

**FRONT-TRENDS 2014**  
Conference Booklet



# Welcome to

# FRONT-TRENDS 2014

## Conference

**02** Conference Day 1

**04** Interview with Jed Schmidt

**07** Conference Day 2

**10** Interview with Lea Verou

**12** Conference Day 3

**16** Interview with Mary Rose Cook

**18** Interview with speakers

**20** Sponsors

# 1

## SCHEDULE

# DAY

Wednesday, May 7

<b>08:00</b>	Registration and breakfast
<b>09:30</b>	Open and Agenda
<b>09:45</b>	Julian Cheal Dancing with Robots
<b>10:20</b>	Tero Parviainen Build Your Own AngularJS
<b>10:55</b>	Coffee Break
<b>11:25</b>	Jed Schmidt API Have A Dream
<b>12:00</b>	Tim Messerschmidt Supercharging Node with Kraken
<b>12:35</b>	Lunch Break
<b>14:10</b>	Rachel Andrew We're Not "Doing a Startup"
<b>14:45</b>	Zeno Rocha A future called Web Components
<b>15:20</b>	Coffee Break
<b>15:50</b>	Various speakers Lightning talks (Rooms A / B / C)
<b>17:30</b>	End of day

THE  
**BIGGEST**  
THREAT TO  
**JAVASCRIPT**  
is its own  
**UBIQUITY**  
AS IT BECOMES A COMPILE  
**>TARGET<** for other  
**LANGUAGES**

Interview with Jed Schmidt

Jed is a very active person in JavaScript community. He opens up conferences, sings and deliver fun talks among other things.

Being a Japanese translator he explains how you can links these two languages and what his opinion of the future of JavaScript is.

What needs to happen to a Japanese translator that he falls in love with JavaScript?

I think it's a pretty natural thing to happen to a late-learning bilingual like myself. I started studying Japanese seriously when I turned 20, and learning a new natural language post-adolescence is a very different endeavor than learning it as a child in a multi-lingual home. For better or worse, your "hardware" is already set; so you have to learn in "software".

So while you'll never be as convincing as a speaker who grew up with the language, you'll have a much better grasp of language itself, because you're forced to reconcile new words and concepts with old ones. This means you learn much more about language in the abstract.

Which is how I came to enjoy programming, because computer languages are inherently more abstract than natural languages. It also didn't hurt that JavaScript is basically the computer's answer to English; a warty hodgepodge that somehow became the lingua franca of its time.

What translation skills are helpful for programmers?

As a process, translation is most specifically like porting code between platforms. First you have to read and parse the source, figure out what it means, and then serialize it into the target.

This means you need to be good at disambiguating source code, which I think has been a useful skill. For example, Japanese is very context-dependent. It's totally cool for authors to omit subjects and objects. So, you have to maintain more state in your head than you would otherwise, which is a useful skill when reading other people's code.

Should languages other than JavaScript be allowed in browsers?

Is there anything preventing browsers from adding languages? Personally, I don't think arbitrary languages are worth the additional complexity in a browser runtime, especially given the success we've seen from folks doing great work in sculpting the environment they want to see out of JavaScript. CoffeeScript and ES6

# JED SCHMIDT

Jed is a JavaScript developer and recovering Japanese translator based out of Brooklyn, NYC. When he's not geeking out on the mobile web at UNIQLO, he enjoys running BrooklynJS and singing tight barbershop harmonies.

## Talk: API Have A Dream

As JavaScript continues to eat everything, it seems like the lifespan of any new JavaScript library ends up at about half of the one it replaced. While each of these tools serve to make our lives easier, the churn from switching sometimes makes

transpilers are a decent path for those looking for a better JavaScript, while ambitious projects like ClosureScript and asm.js seem promising at the other end of the abstraction spectrum.

Are you happy with the job TC39 is doing?

I'm grateful for the work they do. To be honest, I don't think I'd have the patience to see things through the way they have. That said, I'm not really the type to get too excited by what's coming down the pike anymore. I'd rather wait for things to shake out in implementations before caring too much.

What are the biggest challenges JavaScript will have to face in the next 5 years?

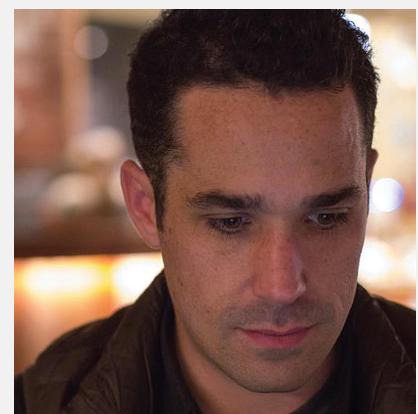
JavaScript may have its share of challenges, but I think things look pretty good compared to other languages. I agree with @garybernhardt's thesis in his "The Birth and Death of JavaScript" talk at

---

**CoffeeScript and ES6 transpilers are a decent path for those looking for a better JavaScript, while ambitious projects like ClosureScript and asm.js seem promising at the other end of the abstraction spectrum.**

---

PyCon earlier this year. The biggest threat to JavaScript is its own ubiquity, as it (through asm.js) becomes a compile target for other languages. Perhaps JavaScript will meet the same fate as Java, where the runtime thrives more than the language.



it hard to tell progress from fashion. In this talk, Jed talks about how the interfaces of these libraries and frameworks are evolving, and how we can design the interfaces to survive the inevitable front-end Singularity.

In the shorter term, I worry a bit about fragmentation and the complexity of API surface area. For example, there are so many approaches to async right now, from callbacks to promises to generators to streams to async functions. Hopefully at some point the community will rally around one particular abstraction, but until then, it seems pretty daunting for someone just getting started.

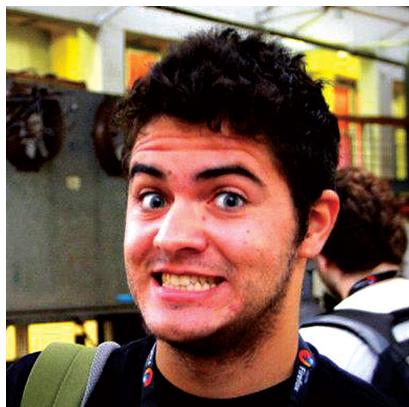
**PayPal™ | Developer**



WARSAW, JULY 12-13

**WIN  
\$100,000 USD**

2014.battlehack.org



## Julian Cheal



A British Ruby/Rails developer, with a penchant for tweed, fine coffee, and homebrewing. When not working for clients, I help organise fun events around the world that teach people to program flying robots. I also occasionally speak at international conferences on the intersection of programming and robotics.

