STEPAN PARUNASHVILI

San Francisco, CA | 646-397-8797 | stepan.p@gmail.com | www.stepanp.com

CORE EXPERIENCES

Senior Software Engineer, Wit.ai, Facebook (2017-present)

Wit is an API that allows developers to use NLP in their apps, without any knowledge of machine learning

- Patented, pitched, and led the team to create Smart Replies, a tool that helped Page owners automate conversations. Launched at F8 2017.
- Patented, pitched, and led the team to create Built-In NLP. Built-In NLP allows developers on Messenger Platform to instantly leverage Wit.ai's NLP. Led the launch of Internationalization support for Built-In NLP, and supported it's extension for F8 2018.
- Migrated Wit.ai's Frontend into Facebook, to leverage automatic deployments and Facebook's Frontend Infra. Created a plugin to convert CoffeeScript to ES2015 JavaScript and JSX. The project was landed without any errors.
- Led the migration of Wit.ai's business logic and underlying database from Clojure to Facebook Infra. Layed out a plan of multi-phase deployments, scoping out projects for a team of 5. Instrumented systems to run consistency checks, fixes, and shadow-tests. Created an effects-based framework to migrate the logic for writes. The migration was completed without downtime.

Software Engineer, M/AI, Facebook (2015-2016)

M is the an AI assistant that answers your questions in Messenger.

- Core contributor to the task force that launched BotEngine for Wit.ai. Built the user interface that allowed users to create conversation trees, to be used for dialog training. Designed, patented and architected branching and quick replies for BotEngine
- Architected complex text editing UIs in both Wit.ai and M. This included the AI Composer, which dynamically suggested answers and structured messages in M, and Tagger, which was the tool that users used to tag specific portions of sentences to build NLU models in Wit.
- Designed and implemented WittyFiddle, a UI for deploying Wit.ai apps on the Messenger Platform.
- Led the team that built the console that humans used to train the AI behind M. Architected the system with React.js, Immutable.js, Flux.
- Built the Clojure backend and the frontend for adding quick replies, intent requirements, and peer feedback for trainers.
- Profiled and increased web app performance for a complex React Application. Transitioning between threads decreased from 900ms to 100ms
- Owned and launched support for VOIP, and dynamic emails for M Trainers.
- Launched two UI engineering training courses for React.js and ES2016 in FB Menlo Park and FB NYC.

Software Engineer, Wit.ai (2014)

Wit is an API that allows developers to use NLP in their apps, without any knowledge of machine learning

- Employee #2. Took ownership of the frontend architecture
- Migrated an AngularJS architecture to React and Flux
- Built a ClojureScript application to dispatch requests to mechanical turk, for validating voice data
- Built the documentation for Wit, and for Duckling, in a custom framework written in Ruby
- Introduced end-to-end testing into Wit

Technical Co-Founder, China-Based Software-Engineering Consultancy (2012-2014)

Consulted and built software from the ground up, for companies in China and Taiwan

- Built the recruitment platform for Atlas China, supporting candidate matching, and video interviews
- Built a custom booking system for Opohills in Taiwan, and for the Whitehouse Hotel in Mongolia
- Founded a startup that helped Dentists get more patients. Built a call-tracking application in Ruby.

SPEAKING

Destructuring ES2016 - FB NYC Intensive (Sep 2016)   How to React - FB NYC Intensive (Sep 2016)
On Engineering Growth - GGW UChicago (Aug 2016)   On Scaling React Applications - BrazilJS (Aug 2016)
On Scaling React Applications - Applicative (Jun 2016)   React Application Structure - Learn@Facebook (Jun 2016)
Universal Applications - Spotlight React (Mar 2016)   Universal Applications - FEDay Guangzhou (Mar 2016)
ES2015 - Presentation at Wealthfront (Dec 2015)   ES2015 & React - Presentation at FutureAdvisor (Jan 2016)