Ustun Ozgur Software Technologies Company Overview

Üstün Özgür, Founder

2017

Software is Hard. We Make it Easy.

Contents

1 What We Offer **1**
- 1.1 Product Development . . . 1
- 1.2 Modernization & Rearchitecture . . . 1
- 1.3 Education . . . 2

2 What We Believe in **2**

3 What We Provide and Use **2**
- 3.1 Language Ecosystems . . . 2
- 3.2 Databases . . . 3

4 Who We Are **4**

5 Contact Us **4**

1 What We Offer

1.1 Product Development

At Ustun Ozgur Software, we pride ourselves on understanding our customer's needs and offering solutions best suited for the task at hand. We have extensive experience building great software applications, and we offer end-to-end solutions, from inception of an idea to final product on the market.

1.2 Modernization & Rearchitecture

We also have experience removing technical debt to bring mature but old applications to modern architectures and offer rearchitecturing services that will revive your codebase and company. Most software development comes to a halt after some time, due to rapidly changing requirements and the failure of software teams to accomodate change. We offer solutions to fix this.

1.3 Education

For technical companies, quality and skillset of people is of prime importance; and we offer training services so that you can modernize the skillset of your staff. We offer tranining on software development, JavaScript, React, backend and frontend Web application development and databases.

2 What We Believe in

We believe software engineering is a young discipline that is changing rapidly, with new technologies emerging old the time, old ideas being rediscovered and rebranded, and new ideas being discovered and implemented. As a community, we are not there yet, however we adhere to some principles to keep it simple and robust:

- We believe functional programming is an important paradigmn that is being rediscovered and getting more mainstream. We are experts in functional programming and favor the paradigm whenever is possible.
- Immutability and immutable by default architectures force engineers to re-think how they approach state in applications, management of which is the single-most source of complexity and bugs in computer programs. We are true believer in immutable by default, mutable by request architectures.
- We also belive object-oriented programming has important assets, but is widely misunderstood and implemented incorrectly by the masses. We believe OOP, coupled with immutability brings abstraction benefits to the table that functional programming cannot provide on its own, hence OOP done right is a vital tool in our toolbox.
- Domain-driven design: Most of the complexity in software arises because the implementers have vague knowledge on the domain they are building applications for. We believe computing is a way of advanced thinking that brings tools to re-think how we understand the world and the domains we are implementing software for.

3 What We Provide and Use

We provide the following services, with the related platforms:

3.1 Language Ecosystems

- For rapid application development for MVPs, we use and recommend Django and Python for its built-in good defaults and widespread community support. Coupled with Django Rest Framework, building API's for CRUD apps is a breeze with Django.
- For applications where microservice architecture is more apt, realtime connectivity is desired, or where performance is a higher priority, we prefer to use node.js and its ecosystem. It provides better performance thanks to its async nature, and the skillset can be reused on the frontend as well.

- We believe node.js provides the best environment for building microservices thanks to the following builtins:
  - Async by default: Services communicate with each other via async mechanisms, and the whole ecosystem is built around that.
  - Ease of getting started and significant momentum: There is a Cambrian explosion of frameworks, libraries and ideas in the node ecosystem and we believe, while the pace of this progress can be daunting and tiresome to some, it is getting more mature, approachable and useful everyday and it is a safe choice to bet on JavaScript for the future.
  - Rapid prototyping thanks to fast startup and reload times: Compared to other popular dynamic languages like Python and Ruby, JS is significantly faster thanks to the competition in browsers and the wealth of ecosystem is immense. Compared to more heavyweight frameworks, the fast reload times of these applications make building microservices a breeze.
  - We favor cote.js to build our microservice architectures on node.js, a relatively unknown pearl in the node.js ecosystem.
- For more ambitious and ground-breaking apps, we prefer to use Clojure, as it provides the best development experience for us on the server and on the client, combining high performance with rapid development. The interactive development environment, the immutable, functional but pragmatic foundations,coupled with the hosted nature of Clojure which provides the ability to use three significant platforms, JVM, .net and JavaScript as compilation target makes it the best choice for significant application development.
- We owe a great gratitude to Clojure and its ecosystem, and most of the ideals we believe in are rooted in the philosophy of Clojure, and its leader Rich Hickey.

On frontend, we are true believers that JavaScript and languages that compile to JS are the way to build ambitious apps. To this end, we prefer to treat the backend as an API, and build ambitious apps using React and its ecosystem. Depending on the complexity of the app, we use various state management techniques, including, but not limited to Redux. We are also experienced in building ClojureScript apps that use various libraries that wrap React and its friends.

3.2 Databases

For data management, we believe that for most use cases, relational databases suit the purposes fine. However, depending on the requirements, sometimes a NoSQL databases suits the task better, and we are skilled in various NoSQL databases like Datomic, MongoDB, Redis and ElasticSearch (as a non-primary store for searching purposes).

For RDBMs, addition of an ORM layer as in Django provides a simple way to manage this data for most of the basic tasks, however we believe that ORM's usually provide an opaqueness that harm productivity and performance. We are skilled in performance tuning for these problems, and also find dropping to the SQL level whenever needed, a vital skillset. For Clojure applications, we eschew ORMs in favor of more data-centric layers as offered by JDBC coupled with SQL statement generators like HoneySQL.

Sometimes though, a document is just a document, and we find that just storing it as a composite document removes a lot of the barriers between the mapping of data from DB to application layer. To this end, we favor the JSON capabilities of Postgres and of Mongo, depending on the use case.

For more ambitious, that look onward to the future, we believe immutable databases like Datomic bring a lot to the table. We find that storing historical data is such a common task and disk space so cheap that deleting historical data to save the current state of the world is somewhat of an historical relic from the 1970s. We are true believers that the immutable data model of Datomic has a lot to offer in the following years, and hopeful that append-only data stores like Datomic become more widespread in the future.

4 Who We Are

We are a team based in Istanbul, Turkey, comprising Ustun Ozgur, and Serkan Ozer, both with MSc degrees in engineering from Bilkent University and University of Barcelona respectively.

We also have an advisory board whom we consult with on architectural and business decisions, consisting of Armagan Amcalar, Head of Engineering at UNU, MSc Sabanci University and Mehmet Onsiper, MSc UC Davis.

5 Contact Us

If you are interested in transforming:

- Your idea to a real, successful product
- Your company to a modern software stack
- Your staff to better software engineers,

Contact us at contact@ustunozgur.com

We will be providing you the services to prepare you for the next world, the world understood, governed and made better by software.

`http://ustunozgur.com`