### Talk: Dancing with Robots

Ever wanted to control your own robots, using just code and your imagination? Did you know you can make a robot fly with only 16 lines of JavaScript? With a little Node.js and a variety of hardware controllers we'll see flying robots, and we may even be able to make them... dance! So let's get coding: `drone.doAwesomeStuff();`

## Zeno Rocha

Zeno Rocha used to work as a software developer at Petrobras (the largest company in Latin America) and Globoesporte.com (the most accessed sports website in Brazil), now he's a front-end engineer at Liferay. Besides that, he's a writer at Smashing Magazine, host at Zone Of Front-Enders podcast, member of the Google Developer Expert program and co-founder of BrazilJS Foundation. With only 23 years-old he's one of the top 30 most active

users on GitHub and a reference in Brazil when the subject is front-end development.

### Talk: A future called Web Components

Discover how this new concept formed by five new specifications (Templates, Decorators,Shadow DOM, Custom Elements, HTML Imports) will revolutionize the way we develop and interact on the web.

## Tim Messerschmidt

As a long time mobile and web developer specializing in Android, Tim channels his knowledge and experience as PayPal's Lead Developer Evangelist in Europe. He is very passionate about startups, and serves as a mentor of Seedcamp, Wayra, Startup Wise Guys & Rockstart. In his spare time, he leads and creates training classes in all sorts of Android related topics, and is the author of the Android chapter of the Mobile Developer's Guide to the Galaxy, as well as numerous articles published in magazines like Web&Mobile Developer. His ongoing work on this topic and having an app in Google's Play Store with more than 300,000 downloads makes him a great resource for mobile related questions.



Learn some key findings that PayPal made when working on Node and how we structured the introduction of it into our existing infrastructure. Being more efficient, needing less people working on projects and being way more responsive than classical apps Node and Kraken have become PayPal's favourite stack.

### Talk: Supercharging Node with Kraken

Scaling Node applications from prototyping stacks to production can be very difficult as there is a big lack of conventions and best practices that work across companies and even communities. In this talk Tim will be showcasing his favourite 4 modules of KrakenJS – a thin convention layer on top of Express – which helped PayPal pushing out Node apps.

# RACHEL Andrew

Rachel Andrew is a web developer, writer, speaker and one half of edgeofmyseat.com - the company behind the CMS Perch. When not working on her own product, Rachel likes to help other people transition from consultancy and client services to profitable product businesses - without needing to seek investment or risk everything by "doing a startup".

## Talk: We're Not "Doing a Startup"

You would love to launch your own product but you have a nice life, a family, a mortgage and don't want to buy into putting it all on the line chasing some startup dream. This presentation is about the practical steps developers and designers can take to get from idea to launch – no sleeping in the office required.



# TERO Parviaainen

Tero Parviaainen is an independent software maker with 12 years of enterprise-grade yak shaving under the belt. During this time he has worn many different hats, ranging from consultant to startup CTO. His weapons of choice are JavaScript, Clojure, Ruby, and Java. Tero is the creator and organizer of the Clojure Cup programming competition and author of two books: The self-published "Build Your Own AngularJS" and "Real-time Web Application Development with Vert.x" from Packt Publishing.

## Talk: Build Your Own AngularJS

AngularJS is a powerful JavaScript framework with a steep learning curve. The inner workings of features like directives, transclusion, data binding, and dependency injection are shrouded in mystery. When you're trying to get things done, mystery is not helpful.

This talk is all about demystifying the internals of Angular. We'll take some of the tricky aspects of Angular and build them from the ground up, live on the spot. In the process we'll gain a deep understanding of what makes this framework tick.



Schibsted Tech Polska was established in Poland as a core R&D center for Norwegian biggest media houses. There are over 115 of us now in two locations (Krakow and Gdansk), but we are still missing YOU. If you have an unparalleled passion for front end software technology and a reputation for site performance and optimization focus than we would love to hear from you!



In Schibsted Tech Polska (STP) we have freedom to do great work and the challenge of using our skills and ambitions to drive the future of the web. We have an inspiring culture which provides room for growth and innovation. We work for Scandinavian market which is very unique and special also ensuring us to have stable jobs in a very supportive environment.

In STP you can voice your ideas and affect transformation. We work in small and stable SCRUM teams, daily cooperate with friends from Norway and Sweden, travel to Scandinavia and most importantly we do develop the best known and most visited websites in the whole Scandinavian market.

# DAY 2

## SCHEDULE

Thursday, May 8

<b>08:00</b>	Registration and breakfast
<b>09:30</b>	Lea Verou The Chroma Zone: Engineering Color on the Web
<b>10:15</b>	Coffee Break
<b>10:30</b>	Brian Ford Zone.js
<b>11:05</b>	Kyle Simpson CSS: Yawn to Yay!
<b>11:40</b>	Coffee Break
<b>12:10</b>	Sara Vieira CSS as a programming language
<b>12:45</b>	Martin Ringlein The ROI on Front-End Experimentation
<b>13:20</b>	Lunch Break
<b>14:50</b>	Alicia Liu Slaying the Dragon: Refactoring CSS for Maintainability
<b>15:25</b>	Patrick Hamann Breaking news at 1000ms
<b>16:00</b>	Coffee Break
<b>16:30</b>	Discussion Panel JavaScript Frameworks – The Good, The Bad, and The Beautiful
<b>17:30</b>	End of day
<b>19:30</b>	Party

# KYLE Simpson

Kyle Simpson is an Open Web Evangelist from Austin, TX. He's passionate about JavaScript, HTML5, real-time / peer-to-peer communications, and web performance. Otherwise, he's probably bored by it. Kyle is an author, workshop trainer, tech speaker, and avid OSS community member.

## Talk: CSS: Yawn to Yay!

Maybe you think moving from static CSS files to (mostly-static) CSS preprocessing was a huge revolution for your app's styling. I've got news for you. You're still stuck in quicksand.

If you want more out of your CSS, it should be dynamically templated, not just build-processed. Render the CSS on-the-fly, responding to run-time conditions, either on the server or in the browser. And that's just the beginning. Imagine modifying CSS structure and rule-relationships in process, re-compiling then re-rendering? Yep.

