

# Pooria Azimi

## Curriculum Vitæ

<http://bit.ly/pooria-azimi-cv>



### RESEARCH INTERESTS

#### Database and Information Retrieval Systems

#### Operating Systems

(File systems – Microkernels)

#### Concurrent Programming

(the “Actor Model”)

#### Human-Computer Interaction

### EDUCATION

Amirkabir University of Technology (Tehran, Iran)

B.Sc. in Computer Science, 2014 (expected)

GPA (last 60 CS credits): **16.5/20**

### AWARDS AND HONORS

- **2<sup>nd</sup> place** at **13<sup>th</sup> International Data Mining Cup** (Berlin)  
*Our team created a bidding agent in Java for the “online” task, and used a combination of statistical models, neural networks and SVMs (with **MATLAB** and **Weka**) for predicting the results of the “offline” task.*
- Ranked 590 (among ~400,000 participants) in 2008 National Matriculation Exam (**top %0.2**)

### COMMUNICATION SKILLS

ENGLISH    Fluent

PERSIAN    Native speaker

GERMAN    Basic understanding

📍 365, E2, Ekbatan, Tehran, Iran

☎ + 98 (935) 431 26 45

✉ [pooriaazimi@gmail.com](mailto:pooriaazimi@gmail.com)



### TEACHING EXPERIENCES

#### Teacher Assistant

Database Design (Winter '11)

Database Design (Winter '12)

Artificial Intelligence (Winter '12)

Database Design (Winter '13)

Data Mining (Spring '14)

Internet Engineering (Spring '14)

#### Presenter

4th National Linux Festival (Spring '12)

#### Advanced Unix Command-line

AUT-CEIT Computing Festival (Spring '12)

#### Introduction to Node.js

AUT Database Workshop (Spring '13)

#### PostgreSQL vs. MySQL

AUT Database Workshop (Spring '13)

#### Introduction to MongoDB

AUT Database Workshop (Spring '13)

#### Basics of Neo4j and Redis

### TECHNICAL SKILLS

#### Programming Languages

Experienced: **Java** – **Ruby** – **Objective-C** – **CoffeeScript**

Familiar: **Scala** – **C** – **Erlang** & **Elixir** – **Scheme** – **PHP**

#### Databases

Used extensively: **MongoDB** – **PostgreSQL** – **Redis**

Used occasionally: **Neo4j** – **MySQL** – **MS SQL Server**

#### Server-side Web development

Experienced: **Ruby on Rails** – **Node.js** – **Sinatra** – **PHP**

#### Client-side Web development

Limited experience: **Backbone.js** – **Ember.js**

#### Miscellaneous

Source Control Management (**git** & **hg**)

UNIXish tools (**vi**, **awk**, **sed**, etc.)

**BDD** – **ANTLR** – **nginx** – **WebKit** – **LaTeX**

TOEFL iBT		GRE GENERAL	
Reading	<b>30</b>	Verbal Reasoning	<b>164</b> (%93)
Listening	<b>30</b>	Quantitative Reasoning	<b>170</b> (%98)
Speaking	<b>27</b>	Analytical Writing	<b>4.5</b> (%78)
Writing	<b>29</b>		
Total	<b>116</b>		

## NOTABLE ACADEMIC PROJECTS

---

November 2013 – Present

### Translatr

**My Thesis project** – An online, computer-assisted translation manager software that would help teams of translators with their jobs by providing features like automatic translation, spell- and grammar-checking, translation memory, and terminology manager.

*(Ruby on Rails – Node.js – PostgreSQL)*

July 2011 – April 2012

### Kavandeh Search Engine

Improving link-based Web page ranking algorithms in a Persian-only search engine, using various statistical and heuristic methods.

Our team of two heavily improved upon Nutch's scoring, parsing, crawling, and spam detection submodules, and used WebKit's rendering engine for detecting semantically significant parts of a (rendered) Web page and assigning more weight to links in such areas.

*(Apache Nutch & Solr – WebKit – Java – C++)*

March 2012 – June 2012

### Visual WebPage Segmentation (VPS)

Detecting Web page structure with statistical analysis of the visual representation of the rendered page content, and using that structure for improving ranking algorithms in a search engine.

*(Node.js – PhantomJS – MongoDB)*

### Baygan Database

January 2011 – June 2011

An extendable and clearly-modulated framework for introducing students to the intricacies of relational database design. Inspired by [pintos](#).

*(Java – ANTLR)*

June and July 2010

### Embedded Search Engine

A complete, single-purpose search engine (all parts written from scratch), designed to use minimal online memory (60 MB), for indexing and searching the contents of the English Wikipedia.

*(Java)*

## REFERENCES

---

**Shahram Khadivi** (Assistant Professor, Amirkabir University of Technology, Iran)  
<http://ceit.aut.ac.ir/~khadivi/> – Email

For much of the past 3 years, and also the coming year, I have been working closely with Dr. Khadivi on multiple projects (Kavandeh, VPS, Baygan), Berlin Data Mining Cup, and now my thesis project. I have also been Teacher Assistant to a couple of Dr. Khadivi's courses ("Database Design": three years, "Data Mining": one semester).

More references are available upon request – but I've included only Dr. Khadivi's address above because I think he would be in the best position to evaluate my academic and technical expertise.

## OTHER PROJECTS

---

2010 – 2013

### Open Source Contributions

Contributing to multiple Open Source projects (code, documentation, bug report, and IRC support), including [PhantomJS](#), [Ag](#) (the silver searcher), [hg-prompt](#), [Kiwi](#), [shadowsocks](#), [fish shell](#), and [Homebrew](#).

*(JavaScript – awk – Python – Objective-C – Ruby)*

### BetterDictionary

*(Objective-C – Cocoa)*

### MiniJava Parser

*(CoffeeScript – jison – d3.js)*

### Readr

*(CoffeeScript – HAML – SCSS – Foundation )*

### A Twitter Clone!

*(Ruby on Rails – PostgreSQL – CoffeeScript – Sass)*

### Farhang (partial contribution)

*(Objective-C – Cocoa Touch – Core Data)*

### Secure File System

*(PHP – MongoDB)*

### Ad Server

*(Ruby – Sinatra – SQLite)*

## ONLINE EDUCATION

---

In addition to attending my normal classes, I have watched the videos, and finished the assignments, of more than a dozen freely-available online courses, including the following CS-related ones:

- MIT's [Structure and Interpretation of Computer Programs](#) (1986 – by [Harold Abelson](#) and [Gerald Jay Sussman](#))
- UC Berkley's [Operating Systems and Systems Programming](#) (2008)
- Harvard's [Introduction to Computer Science](#) (2010) and [Building Dynamic Websites](#) (2010)
- Stanford's [Programming Methodology](#) (2007), [Programming Abstractions](#) (2008), [Programming Paradigms](#) (2008), and [iPhone Application Programming](#) (2011 & 2013)

I'm also taking the following Coursera courses this semester:

- EPFL's [Functional Programming Principles in Scala](#) (2013 – by [Martin Odersky](#))
- EPFL/Typesafe Inc.'s [Principles of Reactive Programming](#) (2013 – by [Martin Odersky](#), et al.)