

SUMMARY

I am a software engineer at wikiHow, where I build web products that reach millions every day and create tools to enhance business processes. Previously, I helped Ryanair grow its online platform as it expanded rapidly, playing a central role in improving old systems and building new ones.

With over 8 years of experience in the industry, and having designed, built, extended, and maintained large systems in a variety of languages, I am a versatile engineer comfortable at all levels of the web stack, from servers to stylesheets. I am particularly adept at translating business needs into reliable, efficient, and extendable back-end code. The largest projects I've worked on were built on PHP and Java, and I have also written a fair amount of Python, Bash and JavaScript.

Here are some of the tools and techniques I've used:

- Java EE: Spring, Hibernate, Maven, JUnit, Mockito, Liquibase, JSTL, EL, AspectJ, JAX-RS, JAXB.
- PHP: Zend framework, MediaWiki, Laravel, Composer, PHPUnit, Mockery, Doctrine, Xdebug.
- Python: Flask, Bottle, virtualenv.
- Data: MySQL, Redis, Memcached, Amazon Redshift.
- Infrastructure: AWS, Rackspace, Fastly, Chef, Puppet, Vagrant.
- Linux: shell (Bash) scripting and a plethora of command line tools.
- Design patterns, algorithms, data structures, Scrum, Kanban, Git, Mercurial.
- Public APIs: Google, Facebook, Twitter, Yahoo, etc.
- Client-side: Javascript, jQuery, HTML, CSS, Bootstrap, AMP.
- To a lesser degree: C, TCL, Ruby, Go, PostgreSQL, MongoDB, Oracle DB.

EXPERIENCE

Software Engineer at wikiHow

February 2015 → Ongoing

Palo Alto, California (remote)

- Full-stack software engineer with a focus on the back-end; I work with many tools and languages: PHP, Python, Bash, MySQL, Redis, Memcached, Varnish, Chef, Vagrant, AWS, JS, HTML, CSS, etc.
- Develop new features for wikihow.com (mobile and desktop) and fix security vulnerabilities.
- Create software to detect and prevent malicious bot activity like denial-of-service attacks.
- Configure and provision new server infrastructure using Chef to install Varnish, Memcached, etc.
- Prepare a Vagrant VM to make it easy for engineers to run and develop code for wikihow.com.

Software Engineer at Ryanair

September 2011 → February 2015

Dublin, Ireland

- I was a core Java back-end developer working on ryanair.com. My colleagues and I rewrote the website from scratch and extended it with new features such as geolocation, public APIs, etc.
- Create an email marketing application in Java, backed by a Hadoop database.
- Develop the core of a very large PHP project (old ryanair.com; ~15 million hits/day at the time).
- Coordinate code releases, manage 10+ Linux servers (build automation, Tomcat, Git hooks, etc).
- Create developer/sysadmin tools and automation scripts with Python and Bash.
- Design MySQL database schemas, optimize queries, configure MySQL Server.
- Develop JavaScript modules; e.g. an analytics plugin used to track user behavior, bookings, etc.

Web Developer at Château de la Tuilerie

April 2011 → September 2011

Nîmes, France

- Create their new online shop from scratch and an admin panel to manage it.
- Develop my own PHP microframework (router, templating system, helpers, etc.).
- Modify existing Java inventory software.

Freelance web developer

2010 → 2011

Toledo, Spain

- As a student, I created PHP websites for local businesses and updated their existing sites.

EDUCATION

Associate Degree in Computer Science (Tertiary education / ISCED 5B) Azarquiel Institute of Technology	Toledo, Spain 2009 – 2011
Web Development Course M.P.R. Escuela Informática (programming school)	Toledo, Spain 2008 – 2009
Baccalaureate in Science (Upper secondary general / ISCED 3) Alfonso X el Sabio (secondary school)	Toledo, Spain 2006 – 2008
Secondary Education (Lower secondary general / ISCED 2) La Sisle (secondary school)	Sonseca, Spain 2002 – 2006

Courses completed online:

UC Berkeley: CS 61B - Data Structures	Prof. Jonathan Shewchuk
MIT: 6.042J - Mathematics for Computer Science	Prof. Tom Leighton, Dr. Marten van Dijk

Some of my favourite technical books:

Algorithm Design Manual, The	Steven Skiena
Computer Networking: A Top-Down Approach	James Kurose, Keith Ross
Effective Java	Joshua Bloch
Head First Design Patterns	E. Freeman, B. Bates, K. Sierra, E. Robson
High Performance MySQL	B. Schwartz, P. Zaitsev, V. Tkachenko
Java EE 6 Pocket Guide	Arun Gupta
Pro Git	Scott Chacon
Silence on the Wire: A Field Guide to Passive Reconnaissance and Indirect Attacks	M. Zalewski
SQL Antipatterns: Avoiding the Pitfalls of Database Programming	Bill Karwin
Systems Performance: Enterprise and the Cloud	Brendan Gregg
Tangled Web: A Guide to Securing Modern Web Applications, The	Michal Zalewski
Well-Grounded Java Developer, The	Benjamin J. Evans, Martijn Verburg

CODE SAMPLES

I like solving algorithmic puzzles, like the ones in projecteuler.net, here are some examples:

- An efficient [Sudoku solver](#) (a backtracking algorithm with search pruning).
- An [optimal path finder](#) for matrices (an implementation of Dijkstra's algorithm).
- A [Pell's equation solver](#), which uses the convergents of continued fractions to find solutions.