Are you still excited because your CSS automatically re-compiles on save? CSS.past. Open your eyes to the next big things in CSS. You don't need more fancy features and coding constructs IN your CSS file, you just need super-charged CSS tools. Welcome to CSS templating.

We haven't scratched the surface of what CSS can do for us yet. It's time we get scratching.



# Sara Vieira

Sara is a front end developer from Portugal. She's one of those few people in the world who love what they do, wanna do it more and even work on their free time.

When she isn't in code she's a total TV Show addict and everything she says is a TV Show reference.

## Talk: CSS as a programming language

CSS will be sooner or later its own preprocessor, we have amazing functionalities in CSS that allow us to go further with it than we ever did.

In this talk Sara walks you throughout these new functionalities that will make you look at CSS in a different light.



Alicia Liu is a full-stack web and mobile engineer, currently with a focus on front-end at Lift. She has been building web apps since IE 5.5, and honing her craft in early-stage startups since 2007. Alicia writes at [alicialiu.net/writing](http://alicialiu.net/writing) and tweets @aliciatweet.

## Talk: Slaying the Dragon: Refactoring CSS for Maintainability

Refactoring is a practice that is often overlooked when it comes to CSS. As a result, CSS seems to just grow and grow like a hydra, until no one knows for sure which styles apply where, or whether many rules are still used at all. We'll cover methods for refactoring CSS including code organization and naming, dealing with incremental change, and special considerations for responsive styles. Rewrites are costly and high risk, refactor instead!



BECAUSE  
**CSS**  
**IS DYNAMIC,**

WE OFTEN END UP

WITH CONSIDERABLY  
**MORE USEFUL**  
**FEATURES**  
THAN WHAT  
**PREPROCESSORS**  
**CAN OFFER.**

---

Interview with Lea Verou

## INTERVIEW WITH LEA VEROU | 2ND DAY

Over last few years, Lea has spoken around the globe and made an impact on developers worldwide. We asked her about the future of CSS.

Should CSS become a programming language with features like variables, mixins and functions used in tools like Sass?

It's not a matter of if, it's a matter of when, and with what syntax. The CSS WG is already using preprocessors as inspiration for new features, similarly to how TC-39 uses CoffeeScript as inspiration for the next generation of ECMAScript. After all, preprocessors are an excellent testing ground for new features without the usual risks of experimental implementations in browsers (= getting stuck with one, because authors used it in production too much).

However, unlike preprocessors which only run once, generate output and be done with it, CSS has to be constantly re-evaluated based on user actions or DOM mutations, so the performance considerations are different.

However, exactly because CSS is dynamic, we often end up with considerably more useful features than what preprocessors can offer. For example, calc() in CSS is much more powerful than calculations in preprocessors, because it's aware of the browser environment, so it can resolve em values, viewport units, percentages etc. Similarly, custom properties for cascading variables are much more powerful than preprocessor variables because they are dynamic, which means that you can change the variable with JS and see every value that depends on it be updated as well. Take that, preprocessors!

However, because CSS is baked into browsers, it means that we need to think of everything we add very, very carefully, because once authors start using it, it cannot be changed, ever, as it will break existing websites. This leads to the impression that the CSS WG is "slow" in adding these highly requested features compared to the development speed of preprocessors & other tooling. Also, whereas in preprocessors one implementation is all that's needed, in browsers you need to wait for at least some degree of interoperability and that's \*after\* the specification becomes stable-ish. This also gives a false impression of sluggishness, where in fact, it's just different things that need to be done.

How do you manage to travel so often while giving top-rated talks for people from different countries and cultural backgrounds? How do you fulfill their expectations?

I think that developers around the world have more in common than what separates us. Even though I've talked to audiences from completely different cultural backgrounds, I can't say there was much of a difference in expectations. Actually, in my experience, difference between technical backgrounds (e.g. designers vs

programmers or advanced devs vs beginner devs) accounts to a much bigger difference in perspectives & expectations than cultural background ever could. To account for that, I try to include bits in my talk for every kind of technical background, instead of targeting only one specific kind of professional for its entire length.

That said, there is an important difference that different cultural backgrounds can cause and it's the so-called "language barrier" in countries where the main language is not English. Frankly, I'm still not quite sure if there's a good solution for that. I've seen simultaneous translation being used to mitigate the problem, but there is often a lot that's "lost in translation", especially jokes. Another solution I've heard from other speakers is to try and lower the barrier by using "simpler" words, but that sounds a bit patronizing to me. I'm not a native English speaker myself, and I hate it when I realize that somebody is trying to dumb down their vocabulary when talking to me.

If you had two more days in every week, what would you do with them?

I'd love to work more on my open source projects and do more research. I have so many ideas of things I want to make at some point, but never seem to find the time anymore. Sometimes I even end up not reviewing pull requests on Github for months, which makes me feel really guilty for the time its sender spent on it. I'd also want to blog more. Sometimes while coding something I discover a new technique that might be worth blogging about and I keep postponing to blog about it. Eventually I forget, until I read someone else's blog post about it months or years later. Lastly, I'd love to have more time to report browser bugs. I used to do it a lot more, and now I rarely find myself being able to take the 5 minutes required to make a testcase and report the bugs I find. In fact, thinking about it again, I think two extra days in a week wouldn't be enough, can I haz an extra week instead? :)

# LEA Verou

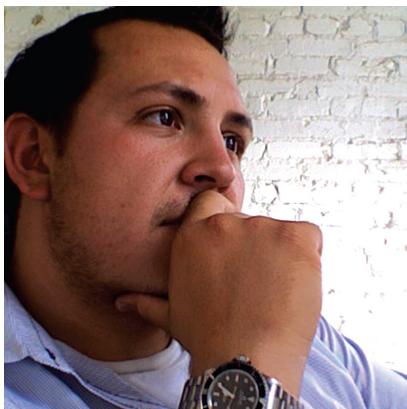


Lea previously worked as a Developer Advocate at W3C and currently spends her days writing & designing her first book ("CSS Secrets"), published with O'Reilly in 2014. She has a long-standing passion for open web standards, which she fulfills by researching new ways to use them, blogging, speaking, writing, and coding popular open source projects to help fellow developers. She is an Invited Expert in the CSS Working Group, which architects the language itself. Although she holds a BSc in Computer Science, Lea is one of the few misfits who love code and design equally.

