

29 PAGES OF EXPERT TUTORIALS

FREE 120 MINS OF CSS VIDEO

web designer™

HTML5 CSS3 jQuery WordPress

www.webdesignermag.co.uk

CREATE CUSTOM PLAYLISTS

A LOOK INSIDE THE LAST.FM API



SUPERCHARGE WEB VIDEO

GO INTERACTIVE WITH POPCORN.JS

DESIGN

UNIQUE CSS MENUS

CODE SMARTER NAVIGATION

LIKE A PRO
CODE

INSTANT WEB APPS

FAST BUILDS WITH ELECTRON FRAMEWORK

ANIMATED TEXT

USE CSS AND JS TO CODE POP-UP TEXT

LIKE AN EXPERT
20 HTML5 TIPS, TRICKS & TECHNIQUES TO BUILD BETTER WEB EXPERIENCES

ON-SCROLL EFFECTS

ADD A FULLPAGE SHRINK ANIMATION

IS YOUR CONTENT KING?
7-STEP PLAN

54

DESIGNER RESOURCES

WEB TEMPLATES, FONTS, BACKGROUNDS & ACTIONS

ip DigitalEdition
GreatDigitalMags.com
IMAGINE PUBLISHING ISSUE 237

GITHUB UNLEASHED

THE 20 BEST PROJECTS YOU NEED TO KNOW





Reliable UK Hosting
from Specialists.

24/7 UK Support • ISO 27001 Certified • Free Migrations

Managed Hosting • Cloud Hosting • Dedicated Servers

SUPREME HOSTING.
SUPREME SUPPORT.

www.CWCS.co.uk

T: 0115 740 1234 • E: info@cwcs.co.uk • W: www.CWCS.co.uk

Welcome to the issue

THE WEB DESIGNER MISSION

“To be the most accessible and inspiring voice for the industry, offering cutting-edge features and techniques vital to building future-proof online content”



Steve Jenkins

Highlight



We hate everything about the old waterfall approach

Straight out of Shoreditch comes digital forward thinkers Clubhouse. **Web Designer** gets chatty. **Page 32**

Everyone wants better HTML5, don't they?



HTML5 is the lynchpin of the modern web. Its ever-evolving nature means that the latest incarnation is always revealing more of itself. As browser support improves so do techniques, and our latest lead feature dives into HTML5 from two angles. First, it takes a look at key front-end techniques that should be used. Plus, it reveals how a host of HTML5 tags are misunderstood and misused, and demonstrates how to remedy the situation. On the flip side it's a look at the army of HTML5 APIs out there. Discover how to use the popular and get friendly with the unfamiliar.

Code may be the creator but content is the layer that engages and entices users. A great-looking site with poor content is no substitute for a site with great content. Content-led design is the future and perfecting a focused content strategy is critical for brand success. Our 'Is your content king?' feature on page 64 provides the 7-step plan you need to follow.

The web is all about creativity, whether it is prettifying a frontend or powering up a backend, the community are always here to help. We pick a collection of contemporary GitHub projects that you need to add to your creative arsenal.

Plus, as always we have our staple selection of CSS, HTML and JS tutorials to keep you busy. Enjoy the issue.

We want to minimise the time a user spends looking at a white screen by optimising assets and the order in which they're loaded

Follow us on Twitter for all the news & conversation @WebDesignerMag

Visit our blog for opinion, freebies & more www.webdesignermag.co.uk



FREE - exclusive with this issue

54 Designer resources

- Video Tuition** - CSS3: 120 Minutes of expert video guides from CartoonSmart
- Assets** - 30 Hi-res blurred backgrounds from SparkleStock
- Actions** - 19 Instant hipster Photoshop actions from SparkleStock
- Fonts** - Nyxali and Octynaz font from Typodermic fonts

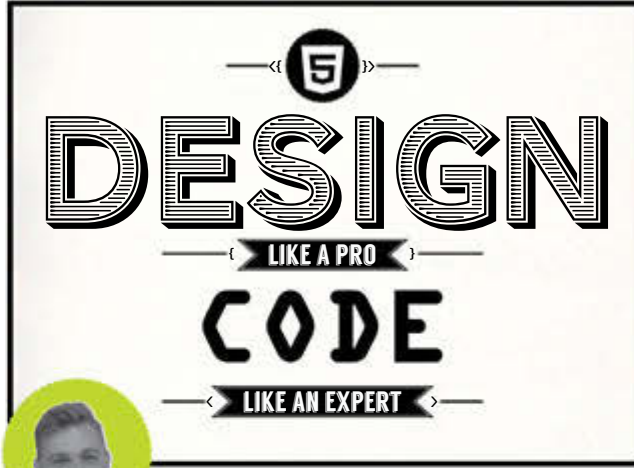


www.filesilo.co.uk/webdesigner

This issue's panel of experts

Welcome to that bit of the mag where we learn more about the featured writers and contributors...

☞ The <article> tag can contain the HTML5 elements header, footer, aside and more instances of itself - allowing for further semantic structure ☞



Ralph Saunders

Ralph is a front-end developer who abandoned graphic design after building his first site eight years ago. This issue he digs deep into HTML5 to unveil powerful, little-known features, explains the misunderstood and teams up with Tam Hanna to reveal the might of HTML5 APIs. **Page 38**

Mark Shufflebottom



Mark Shufflebottom is professor of Interaction Design at Sheridan College, Ontario. This issue he works with Popcorn.js, a library for syncing events with music or video to create interactive audio/visual presentations. **Page 46**



Tam Hanna



Tam is a veteran in the world of coding and has worked with a host of coding languages. This issue he dives into the depth of GitHub and finds the best new and most exciting projects that you need to check out. **Page 70**



Jo Cruickshanks



Jo is the cofounder of online retail specialists Folk. This issue she teams up with Louis Pollard, Folk's in-house wordsmith to reveal why a content strategy is essential, what they are and how to create one. **Page 64**



David Boyer



David has over ten years of experience and is a senior developer within NHS Wales using ColdFusion, PHP and Node.js. This issue he offers an insight into the Electron framework and shows how to build apps fast. **Page 58**

Sean Tracey



Sean is a creative technologist living and working in the south coast. He loves playing with the latest JS APIs he can get his hands on. This issue he demonstrates how to use NW.js to create JS-powered desktop apps. **Page 82**

Mark Billen



Mark is a freelance technology journalist with a background in web design and development. This issue he tackles the Lightbox pages and reveals how to create a host of quick-fire techniques you need to know. **Page 14**

Tim Stone



Tim is a lead front-end developer at Redweb and has a first-class degree in Interactive Media Production. This issue he demonstrates how to create a web app backed by the Last.fm API and the new HTML5.1 Fetch API. **Page 76**

Leon Brown



Leon is a freelance developer and trainer who assists web developers in creating efficient and secure code for their web projects. This issue he shows us how to create an expanding information card box using CSS. **Page 52**

Got web skills?
We're always looking for the hottest web-design talent. Email webdesigner@imagine-publishing.co.uk with examples of your creative work

Imagine Publishing Ltd
Richmond House, 33 Richmond Hill
Bournemouth, Dorset, BH2 6EZ
☎ +44 (0)1202 586200
Web: www.imagine-publishing.co.uk
www.webdesignermag.co.uk
www.greatdigitalmags.com

Magazine team

Editor Steve Jenkins
steve.jenkins@imagine-publishing.co.uk
☎ 01202 586233
Senior Designer Benjamin Stanley
Production Editor Carrie Mok
Photographer James Sheppard
Senior Art Editor Will Shum
Editor In Chief Dan Hutchinson
Publishing Director Aaron Asadi
Head of Design Ross Andrews

Contributors

Rob Bateman, Evgeny Shadchnev, Mark Billen, David Howell, Ralph Saunders, Tam Hanna, Mark Shufflebottom, Richard Lamb, Leon Brown, David Boyer, Tom Wittlin, Jo Cruickshanks, Louis Pollard, Tim Stone, Sean Tracey

Advertising

Digital or printed media packs are available on request.

Head of Sales Hang Deretz

☎ 01202 586442
hang.deretz@imagine-publishing.co.uk

Advertising Manager Alex Carnegie

☎ 01202 586430

Account Manager Luke Biddiscombe

☎ 01202 586431

FileSilo.co.uk

Assets and resource files for this magazine can be found on this website.

Register now to unlock thousands of useful files.

Support filesilohelp@imagine-publishing.co.uk

International

Web Designer is available for licensing. Contact the International department to discuss opportunities.

Head of International Licensing Cathy Blackman

☎ +44 (0) 1202 586401
licensing@imagine-publishing.co.uk

Subscriptions

subscriptions@imagine-publishing.co.uk

To order a subscription to **Web Designer**:

☎ 0844 848 8413 ☎ +44 1795 592 878

Email: webdesigner@servicehelpline.co.uk

13-issue subscription (UK) – £62.30

13-issue subscription (Europe) – £70

13-issue subscription (ROW) – £80

Circulation

Head of Circulation Darren Pearce

☎ 01202 586200

Production

Production Director Jane Hawkins

☎ 01202 586200

Finance

Finance Director Marco Peroni

Founder

Group Managing Director Damian Butt

Printing & Distribution

Printed by Southernprint Ltd, 17-21 Factory Road Upton Industrial Estate, Poole, Dorset, BH16 5SN

Distributed in the UK, Eire & the Rest of the World by Marketforce, Blue Fin Building, 110 Southwark Street, London, SE1 0SU, ☎ 0203 148 3300, www.marketforce.co.uk

Distributed in Australia by Network Services (a division of Bauer Media Group), Level 21 Civic Tower, 66-68 Goulburn Street, Sydney, New South Wales 2000, Australia, ☎ +61 2 8667 5288

Disclaimer

The publisher cannot accept responsibility for any unsolicited material lost or damaged in the post. All text and layout is the copyright of Imagine Publishing Ltd. Nothing in this magazine may be reproduced in whole or part without the written permission of the publisher. All copyrights are recognised and used specifically for the purpose of criticism and review. Although the magazine has endeavoured to ensure all information is correct at time of print, prices and availability may change. This magazine is fully independent and not affiliated in any way with the companies mentioned herein.

If you submit material to Imagine Publishing via post, email, social network or any other means, you automatically grant Imagine Publishing an irrevocable, perpetual, royalty-free license to use the materials across its entire portfolio, in print, online and digital, and to deliver the images to existing and future clients, including but not limited to international licensees for reproduction in international, licensed editions of Imagine products. Any material you submit is sent at your risk and, although every care is taken, neither Imagine Publishing nor its employees, agents or subcontractors shall be liable for the loss or damage.



© Imagine Publishing Ltd 2015
ISSN 1745-3534



Where ideas take shape.

Bring your projects to life with Shutterstock's diverse library of over 50 million images, videos, and audio tracks. Flexible payment and subscription options make creating even easier. **Save up to 20% now.**

[shutterstock.com](https://www.shutterstock.com)

NEW! Convenient monthly billing • **NEW!** No daily download limit • Innovative search tools



shutterstock[®]

contents

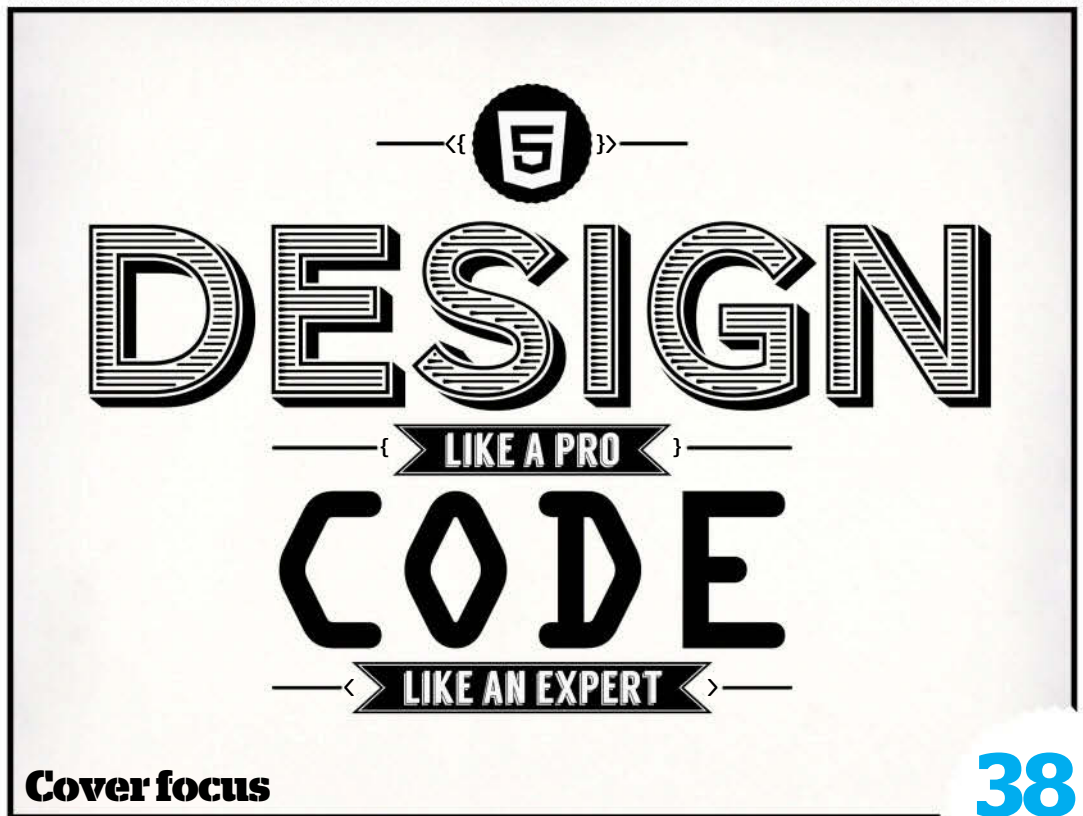
Cutting-edge features, techniques and inspiration for web creatives

Chat with the team and other readers and discuss the latest tech, trends and techniques. Here's how to stay in touch...

✉ webdesigner@imagine-publishing.co.uk 🐦 [@WebDesignerMag](https://twitter.com/WebDesignerMag) 🌐 www.webdesignermag.co.uk

Quick look...

- 08 Are you Google-friendly?**
The world's favourite search engine recently changed its search algorithm. **Web Designer** investigates what this means
- 10 Webkit: The best must-try resources out there**
Discover the libraries and frameworks that will make your site a better place to visit
- 11 Is WebGL the saviour of the mobile web?**
The Away Foundation director Rob Bateman explains how the graphics API will push mobile to the next level
- 14 Lightbox**
A showcase of inspirational sites and the techniques used to create them
- 26 On the edge of extinction**
CSS and its shard-shifting abilities add a voice to the world's endangered species in Bryan James' In Peices project
- 32 Digital alchemy**
From a small studio in Shoreditch comes Clubhouse, an agency that brings excitement and energy to the digital arena
- 38 Design like a pro
Code like an expert**
20 HTML5 tips, tricks and techniques to build better web experiences
- 64 Is your content strategy king?**
Content-led design is the future. Get ready for your brand to succeed with our 7-step plan
- 70 20 best GitHub projects**
A contemporary collection of libraries and tools to add to your creative toolset
- 88 Hosting listings**
An extensive list of web hosting companies. Pick the perfect host for your needs
- 98 Next month**
What's in the next issue of **Web Designer**? Find out what you can look forward to



Cover focus

38



32

ProFile: Clubhouse
Giving a thumbs down to the old and adopting the new



14

Lightbox: Alexander Engzell
A supercool portfolio site to admire

FileSilo

94 Get the latest must-have resources and videos

A comprehensive collection of free designer resources!

- 120 Minutes of CSS3 video guides
- 3 Web templates from w3layouts
- 19 Instant hipster PS actions
- 30 Blurred landscape backgrounds



64

Content-led design is the future. Get ready with the perfect strategy plan

<header>

The tools and trends to inspire your web projects

- 08 Are you mobile-friendly?**
Google's recent algorithm update concentrated on being mobile-friendly. **Web Designer** takes a closer look at what this means to designer and developers
- 10 Webkit: The best must-try resources out there**
Need a new framework or library? Then this is where you need to start
- 11 Comment: Rob Bateman**
Has the full power of WebGL been adopted by mobile developers yet?
- 12 Interview: Evgeny Shadchnev**
Makers Academy CEO reveals how they turn novices into web development experts



£30,000 to £35,000 is impressive for a first job with no experience

Evgeny Shadchnev

Never miss an issue
Subscribe
Turn to page 62 now
USA readers turn to page 81 for an exclusive offer

Visit the **WEB DESIGNER** online shop at **imagineshop.co.uk**
for back issues, bookazines and DVDs

<tutorials>

Web gurus take you step-by-step through professional techniques

- 46 Supercharge web video**
Learn how to sync animations to video and audio using Mozilla's Popcorn.js
- 52 Create expanding boxes with CSS**
Animate sliding panels to reveal and hide information
- 58 Build desktop apps with Electron**
Use pseudo-elements, HTML, CSS and JavaScript to create cross-platform desktop applications



web workshop

- 50 Animated slide-up titles**
onedollarlesson.com
Introduce quick and simple engagement
- 56 Create a shrinking logo effect**
devstars.com
Add attention-grabbing on-scroll animation

web developer

- 70 20 Best GitHub projects**
A contemporary collection of libraries and tools that you need to add to your creative arsenal
- 76 Create custom playlists**
Master Fetch and Last.fm's API by making a playlist from you and your friends' mutual music appreciation
- 82 Making desktop apps with NW.js**
Learn how to use NW.js to create JS-powered desktop apps that run on Mac OS X

<header>

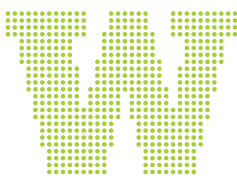
The tools and trends to inspire your web projects

If you have a creative project, new web product or great designer story, contact the editorial desk

✉ webdesigner@imagine-publishing.co.uk  [@WebDesignerMag](https://twitter.com/WebDesignerMag)

Are you Google mobile-friendly?

