservices

- Solution architect for single-codebase multi-country microservices transformation project for DigiBank
- End-to-end solution architect for AADHAAR base e-KYC for DigiBank India.
- End-to-end solution architect for DigiBank India overseas remittence project.

Software Consultant – Ölçgen Software *Mar 2015 → Feb 2016*

- Business process analysis.
- Full stack web development.
- Experienced with e-commerce & financial applications.
- Agile/lean development practises.

Software Consultant – NVest Singapore *Dec 2011 → Dec 2014*
scala, guice, guava, java, mysql, activemq, selenium, berkeley-db, php, imacros, ofx

SigFig

Senior Software Developer – Brandsfever Pte. Ltd. *Dec 2010 → Nov 2011*
django, sqlalchemy, jquery, jenkins, python-sphinx, rest, payment-gateway

Brandsfever

- Developed a **data warehouse** application.
- Contributed to in-house **e-commerce** engine.

Founding Partner – Ortak Us Ltd. *Jun 2010 → Aug 2010*
django, serious-games, simulation, training-simulation

Contributed to Reflex Training Simulation development.

Programmer – Ölçgen Bilişim *Jan 2009 → May 2010*
django, django-south, django-orm, django-models, django-forms, django-templates, django-admin

Developed Nakliyeci Rehberi, a portal for movers and forwarders.

Freelance Programmer *Jan 2008 → Jan 2009*

Türkiye Country Manager – Arukereso.hu *Jul 2006 → Jan 2008*

I was responsible from sales, marketing and content quality of ucuzu price comparison site.

Education

BS in Environmental Engineering – Yıldız Technical University *1999 → 2006*

High School – İzmir Atatürk High School *1995 → 1999*

Projects & Interests

Stack Overflow – https://stackoverflow.com/users/42188/muhuk *Dec 2008 → Current*
Written 148 answers. Active in django, django-models, forms, python, clojure and 1 other tags.

z-buffer-game – https://github.com/muhuk/z-buffer-game *Feb 2019 → Current*
rust, libtcod, entity-component-system, roguelike

Hobby game project written in Rust

clecs – https://github.com/muhuk/clecs *Apr 2014 → Current*
clojure

Entity-component-system for Clojure.

triangulate – https://github.com/muhuk/triangulate *Oct 2013 → Current*
clojure

A Clojure implementation of "Efficient Triangulation Algorithm Suitable for Terrain Modelling" (Bourke, 1989).

lesscss-cli – https://github.com/muhuk/lesscss-cli *Jan 2013 → Current*
java

Command-line interface for lesscss-java

Public Artifacts

First Order Thinking and Pragmatism Are Not the Same Thing — muhuk's blog – *Jan 2019*
http://blog.muhuk.com/2019/01/13/first_order_thinking_and_pragmatism_are_not_the_same_thing.html

Most non-technical people think software development is a standardized, repetable process. That is why they come up with enterprise agile frameworks. If it was standardized we would not need to write software but we would generate it from specifications[1]. Every new project, every new task has a component of discovery. It requires us to think outside of the box. Therefore assembly line approach to software development produces mediocre results at best.

Getting a Little Further Than Hello World With Rust - Part 2: Test Driven Development — muhuk's blog – *Apr 2018*
http://blog.muhuk.com/2018/04/03/galfthw_rust_part_2_test_driven_development.html

Second installment of introductory Rust tutorial.

Getting a Little Further Than Hello World With Rust - Part 1: Ownership & Mutability — muhuk's blog – *Nov 2017*
http://blog.muhuk.com/2017/11/28/galfthw_rust_part_1_ownership_mutability.html
rust, rust-cargo

Introductory tutorial for Rust language, its standard library and toolset.

Presentation: Building a Simple DSL in Clojure — muhuk's blog – *Nov 2017*
http://blog.muhuk.com/2017/11/01/presentation_building_a_simple_dsl_in_clojure.html
clojure

A presentation I did for Singapore Clojure Group

Laziness in Clojure — muhuk's blog – http://blog.muhuk.com/2015/12/22/laziness_in_clojure.html *Dec 2015*
clojure, delayed-execution

Lazy evaluation is delaying known operations until they are forced. What I mean by laziness in this post is a bit more general than that, for some constructs what operations are delayed is not known and some are evaluated before a value is forced. When used correctly these differences should not matter. A significant detail, however, is that they all evaluate at most once.

Infinite Sequences in Clojure — muhuk's blog – http://blog.muhuk.com/2015/11/10/infinite_sequences_in_clojure.html *Nov 2015*

Who in Their Right Mind Would Use Monads in Clojure? — muhuk's blog – http://blog.muhuk.com/2015/10/01/who_in_their_right_mind_would_use_monads_in_clojure.html *Oct 2015*
clojure, monads

TL;DR; You might find monads helpful in expressing large computations even in a dynamic context.

Performance Tuning Clojure Generative Tests — muhuk's blog – http://blog.muhuk.com/2015/09/17/performance_tuning_clojure_generative_tests.html *Sep 2015*

Using Guice Effectively — muhuk's blog – http://blog.muhuk.com/2015/05/28/using_guice_effectively.html *May 2015*

Mind Your Form — muhuk's blog – http://blog.muhuk.com/2015/04/07/mind_your_form.html *Apr 2015*

When Decent Programmers Fail — muhuk's blog – http://blog.muhuk.com/2015/01/26/when_decent_programmers_fail.html *Jan 2015*

Is Clojure Homoiconic? — muhuk's blog – http://blog.muhuk.com/2014/09/28/is_clojure_homoiconic.html *Sep 2014*

What is Expression Problem and Why Should We Care? — muhuk's blog – http://blog.muhuk.com/2014/03/12/what_is_expression_problem_and_why_should_we_care.html *Mar 2014*
clojure, expression-problem

The Expression Problem is a new name for an old problem. The goal is to define a datatype by cases, where one can add new cases to the datatype and new functions over the datatype, without recompiling existing code, and while retaining static type safety (e.g., no casts).

A Makeshift Solution to Expression Problem — muhuk's blog – http://blog.muhuk.com/2014/03/27/a_makeshift_solution_to_expression_problem.html *Mar 2014*
clojure, expression-problem

In the previous post I tried to demonstrate the expression problem and its implications. In this post I will present a solution. As I mentioned last time I interpret the static type safety clause as no monkeypatching. Otherwise there is no solution for expression problem in Clojure, since it's a dynamic langulage with no static type checking.

Unified Support for Numbers in Scala — muhuk's blog – http://blog.muhuk.com/2013/09/29/unified_support_for_numbers_in_scala.html *Sep 2013*

Using Layouts In Qooxdoo - Index — muhuk's blog – http://blog.muhuk.com/2009/04/30/using-layouts-in-qooxdoo-index.html *Apr 2009*

Apps & Software

yurt - Functional Relational Programming – http://www.yurt-persistence.com/
scala, jvm, functional-programming, clojure, kotlin, java, persistence, akka

Yurt is a Functional Relational Programming platform as described in Out of the Tar Pit [Ben Moseley & Peter Marks, 2006]

This is my personal project.

Tools

Favorite Editor:	Emacs