## Talk: The Chroma Zone: Engineering Color on the Web

We use color every day, but how well do we really understand it? More often than not, we are merely scratching the surface of a large and complicated discipline. In this dynamic session, we will scratch a little deeper, and you will be surprised at how deep the colorful rabbit hole goes. How does color work on our screens? What's the difference between color models and color spaces? Which existing features of CSS Color are we underutilizing? What's in store for CSS Color level 4? This is not a design talk, it's a technical talk about the inner workings of one of the most important design aspects. Whether you identify as a designer or a developer, you will walk out of this session with a newfound confidence about anything color related.

2ND DAY



# Martin Ringlein

An award-winning designer, developer, entrepreneur and angel investor; Martin most recently was at The White House as a Presidential Innovation Fellow. Prior, Martin exited his first start-up through an acquisition deal with Twitter where he joined Twitter as their first Design Manager, with a primary focus on designing the design team. Martin studied Business Management at Columbia Business School and Yale School of Management. Currently, Martin is founder and CEO of nvite.

Martin's influence and success ranges from designing and directing major strategic interactive marketing initiatives with organizations large and small such as Apple, Google, Adobe, British Telecom, Mozilla and Oracle. Initiatives including the redesign of Ogilvy, Mashable, NASDAQ;

developing the initial mobile strategies for The Washington Post, POLITICO, National Journal, Kiplinger, C-SPAN, PUMA, OPOWER; and even designing the original brand identity for start-ups such as LivingSocial.

## Talk: The ROI on Front-End Experimentation

Based on his work at includ, Twitter and The White House, Marty will share how early experiments over the past decade have shaped our industry from the first experimentations with CSS for layout, liquid layouts impact on responsive design, to transforming WordPress from a blogging tool into a CMS. The results of experimentatioin have fundamentally shifted how Apple develops products, how Oracle thinks about mobile, how Mashable is run, and how Google is inspired by design.



# Patrick HAMANN

Patrick is a senior client-side developer at the Guardian in London, where amongst other things, he is helping to engineer the next generation of their web platform. When not speaking or ranting about performance or CSS architecture on twitter he enjoys spends his spare time discovering new food and craft beer.

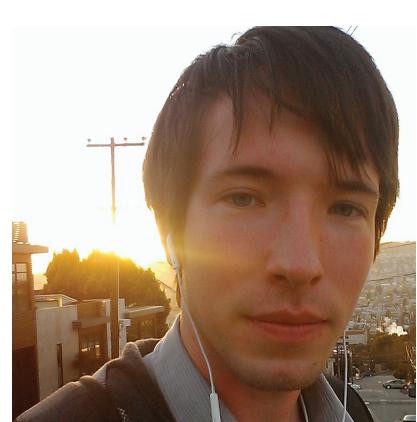
## Talk: Breaking news at 1000ms

For a website to feel instant, a perceptible response to user input must be provided within hundreds of milliseconds.

This perceived speed of your website relies heavily on the browser being able to paint to your user's screen. For this they must construct a "render tree" which consists of the DOM and the often forgotten CSSOM. The critical path which the browser takes to gather this information is the only thing standing between your server and the user's screen.

Using new research and real world examples, Patrick will cover a range of techniques – from the controversial to bleeding edge – the Guardian are using to make their next generation website load as fast as possible, and ultimately breaking the news to the user within 1000ms.

During the talk you will discover why performance matters, what are the common performance bottlenecks in the browser from networking to painting and learn how to best optimise and monitor each stage of the critical path to create fast, jank free websites.



# BRIAN FORD

Brian Ford is a software engineer working on the AngularJS core team at Google.

## Talk: Zone.js

Zones are a powerful new way to deal with asynchronous JavaScript in both the browser and node. Learn how zones simplify error handling, profiling, debugging, and more.

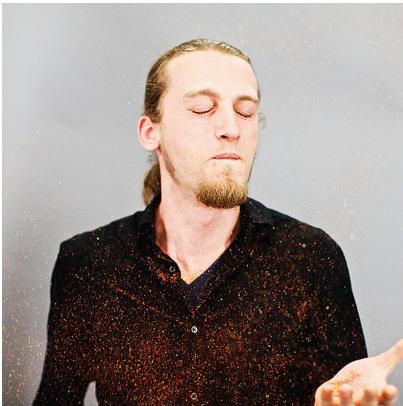
Friday, May 9

08:30	Registration and Breakfast
10:00	Horia Dragomir The Mobile Development We Leave Behind
10:35	Ana Tudor Pure CSS 3D Solids
11:10	Coffee Break
11:40	Mary Rose Cook Mary live-codes a JavaScript game from scratch
12:15	Mariusz Nowak Full Stack JavaScript Applications, Reactive Way
12:50	Lunch Break
14:20	Kornel Lesiński Modern Image Compressors
14:55	Max Ogden Browserify - Using Node In Browsers
15:30	Coffee Break
16:00	Garann Means Progressive Enhancement for JS Apps
16:35	Gunnar Bittersmann Patterns & math with Sass
17:10	Close
17:30	End of Conference

DAY

SCHEDULE

3

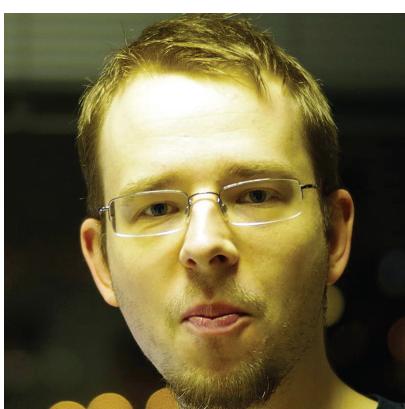


## HORIA Dragomir

UI Developer, Hungry and Foolish. He has worked in the empty spaces between application programming, product design, game development, data analysis, interface optimisation and project management. He likes to spend his evenings wearing users' shoes, taking long walks on the beach.

### Talk: The Mobile Development We Leave Behind

Let me walk you through the influence our past has on us while developing for mobile. Let me walk you through what we are leaving behind for future developers. And let me explain to you how mobile is a lie.



# MARIUSZ NOWAK

BWeb applications full-stack programmer, with JavaScript as a main programming language since 2007. Three years ago Mariusz abandoned ES3 based, legacy engines world, and started exploration of latest standards focusing strictly on development for modern engines.



Born and raised in Warsaw, Mariusz is currently working as an UNCTAD consultant on Open Web solutions of which long-term goal is reduction of

grey market in developing countries. Most of his work is open sourced and can be found on his Github profile.