Has Google's recently introduced mobile algorithm affected your ranking? Now is the time to take action



Whether you like it or not Google rule when it comes to search. Over 70 per cent of the world's

searches are performed via the Google Search Engine. Looking to mobile and tablet this figure rises to over 90 per cent. To put this into more easily digestible figures, Google is performing over 3.5 billion searches daily. This is certainly not a figure to be sniffed at and any time the search giant decides to update its search algorithms the world needs to take notice.

It has been well documented that Google changed its mobile search algorithm back in late April, with the changes being dubbed 'Mobilegeddon' by many in the media. In the aftermath of the update it seems there hasn't been a seismic shift in how many contemporary sites rank. Let's be honest most sites that are owned and maintained by people that live in 2015 are already mobile-friendly. We could say 'Well who doesn't have a responsive or bespoke mobile web experience in this day and age?'. But, then we remember that the web is a vast space with millions still stuck with a desktop-only behemoths. A study conducted by TechCrunch reveals a surprising set of results, suggesting that around 50 per cent of Fortune 500 companies websites were

not mobile-friendly. Nowadays, a site needs to be conceived so that it works on all devices - that is a given. This is obvious for a company/business to succeed or at least gain a competitive advantage. It stands to reason... if a site works.

We expect that most of our readers are already 'mobile-friendly', but the introduction of the new mobile algorithm provides the ideal opportunity to go back and revisit your current setup. The premise of the new Google algorithm is that it will rank mobile-friendly sites higher than those that are not as friendly.

The first and very simple step is to use Google's very own Mobile-Friendly Test (google.co.uk/webmasters/tools/mobile-friendly) tool. This will analyse a URL and determine whether a site is mobile-friendly or not. This is a simplistic tool but it is a quick and easy starting point and may reveal issues that you didn't know about. Note that this will only analyse the page URL added and not an entire site.

Now, we have got past the first hurdle the next step is to follow a few simple(ish) steps. Google has a host of tools to check the performance of pages. The load time of a page may be okay for desktop but mobile is a whole different challenge - 4G may be with us but the vast majority of users won't necessarily have it. You can check your page-loading speed using PageSpeed Insights (developers.google.com/speed/pagespeed/insights). The Insights results will

give you suggestions for both mobile and desktop. Does your site still have Flash (mostly adverts)? Get rid of these, mobile support is very poor. Just make any Flash-based content with HTML5, it's the way forward apparently.

Another almost obvious move is to optimise images. Big, beautiful images on desktop are great, but if they are set up as 100 per cent width then they will be a definite bandwidth hogger. Try Foundation 5 Interchange or take a look at these insights liquidlight.co.uk/blog/article/responsive-image-solutions-picturefill-and-the-picture-element or davidwalsh.name/responsive-images for a solution. If you are using a CMS eg WordPress, Joomla, Drupal, Blogger and so on, Google offers dedicated guides. Find WordPress advice at developers.google.com/webmasters/mobile-sites/website-software/wordpress, Joomla at developers.google.com/webmasters/mobile-sites/website-software/joomla, Drupal at developers.google.com/webmasters/mobile-sites/website-software/drupal and Blogger at developers.google.com/webmasters/mobile-sites/website-software/blogger.

Finally, check out Google's very own Mobile Guide (developers.google.com/webmasters/mobile-sites/website-software) for a few more tips. Try these suggestions and it won't be long before you are on the first page on Google and all the goodness that comes with it.

Get Web Designer digital editions

How did they build that amazing site? Check out Web Designer's latest digital special

Web Designer's latest digital additions to our burgeoning collection is **10 Amazing Websites and How to Build Them Vol 02** and **HTML5 Arcade**. Get an insight into the different games engines out there, plus discover how to start creating your very own.

Discover great-looking sites and the techniques used to build them.

Plus, check out **Web Design Superstars**, **225 Best Web Tools and Resources** and **Amazing Websites and How to Build them Vol 01**. How can you get your hands on a Web Designer digital edition? Head to bit.ly/1hsGYgl, download the free Web Designer app and find them waiting for you as an in-app purchase.

66 Does your site still have Flash (mostly adverts)? Get rid of these, mobile support is very poor. Just make any Flash-based content HTML5 99



<design notes>

A collection of inspirational visuals

A world of languages

bit.ly/IPNrBRI

1 There are at least 7,102 known languages across the globe. The circular nature of the infographic reinforces the idea that the information is of a global nature.

2 Simple colour coding, along with essential data, helps establish which are the most popular languages on the planet out of them all.

A world of languages

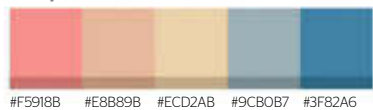
There are at least 7,102 known languages alive in the world today. Twenty-three of these languages are a mother tongue for more than 50 million people. The 23 languages make up the native tongue of 4.3 billion people. We represent each language within black borders and then provide the numbers of native speakers (in millions) by country. The colour of those countries shows how languages have taken root in many different regions.



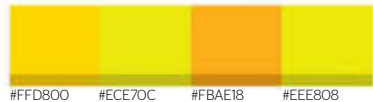
Colour picker

Hot hexadecimal codes

Salty fun
bit.ly/1ROMEka



Mega
bit.ly/1GbFOMB



Typesetter

The best fonts you need



Silver
fonts.com/font/fenotype/silver
A clean and elegant script font ideal for creating elegant headlines or logos.

ABC abc

Stadt
bit.ly/1dj4Cfb
A combination of Fifties typographic aesthetics, adding a contemporary twist.

Graphics

Great visuals to inspire



SkyAdventures
on.be.net/1MabB5G
Jonathan Quintin, founder of STUDIOJQ (studiojq.co) has kept a vibrant palette throughout and added to the design with beautiful location photography.

Themematic

Style up your WordPress



Borderland
bit.ly/1dRDNQI
There's plenty of potential for gorgeous big-screen graphics here, with a whole host of great-looking, clean and constructive layouts.



For Better Coffee forbetter.coffee
One-page scroll animation with cute illustrations



My Big Tomorrow mybigtomorrow.com.au
Big, bold, brash design for careers

Sites of the month



Weber - BBQ Cultures bbqcultures.com/en_GB/intro
A video journey via the medium of BBQ-ing



ETQ Studios etq-amsterdam.com
Minimalist and clean with a clear focus on product

<webkit>

Discover the must-try resources that will make your site a better place



CSS gradient generator

paintbycode.github.io/gradient-generator
In need of a CSS gradient but can't be bothered to do the hard work? Click Roll, wait for one you like and get the code.



Bone.io

bone.io
Bone.io is a lightweight framework that lets users build real-time HTML5 apps using WebSockets. Users can easily construct rich, highly responsive user interfaces.



Crosswalk

crosswalk-project.org

Need to build powerful cross-platform apps that work on all devices? Of course you do, and Crosswalk helps. The big issue is the differences between JavaScript APIs, support for specific CSS properties and how a UI will render on devices. Consistency is what designers and developers are looking for.

Crosswalk is looking to provide consistency using WebView based on the latest Google Chromium version. Crosswalk's focus is its compatibility with Cordova/PhoneGap APIs and Android. It uses the latest innovations and APIs, and in turn developers can experiment and bring the latest tech to their builds.



Pintsize

pintsize.io
Pintsize is a Sass front-end template that is an ideal starting point for any web project. It is simple, scalable, lightweight and configurable.



ReadRemaining.js

aerolab.github.io/readremaining.js
A simple but useful tool for pages with long reams of text. It informs the reader how much longer they'll need to finish an article.

TOP 5 HTML & CSS experiments

Code that is pushing the boundaries of web technologies

01



Circle, square, triangle

bit.ly/1j2vamN
A neat CSS keyframe animation that transforms from shapes into a dolphin as it crashes into the opposing colour.

02



Flipside

lab.hakim.se/flipside
A button that seamlessly transitions from one state to another using a flip animation. The clicking position determines the animation.

03



HTML5 Canvas loader

cssdeck.com/labs/4do6cnjm
Page loaders are simple and boring, not this one. HTML5 Canvas brings the loader to life with a simple animation.

04



Crosscode

cross-code.com/en/start
This is a retro HTML5 game that will demonstrate the power of the browser. Take it for a test run and head back to your childhood.

05



3D CSS typography

bit.ly/1t2v4jN
Take a half-decent font, add a pure CSS 3D effect, include some simple animation and admire the end result.

WEBGL: SAVIOUR OF THE MOBILE WEB?

The Away Foundation's director, Rob Bateman, explains how the graphics API can be utilised to push mobile to the next level



A common dilemma for the mobile marketer is whether to create an app or a website. Do you go with a single-codebase webpage or a bespoke, native application, written for as

many platforms as your budget will allow to distribute and maintain on app stores?

At face value this seems ludicrous, given the effort and cost involved. But the latter option has frequently won out because of a major hurdle facing browsing experiences to date: performance. The modern mobile browser has suffered a major disadvantage in performance over native applications.

HTML was already a highly complex stack of visual elements when it arrived on mobile, and despite improvements in the run-time execution speeds of JavaScript, the visual compositing and animation of mobile web browsers lags behind native apps. As a result, most web content created for mobile is static, text-heavy content; great for reading news articles, but hardly the platform of choice for engaging applications.

The steady advance in mobile hardware has paved the way for better performance, with a noticeable effect on what is possible in the browser without closing the gap between native and web.

The problem is only set to get worse. To increase hardware performance, more focus is being placed on the graphics processing unit (GPU), which offers higher speeds from parallel processing data. The architecture of the GPU enables this to occur with less energy - an important statistic in the world of constrained mobile power requirements.

This is great for native applications, but creates a problem for browsers which tend to favour the CPU for most tasks, including a large part of the built-in graphics pipeline.

To break the mobile browser out of its CPU shackles, a new graphics API was required. This was



ROB BATEMAN

Director of The Away Foundation
theawayfoundation.org

Rob Bateman is a director of The Away Foundation, which is a

non-profit Community Interest Company (CIC). The Away Foundation promotes openness, innovation and participation in and around freely available software resources.

that browser vendors feel is worth implementing, including all the major desktop browser vendors and the larger handset vendors including Apple, Google and Microsoft. WebGL is rapidly becoming the best choice for high-performance cross-platform web apps that require the fast graphics capabilities of a native application.

So what practical application does WebGL hold for mobile browsers? One obvious area is in games, where fast-paced motion graphics are required. Responsive user interfaces also stand to benefit as new, real-time interactions are dreamt up that before could only have been done piecemeal. Perhaps most ground-breaking is the field of interactive data visualisation.

When describing large data sets, nothing says it like you mean it more than an infographic. They convey meaning on a macroscopic scale, enabling data to be revealed and consumed naturally. The design process involves a balancing act between clarity and aggregation, and one way to enhance the experience is by graphically interacting with the data so that areas of interest can be explored. WebGL now not only makes this possible, but easily accessible across devices.

As a recent example of interactive data visualisation, open source companies, The Away Foundation collaborated with **51degrees.com** to build a proof of concept. With the amount of momentum behind the WebGL standard, we are now on the verge of significantly reducing the performance gap between the browser and native run-times on mobile. Combined with the browser's unflinching promise of cross-platform authoring, cost-free

publishing and frictionless distribution, this might just be enough to tip the balance in favour of browser-only apps for mobile developers. It should give the app stores plenty to think about when considering the mobile app landscape of the future, and their role within it.

JOIN THE WEB DEV REVOLUTION NOW

Makers Academy CEO Evgeny Shadchnev reveals how they turn novices into web development experts in just 12 weeks

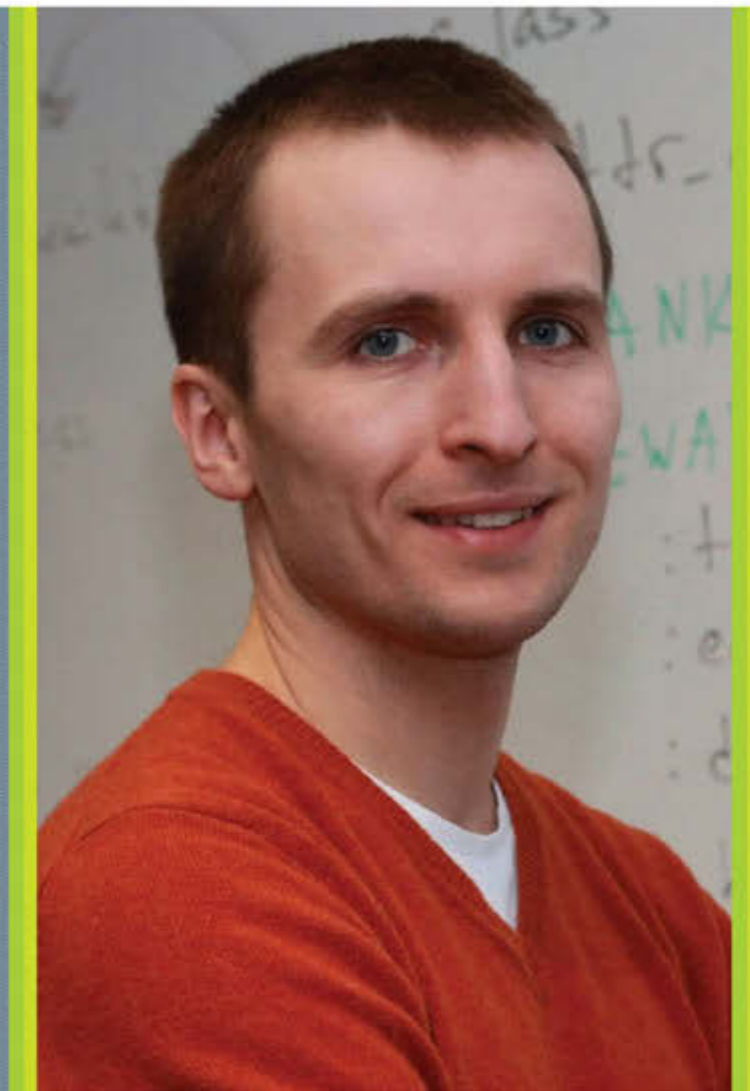
Q. The importance of web development is demonstrated by the fact that it is now taught in schools. How do you see the future of web development evolving, what kind of route do you think this will take and what sort of courses do you expect Makers Academy to be offering in five years time?

Software development is becoming an increasingly important skill, not just for software developers but for all kinds of professions. Web development is particularly exciting since web technologies are flexible enough to power everything from websites to mobile apps, to desktop apps, to Facebook apps and even more. I expect web development to become even more accessible to beginners and more widely used. Although we started Makers Academy off with teaching about web development, in five years we'll be teaching other technologies: data science, hardware programming and so on.

Q. Your courses run over 12 weeks. How intensive are they and to what level can a Makers Academy graduate reach in such a short space of time as it's essentially three months of learning?

They are as intensive as it gets, morning to evening, every working day for months on end, with homework for evenings and weekends. This is not easy but it ensures that our students are ready to get their first job as a software developer in a company like Sky, Deloitte, Comparethemarket and so on at an average salary of £30-£35,000. To me, this is pretty impressive for a first job with no commercial experience!

Q. You offer a £500 discount to any female graduates who want to take up the course. What is the ratio of men to women joining your



EVGENY SHADCHNEV
Cofounder and CEO of Makers Academy
makersacademy.com

Makers Academy is a full-time computer programming bootcamp that turns people with no web-dev knowledge into job-ready junior developers in 12 weeks.

courses and have you noticed a rise in the number of female applicants since you started?

Almost one out of three applicants and students are women, which is better

than the industry average but it's not exactly the 50/50 balance we'd like to see, although it's getting better. However, some of the best graduates we ever had are women, for example

we can look at Sroop who went from a successful career in design to starting as a web developer at New Bamboo or Margo who was focusing on raising children for years before she went through the course and became a developer at M&S Venture Labs.

Q. You currently teach courses in a host of web technologies. Do you have any plans to expand the range of technologies that you are offering and to teach to a higher level of students as well?

We're using a number of web technologies to help our students to learn how to learn, instead of focusing on technologies themselves. We're often updating the curriculum to reflect the needs of the market but the technology stack changes so often that the main skill of a software developer is often the ability to learn whatever technology they need for the current project. In fact, in their final projects our students often use technologies that we never taught on the course!

Q. Graduates who complete a course will be looking to get into employment after they finish. What kinds of job-seeking paths do you offer to your Makers Academy graduates?

Almost everyone who wants to get a job - works hard, attends interviews, does tech tests and so on - gets a job within a month or two after graduation. We maintain connections with businesses around London, knowing what exactly they are looking for, setting up interviews, advising our graduates and so on. In fact, we spend a full week full-time on preparing them to find the first job: doing mock interviews, information sessions, individual coaching and so on. We pride ourselves on our placement record, so our work doesn't stop until the contract is signed.



BECOME A PROFESSIONAL DEVELOPER *in* 12 WEEKS

makersacademy.com



Achieve more with our
MSc in Creative Technologies
and Enterprise

Not all postgraduate degrees are created equal. Study for yours at Bath Spa University and give yourself a head start in the creative industries.

Our course will help you hone your creative skills, making sure you're prepared for the rapidly-changing global digital economy.