### Talk: Full Stack JavaScript Applications, Reactive Way

I will expose important details and ideas behind fully reactive stack that clarified through out my journey of working strictly with modern engines (ES5+ world). It'll be about JavaScript architecture that takes full advantage of latest ECMAScript and DOM standards and which is largely composed from environment agnostic modules which puts code share between client and server to next level.

Part of the talk will be dedicated to presentation of in-memory database engine written in JavaScript,

that works on both client and server-side, assures offline synchronization and two-way bindings for view components.

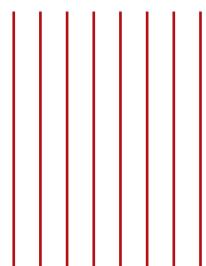
Last but not least, mentioned stack will be showcased with real world project built for UNCTAD, and "How come, it's fully functional also in IE6?" questions will be answered.

## Kornel Lesiński

Developer at FT Labs — an emerging web technologies team at the Financial Times. Independent image compression researcher. Invited expert in W3C HTML WG and WHATWG participant.

### Talk: Modern Image Compressors

Breathing new life into Web-compatible image formats. You will learn about new image compressors and supporting tools that enable delivering images faster, with higher quality and with alpha transparency for graphically richer websites, while maintaining backwards compatibility with all current browsers and staying within limits of mobile networks.



# m a x OGDEN



# Gunnar Bittersmann

Having programming and UX backgrounds, Gunnar's passion is where both meet: at the frontend. Building web pages (\$whatever first ;-)), he aims at a better understanding between humans and machines. He translates articles on the W3C i18n website, helps organizing the World Usability Day Berlin and struggles not to forget how to play the guitar. And he loves to come back to the place that had been his 2nd hometown for a while: Warsaw.

Max Ogden is a JavaScript programmer who lives in Oakland, CA, home of the Node.js project. Some previous projects include Code for America, JavaScript for Cats, TacoConf and most recently Voxel.js, a WebGL voxel rendering engine.

## Talk: Browserify - Using Node In Browsers

Lots of people think that since npm has 'Node' in the name that it must only be used for server side JS modules. This is completely untrue! npm actually stands for Node Packaged Modules, e.g. modules that Node packages together for you. This talk will cover how to use the node module system in the browser using Browserify and show off lots of cool demos using <http://requirebin.com>.

# GARANN Means

Garann has been making websites since the early days. She spent time as a full-stack developer writing Java and .NET before refocusing on the front-end and, eventually, JavaScript. In addition to her full-time work, she's an acquisitions scout for A List Apart, and is always looking to meet developers who also share her passion for writing and sharing their knowledge.

## Talk: Progressive Enhancement for JS Apps

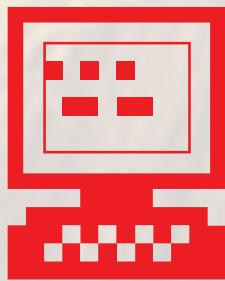
When progressive enhancement—offering a basic version of a feature that all clients can use as a default—was introduced as a concept, JavaScript applications seemed as relevant as flying cars. As JS became more powerful, it seemed we'd reach a point where we could forget about PE entirely. For the things we originally used it for, we now have rock-solid libraries and polyfills to provide

abstractions that make PE easy. But as JS has advanced, we've started writing things that can't be polyfilled. We know now how to progressively enhance widgets and user interactions. We'll talk about how we progressively enhance entire applications, and why it's potentially more important than ever that we do so.

## Talk: Patterns & math with Sass

From bicolores (Polkska) over crosses to stars and stripes – using flags as leitmotif we'll build patterns with a single HTML element and CSS gradients. As much as about the geometrical patterns it's about the patterns in the code. And we'll calculate square roots like the ancient Babylonians – in Sass. A short trip into geometry. Bring your rulers and compasses!





# HACKER SCHOOL

---

# DESIGNED

---

**AROUND THE IDEA**

that

**PROGRAMMING**  
is intrinsically  
**MOTIVATING**

Interview with Mary Rose Cook



# MARY ROSE COOK

Mary teaches programming on her daily basis. She holds a very responsible role and tells us more about her professional life and how she perceives teaching technology.

## Why do you like to teach programming?

I don't really think of what I do at Hacker School as teaching. I think of it as helping people learn. Sometimes, that means demonstrating a programming language feature that I think is cool. Sometimes, that means pair programming with someone on their project. Sometimes, that means running a workshop where we explore a topic like functional programming.

---

I like speaking at conferences because it gives me the push to dive deep into a topic I am interested in.

---

I have learnt way more from other people than they have learnt with me. By working with other people, I've used way more technologies than I would have on my own. I've learnt to understand unfamiliar code fast. I've read a ton of code and seen many new techniques. I've seen others people's workflows and learnt great tips. Most importantly, I've often discovered when trying to explain something that I don't understand it fully myself. This has led to me diving deeper into the topic and coming out understanding it way better.

## How do you make programming a fun experience?

In this regard, my job is kind of easy. People come to Hacker School because they are super into programming. Hacker School is designed around the idea that programming is intrinsically motivating. That said, I do contribute a bit towards helping people find the fun in programming.

First, I sometimes help people find projects they are excited about. But my only input is to talk through what they want to learn. Second, I spend time working on my own projects. This contributes to the culture of Hacker Schoolers



being inspired by the excitement of other Hacker Schoolers.

You must meet a lot of advanced developers at different events. What do you and your peers do in order to constantly improve and learn new things?

The learners I really admire are the ones that kick their own ass. That is a phrase coined by my friend and colleague, Alan. It means to be constantly turning towards the subjects you find hardest and scariest. I try to do this, though I am not as good at it as Alan.

I am incredibly lucky. I get to spend most of my time learning things that interest me. So, recently, I have written a Lisp interpreter in JavaScript, and an iPad synthesiser in ClojureScript, and a game framework in JavaScript. These projects grew out of the desire to understand how a type of software works underneath, or the desire to understand a programming technique or new technology.

I like going to conferences because the talks give me quick overviews of things I haven't heard of. I like going to conferences because I meet people who are working on cool projects. I like speaking at conferences because it gives me the push to dive deep into a topic I am interested in. conferences because I meet people who are working on cool projects. I like speaking at conferences because it gives me the push to dive deep into a topic I am interested in.

Mary Rose Cook writes code, makes music, works at Hacker School and lives in New York City. She's made two games: Pistol Slut and Empty Black. She's made a micro framework for JavaScript games: Coquette. She's made a programming language for children: Isla. She's spoken at JSConf, BerlinJS, BrooklynJS and EmpireJS.