Get more from your future,
visit bathspa.ac.uk/cte

**DIFFERENT
THINKING**



<Above>
 • Project case studies like this one for FitWhey offer enviable detail on assets created, with desktop and mobile screenshots



<Above>
 • Restrained colour use, moody imagery and sharp typography gives Alexander's portfolio a strong sense of professionalism

<Above>
 • Scrolling effects add subtle hints of motion while a distinctive 'guillotine' motif is applied to the project selection menu

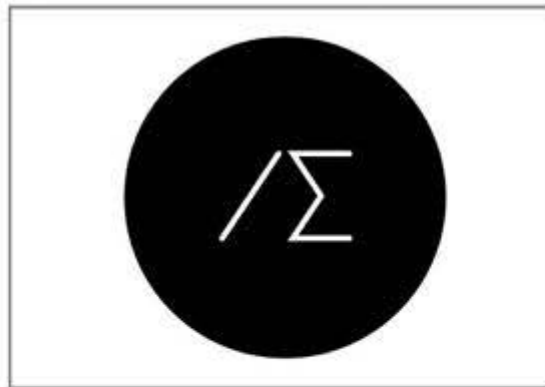
<Above>
 • A commitment to responsive design across client work, Engzell ensures his portfolio looks just as great across multiple devices

<Above>
 • Navigation between long, sprawling project pages is assisted with simple footer navigation and options for social sharing

Alexander Engzell

engzell.me

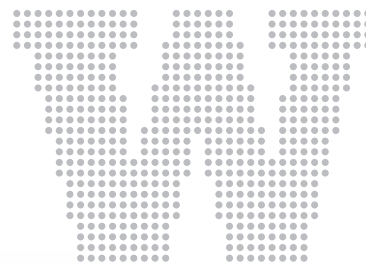
Development technologies HTML5, CSS3, jQuery, TweenMax, SVG, WordPress, JavaScript, After Effects



Designer **Alexander Engzell (art direction and web design), Antoine Wodniack (code), Jimmy Raheriarisoa (motion)**

engzell.me, wodniack.fr, en.nerisson.fr

A supercool portfolio from over ten years at the top flight of the art-direction game



welcome to the new online home of Alexander Engzell, founder and art director at Bonne Marque and a man with “over a decade in the game” of high-profile creativity. A proud member of the CSS Design Awards jury, the talented Swede showcases his numerous achievements here within a

tastefully dynamic design. Energetic typography bursts from a stark monochrome template bubbling with impressive projects for brands such as FitWhey, ProDo and Frasers Mall. Visitors get a real sense of the breadth of services provided across image-rich case studies, alongside touches of personal creative expression. “My original inspiration when

setting out to build my new portfolio was the guillotine, you see it clearly throughout the website, [with the way that] the angle of the portfolio thumbnails are cut, the Bezier curve animation, the hover effects for buttons and the lazy loading effect for content,” Engzell explains. “My other inspiration came from Eighties movie posters for the custom typography on all the portfolio titles, which are personally developed slogans for representing the work in just a few words. The animations for the titles were first made in After Effects and then exported as sprites and animated with JavaScript.”

In addition, **engzell.me** looks and performs as impressively inside mobile browsers as it does on the desktop. Seeing as it’s built to be responsive and wholly consistent, it’s no surprise that the site has claimed two Site of The Day awards in just two weeks after its launch.

ABCABC

1234567890

<Above>
• Bebas Neue Bold by Ryoichi Tsunekawa for Dharma Type Foundry is applied to page section headings

abcABC

1234567890

<Above>
• The Voyager typeface by Martin Van’t Wout covers subsequent page text in Medium, Bold, Light and Light Italic

• Inspiration came from Eighties movie posters for the custom typography



How to emulate engzell.me's hand-drawn typography

Alexander Engzell describes the art direction and creative process behind his site's animated Eighties movie poster-inspired typography

01 Sketching lettering by hand

The distinctive animated lettering on engzell.me begins hand drawn on pen and paper. When doing this you need to take into consideration the overall 'slogan' for each of the lettering, the overall feeling for the work, plus the eventual background image when sketching. This all helps the lettering to complement the design as effectively as possible. You'll also want a varied selection of cool marker pens!



02 Vectorise the sketches

With your sketches produced, the next phase is in capturing and digitising them for processing. At this point it's a matter of spending adequate time vectorising the sketches in an application such as Illustrator. The effort you put in here will help you get a more perfect result, using as few points as possible to get a smooth and clean graphic for importing into After Effects later on.



03 Animate the letterings

We explored hundreds of options and probably around ten revisions for each animation once decided on the overall flow. Each animation sequence is therefore totally unique and realised at this stage using expressions within After Effects. This is a particularly useful tool for prototyping UI animations, given that the expression language is based on JavaScript that will eventually power the final animation within the browser.



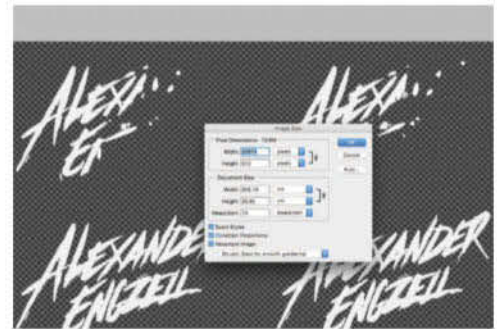
04 Export the animations

Once satisfied with the animation sequences you need to optimise the animation and export it as a PNG sequence so it can be implemented via code. It's very important to keep the file sizes low here and lightweight in order to provide a great user experience across browsers, devices and platforms. This PNG sequence will then be stitched together to form a sprite or sprite sheet.



05 Implement the sprite

Once your animation is exported into a PNG sequence, you have to combine those files into one sprite image. You can find online tools for assisting here such as SpritePad (wearekiss.com/spritepad) or Stitches (draeton.github.io/stitches) for HTML5 and CSS, or simply combine them manually in PhotoShop. In the case of engzell.me, a PHP script was written to generate the required sprites.



06 Power on the page

With the sprite readied, you'll need an HTML/CSS container sized to fit your animation and with your sprite set as a background. Then write a JS function that shifts the background-position property of your container, moving it from one frame of the sprite to the next from the time called. A `setInterval()` method is used here to call the function at the right interval, so for example 40ms for a 25FPS animation. Don't forget to stop or loop animation once it's over!

```
001 //keeps calling function each 40ms
002 until stopped:
003 setInterval(function(){ shiftBg()
004 }, 40);
005 function shiftBg() {
006 //Shift container background 0-24
007 frames here
008 }
009 }
```



Receive £25k
tax-free to retrain as
a computing teacher

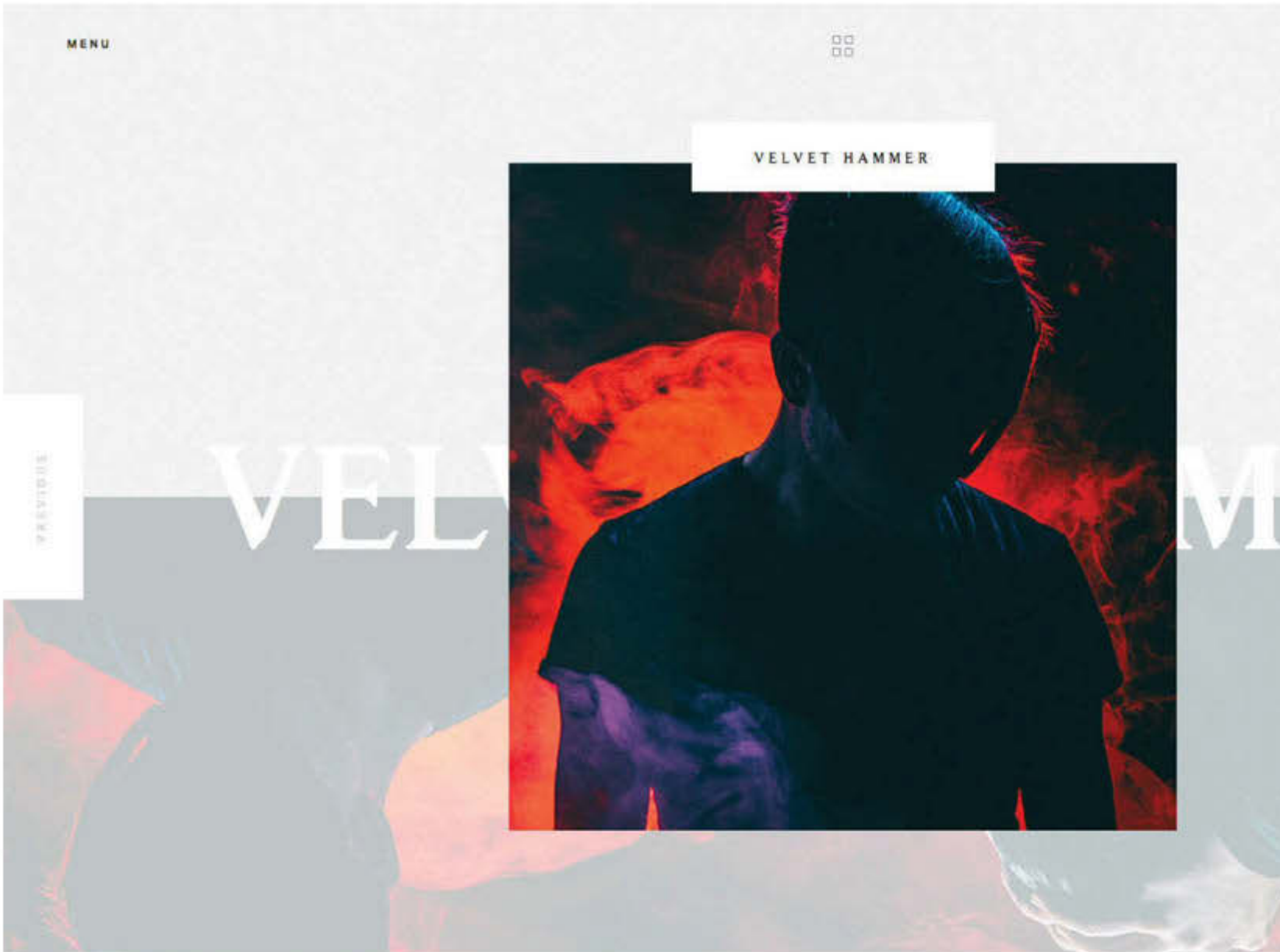
and help develop
the great minds
of tomorrow.



Switching to a great career as a computing teacher is more achievable than you think. You could receive a £25k* tax-free bursary or a £25k* scholarship provided by BCS, The Chartered Institute for IT. Alternatively, you could earn a salary of up to £25k* while you train. Then you can inspire young people to fulfil their potential.

Apply now.

Visit education.gov.uk/teachcomputing



<Above>
 • The site revolves around an unusual shuffling menu blending a split background and rollover elements



<Above>
 • A square design motif is everpresent, beginning with the opening preloader element and appearing throughout

<Above>
 • Five featured projects are used to illustrate the main navigational element, and these prove easy to cycle through

<Above>
 • Project pages inject much more colour and stack sumptuous assets down a lengthy two-column structure

<Above>
 • Never afraid of keeping things simple, main navigation is also made easy via bold rollover text links on the Suspended Animation site

Suspended Animations

sa-studio.fr

Development technologies HTML5, CSS3, Jekyll (jekyllrb.com), npm, gulp, page.js, GSAP, Typekit



Designer **Baptiste Briel (creative developer), Flavien Guilbaud (art director)**

bbriel.me, flavinsky.com

French flair meets refreshing subtlety in a studio site showing that in design terms, it's hip to be square



Suspended Animations is a multiple award-winning creative studio based in Nantes, France. Founded in 2011 by Flavien Guilbaud and Baptiste Briel, this latest portfolio was also crafted by the pair and features a selection of works across digital art direction, design, branding and of course web

development. The site juxtaposes subdued colours against sumptuous, colourful project imagery to deliver an experience both understated and confident. Perhaps most notable though is the navigation and a reflective scrolling menu for accessing projects. "Design-wise, we wanted to create a portfolio with an unusual and original navigation," explains art

director Flavien Guilbaud. "Every piece and page of the website is based on a square, like the loading animation, the menu and homepage. We used a massive serif font called Rasmus and a Typekit font, Aktiv Grotesk." For all that style, **sa-studio.fr** isn't all surface with plenty going on development-wise to drive some slick front-end transitions. "The site was built using Jekyll, a static site generator that uses Markdown files as data," elaborates creative developer Baptiste Briel. "I would highly recommend this when building your own portfolio and stuck on the back-end development. If you're into the front-end development thing, my workflow consists of npm as package management with gulp for building minified CSS and JavaScript files. The site also uses page.js, a tiny JavaScript router and the tween engine GSAP."

Design-wise, we wanted to create a portfolio with an unusual and original navigation



abcABC
1234567890

<Above>
• Rasmus by Markus John and available via Ten Dollar Fonts is used in Regular and Bold styles

abcABC
1234567890

<Above>
• Aktiv Grotesk by Dalton Maag via Typekit handles subsequent page links throughout the site



Recreate the smooth scrolling effects on sa-studio.fr

Creative developer Baptiste Briel explains how to get started with the smooth scroll JavaScript library at the heart of the Suspended Animations site

01 Get started

A popular technique amongst many websites is to override the default body scroll behaviour in favour of custom scrolling. In many cases this is achieved using JavaScript and specifically the `requestAnimationFrame()` method for applying CSS transforms across several HTML elements. This smooth scroll is what's used on [sa-studio.fr](#) and utilises a library script on GitHub.



02 Smooth scroll library

As mentioned, the smooth scrolling scripts used on SA are available via GitHub at [github.com/BaptisteBriel/smooth](#). Download by clicking the Download ZIP button. Save and unzip the 'smooth-gh-pages.zip' repository to find a selection of very simple demos to see how smooth scroll works. The three main library JS files can be found within the `src` folder.

03 Set up the HTML

To start, copy those core JS files into the root of an HTML document and attach just before the closing `</body>` of the page. Next we'll need some simple HTML elements, starting with a wrapper `<div>` with the class `vs-section`. This is very important as the `Smooth.js` script gets the max scrolling value from this wrapper's height. Be sure that there are no floating `<div>`s without `clearfix` or other CSS bugs that would make the wrapper smaller:

```
001 <body>
002 <div class="section vs-section">
003 // Nested <divs> to go here
004 </div>
005 <script src="rAF.js"></script>
006 <script src="vs.js"></script>
007 <script src="smooth.js"></script>
008 </body>
009
```

04 Add additional divs

Basically, the default scroll is applied on this section wrapper. This means that if you want to apply

parallax effects on your page, you'll need extra `div`s within this wrapper. So inside, create a few `div`s with the class `vs-transform` - for instance we've added five here. You might also set the `data-speed` attribute to define how fast each element will move on scroll. In addition, we'll add some minimal CSS styles:

```
001 CSS:
002 body {
003 overflow: hidden;
004 }
005 .section {
006 position: absolute;
007 /* if vertical scrolling */
008 width: 100%;
009 height: auto;
010 /* if horizontal scrolling */
011 width: auto;
012 height: 100%
013 }
014 HTML:
015 <body>
016 <div class="section vs-section">
017 <div class="vs-transform" data-
018 speed="0.2"></div>
019 <div class="vs-transform" data-
020 speed="0.4"></div>
021 <div class="vs-transform" data-
022 speed="0.6"></div>
023 <div class="vs-transform" data-
024 speed="0.8"></div>
025 <div class="vs-transform" data-
026 speed="1"></div>
027 </div>
028
```



05 Initialise with JavaScript

With our JavaScript we now need to initialise a new object with some parameters: the wrapper section `vs-section`, the extra `div`s `vs-transform` (which is optional) and as you can see, an easing effect. This is like the `data-speed` attribute on HTML elements, but for

the section wrapper. As soon as we have defined our variable `scroll`, `scroll.init()` launches all event listeners for touch interactions on mobile, wheel interactions on desktop and will start applying the CSS transforms using `requestAnimationFrame()`. Using `scroll.destroy()` stops the events and `requestAnimationFrame()`.

```
001 //get the section
002 var section = document.
003 querySelector('.vs-section');
004 //get the extra divs
005 var divs = document.
006 querySelectorAll('.vs-transform');
007 //initialize the object w/ some
008 parameters
009 var scroll = new Smooth({
010 direction: 'vertical',
011 section: section,
012 ease: 0.1,
013 els: divs // optional
014 });
015 //kickoff the smooth scroll
016 scroll.init();
```

06 Extra scroll methods

Notice that if you want a parallax effect, the wrapper section will not be transformed. If we want to transform the wrapper and a few more `div`s, the wrapper will need the `vs-transform` class and `data-speed` attribute as well. In addition, other `scrollTo` functions are available for invoking optional behaviours:

```
001 HTML:
002 <!-- the div we want to scroll-to
003 -->
004 <div class="vs-transform js-
005 referer" data-speed="0.2"></div>
006 <!-- the link (could be a nav menu)
007 -->
008 <span class="vs-scrollto" data-
009 scroll="js-referer">text</span>
010 JAVASCRIPT:
011 // scrollTo with a fixed value
012 (useful for simple scrollTop)
013 smooth.scrollTo(0);
014 //scrollTo a specific div
015 var div = document.querySelector('.
016 js-referer');
017 var offset = div.
018 getBoundingClientRect().top;
019 // this should also work :)
020 smooth.scrollTo(offset);
```



Authorized
Reseller

NOW YOU CAN AFFORD A BRAND NEW MAC



LESS THAN
£2 PER DAY



YOU DO THE MATHS

ALL OUR FLEXI-LEASE PACKAGES INCLUDE:

MAINTENANCE

- **3 Years** full cover for all hardware with remote and telephone support, plus loan equipment

LEASE RENEWAL

- after only **2 years** we will offer you a renewal to latest equipment and options to keep or return the old machine

OWNERSHIP

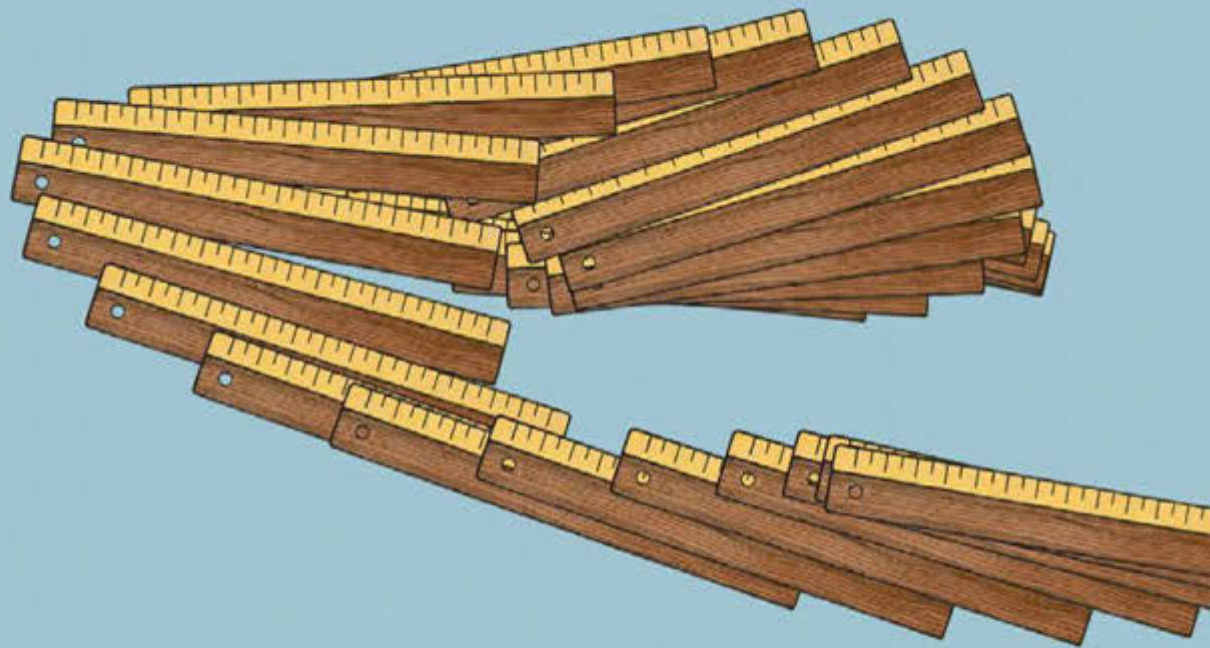
- at the end of the lease the equipment is yours for just **£1**

020 7111 1643
WWW.HARDSOFT.CO.UK

HARDSOFT
Established for over 25 years

FORNASETTI

HISTORY OBJECTS VINTAGE RETAILERS EXHIBITIONS NEWS CONTACTS



CLICK

<Above>
 • Fornasetti.com introduces itself as a slick, stylish and dynamic presence, if a little mysterious regarding content



<Above>
 • Evocative historic photography is presented inventively throughout, drawing you into Fornasetti's time, world and work

<Above>
 • Assets are presented dynamically to make the overall experience much closer to an immersive journey than a conventional, flat website

<Above>
 • The Fornasetti archive of famous works and sketches were brought together and reimagined by a team of creative agencies

<Above>
 • Navigational elements and exhibition sections feel more playful and engaging due to innovative approaches to animation

Fornasetti

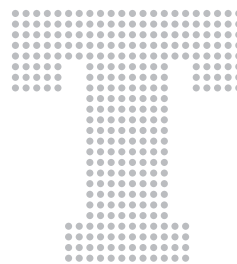
fornasetti.com

Development technologies HTML5, CSS3, Modernizr, jQuery, SVG, Underscore, Backbone, TweenMax, Create



Designer **POMO (art direction)**
Random Studio, AQuest, Luciano Foglia, Vito Tardia, Cesare Picco (music) thepomo.com

This modern web experience reimagines the past with a truly 'bello' site



his lavish online shrine to celebrated Italian painter, sculptor and interior decorator Piero Fornasetti is a timeless masterpiece in its own right. A collaborative project harnessing Fornasetti's extensive archives, the site was crafted by numerous agencies headed by Milan studio

POMO. The shared aim would be to leverage a wealth of artistic assets to deliver something appropriately original. "We were planning something different which would go beyond any branding logic," explains creative director and POMO founder Marco Cendron. "We didn't want a common well-designed website but a metaphysical window founded on innovation

and a deep aesthetic research. Our desire was to carry the imagination of the viewer into the captivating world of Fornasetti, enhancing the unique and unexpected style; unconventional elegance... [that is] impossible to define." The result is an incredibly rich and immersive experience, behaving much more like an art book or gallery brochure. Fornasetti's work and life unfurls itself across a series of dynamic chapters before valuable museum pieces float through your browser vividly. "We based the project on the newest technologies making use of several libraries like Underscore, Backbone, TweenMax and Create, not to mention a lot of custom code," reveals art director Alessandro Barbieri. "But it's the way we painstakingly wrapped Fornasetti's awesome decorations with unusual yet elegant animations that really differentiates this project from any other."

Our desire was to carry the imagination of the viewer into the captivating world of Fornasetti, enhancing the unique style

abcABC
1234567890

<Above>
 • Albertus font by Berthold Wolpe styles the Roman-esque typography found on chapter headings and navigation

abcABC
1234567890

<Above>
 • Halis Grotesque Book by Ahmed Alton provides a much more straightforward and readable font on subsequent text



Draw images on event to full-page <canvas> elements

Inspired by the cool canvas animations of Fornasetti.com, learn some beginner image drawing basics with JavaScript/jQuery events

01 Get started

Fornasetti.com exhibits some lovely interactive animations with HTML5's <canvas> element, opening to a trio of great examples. Inspired by this more sophisticated work, we're going right back to basics on using Canvas and some beginner tips for drawing images or sprites on certain events. So let's get started with simply adding a <canvas> tag to the body of our HTML page - it couldn't be simpler!

```
001 <canvas id="myCanvas"></canvas>
002 <div class="header"><h1>HTML5
CANVAS</h1></div>
003 <div id="clearMe"
class="clear"><h2>CLICK</h2></div>
```

02 Setting up the styles

You don't need our header <div> containing our <h1> header, although we'll use the clearMe <div> later for clearing the Canvas. Notice however, we haven't set the <canvas> height and width inline properties. Normally you must do this to define size, however we want our canvas to be fullscreen, which we'll do with JavaScript. First, our core page CSS looks like this:

```
001 html, body {
002 margin: 0px;
003 padding: 0px;
004 height: 100%;
005 width: 100%;
006 overflow: hidden;
007 }
008 div.header {
009 top: 0px;
010 left: 30%;
011 }
012 div.clear {
013 bottom: 0px;
014 left: 45%;
015 }
016 #myCanvas {
017 background-color: #009681;
018 display: block;
019 }
020 </style>
```

03 Find, initialise and resize

In our JavaScript we first find the <canvas> within the DOM using its myCanvas id, before getting the 2D context object required for drawing. Next, as we'll be drawing an image, we create an Image() object and set the src to our local image path. With those

initialised on page load, we can also resize our Canvas to fullscreen using window width and height:

```
001 <script>
002 var theCanvas = document.
getElementById("myCanvas");
003 var theContext = theCanvas.
getContext("2d");
004 var theImage = new Image();
005 theImage.src = 'ruler.png';
006 theCanvas.width = window.
innerWidth;
007 theCanvas.height = window.
innerHeight;
```



04 Bind some events

You could now simply use drawImage() on the Context object to draw our image to the Canvas on page load. However we're going to perform drawing and clearing on cursor events so we'll use jQuery's bind to attach click functions to our Canvas and our clear div:

```
001 $("#myCanvas").bind("click",
drawImg); //Binds click call to
canvas
002 $("#clearMe").bind("click",
wipeImg); //Binds click call to
div
```

05 The drawImg() function

Our drawImg() function starts by offsetting x and y coordinates based on the event and half the image width/height. This will be used to draw the image sprite centrally below the cursor. Below this we can call our wipeImg() function, clearing the Canvas before each drawing. Simply omit this if you wish to retain the Canvas content as you draw:

```
001 function drawImg(event){
002 var posX = event.clientX -
theImage.width / 2;
003 var posY = event.clientY -
```

```
theImage.height / 2;
004 wipeImg(); // OPTIONAL! Clears the
canvas before drawing
005 theContext.drawImage(theImage,
posX, posY);
006 };
```



06 The wipeImg() function

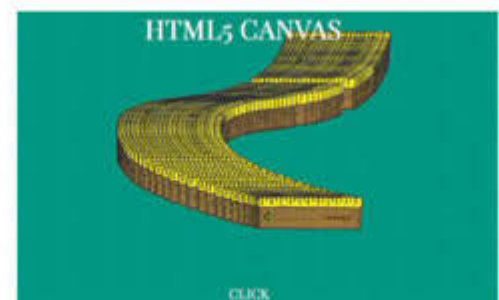
To wipe a <canvas> of any drawing, you basically draw a clear rectangle to the drawing Context, covering the whole Canvas element's width and height. We'll pop this in our wipeImg() function like so:

```
001 function wipeImg(){
002 theContext.clearRect(0, 0,
theCanvas.width, theCanvas.height);
003 };
004 </script>
```

07 On mousemove instead?

If you test the page now, the image will draw on clicking the Canvas, wherever that click occurs. It also clears the Canvas before each new click, while clicking the CLICK div also wipes. If you now remove the call to wipeImg() the Canvas will retain each drawing. In addition, if you swap our event binding to mousemove, the image will draw as you hover the Canvas!

```
001 $("#myCanvas").bind("mousemove",
drawImg); //Binds hover call to
canvas
```



catalyst2

managed hosting



www.catalyst2.com

0800 107 7979

Providing helpful, reliable, **hassle free hosting for 15 years**

We do all the server management so you don't have to

Our **Managed Hosting** is...

- **UK based** (support and servers)
- **Value for money**
- **A worry free service**
- **Reliable and Stable**
– 99.97% uptime since May 2007

Winner
ISPA CUSTOMER CHOICE AWARD
2013 + 2014

99% of our customers rate us **good, very good or excellent**

Try us today.
30 day no quibble money back guarantee.

0800 107 7979 **info@catalyst2.com**

Sign up today and get **10% off** any package using the code **WEBDESIGN**

ON THE EDGE OF EXTINCTION

CSS and its shard-shifting abilities give a voice to 30 of the world's most interesting endangered species



Ever heard of the Javan slow loris? Chacoan peccary? Sumatran rhino? Diademed sifaka? No? They are all under the threat of extinction and unless action is taken they will simply end up being assigned to the history books.

So, how do you go about making people aware of the plight of these endangered species? Bryan James, the brains behind the In Pieces project, decided to embrace the medium of new and experimental code to grab the world's attention of these animals. His desire to add a veneer of the fresh, intriguing and interactive, were combined to communicate a concept with an underexposed topic.

So, now we know why, but what is the project that Bryan built all about? In Pieces is an interactive exhibition of 30 of the world's most unique but sadly endangered species, their survival laying literally "in pieces". How was the site developed? Well, it uses pure CSS (and more specifically, taking advantage of the clip-path property), and this is the big player throughout the project. Each endangered animal is represented by one of 30 triangular pieces, which flow and form into one another, and this movement echoes the common struggles they share and the united nature of many of their survival stories.

The user is able to coast through a collection of species they may never have heard of. They will learn a little about the species and discover why they are in



PIECE
3

GOLDEN LION

IN PIECES | HOW IT'S MADE | CAUSES & SOLUTIONS

Project
Web

In Pieces

species-in-pieces.com

Agency
Web

Bryan James

bryanjamesdesign.co.uk

People
involved
Total
hours

One

Five months in personal time



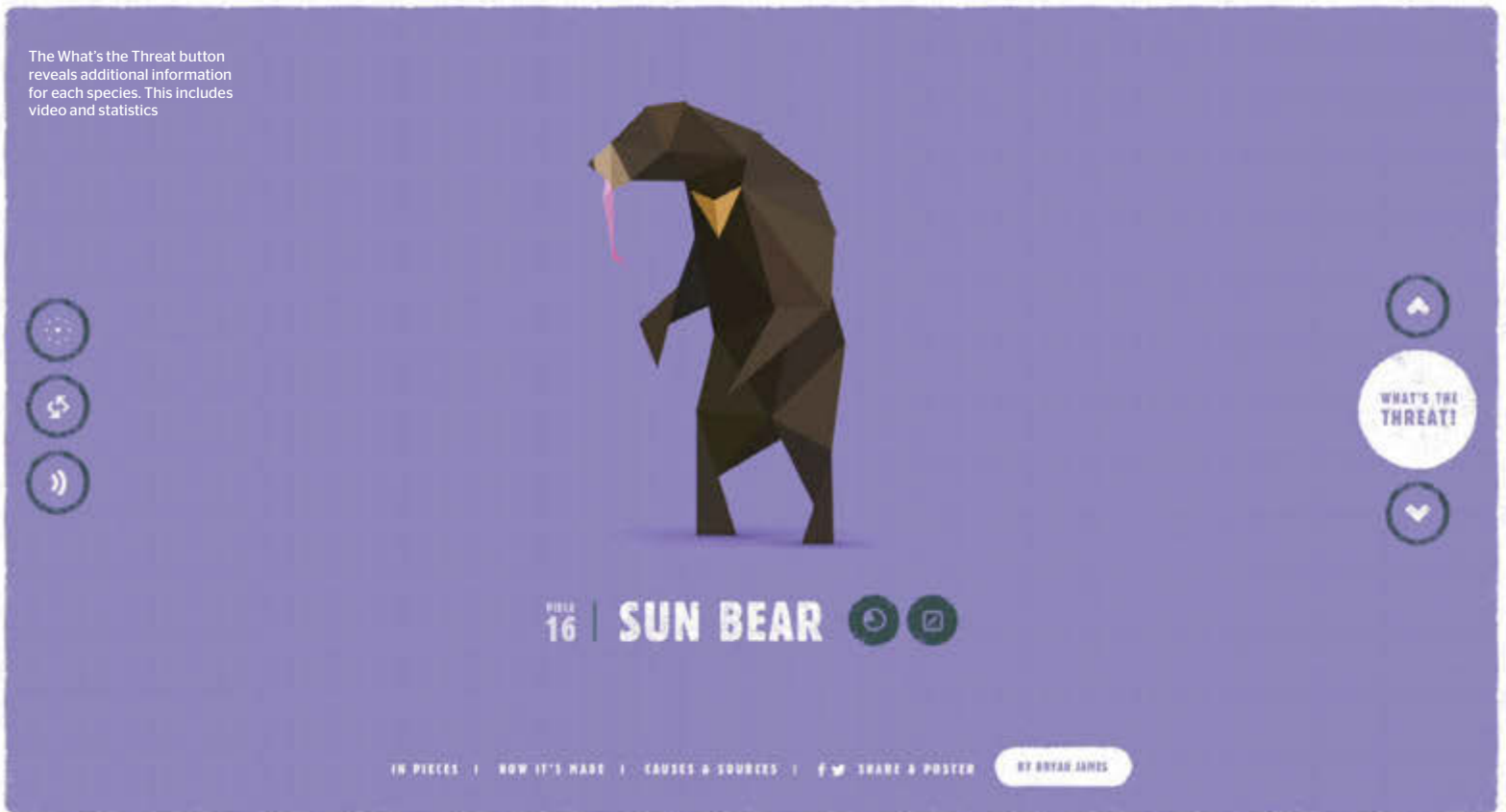
TAMARIN



RESOURCES |   SHARE & POSTER

BY BRYAN JAMES

The What's the Threat button reveals additional information for each species. This includes video and statistics



their current predicament. A third layer of engagement is supplied through the data visualisation of each species' plight, along with a set of desktop and mobile downloadable wallpapers.

The lack of limelight for the vast majority of the species means that people rarely have a connection with them. This is why Bryan wanted to provoke an emotional reaction with the project and remind people of the beautiful creatures that the Earth is on the verge of losing

As the sole creator, designer, developer and artist, it was Bryan's love for CSS that initially kicked off the project, "I had known of CSS clip-path polygons for around half a year but hadn't seen them used at all within real projects, only the odd experiment here and there on coding platforms like JSFiddle." However, it was the lack of interest in the techniques that took the level to the next stage, "Surprised by the lack of furor, I wanted to see how they would react when combined with CSS transitions, and upon initial experiments feeling really smooth I began crafting a polygon animal.

"While designing the first species which ended up being 'Piece 1', Hawaiian crow, I thought it would be interesting to depict an endangered species with the topical value that brings, but at this early stage I had no idea of the monster period of time I was about to embark on.

"The first 'elements' to the idea came from the thought of designing and developing the same amount of animals as if they were pieces, as well as using the same amount of pieces for each animal which was the only way to savour seamless transitions for every triangle.

"I was doing a lot of idea generation on trains and planes through sketches and [the moment] I hit upon the 'pieces' element to the idea was the moment I thought this

could be something which really worked and could communicate a powerful message – the idea that the species' survival lay 'in pieces', so are depicted as a set of 30 'pieces'. After producing the first three species and coming up with what was the true heart of the idea, I designed and developed the 'structure', before moving onto the other species. I was just doing crazy amounts of hours at every opportunity, at home, through the night, on the train to work and in lunchtimes. It was a fairly intense period of time!"

The power of an idea can be immense to the creators involved, but it is the execution where the hard work starts.

“'IN PIECES' JUST FELT RIGHT... IT HAS A CERTAIN GRANDEUR IN THE WAY IT'S SAID AND THE EMOTIONAL CONNOTATION IT EMITS”

How it evolves with a single person at the helm is interesting and intriguing, rarely is it the same process as a multiple person partnership. We asked Bryan how he came to start the very first phase of the In Pieces project: "The initial few weeks of the idea being born were explored in the form of sketching and personal brainstorming – anytime I had free to myself really.