## Talk:

## Mary live-codes a JavaScript game from scratch

When I made my first game, I was scared of writing graphics code and dealing with browser quirks and player input events. So, I used a game framework to handle that stuff for me. Later, I discovered that stuff is not so scary. I will live-code an action game from scratch without using any libraries. We will cover keyboard input, graphics, collision detection and sound.

**it KONTRAKT**

Your manager in the IT market



IT KONTRAKT SPECIALIZES IN OUTSOURCED IT PERSONNEL. WE HAVE MANY YEARS OF EXPERIENCE IN BUILDING EXPERT IT TEAMS FOR THE MOST PRESTIGIOUS IT PROJECTS IN ALL MAJOR SECTORS OF THE ECONOMY.

The IT Kontrakt team consists of 500 professionals, working on IT projects across the globe. We link IT experts with exciting projects in the labor market, and our teams enjoy competitive pay and great working environments. We put great care into developing each associate by organizing frequent trainings, workshops, and conferences.

## JOIN US

If you are interested in cooperating with IT Kontrakt, please send your CV to [rekrytacja@itkontrakt.pl](mailto:rekrytacja@itkontrakt.pl) or fill in the application form on our website [www.itkontrakt.pl](http://www.itkontrakt.pl).

# ANA Tudor

Loves maths, especially geometry. Enjoys playing with code. Passionate about experimenting and learning new things. Fascinated by astrophysics and science in general. Huge fan of technological advance and its applications in all fields. Shows an interest in motor sports, drawing, classic cartoons, rock music, cuddling toys and animals with sharp claws and big teeth. Dreams about owning a real tiger.



## Talk: Pure CSS 3 Solids

- Now that's cool! Haven't experimented with WebGL really yet!
- Actually, that's pure CSS.

The above is from an actual conversation I recently had. WebGL is great, but really cool 3D can also be created using nothing but HTML and CSS.

How? Well, obviously using CSS 3D transforms. Chaining them and applying them on nested elements allows us to create more than the basic cube example everyone is probably familiar with by now. We can recreate regular polyhedra as well as Archimedean, Catalan or Johnson solids.

We can also make these 3D shapes transform into others, either by breaking them into smaller ones which then recombine into something else or by making their simple polygonal faces morph into more complex ones. We can even use CSS gradients for shading.

Best thing about this? Preprocessors introduce logic and computation in CSS, keeping these demos extremely customizable while writing little code and all the Maths reduces to a right triangle and a simple equality:  $x = y/z$ .

Introducing the world's first attendee-focused events platform.

You're invited to the private beta:

Use access code **warsaw**

Try it today @ [nvite.com](http://nvite.com)

**nvite.**

Your next event is going to be epic!



Max Ogden

# INTERVIEW



Mariusz Nowak



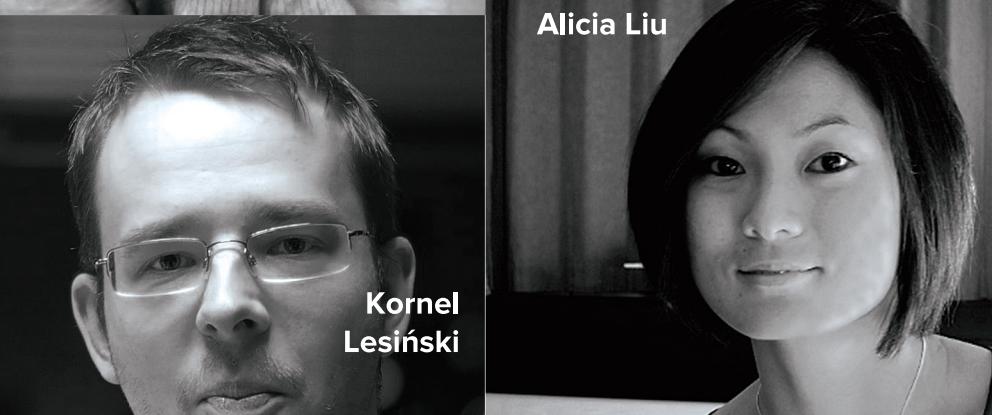
Sara Vieira

Zeno  
Rocha

Garann Means



Alicia Liu

Kornel  
Lesiński

We asked some of the speakers to share their predictions and thoughts on upcoming trends in front-end technologies, web development, and the web dev market. See what they have to say!

**With ECMAScript 6 around the corner, do we still need tools like CoffeeScript?**

**MARIUSZ NOWAK:** When speaking of CoffeeScript I think it'll be dead shortly after adoption of ES6 by major engines, as main issues that CS aimed to solve on top of ES5, were solved in well thought and better manner in ES6. There would be no point to use CS over something that's better and provided natively. Sure, maybe there would be some new version of CS, that would adapt to new environment, but I think it'll never be as tempting as when CS was put against ES5.

Concerning any other dialects compiled to JS, as long as we have different opinions on how language should be designed as long new compilers would arise. Whether they would be worth a try, would depend strictly on opinions and individual needs.

Personally I don't see a need for that, ES6/7 is great enough to be used on its own, anything on top even if it improves things also becomes an additional baggage, and journey is easier when you have less to carry.

**MAX OGDEN:** I actually don't pay attention to what is going on in ES6 or CoffeeScript anymore, because I know enough about both to know that they won't make a significant impact into the kinds of things I can do with JavaScript. I get very excited by new low level capabilities (ServiceWorkers, FileWriter, WebRTC, IndexedDB, etc).

Things like CoffeeScript and Promises are very high level, and I think it's unfortunate that a (perceived) majority time is spent by people trying to come up with the ultimate language syntax instead of focusing on limitations of the web as a low level platform.

**What are the biggest challenges facing web development in the next 5 years?**

**ZENO ROCHA:** One of the biggest challenges is designing and developing applications for dozens of different platforms. We now have desktop computers, tablets, mobile phones, and televisions. Imagine how many different devices we'll have in 5 years. We're quite comfortable

displaying things for different resolutions, but they are all box-shaped screens in the end. We'll soon have interfaces for watches that are rounded, interfaces with less colors and a completely different user experience. Things are getting more challenging, and this is good because it opens doors for creative people to solve new problems.

**KYLE SIMPSON:** The biggest change to web development will be that we stop designing for browsers and start building for peer-to-peer connected devices. The client-server model is going to give way to a de-centralized peer-grid web, and that will drastically change the types of services we build and how we present it to users.