"Possibly unlike a lot of commercial work, I feel the idea had a lot of time to evolve and be caressed if you like. Especially at the beginning, I had a key technology I wanted to play with and an animal created, and needed a place to take them to. There were a lot of ideas which I had thought were perfect, then scrapped the next day. But when the 'pieces' element of the idea was hit upon, the project quickly took a lot more conceptual shape.

"The 'pieces' idea came from a naming exercise for the species, which ended up being the total wrapping up of the whole concept in the word-play connecting to the visual style and the emotive concept. I was looking at things like 'the 30' and '30 pieces'. 'In Pieces' just felt right – not just because it was a perfect explanation of the visual execution of the species, but it has a certain grandeur in the way it's said and the emotional connotation it emits. It's normally used in a very human way, such as 'the ordeal left her in pieces', so I think it's this usage of a very humanised term which works well to encompass the project.

"The second ideas phase was more about individual elements. In essence I readied a design on paper both visually in my head, as well as interactively trying to think of how I could create things from pure code and animate them accordingly as a consequence."

The visuals throughout In Pieces are stunning, clever and interactive, bringing a new edge to the technologies used.

However, their development and production process is not as obvious and as straightforward a process as they may first seem as Bryan explains: "The visual design of the interface and structure was contrary to how it may look from the outside as it's a very visual project, one of the least intense parts of the whole process.

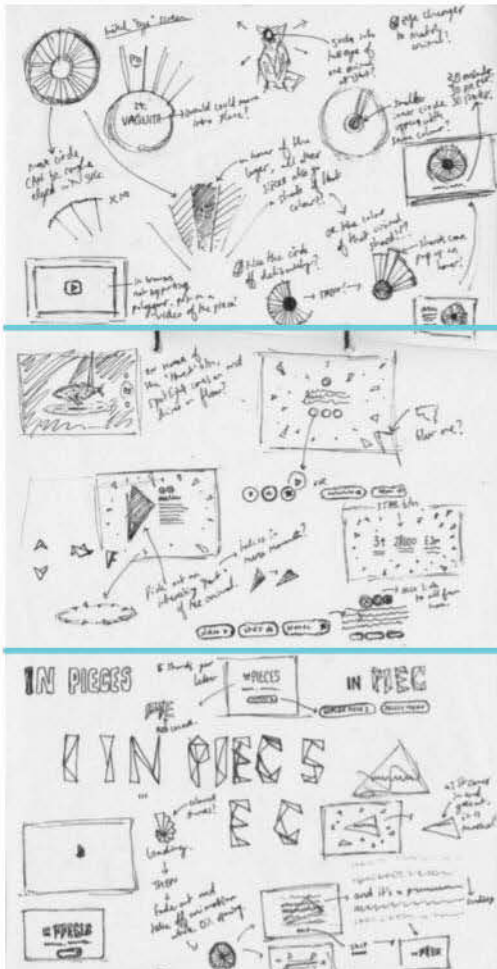
"Within the ideas phase I would think up and sketch what I wanted to do with say, a button, but would also think of how it would be coded and thus how I could animate it. With much of the design phase of elements, it was normally just about placing the textured aesthetic I created into the context of a simple shape to make sure it worked before diving directly into the code to execute."

"The design of the species themselves was a different story. Firstly, having just 30 pieces to play with was

The In Pieces project foregoes WebGL and Canvas, instead it is made entirely with CSS



Bryan has worked on a host of projects including Sounds of StreetView and won numerous design awards



Bryan would first work on the visuals in sketches before delving into the developer side to figure out how to animate it all in code

TOTAL UNITY

As the sole creator Bryan shunned the two-party involvement, but he believes that the typical client-agency relationship still has plenty to offer: "I do think it's an interesting topic in regards to this project because I feel personally that it's the perfect showcase of why designers and concepteurs who traditionally lie agency-side and interactive designers and developers who are normally in production, could be a lot closer within a project's lifespan."

"I've mentioned to many friends that even though the project has been a success, I still don't personally believe that it has the best design, best development, best concept or best anything. But what the project does have is complete and total unity between the different elements, which make up an interactive experience and I believe much like a sports team would, it becomes bigger than the sum of the parts through this unity."

"In my eyes, this is where especially designers and developers can work a lot closer than they do both in terms of what they do, but also their actual goals and wills for a project. I think if you mix genuinely world-class parts of the process within a relationship which is completely united, then the results end up spectacular."

tough-going for a few of the species and I had to be very careful to maintain the same style across a large set. Each animal was illustrated traditionally using Illustrator, then placed into the browser as a flat image for me to code up on top of, within the browser.

"Each species was then also produced into different sizes of wallpaper, and also the In Pieces poster design. However, in terms of visual design of the experience and especially the animation of elements, this was conceived on paper and executed straight in the code - and experimented with until the feel was correct."

Code is the backbone of any site and you can guarantee that every project is different. Every project will throw up a host of challenges though and of course In Pieces was no different. Even though CSS is the core component, the project couldn't be completed without the help and assistance of the ever-popular JavaScript: "The project had a major development hurdle to crack [like learning] how to draw so many species, when CSS polygons are created from three points made up of x and y percentages within a particular rectangle div. They couldn't simply be done in the browser until matching a flat illustration because it would have taken such a long time.

"So, I created a bespoke JavaScript function which essentially enabled me to trace over my flat illustrated images - three clicks or points at a time - to create each and every triangle making up the species. Each click would gain where the mouse cursor was within the context of a rectangle div at the point of click and place it into a line of

“I CREATED A BESPOKE JAVASCRIPT FUNCTION WHICH ESSENTIALLY ENABLED ME TO TRACE OVER MY FLAT ILLUSTRATED IMAGES – THREE CLICKS OR POINTS AT A TIME”

code. After the third click, I had an alert pop up to give me the line of code I required, something like this:

```
-webkit-clip-path: polygon( 47.470% 13.626%, 50% 50%, 52.9% 13.867%);
```

"No backend is in place in the project, but I do use a set of variables for all of the information within the website. When a species is changed for example, I use the 'number' of the animal you're going towards and match that up with an array of animal names to then change the name. That is one example of what is happening all over the site when a change of species occurs.

"The biggest developmental challenge to the project was undoubtedly generally dealing with a line of code so underused. This meant that when I had problems, there was nothing really online available to read to help me with issues. One example was that the polygon animations were much more sluggish and sometimes even glitchy, in Safari and on Retina devices.

"It took a long time for me to work things like that out - in this case it was that I use a vignette behind the species, which I ended up using only on non-Retina in Chrome. There is a lot of optimisation to make sure the fluidity of the story is kept as much as possible across devices."

RICHER REWARDS

Launching and marketing a personal project, can be very different to one being decided on by a committee. The creator has a far more personal connection to the project and the success of In Pieces was built on the back of a single tweet: "I'm personally marketing the site myself but as is the case when you design and develop something personally, the rewards are rich personally when things go well. I did worry for a long time over how I would get the project seen, and whether the topic - though important and highly topical - would also possibly be seen by online users as a little preach-y. It's genuinely a tough task to get people outside of digital interested in a topic or interactive project but this is where it's had particular success and I couldn't be more delighted at that being the case.

"This is one project which completely blew me away at how it took off though. Normally it's a case of some small-scale social success before needing to chip away at some online publications and trying to sweep up some traction that way. With In Pieces, I sent out a single tweet and it just seemed to spread from there which took my breath away. To see something fly so unexpectedly when you put so much time into something was really quite something mentally, and I'll never forget that rush!

"The best part of seeing how people have reacted however, is when people appear to have been genuinely moved. There have been some tweets saying things like they had goosebumps, they were almost moved to tears, that they were left heartbroken. Those sorts of emotions are difficult to engage with and that is I think where the real success in the project lies - outside of statistics or target numbers, that people actually had an emotional connection or reaction."



THE BIG QUESTION...

If you could only save one endangered species. Which one would it be?

🐼 The Vaquita. Vaquita is possibly the one species most perilously in trouble because they are impossible to maintain in a human-controlled environment, and so their survival lays completely with their own small numbers. It's a sad situation. 🐼

BRYAN JAMES



Digital alchemy

From their small studio in Shoreditch, Clubhouse's collaborative and open-minded approach bridges the gap between creative strategy, innovation and design to deliver crafted interactive products

Clubhouse Studios was set up to do things differently from day one, and the story of the four directors meeting and self-launching the company is no different. Back in 2011, cofounders Dan Coppock and Nick Mulley had been working at AKQA London for three years, and at the time held lead roles within a creative team tasked with redesigning the **Volkswagen.com** platform.

Like most creatives in large agencies, they'd been talking about making products and services outside of client work for some time. The two had been running small multidisciplinary design teams to great effect for global brands, but with **Volkswagen.com** launching after an intensive 12-month sprint, they decided the time was right to leave AKQA behind and see where their ideas and ambitions might lead.

Dan outlined their approach to the new agency: "This wasn't 'Field of Dreams', so we knew we couldn't just start a company and the work would roll in by itself. But we also didn't want to go down the investment route, primarily because it's kind of pointless to give parts of your company away before you've even started, and secondly because if we were going to do this, we wanted to do it our way, with work at the heart."

"We had decent personal portfolios, and although many companies use them in early new biz conversations, we didn't want Clubhouse trading off work completed at other agencies. Instead we launched with a portfolio piece that stood out from the crowd, showed we go the extra creative mile and that executionally we were up there with the big boys, and also to showcase an important part of the studio - making and releasing our own products."

"So we set about creating Equilibrium - a transmedia iOS release that would go on to be featured globally by Apple, but without investment, the only other commodity we had to trade was our time. So with a timeline agreed between the two of us, Ogilvy stepped in and offered us part-time positions while we got the company off the ground. We were there for nearly 18 months while development of the game progressed."

Dan continued: "At Ogilvy we helped run the creative accounts for Ford and five Unilever brands, during which we worked heavily with what would become the final pieces in the director puzzle: Rory O'Connor and Will Orrock. Rory was a creative director, and Will an art director at the time. They shared the same work ethic, the strategic approach to creative development, and our 'small-

who Clubhouse Studios
what The agency blends strategic thinking with world-class technology and interaction design, to create digital experiences people talk about
where 42 Redchurch Street Shoreditch London, E2 7DP
web weareclubhouse.com

Key clients

adidas
Ben & Jerry's
Dishoom
Unilever
Argos







team-big-things' methodology. Rory had been in the industry for ten years and Will had strong strategic and business interests stemming from his Product Design degree.

"Equilibrium launched globally in May 2013, and the four of us left Ogilvy, rolled up our sleeves and began pitching for business with the buzz of the game still resonating. Our first branded work with Lipton and Omo soon followed, and against the odds the studio was up and running in Shoreditch without ever needing that financial 'leg-up'."

Naming a new agency is sometimes achieved by accident or through many hours of agonising thought and debate. Nick explained their approach: "The Clubhouse Studios name definitely came before the URL. I think something like that has to really. For the studio we wanted a name that wouldn't date easily, wasn't overconfident, didn't necessarily pigeon-hole what we do, and nodded to the 'community' feel we try to cultivate in our teams and extended client family. Even the fact we call ourselves a studio sets the tone for the type of work we do and the environment we like to do it in."

"We threw a bunch of names around for a few weeks, trialed a couple with our friends and family, and Clubhouse was the one that stuck. URL-wise, we secured weareclubhouse.com because when you have a name like we do, it's often used outside of the context of your company, so we needed something that not only worked as our URL, but for any future services the company might want."

"Weareclubhouse as a handle has given us that flexibility. Our handle matches our domain cross-channel, which is the important thing. On Twitter, Facebook, Instagram, you name it, we're 'weareclubhouse' on them all."

An agency's website is its calling card, but how much maintenance does Clubhouse Studios' site require? "Our website is intentionally simple," said Will. "It focuses purely on our work as it's the truest reflection of what we do, the sort of people we are and obviously, the clients we are working with. We also share where we live, both in the real world and online, which all in all feels like enough to know if you like us or not."

"The 'agency website world' is pretty formulaic. There are definitely some great ones out there, but the vast majority we've all seen a million times before have paragraphs about what they do, icons above headline services and pretty profile pictures, which all generally proceeds the work. Everything is wrapped up with a wordy statement that is all too often meaningless - the infamous 'point of difference'. So in short, we use our work to tell our story. What we put out in the world is the best reflection of our reason for being."

From the very beginning Clubhouse wanted to trade on their ability to create new and innovative digital experiences. Turning their back on their previous lives and work was a bold step, but one

that has proven to be right judging by the calibre of clients they have attracted in the last few years.

"If we are honest, every client we have had has a different new business story," explained Will. "Overall, we have definitely learnt to keep our approach as fluid as possible. For example, due to our studio culture and team, ideas come from anywhere and everywhere. Often one of the guys will discover an insight or idea, which, if it's not relevant to a current client, we then share with a potential client to start a conversation. This beats the cold shout out, 'heard of us? See attached creds' hands down, and also allows them to visualise what the working relationship might look like before we're even through the door."

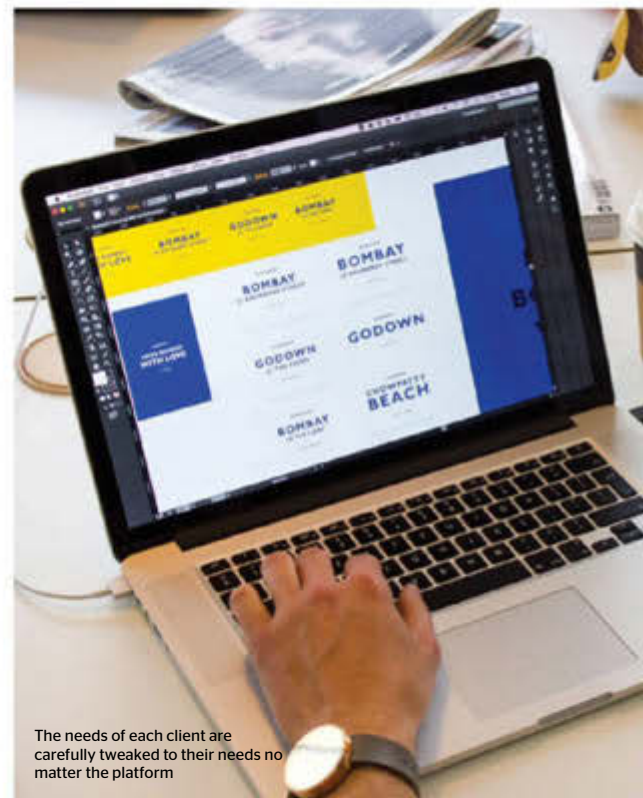
Dan also commented: "New business is also directly related to the type, ambition and quality of work you are putting out there. Pretty much like our site, the work is the best way to inspire, so sometimes we'll receive cold calls from clients looking to work with us, which is what happened when adidas invited us to pitch. On top of that we also do the standard pitching and responding to RFQs. If the client or project is a good fit for us and



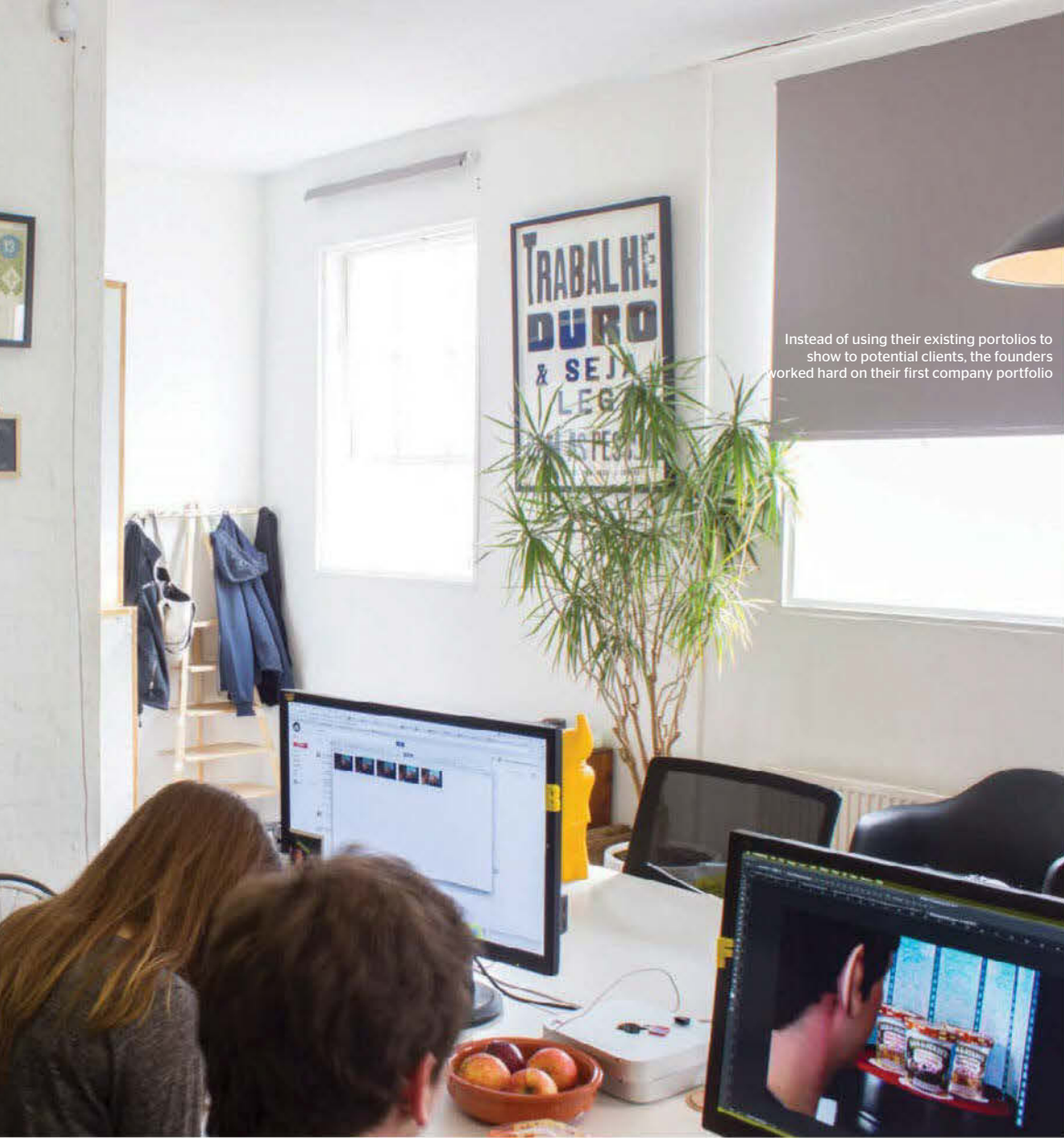
industry insight

Will Orrock
Managing partner

“Overall, we have definitely learnt to keep our approach as fluid as possible. For example, due to our studio culture and team, ideas come from anywhere and everywhere. Often one of the guys will discover an insight or idea, which, if it's not relevant to a current client, we then share with a potential client to start a conversation”



The needs of each client are carefully tweaked to their needs no matter the platform



Instead of using their existing portfolios to show to potential clients, the founders worked hard on their first company portfolio



adidas StellaSport Tumblr

adidasstellasport.tumblr.com

As part of adidas StellaSport's global launch we kick-started an initiative, including a bespoke zine with London creative Lotte Andersen, featuring interviews with empowered creative women exploring their interpretation of StellaSport. Lotte made the final pages live at a Topshop pop-up event.

Following the success of the zine we were tasked by adidas to create a digital hub for StellaSport. Inspired by the audience and our collaborators, we created and launched a brand-owned Tumblr site. It



was important for us that the design and UX was centred around native Tumblr functionality, using as little hard coded front-end as possible. It was also paramount to ensure we could control the narrative and the order content is read. In order to achieve this without adapting the front-end build each time, we



The team will spend time on R&D or developing interesting projects, so that their knowledge is always improving



'hacked' how Tumblr works using custom control hashtags. It lets us up-weight certain articles, overriding Tumblr's standard chronology. Finally, to make the content truly shoppable, we added shopping hashtags tagged with the outfit URL that add 'get the look' buttons to the posts dynamically.



the team are excited about the work, we're happy to dedicate resource and time to it with the risk it may never come in."

Choosing new clients to work with can be a multifaceted decision. Are there projects Clubhouse Studios wouldn't take on? Will explained their attitude: "So far, no. But, we wouldn't ever enter into a conversation with a client that we didn't want to work with. We're independent, so what's the point."

"In terms of a project size, we use the same ethos. Sometimes a low-budget project will come in and for one reason or another it strikes a chord with the studio. That happened with Dishoom, the London-based 'Bombay Café'. We loved the brand, their attention to detail, story and ambition so found a way to make it work from a business perspective. It all resulted in putting out some award-winning work we are all thoroughly proud of - so it's never black and white. New business has also directly fallen into the studio off the back of that work, so if we keep releasing products like Dishoom, it affords us the opportunity to be more choosy with other clients for sure."

Amongst the many projects completed by the agency is there one that typically defines the Clubhouse Studios ethos? "We haven't been around long," said Rory. "And we're pretty small, so every project has to be a home run, and for that reason we can

see our ethos in everything we have worked on. The clearest representation of that ethos is probably the adidas StellaSport brand launch in January 2015. When you are launching a brand globally you have to focus hugely on the consumer and let that mould your thinking and ultimately the work you put out. For StellaSport so far, we've launched a unique Tumblr site, created a printed zine, collaborated with Topshop and published a host of social activations - all because that's what our Generation Z audience can relate to."

How a studio is organised and which tools they favour is always a fascinating question to ask, as no two companies are the same. Nick explained: "The studio is one big room so the majority of our communication is just done face to face, which is the way we prefer it. For other bits we use the Clubhouse Slack account - it's great for file sharing, having dedicated project channels with the right people tuned in, and a pretty constant flow of GIFs!

"We have also just started testing out Wake as a hook up to Slack - it's pretty useful for sharing design work between the team. After using our sketchbooks and the enormous studio blackboard wall, we fire up 'old faithful' - the Adobe Suite - and

when needed, our designers are pretty handy with Unity and Cinema 4D for prototyping and 3D work. On the project management front, Jen (Harvey, project manager) works in Google Docs and OmniPlan so everything is always up to date and 'live'. Keynote is always the go-to for presentations."

As each project comes into the studio, how its particular nuances are approached is always different. Dan outlined their working practices: "Every timeline is different and driven entirely by what needs to be done. As we've delivered a broad range of projects over the last couple of years, it is impossible to pin down a particular formula."

"Our working process is a bit different. We hate everything about the old waterfall approach, or even what many teams think is 'agile' or 'collaborative', when they absolutely aren't. Instead of playing pass-the-parcel across the studio, we form a core team on day one that carries the parcel through the entire project. That means, a creative, a developer, a designer, a strategist and a 'whatever-is-needed' attend the creative briefing session and work through the project stages as a unit right up

"Taking Dishoom as an example, by using existing analytics, surveys and focus groups we identified behaviours across their three core segments; new customers, return customers and brand advocates. Each segment had a unique set of content requirements across different scenarios of use. For example, new customers on desktop were after menus as a priority, followed by imagery and locations, whereas the return customers were more interested in making a booking. In contrast, the third tier, brand advocates, typically visited on mobile (often directed via social) to read the journal or discover future events. Those needs and requirements were carefully considered during the UX and the structural design stage to ensure the experience felt tailored."

Using small well-focused teams is at the heart of Clubhouse Studios' success, but what are they looking for when recruiting? Rory explained: "Our belief is to start by looking at ourselves before finding the right people to join Clubhouse. We work hard to cultivate an environment in which creativity thrives, allowing everybody to share their unique

perspective. Being a smaller studio, we need people who can ultimately deliver but above that we are after the creatively unique; the sort of passionate people you put in a room together knowing they will come up with something genuine

We hate everything about the old waterfall approach, or even what many teams think is 'agile' or 'collaborative', when they absolutely aren't

to go-live and beyond alongside the client team. It ensures the final delivery looks great from the team, the consumer and the client's point of view - leaving everyone proud."

With a rapidly expanding toolset available to digital designers, which do Clubhouse Studios favour? "We use the same as everyone else for the good bulk of web projects," said Nick. "Things like three.js and Canvas enable us to do some pretty special things (like Argos Gift Guide and thehappyforecast.london). The web technologies toolset is versatile enough to cover anything from a standard responsive site, to a 3D immersive world. The tech guys really like React at the moment. It's maintained by Facebook, it's really fast and it's great for single-page applications. Tenon.io is also great for quickly testing your site's accessibility, which saves a lot of hassle as it's usually the 'boring' elephant in the room on any development project."

Design often begins with mobile. Is this how Clubhouse Studios approach their projects? Dan responds: "We use the same approach for mobile as we do for desktop and tablet - we start with the user's needs per channel, then tailor the content and design to best suit those needs."

and interesting. It's a little nonsensical, but we always know when we've just met a 'Clubhouse person'. Our advice to anyone looking to get into our industry is to become really good at something, but interested in everything. If you can inspire someone whilst showing a willingness to learn you're probably onto a winner."

And what does the future hold for Clubhouse? Dan concluded: "We've put a lot of the groundwork in place over the last couple of years. We've got a great studio and people, and the work we're producing for not only our amazing clients, but also ourselves, is something we're incredibly proud and excited about. We'll be looking to continue growing the team for sure. The next 12 months of client work is mapped out already with some really exciting stuff happening. The possibilities for what the team might do are endless, and most importantly, in their hands."

The sense of excitement permeates everything that Clubhouse Studios produce. Having strong innovation and a drive to build their own digital projects marks this studio out, as a group of exceptionally talented individuals that come together to create digital alchemy.

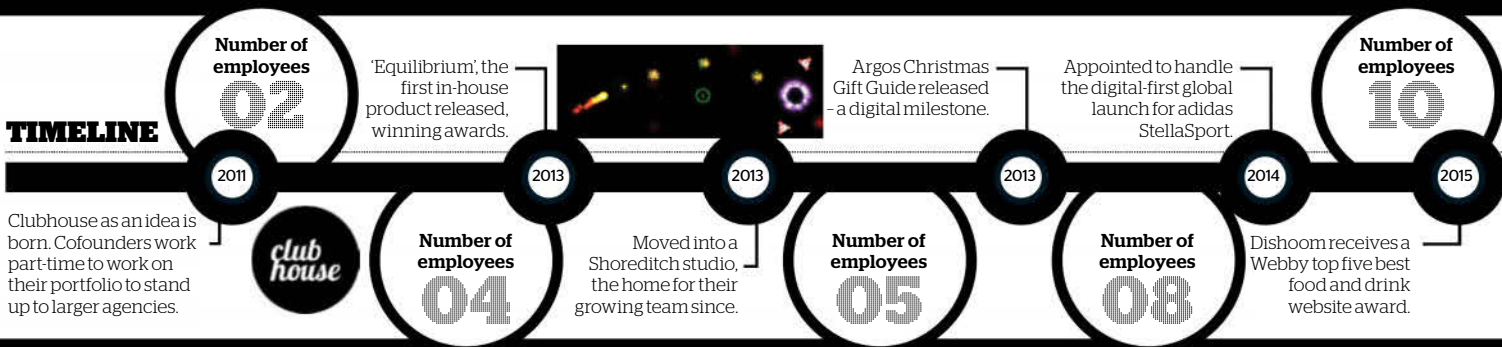
In terms of their working time, Clubhouse will try to split it into a 80:20 ratio for client and studio work



WEBweareclubhouse.com
FOUNDERS..... Dan Coppock, Nick Mulley, Rory O'Connor, Will Orrock
YEAR FOUNDED 2011
LOCATION Shoreditch, London

SERVICES

- > Digital strategy
- > Interactive design
- > Creative
- > Social
- > Product development





DESIGN

{ LIKE A PRO }

CODE

< LIKE AN EXPERT >

**Get your hands on a host of HTML5 tips, trick
and techniques to empower your frontend
and backend**



Using <article> for semantic components

THE <ARTICLE> TAG IS ONE OF THE MOST MISUNDERSTOOD ELEMENTS OF HTML5

Marking up widgets and components can be tricky to get right semantically. Components will vary significantly in structure and design, even when it's throughout the same build.

Unfortunately, this is at odds with what we actually want as developers: a consistent maintainable approach with semantic value. Well this is where the <article> tag comes into play.

"The article element represents a complete, or self-contained, composition in a [...] site [...] that is, in principle, independently distributable or reuseable, e.g. in syndication. This could be [...] a magazine or newspaper article, [...] an interactive widget, [...] or any other independent item of content" as defined by the HTML5 specification at w3.org/TR/html5/sections.html#the-article-element.

The specification makes it very clear that the <article> tag has a much wider use case than is first obvious. This makes it the perfect semantic tool for components, which we often design in modular fashions.

The HTML5 specification also details that the <article> tag's content model is flow content, which is the way we mark up most of the elements that are used in the body of documents. This means that the <article> tag becomes the semantic choice when we want to contain modular content and even functionality such as things like forms and navigation.

Of particular note is that the <article> tag can contain the new HTML5 elements header, footer, aside and more instances of itself - allowing for further semantic structure inside your reusable component. This provides us with semantic building blocks for complex modular UIs such as faceted search, accordion menus and tab systems - perhaps even all three at once (though this is not advised).

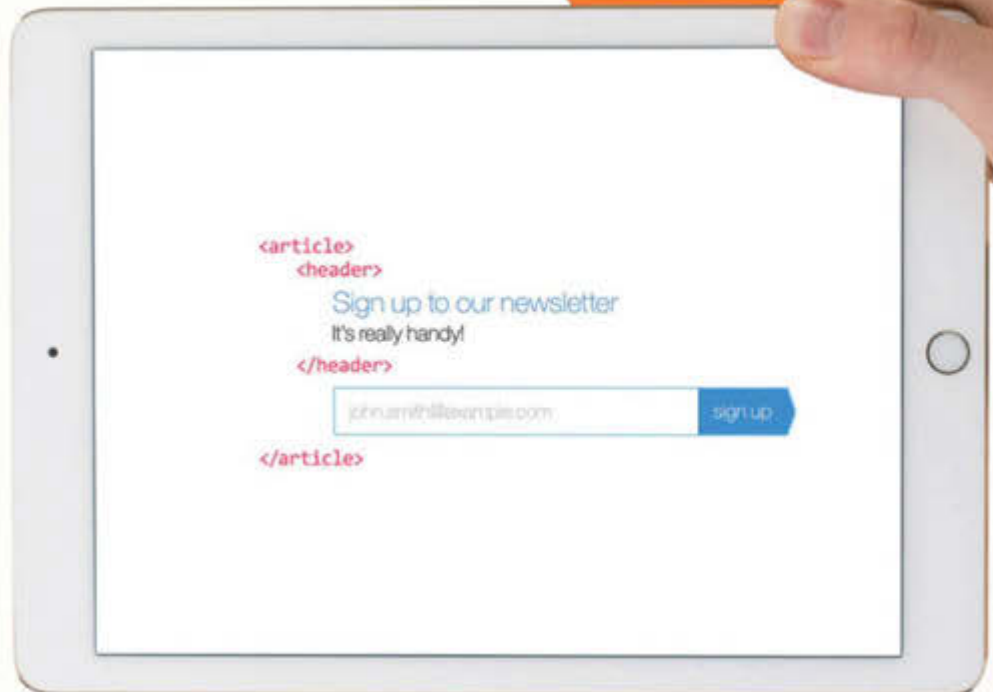
In many ways <article> is a 'do everything' tag. It's important to remember that the <article> tag brings semantic value only to components that can be isolated from the rest of the page.

It is not the right tag for supporting 'pods' or other components that only make sense in situ or are otherwise relying on surrounding content.

Likewise, the <article> tag is very unlikely to be the right tag in cases where the component is semirelated but also independent, for example in a pull quote. In these scenarios then, the <aside> tag is likely the best choice. Though it's worth considering that aside elements can contain article elements - enabling the semantic markup of any tangentially related content which also containing reusable components.

AUTOCOMPLETE TURNED OFF

Most browsers now have an autocomplete defaulted to 'on' for <input>. Turn this off for things like reset password fields.



“The <article> tag becomes the semantic choice when we want to contain modular content and even functionality”



“It’s a little detail a lot of websites miss, but form fields should only autocomplete information that is likely to be correct, lest users spend time deleting text that shouldn’t be there in the first place.”

Ralph Saunders
Front-end developer
at Redweb

Misunderstood HTML5 tags

SOME HTML5 ELEMENTS AREN'T WHAT THEY FIRST SEEM

Section

The `<section>` tag provides a semantic way to markup thematically grouped content that should be in the document outline (so it should have a heading). The `<section>` element would be perfect for marking up chapters in a book for example. We can also make use of it inside flow content, where we enable the clear separation of many sections in a lengthy article - such as a thesis or dissertation.

Footer

The `<footer>` tag is not for website footers but for footer content that relate to the nearest sectioning element (these are the four sectioning elements: section, article, aside and nav). The footer can also be used for content such as that of bylines, related documents and copyright data. It can also contain entire sections and this will then lend itself to things like appendices, indexes and licence agreements.

Header

Like the `<footer>` tag, the `<header>` tag isn't for website headers - as you would be forgiven for thinking. It's for introductory content relating to the nearest sectioning element. It can be used for larger-scale content such as a table of contents or foreword, but is more commonly used for headings and standfirst paragraphs.

Address

The `<address>` tag is not just for addresses you can post letters to. It is also contact information for the nearest `<article>` or even `<body>` when it's applicable to the document as a whole. While this may be a postal address sometimes, it's more likely an email address, phone number or link to the author's page.

Nav

The `<nav>` tag can be used for primary navigation, but it should also be used to markup any navigation on the page. Secondary navigation, even when internal to the page, should be marked up with `<nav>` - like jump links. If the nav contents represent a list of items they should be marked up with a list as well.

HTML5 CHEAT SHEET
 There are a lot of tags in HTML5, some popular and some not so well known. A quick reference to a host of HTML5 tags is just the job. Get your hands on one at websitesetup.org/html5-cheat-sheet.



ARIA landmarks

USE ARIA LANDMARKS TO IDENTIFY REGIONS OF A PAGE

ARIA stands for Accessible Rich Internet Applications and is a set of attributes for making markup more accessible. The basic premise is that these attributes can help describe the components on a page for screen reader applications. A subset of these attributes are called landmark roles, and they indicate navigational landmarks like in this code example:

```
<article role="main">
  <header>
    <h1>Awh, kitteh</h1>
    <p>We've just named our new cat Jimmy!</p>
  </header>
</article>
```

Here we're declaring this section of the page the 'main' content of the document - it contains the content you came to the page for. There are a few more landmarks that should go into any web project.

'Banner' declares that the region is mostly site oriented as opposed to page oriented. This can be applied to site headers where the logo, primary navigation and search are site specific.

'Complementary' declares a region that's outside the main content but complements it and would remain meaningful in isolation. Most sidebars fit this criteria.

'Navigation' declares the region contains navigational elements for navigating the page or related pages. If you're using `<nav>` you should be using `role="navigation"`.

Optimise page structure for Time To First Paint

OPTIMISE FOR TTFP TO GET MORE PAGE VIEWS AND IMPROVED CONVERSIONS

TTFP is the metric for measuring how fast the browser was able to paint the page. We want to minimise the time a user spends looking at a white screen by optimising assets and the order in which they're loaded.