### What are the trends in market demand for front-end developers?

**ZENO ROCHA:** There are lots of trends going on right now. On the other hand, there are couple software engineering challenges that the front-end environment still needs to improve. Encapsulation and code reuse are one of those things. The good news is that Web Components is coming as a solution to address those problems and I really believe that this is going to change the way we develop front-end code.

### Is responsive web design the only right answer for web development these days?

**ALICIA LIU:** The term "responsive" web design is relatively new, coined in 2010, but the concept is not. HTML has always had amazing text reflow capabilities, so you didn't have to do anything special to make your web content fit inside the width of any browser. As the web matured, web design turned to web "apps" and required more complex layouts inherited from desktop applications. At the time, the only way to create these layouts was with tables, which prevented text from reflowing, and sites became very fixed.

In the last few years, we've finally come around again, as CSS techniques have evolved and support for CSS media queries have been built into browsers, to once again support varying device sizes in one design. If your audience is the general web, you can assume your site could be accessed from devices of all different sizes, and you should make sure your site is, at the very least, usable at varying sizes. Call that responsive web design, but to me, it's just part of designing for the web.

**SARA VIEIRA:** Like everything in web development there is never a single right answer for all websites. Of course you should have a website that looks great on mobile. That is a necessity now-a-days, but depending on your website you could go responsive or create a website design entirely for mobile. It all depends on how much you want you want to focus on your mobile audience.

### Is JavaScript a reasonable alternative for ObjC and Java for advanced mobile apps yet?

**KORNEL LESIŃSKI:** In many cases it's a great solution already. Our team at Financial Times was

able to create an offline-capable, smooth and responsive web app that works on a wide range of devices. We don't need these annoying "download our app" banners — you can just open [app.ft.com](#).

The platform isn't perfect yet, but it's improving. Old browsers fade away, and thanks to improvements in hardware and software it's getting easier to get good performance. These days Firefox can smoothly run a WebGL port of Unreal Engine, in full HD, on a mobile phone — how awesome is that?

However, I don't think the web should go too far trying to be a complete alternative to native applications. So far nobody has found a way to have unrestricted capabilities, good security and openness at the same time. We have curated native platforms that are safe, we have open native platforms that blame users for not understanding tricky technical questions that lead to installing malware, and we have the Web that—at a cost of limited capabilities—is safe and completely open. I would like the web to remain safe and open.

### Do you agree with a statement saying that Google wants a monopoly in the browser market?

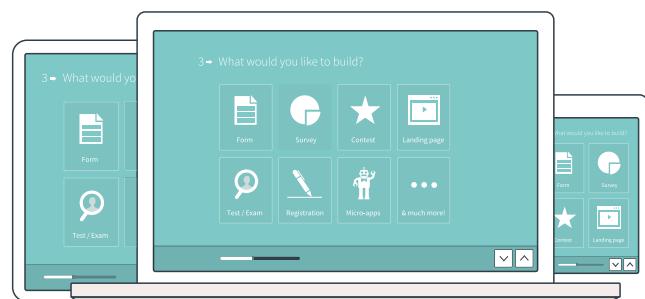
**GARANN MEANS:** While I'm sure they'd like a monopoly, they need an open web as much as any organization creating tools for it does. They could choose to use Chrome-specific features on their own sites and services, but it's unlikely that outside developers would use those things and limit their users to a single browser. Of course there will always be developers who only test their work in one browser, but that's hardly a best practice. I think Google probably recognizes that their goals are best served by implementing standards quickly, putting effort into their developer tools, and in doing so pushing other browser vendors to do the same so that Google's other products work well no matter the browser used to access them.

### Are web components the next big thing in a web development?

**GARANN MEANS:** In terms of speeding up development and making it easier, yes. More people began writing front-end code when IDEs began to include components for complex markup and styling, and the same

happened with jQuery plugins. In both JavaScript and CSS, tools that allow developers to modularize complex code are currently very popular. However, I think the idea with the capacity to change what we develop (not just how quickly we do it) probably looks more like the offline web.

Typeform makes asking questions  
easy, human & beautiful



Get 3 months **FREE** when you upgrade to our **PRO** plan

USE PROMO CODE

**FRONTT\_3months**

[WWW.TYPEFORM.COM](http://WWW.TYPEFORM.COM)



## PONSORS

**Booking.com**

**GOLD SPONSOR**

Booking.com is part of Priceline.com (Nasdaq:PCLN) the World's #1 Online Hotel Reservations Company attracting over 30 million unique visitors every month and offering nearly 170,000 hotels worldwide. Due to the growth of our IT department we need experts in UX, HTML, CSS, JavaScript, Ajax, jQuery, performance, and mobile! Would you like to work in an international, results driven, fun & dynamic work environment? Join us! We are based in Amsterdam, NL.

[workingatbooking.com](http://workingatbooking.com)

nvite is a platform for planning, attending and engaging with events and we're redesigning and reimagining the event experience from the ground-up!

**GOLD SPONSOR**

[nvite.com](http://nvite.com)

**nvite.**

**SILVER SPONSOR**

— schibsted.pl/kariera —

Schibsted Tech Polska is a technological center for innovation and software development located in Krakow at Armii Krajowej 28. We are part of bigger international media group called Schibsted Media Group. In Krakow we design and develop information services mainly for our Norwegian and Swedish partners. We work in AGILE teams and often travel to Scandinavia.

We deliver high-quality solutions based on the latest web, mobile and smart TV technologies, employing lightweight software development methodologies for our Scandinavian partners.

There are over 110 of us now. We hope to hear from you soon since there is some office space we would like to fill with the coolest frontend lovers, java geeks, .NET maniacs, mobile apps masters and PHP gurus or other passionate developers! Also, we hope to have more colleagues to play soccer table with J or swing on our hammock at the roof top terrace.

**SILVER SPONSOR**

— itkontrakt.pl —

IT Kontrakt specializes in outsourcing of IT personnel. We have many years of experience in creating IT expert teams for the most prestigious IT projects in the key sectors of economy. Thanks to the high effectiveness of operating and partner approach towards the Customers, we gained the confidence of the largest banks, financial, insurance, computer and telecommunication companies.

TELL US ABOUT YOUR EXPECTATIONS. WE WILL MEET THEM AND OFFER YOU AN INTERESTING JOB AND AN OPPORTUNITY FOR CONTINUOUS PROFESSIONAL DEVELOPMENT.

IT Kontrakt team consists of more than 470 IT professionals implementing IT projects throughout the country and abroad. We perform a role of Manager for our IT experts in the labour market, we offer them the best projects, and thus – we offer satisfying financial and social conditions. We put emphasis on assurance of continuous development for our associates, by organizing conferences and trainings. The company policy is the patronage over each of the specialists. A patron from IT Kontrakt introduces the new specialist to work and takes care of all formal issues connected with cooperation.

## JOIN US

If you are interested in cooperation with IT Kontrakt, please send us your application documents to: [rekrutacja@itkontrakt.pl](mailto:rekrutacja@itkontrakt.pl) or fill in the application form on our website.

**SILVER SPONSOR**

— developer.paypal.com —

The PayPal Developer program is the face of technology innovation at PayPal. Using open source foundations in their technology stacks, PayPal is constructing new breeds of identity systems, API standards, and payment platforms to support the next generation of disruptive technology.

**SILVER SPONSOR**

Ask awesomely

— typeform.com —

Typeform makes it easy to build and share beautifully designed online forms. We call them typeforms, and they mix human creativity with the power of modern, cross-device web technology to create new ways of asking questions online.

Typeform is already used by people all over the world, from a myriad of backgrounds. We're helping businesses hire the best candidates. Creatives are evolving their craft by receiving great feedback. Professionals are getting the information they need, when they need it. In other words: people are getting to know each other better, and it's awesome!

**SILVER SPONSOR**

— wikia.com —

Create. Collaborate. Be original. 100 Million Fans Strong and ranked a Top 10 Social Network by Nielsen, Wikia operates the world's largest network of collaboratively published games, entertainment, and lifestyle content on the web. Our knowledgeable and devoted fans create hundreds of wikias every day on a trusted and customizable platform designed to help people share what they know and love.

Currently, we maintain over 370,000 wikias across virtually all aspects of popular culture: from console and mobile games, TV, music, and movies to sports, books, fashion, and current events.

Wikia's fan base continues to fuel consistent growth with over 1.5 billion global monthly page views from 223 countries in over 100 languages. Our traffic has grown consistently, 30% year over year on the web and 347% year over year on mobile, including downloads of our mobile applications, MyWikia App and Wikia Game Guides. Over 34% of Wikia's traffic comes exclusively from mobile visits.

Wikia's philosophy and publishing platform is designed to unleash the power of knowledge as influence, enable personal and collective creativity and distribute rich, original content.

**SILVER SPONSOR****mozilla**[mozilla.com](http://mozilla.com)

Mozilla is a global, nonprofit organization dedicated to making the Web better. We emphasize principle over profit, and believe that the Web is a shared public resource to be cared for, not a commodity to be sold. We work with a worldwide community to create open source products like Mozilla Firefox, and to innovate for the benefit of the individual and the betterment of the Web. The result is great products built by passionate people and better choices for everyone.

**SUPPORTER**[qbaka.com](http://qbaka.com)

Qbaka is frontend error monitoring and user analytics. We help web services provide best user experience by smart javascript error tracking, user events and ajax monitoring, problem impact evaluation, combined with powerful analytics engine in the cloud. With just 1 minute setup we provide instant, measurable value. Be aware of client-side problems, save time for your developers and support, give users happy error-free experience with the help of Qbaka. We've got your front(end) covered!

**SUPPORTER**[informit.com/usergroupwelcome](http://informit.com/usergroupwelcome)

InformIT is the online presence of the family of information technology publishers and brands of Pearson, the world's largest learning company. InformIT is home to the leading technology publishing imprints Addison-Wesley Professional, Cisco Press, IBM Press, Pearson IT Certification, Prentice Hall Professional, QUE, and Sams. We also offer products geared toward the professional business audience from our publishing partner FT Press. Here you will gain access to trusted and quality content and resources from the authors, creators, innovators, and leaders of technology and business. Whether you're looking for a book on a new technology, a helpful article, timely newsletters or access to the Safari Books Online digital library, InformIT has a solution for you.

**SUPPORTER****UX Passion**
[uxpassion.com](http://uxpassion.com)  
[walloftweets.net](http://walloftweets.net)

The Front-Trends twitterwall is brought to you by UX Passion, a full service UX design and usability firm with global reach. Wall of Tweets by UX Passion is a world leader in twitterwall solutions for conferences, parties, product launches, festivals, weddings, events, etc.

UX Passion, headquartered in Croatia, takes pride in the fact that some of their clients are big names from the Fortune 500 while some are startups, small and agile. What is constant, is their passion for delivering extraordinary service and build great and valuable user experiences for customers and users. Over four hundred million users see, touch, feel and enjoy UX Passion's work worldwide.

Clients include: Navigon, Microsoft, World Economic Forum, RealNetworks, T-Mobile, Verizon, Sprint, IBM and more.

**SUPPORTER****balsamiq®**[balsamiq.com](http://balsamiq.com)

Balsamiq is the maker of Mockups, the rapid wireframing software that combines the comfort and simplicity of paper sketching with the power of a digital tool, so your work is easier to share, modify, and get honest feedback on. Mockups look like sketches, so stakeholders won't get distracted by little details, and can focus on what's important instead.

Mockups is available for the Desktop (Windows, Mac, Linux), as a hosted web service, or as a plugin for Confluence & JIRA.

**SUPPORTER****typecast**[typecast.com](http://typecast.com)

Design for the reader by putting type first.

The web is 90% text, so getting your typography right matters. Typecast lets you design working prototypes using simple visual controls. Test, compare and style over 100,000 web fonts from top providers to find the perfect fit for your project. Avoid nasty surprises at build by checking for readability, rendering and beauty as you work in the browser. Cut down the back and forth with devs by handing over production-ready HTML & CSS that itemizes each of your design decisions. Share designs with clients and colleagues, and quickly test your designs across devices. Try Typecast now and start making the web a more beautiful place.

**SUPPORTER**[rosenfeldmedia.com](http://rosenfeldmedia.com)

Rosenfeld Media provides deep user experience design expertise in the formats you need: 18 books, over 50 workshops, and consulting from dozens of the field's top experts.

**SUPPORTER****axure**[axure.com](http://axure.com)

Axure is a wireframe software that lets you generate clickable HTML prototypes that can be viewed in any web browser. Use Axure to create lo-fi sketchy mockups or hi-fi prototypes with interactive elements like carousels, tab control, and dynamic error messages. Axure is available for Mac and PC, with pricing from \$289 to \$589.