When visiting a page the browser must download the HTML of that page before it fetches the CSS and JavaScript assets linked within it. There are a couple of things we can do to ensure only the minimum number of assets are fetched before the browser paints. Most people know that JavaScript should go at the bottom of a page just before the `</body>` tag. What's less known is

that if your JavaScript isn't interacting with the CSSOM you can include an `async` attribute to turn it into a nonblocking request, meaning it won't hinder TTFP.

```
<script src="app.js" async></script>
```

We can also use the `media` attribute in links to CSS assets to make them nonblocking.

```
<link href="print.css" rel="stylesheet" media="print" />
<link href="desktop.css" rel="stylesheet" media="(min-width: 1024px)" />
```

Desktop.css won't delay TTFP for devices that don't meet the requirements of the media query, it gets loaded later.



HTML5 input types for better keyboards

BRINGING NATIVE INPUT VALIDATION, AND UX GOODIES TO THE TABLE

Mobile devices often have small keyboards where data entry is made all the harder. We can ease our users' pain by using the new input types that come with HTML5, which can change a keyboard layout intuitively on some mobile platforms out there.

While using these types correctly won't fix a bad form, the different layouts go a long way in making forms less stressful to use on small devices. Some of the more common input types you can include in your forms straightaway are below:

`<input type="tel" name="telephone">`

Tel inputs will bring up a numeric keyboard. No more pesky small keys when entering phone numbers. No format validation is enforced with this type.

`<input type="number" name="amount of pets">`

Number inputs will also bring up a numeric keyboard. Only numbers can be put into this input.

`<input type="email" name="email">`

Email inputs will bring up a keyboard with an @ sign.

Browsers will do email address validation on this input type.

`<input type="url" name="website">`

Url inputs will bring . / and .com buttons to the interface. The browser will validate that this looks like a URL.

`<input type="date" name="date of visit">`

Date inputs will bring up a date picker on modern desktop browsers and a scrolling interface on most mobile devices.

“Different layouts go a long way in making forms less stressful”

Use rel="prefetch" to enhance browsing experience

PREFETCH CAN SPEED UP BROWSING EFFECTIVELY

HTML5 prefetching uses the rel attribute on <link> tags and looks like this:

```
<link rel="prefetch" href="/Page2.html" />
```

Page2.html will be fetched by the browser when nothing else is happening and it's fetched before the user decides to go there. Server response time is often the largest delay in fetching new pages and other assets will be cached, the savings here are substantial.

URL prefetching is a double-edged sword. We're downloading a resource that the user may not ever navigate to and on a mobile network it'll count against your user's data allowance. With this in mind, we should only prefetch resources we're pretty sure the user will visit. There are some common resources that are always worth prefetching:

1. The next page in paginated search results
2. The next page in sequential multipage forms
3. Pages with only one major call to action that the user will likely visit eventually
4. Common images on most pages, like spritesheets.

Along with standard prefetching there's also DNS prefetching which works in a similar fashion.



Darren Hickling

Web engineer at Redweb
“HTML5 introduced new input types, but also new attributes for them. Alongside pattern, these new input types have proven very useful in recent projects. In supported browsers, they can also remove the need for custom JavaScript validation and are enforced by the browser so they cannot be ignored.”



Phil Heywood

Creative director at Redweb
“Web pages using dynamic content and user interface components used to cause problems for people with certain accessibility needs. HTML5, in conjunction with WAI-ARIA, makes it possible for everybody to use rich Internet applications, even if they rely on screen readers and other assistive technologies.”



Ryan Kerry

Global head of development at Lush
“HTML5 introduced the contentEditable attribute, which enables users the ability to alter the contents of a HTML page. This has proven to be extremely useful when required to perform rapid in-browser prototyping of web pages that require real-time updates to the CSS output for clients to see.”



Jonathan Ginn

Developer evangelist at We Are Base
“One of my favourite underutilised new HTML elements is the datalist. It's great for letting someone perform a search while offering find-as-you-type options, all in HTML5 with no JavaScript. It's not fully supported yet but like all HTML5 elements, it has graceful fallbacks for older browsers.”

Back-end APIs for apps

USE HTML5 TO MAKE APPS WITHOUT NATIVE PLUGINS

WebRTC

Don't be fooled, WebRTC has nothing to do with real-time clocks. Instead, brace yourself for a standards-based way to perform voice and video calls using nothing but a web browser. When sufficient traction has been gained, Web Real Time Communication might swipe away Skype, MSN Messenger and other proprietary protocols. Sadly, its adoption has been slow so far - it is limited to Chrome, Opera and Firefox only.

WebWorkers

JavaScript and multithreading are like cat and water: when the language was first designed, no one thought of multicore processors.

WebWorkers seek to remedy this design deficiency by introducing a special and convenient way to run JavaScript logic in a separate thread. A message-passing protocol is then used for data exchange between the worker and the main app.

Shadow DOM

Even though Web Components have not been formalised as of this writing, the shadow DOM feature already deserves attention. It permits the isolation of subcomponents from the rest of the page. When stored in a ShadowRoot instance, your widget's elements are shielded from well-meaning but badly written enumeration routines seeking to wreak havoc on their finely tuned composition.

Typed arrays

If JavaScript wants to become a full-featured programming language, a way must be found to enable interaction with native libraries. Sadly, its untyped memory model makes the management of memory buffers difficult. In essence, a typed array is little more than a broker between a slab of memory and JavaScript's native types. You can use them to create arrays of floats or integers though, and these can then be passed onto any native functions.

File API

HTML5 introduces a plethora of storage systems. Sadly, sometimes there is just no alternative to a good old file. The File API permits you to access pesky little and chunky large files sitting in the file system of your client's workstation. It comes complete with a set of widgets to make selecting the files easy.

Where are we?

THE GEOLOCATION API HELPS YOU FIGURE OUT THE USER'S LOCATION



1. Where is our user?

Even though the HTML5 Geolocation API has seen some impressive implementations, some browsers still don't implement the library correctly. The first step to a working GIS application involves the querying of the presence of the geolocation object. If found, the location getter methods can then be invoked. GetCurrentPosition takes one, two or three parameters: one method which will be called on success, an optional second one which will be called on failure and an optional JSON object with further parameters to set up accuracy and data source selection.

2. Handling all errors!

Getting a user's location is all but foolproof. The optional error handling method receives an object containing four different error codes, eg PERMISSION_DENIED is returned if the user denies the location request shown as the website tries to access the Geolocation API for the first time. Note that some browsers don't return an error if the user refuses to divulge to your application. On such browsers, the success callback simply doesn't get invoked - one way to handle this behaviour is with setTimeout. It can spin a checker to inspect whether success and/or failure have invoked.

3. Parsing user data

If the locationeering process was successful, you are provided with a position object. It contains the set of fields shown in the following table:

Field	Data
accuracy	Accuracy of Lat and Lon data
altitude	Altitude from mean sea level
altitudeAccuracy	Accuracy of altitude
heading	Heading, degrees from north
latitude	Latitude
longitude	Longitude
speed	Movement speed
timestamp	Age of data

4. Caveat emptor

Keep in mind that the actual implementations differ on legacy browsers. For example, Internet Explorer did not support the official Geolocation API until IE9. Fortunately, an intermediary broker layer has been developed. It acts as a relay station between your code and a variety of proprietary implementations. Further information on it can be discovered by visiting the website of the project (github.com/estebanav/javascript-mobile-desktop-geolocation). Another source of grief involves the accuracy of the data: most desktops are located via their IP address. This can lead to significant accuracy issues.

10 HTML5 tips

MAKE THE MOST OF YOUR BACKEND WITH THESE TRICKS

HTML Imports are dead

During the specification of Web Components, Mozilla came up with an include directive for HTML files:

```
<link rel="import" href="myfile.html">
```

Sadly, they never managed to pick up any form of significant traction. Since Firefox dropped support for them some time ago, it's better to leave them alone.

Force some input

HTML5 saves validation effort by informing the browser that some fields must be filled:

```
Name: <input type="text" name="mandatoryname" required>
```

By default, most browsers will display an exclamation mark in the field as long as it is empty.

Force a pattern

For some types of data, a predefined format is very helpful for us as it can be enforced in order for us to weed out any invalid input from our code. Take a look at our snippet below because it shows how HTML5 can be harnessed in this instance to ensure that our users don't end up accidentally inputting non-IPv4-formatted addresses:

```
<input type="text" pattern="d{1,3}\.d{1,3}\.d{1,3}\.d{1,3}">
```

Annotate data

HTML always found itself in a pressure field between verbosity and compactness: adding new tags makes the language richer, but increases complexity.

Microdata solves this by specifying an external format for metadata. The markup is then enriched with tags referencing the metadata stylesheet:

```
<section itemscope
  itemtype="http://data-vocabulary.org/Person">
  <h1 itemprop="name">Mark
  Pilgrim</h1>
```

Specify your own

All kinds of actors have created a large selection of metadata stylesheets, which can be embedded into your designs easily. But of course, a custom stylesheet is needed in some sticky cases. More information (as well as a large range of information on the microdata data model, markups and snippets) and these can be found at the website of Mark Pilgrim, which is available at diveintohtml5.info/extensibility.html.

Using gauges

Digital aeronautical systems have always suffered from a weakness of the human brain: textual information is more difficult to parse. Fortunately, Gauges can be created on HTML5-capable browsers by using the following, widely supported tag:

```
<meter value="6" min="0" max="10">6 of 10</meter><br>
```

<Progress> and <meter>

Even though there is nothing to stop you from using a Gauge as a progress bar, HTML5 provides a dedicated option. Its use is demonstrated in the following snippet - keep in mind, that some versions of Internet Explorer don't support the <progress> and <meter> tags:

```
<progress value="66" max="100">
  </progress>
```

Check your spelling

Typos can and will happen in the best of houses. Some browsers include a spellchecking engine, which can be set loose on all kinds of input fields. Enabling the spellchecker is as easy as setting the following extra attribute: `spellcheck="true"`

Stay compatible

Not every browser supports every part of each and every HTML5 specification. **Caniuse.com** is the semiofficial go-to point for developers wondering whether their feature of choice will impede compatibility with the common and not so common browsers too.

Don't get hacked

Keep in mind that any and all HTML5 components are pure client-side stuff. The availability of open source browsers has enabled hackers to create an 'evil browser' which does not comply to standards - skipping server-side validation is a sure-fire way to ask for pain.

5 Tools

FIVE ADDITIONAL FEATURES TO MAXIMISE YOUR WORKFLOW

Keep them offline

Caching has been around since the glory days of primitive web browsers. HTML5's application cache makes the cache engine addressable. Your code can inform the browser about elements which should be cached. The most common use case involves the creation of web apps, which keep running if deprived of their internet connection. In addition to that, the API can also be used to cache images.

Web storage bingo

In many cases, persistable data consists of but a few small tuples. In this case, the Web Storage API is the tool of choice. It provides a simple KV-Store, of which values can be persisted across sessions or across invocations of the web application. Beware, however, that most browsers tend to put tight constraints on the maximum size of the storage partition.

Database ahoy!

Sometimes, databases are needed to handle large and/or more complex data scenarios. The IndexedDB API provide an object oriented NoSQL database which can be used even in offline mode. As with web storage, be aware that the amount of space available to your application is likely to be limited in order to prevent local denial of service attacks by malicious or badly-designed websites.

Modify the back stack

Windows Phone 7 acquainted users with the back button. After having used one of Microsoft's handsets, users expect the back-button to work coherently. The History API permits you to manipulate the history of the browser. You can modify the address shown in the address bar, but you can also insert steps into the back queue just in case the user ever feels like backtracking through a process.

Even more technology

The web is evolving at an amazing speed. If you want to know more about new technologies, take a look at Mozilla's WebAPIs (developer.mozilla.org/en-US/docs/WebAPI) as it's very interesting to read. The company is in the unique position of having to maintain a completely web-based operating system. This unveils interesting problems normally not faced by browser vendors. Plus, Mozilla's drafts have been accepted as standard frequently.



Go 3D with WebGL

TAKE THE HARD WORK OUT OF CREATING 3D BY FIRING UP THREE.JS

Venturing into the third dimension has always fascinated programmers and designers. Even though 3D engines have been implemented successfully using nothing but JavaScript, their performance was incredibly low due to the lack of hardware acceleration. Fortunately, a technology called WebGL seeks to ameliorate this issue. On a supported browser, WebGL code is passed directly to the GPU and is run at speeds almost comparable to code written in DirectX and/or OpenGL.

Sadly, handling WebGL on its own is tremendously difficult: if you ever bothered yourself with creating OpenGL or DirectX rendering code, you are very well aware of the endless amount of pain hidden in this particular field of computer science.

Fortunately a nice wrapper framework called three.js is available. It acts in a fashion similar to XNA, providing

developers with a user-friendly frontend and taking care of the ugly and gory mathematical and computer scientific details transparently in the background.

While running WebGL on the desktop is by and large not an issue, support in the mobile space is spotty at best. For example Firefox OS was unable to accelerate WebGL scenes until the arrival of Firefox OS 1.2 - developers and users stuck on older versions of the OS need to live with agonisingly slow performance. In addition to that, the very popular Opera Mini browser also has its fair share of issues with WebGL. WebGL is tremendously cool, as can be seen by looking at the rendering code and the screenshot shown above. Sadly, its compatibility in the mobile space is limited - but since Adobe has abandoned Flash for mobile when it met resistance from the late Steve Jobs, developers have little choice.

Code library

For the full code in this library, check out our FileSilo

```

var renderer = new THREE.WebGLRenderer({ alpha:
1 true });
renderer.setClearColor(0x0055ff,1);
renderer.setSize( window.innerWidth, window.
innerHeight );
var loader = new THREE.JSONLoader();
2 loader.load( "Monkey.json", function( geometry
)
{
geometry.computeBoundingBox();
mesh = new THREE.Mesh( geometry, new THREE.
MeshNormalMaterial() );
mesh.scale.set( 10, 10, 10 );
mesh.position.y = 0;
mesh.position.x = 0;
scene.add( mesh );
3 Var pointLight = new THREE.
PointLight(0xFFFFFF);
...
for(i=0;i<9;i++)
{
var geometry = new THREE.TorusGeometry( 10, 3,
16, 100 );
geometry.computeBoundingBox();
var material = new THREE.MeshLambertMaterial( {
color: 0xff00ff } );
4 var torusMesh = new THREE.Mesh( geometry,
material );
torusMesh.position.x=2+getRandomNumber(30);
torusMesh.position.y=2+getRandomNumber(30);
torusMesh.position.z=2+getRandomNumber(30);
torusMesh.scale.set(0.5,0.5,0.5);
scene.add(torusMesh);
} );
camera.position.x = 20;
camera.position.y = 20;
camera.position.z = 80;
document.body.appendChild( renderer.domElement );
function renderAll()
{
requestAnimationFrame( renderAll );
}

```

1. Three.js data

Displaying data with three.js requires the presence of renderer, a scene and a camera. The scene is a keeper element for the various objects - 3D artists refer to them as models - which are to be displayed in the scene. Cameras are a digital representation of the human eye looking into the scene and onto its contents from specified position. Finally the renderer is responsible for generating the picture seen by the virtual camera camera using mathematical processes.

2. 3D models

Displaying 3D scenes requires models and lighting. Our code starts out by using the JSONLoader on a predefined and relatively complex model created with Blender. When the loading is complete, the object is scaled to make it appear bigger and is provided with a set of position coordinates. Finally it is added to the scene, where it will be displayed later. In the next steps, similar processing takes place for the light source and the torus objects, which are to act as surroundings for the monkey.

3. Materials

The generation of the individual torii is interesting for multiple reasons. First, the corresponding geometry class generates a new skeleton of the desired element. This doughnut-shaped object is but a wireframe, bare of any physical appearance. Making it displayable is accomplished by adding a material. You should imagine MeshLambertMaterial to be like a skin of silk paper glued onto the torus. Then, a custom randomisation function is invoked to generate random coordinates.

4. Rendering

In the final step, the renderer is set up so that the scene is actually shown in the DOM tree. Here, the rendering must be requested in a relatively complicated fashion and that's because most browsers don't actually permit code to start rendering whenever it feels like doing so. Instead, a frame must first be requested for the rendering process to happen at all. A rendering handler will then be invoked when the graphics pipeline is ready to accept further data.



Power up HTTP

FIX HTTP AND REAL-TIME DATA EXCHANGE WITH WEBSOCKETS

Establish a connection

Communicating via WebSockets requires the presence of a connection. Fortunately, establishing one can be accomplished by creating a new instance of the WebSocket object. Pass in the URL or the IP address of the socket, and you're set to go.

```
function testWebSocket()
{
  websocket = new WebSocket(<url>);
}
```

Set up event handlers

Socket connections are asynchronous and event-driven by definition. Thus, the next step to WebSocket goodness involves the assignment of event handlers to the various members of the newly created WebSocket instance.

```
function testWebSocket()
websocket.onopen = function(evt) { onOpen(evt)
};
websocket.onclose = function(evt) {
onClose(evt) };
websocket.onmessage = function(evt) {
onMessage(evt) };
websocket.onerror = function(evt) {
onError(evt) };
```

Send some data

Transmitting information to the server is really easy: you can simply invoke the send function with a string or JSON object of your choice. Beware though that some browsers don't actually permit the transmission of objects, and so in that case, you can simply serialise them all by hand.

```
websocket.send("Hello World");
```

Receive server commands

Reacting to data coming from the server is as simple as using the onmessage event handler. Our snippet below prints the data to the console - when we are transmitting objects, the next step would be a classic application for deserialisation.

```
websocket.onmessage = function (event) {
  console.log(event.data);
}
```

Roll your own

WebSockets are supported by most server side development tools. Both Node.JS and Qt can be made to provide a WebSocket server right out of the box. Furthermore, third-party client libraries are available for all the major operating systems.

RESOURCES

THREE ESSENTIALS TO KEEP YOU CODING LIKE AN EXPERT

Website HTML5 Index html5index.org

The HTML5 Index is to JavaScript what the MSDN is to Windows Phone: a never-ending selection of interesting tid-bits, resources and other interesting stuff. The Index has helpful links on each term which corresponds with other technical terms. It even separates the terms by library for easy reading. It furthermore contains a detailed list of parameters and official API specifications which simplify coding.

YouTube channel Google Developers youtube.com/user/ GoogleDevelopers

Even though the big G's channel can become quite Android-centric at times, it nevertheless provides an attractive selection of amusing videos teaching new technologies and tricks of the trade. There are informative tutorial series like Developer How To or there are videos more aimed at the latest updates available like the 2015 I/O keynote speech. Be prepared for a deluge of amusing content - but watch that data cap if you're watching on mobile!

Twitter MozAppsDev twitter.com/mozappsdev

Mozilla's developer advocates are a never-ending source of all kinds of interesting stuff on the open web. Be it a tutorial, an interview, some musings on a RFC draft, a feature on a new standard or library: joining their 30,000 followers is a decision which you will surely not regret. They'll even advertise for job vacancies on there if you think you'll fit the bill.

Graceful degradation

When working on end-user-facing applications, an increase in compatibility tends to lead to a direct increase in earnings. Mobile developers have approached the problem of new APIs with gradual degradation. It is a straightforward technique based on the assumption that some features are not entirely necessary for application execution. Smart apps and websites offer their users a reduced featureset when run on legacy browsers.

Lock your mouse!

Pointer Lock solves the problem of being unable to control the cursor fully in web games. Using this snippet 'standardises' the vendor-specific property names with a common field: `canvas.requestPointerLock` = `canvas.requestPointerLock` || `canvas.mozRequestPointerLock` || `canvas.webkitRequestPointerLock`; This needs a call to `requestPointerLock`: `canvas.onclick = function() { canvas.requestPointerLock(); }`

Make it talk!

Really cool computers like Kit talk to their owners. The Speech API pulls off a similar trick, best accomplished like this:

```
var sayWhat = new
SpeechSynthesisUtterance("Let us
say something!");
var voiceArr = window.
speechSynthesis.getVoices();
sayWhat.voice = voiceArr[10];
sayWhat.pitch = 4;
sayWhat.rate = 10;
window.speechSynthesis.
speak(sayWhat);
```

Listen up, Scotty

Some browsers also go the other way around using a proprietary server provided by Google. Their speech synthesis API usually takes a grammar, which is then analysed against the aural input. This mode cannot be used for free dictation - it is limited to formatted attributes from a known list. Due to the grammar in the background, it tends to be more reliable than free speech mode: entering village names, part IDs or airport codes should never be attempted in 'free scanning'.

Sync animations to audio and video with Popcorn.js

Trigger events easily at any timeframe and play back music or video by using Mozilla's Popcorn.js





nce upon a time syncing up music or video with animations, overlays or other content could be done quite easily with Flash, but since the demise of that

plugin it's become harder to achieve this kind of effect.

HTML5 has given us a great way to natively play audio or video in the browser but it's still a hassle to try and get something to happen at a key point in your media. Fear not though because Mozilla has come to the rescue with their media library Popcorn.js. This has a very simple premise to enable the developer to code content that can be triggered at various times during the playback of their media. If you've ever seen interactive music videos by Arcade Fire and Rome for example, then this is the sort of library that enables the triggering of new scenes or animations at key points in their music videos.

In the tutorial we are going to have a 3D scene in the background of the page, created in WebGL with three.js. Over the top of that will be some regular DOM content, which will be hidden. Popcorn will be used to load audio, play it and then trigger camera moves in the 3D content, while also fading in with CSS transitions, with the regular div-based content over the top.

1. Up and running

Open the start project folder in Brackets or a similar code editor and open index.html. Scroll down to the body tag and add the following div tags for our on-screen messages that will be shown at key points with the music. Copy and paste this code three more times and change the id to two, three and four.

```
<div id="one" class="outer hide">
<div class="middle">
<h1>&ldquo;Globally, loss-related floods
have more than tripled since 1980&rdquo;</>
<br><small>Munich Re Insurance</small></h1>
```

```
</div>
</div>
```

2. Link to popcorn

Change the text inside the <h1> tag for each of those copied sections and you can refer to the text in the finished file or add your own. A little further down the body section you will see a bunch of script tags for the 3D animation, add this at the bottom to link to the Popcorn library.

3. Add the music file

Scroll back up the page to just under the div tags from Step 1 and add this HTML5 audio tag to link to the MP3 audio. Later on, this audio tag will be hidden on the screen using CSS and the Popcorn.js library will link to this so that animation can be triggered based on the current time of the song playing.

```
<audio id="myAudio">
<source src="autumn-leaf.mp3" type="audio/
mpeg">
</audio>
```

4. Start styling

In the head tag on the page, add an opening and closing style tag, then add the following CSS into that. Here the body tag gets the padding and margin removed and the right typeface is set for all content. Any content that overflows the page will be set to hidden, as is the audio from the previous step.

```
body {
padding:0;
margin:0;
overflow:hidden;
font-family: 'Oswald', sans-serif;
color:#ffb000;
```

```
}
audio { display:none; }
```

5. CSS animation

Now the code adds some very simple CSS3 animation. The hide class has been applied to all the content in the body section of the page. A transition for the opacity is applied, which sets the length to half a second. When content needs to fade in, the show class can be applied to the div and it will fade in, taking half a second to do so.

```
.hide {
opacity: 0;
transition: opacity .5s ease-in-out;
-moz-transition: opacity .5s ease-in-out;
-webkit-transition: opacity .5s ease-in-out;
}
.show {
opacity: 1;
}
```

6. Position the messages

As there will be animated 3D content using WebGL in the background, any new content to be displayed needs to be shown over the top of this. As such the z-index, which is like the height, is set to be higher than the rest of the page. This section is set to fill the browser width and height, positioned absolutely in the top left.

Familiar terminology

If you've ever worked with audio or video the term 'cue' means to start a piece of media and hence the code to start something happening is also 'cue' in Popcorn.



Left

The camera is called to start animating immediately as the song starts playing, and you will see the 3D scene zoom closer so that the houses are the focus of attention

Top left

The Popcorn.js library can be downloaded from Popcorn.js.org. There are also some examples so that you can understand how to sync events with the audio

Top right

Once the content is all added to the page, the 3D scene is called to display after the music has finished loading

Tutorials

Sync animations to audio and video with Popcorn.js

```
.outer {  
  z-index: 10;  
  width: 100%;  
  height: 100%;  
  position: absolute;  
  top: 0; left: 0;  
}
```

7. Vertical alignment

Inside the fullscreen div of the outer created in the previous step is another div. The text inside here should appear in the centre of the screen horizontally and vertically. The middle class here will ensure that happens by aligning it on the vertical axis with the page centre.

```
.middle{  
  min-height: 100%;  
  min-height: 100vh;  
  width: 100%;  
  display: -webkit-flex;  
  display: flex;  
  align-items: center;  
  -webkit-align-items: center;  
}
```

8. Set the headings

All of the text on the page is set with headings so here, the CSS for that is set. Because the background has a

number of colours, there is a text shadow on the text to help it stand out against the background. The size of the text is also increased and centred horizontally.

```
h1{  
  font-size: 3.8em;  
  display: inline-block;  
  width: 40%;  
  margin: 0 auto;  
  text-align: center;  
  text-shadow: 3px 3px #000;  
}
```

9. Wait for the audio

Before the audio is set to play, it is important to ensure that it has fully loaded and can be played. Add the code here to just under the existing script tags on the page. This will load all of the scripts and content in the body tags before calling the init function. The init function will show the 3D background, so test that in the browser.

```
<script>  
  document.addEventListener("DOMContentLoaded", function () {  
    init();  
  }, false);  
</script>
```

10. Calling the music

If you've loaded the content from a local server or webhost, the 3D scene will show, but no music is playing just yet. Just under the init(); line in the previous step add the next line, which will call a function to play the music. This hasn't been created yet so don't run it in a browser.

```
■ initMusic();
```

11. Link and play

Before the closing script tag in Step 9, add the following

function. This links to the audio with the id of myAudio, which was added way back in Step 3. Once the link is established, the audio is set to play. Save the document and try this in the web browser, the music will start to play and the 3D scene is visible in the background.

```
function initMusic(){  
  popcorn = Popcorn( "#myAudio" );  
  popcorn.play();  
}
```

12. Call the first animation

The scene needs a little movement so that the first message can be displayed. Add the line below inside initMusic. This is calling a function on line 111 of the scene.js file if you want to look at it. Save this and refresh your browser and the camera will move forward in the scene.

```
■ camMove1();
```

13. Cache the divs

To speed up DOM manipulation of adding and removing classes, variables are created to hold the four div tags that contain text. The 'show' class will be added and removed and this will make the appropriate message fade in or fade out at the appropriate time. Add this code inside the initMusic function.

```
var one = document.getElementById( 'one' );  
var two = document.getElementById( 'two' );  
var three = document.getElementById( 'three' );  
var four = document.getElementById( 'four' );
```

14. Sync to music

All of the remaining code in this tutorial is added before the closing bracket of the initMusic function at each subsequent step. To make something happen in time to music, the following code is used. This calls its own function after one second of the music playing. Test in

Additional plugins

There are additional plugins for Popcorn that allow you to run video from hosted sources such as Vimeo, YouTube and SoundCloud. It's even possible to open Google Maps.



Top left

As the bass starts in the music, lights in the scene are turned on and off to look like lightning flashing. Popcorn makes it easy to sync the timing of this to the music

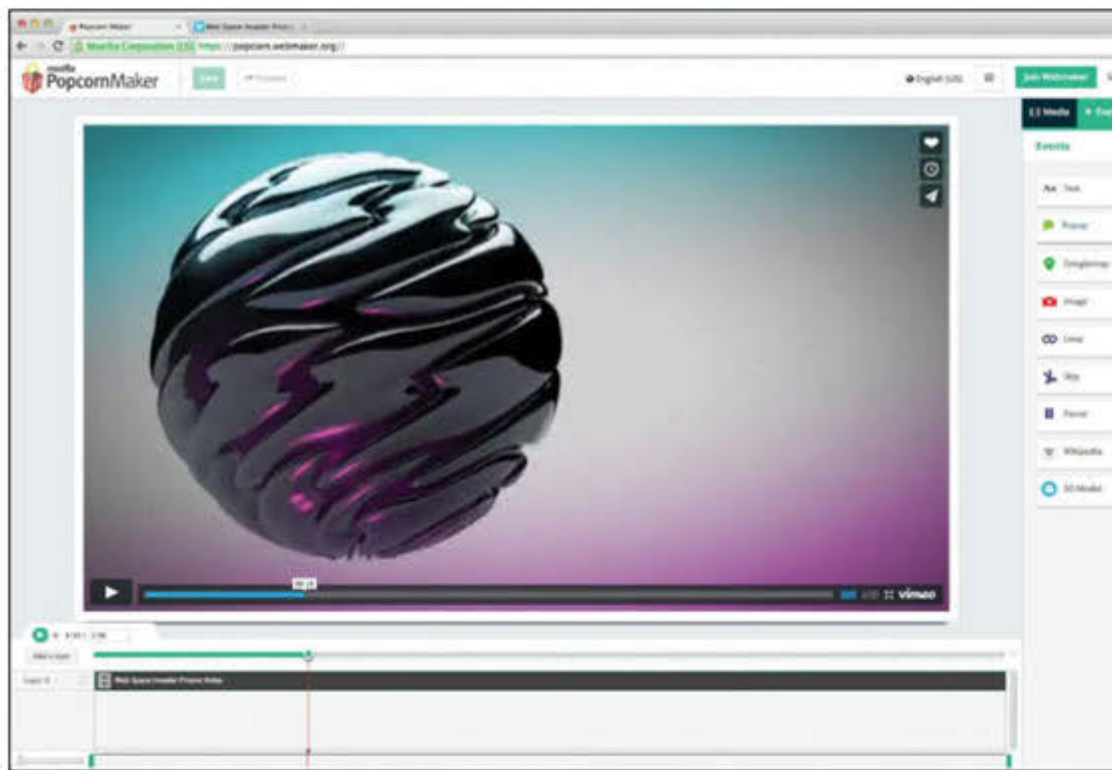
Top right

The camera moves forward again and the third message is displayed to show the relevance of the background. This move is timed to coincide with the car driving past

Right

The camera moves forward to focus on the city with the rain animating. As it does this, the second text content is displayed on the screen





Popcorn maker

While Popcorn is easy enough to cue content, with its straightforward JavaScript API, designers might prefer not to get into the code and instead create content for Popcorn with a graphical user interface. Mozilla has made this possible with Popcorn Maker (popcorn.webmaker.org). By loading this page in your browser it is possible to sync up existing content found on YouTube, SoundCloud and Vimeo, to make events happen at different points during the playback of that content. Popcorn Maker comes with a simple timeline so you can scrub through the content to the section when you want something to happen. Adding events lets pop-ups or other content appear. The only problem is that Popcorn Maker is hosted externally to your site.

the browser to see the message show after one second.

```
popcorn.cue( 1, function() {
  one.classList.add('show');
});
```

15. Move forward

Adding the next code causes the camera to start to move forward to the next section in the 3D scene by calling `camMove2` inside the `scene.js` file. The text on the screen is made to fade out by removing the `'show'` CSS class, and so the opacity is removed to 0 again.

```
popcorn.cue( 13.3, function() {
  camMove2();
  one.classList.remove('show');
});
```

16. Add the message

If you looked at the last move in the browser you will have seen the camera move forward towards the buildings and the rain. Now the next code will show the text in the div with the id of `'two'`. This takes place after almost 16 seconds. Save and test to see the effect.

```
popcorn.cue( 15.8, function() {
  two.classList.add('show');
});
```

17. Lightning flashes

Just after the last message displays on the screen you will hear two bass stabs as part of the music. It's possible to use this with a lightning effect in the 3D scene to make it more dramatic. Here we turn a flash of light on for just less than half a second in time with the first bass stab.

```
popcorn.cue( 17.2, function() {
  lightOn();
});
popcorn.cue( 17.6, function() {
  lightOff();
});
```

18. Slight change to the colour

Let's do the same again on the second bass stab. The code here gets the timing just right for that. There are more bass stabs a little later and you can add more by listening to the audio and noting the times yourself.

```
popcorn.cue( 18, function() {
  lightOn();
});
popcorn.cue( 18.4, function() {
  lightOff();
});
```

19. Continuing the journey

Now we wait for the song to be almost 28 seconds in to call the camera to move again with `camMove3`. The second message is faded out on the screen as the camera moves forward. Save and refresh now.

```
popcorn.cue( 27.7, function() {
  camMove3();
  two.classList.remove('show');
});
```

20. The next message

Just after the camera arrives at the bridge, the next message needs to display on the screen, so in this code

the function is called at just after 30 seconds so that the message displays. You'll notice in the browser window that an animation of a car drives past at this particular point and this is no coincidence as the message is relevant to that.

```
popcorn.cue( 30.2, function() {
  three.classList.add('show');
});
```

21. Final camera movement

Once the car has driven past the screen it's time to move the camera forward again to the final destination. At just after 40 seconds the final camera move is called, and the third message fades out on the screen by removing the `'show'` CSS class. The camera swings up to the mountain for the final text to display.

```
popcorn.cue( 40.5, function() {
  camMove4();
  three.classList.remove('show');
});
```

22. Finish off

As the camera eases into the final position in the background, the final text is displayed on the screen. At 43 seconds the fourth text block gets the CSS class of `'show'` added so that it fades in, and shortly after this the music finishes playing. Now just save and test the document in your browser to see the full animation and music sync up together.

```
popcorn.cue( 43, function() {
  four.classList.add('show');
});
```

web workshop

Slide up titles on page load using CSS

As seen on onedollarlesson.com

Sobering statistics

The animations deliver some frightening stats about the levels of cybercrime targeting user's bank details.

Well-laid out menu

The menu is neatly tucked away to the left. As the user scrolls the menu stretches to indicate position on page.

Claim your prize

Complete all three of the lessons to access a free three-month Kaspersky security trial. A very useful prize.

Hidden animations

Scrolling past the first screen will reveal some beautifully crafted animations. These unfold as the users scrolls down.

On page load

Once the initial load animation (featuring a dollar) is done, the title elements begin their step-by-step animation.





So you're looking for a good way to engage your website users from the moment the homepage loads. Grabbing the diminishing attention spans of web users is an on-going battle for developers. There are all sorts of options, from autoplay videos to pop-ups which demand user interaction. Sometimes, though, you can keep the user's attention for those first few vital seconds with a simple animation. Nothing too over the top, just a little something that keeps them watching.

How about an animation that delivers the site title in steps, over a short period of time?

Internet security company Kaspersky have put together this wonderful-looking microsite, designed to educate users on the dangers of online payments and the many related cyber attacks that users can fall prey to. Featuring some great scroll-activated animations and fullscreen videos (worth checking out for the transport-based demonstrations of Trojans and the like), One Dollar Lesson opens with a perfect example of the effect we are going to re-create in this workshop. Once the homepage background has loaded, the site title and the Kaspersky logo ease into view, one at a time. Simple effect, but one that keeps you watching.



Make the wait interesting

"These days it's a brave website that asks the user to sit through a content-loading percentage bar, however short the wait. The desire for instant content is relentless. If this is unavoidable you better make sure your loading icons are interesting and different enough to hold users fast."

Richard Lamb, owner and web designer at Inspired Lamb Design

<comment>
What our experts think of the site

Technique

1. Background image

The first step for our food-related example is to put in place the fullscreen background image. We've selected an image which places content focus on the left, so that we can place our animated content on the right. Ensure that you include your vendor prefixes.

2. The base HTML

Create an intro div that will sit on the left side of the canvas (using Bootstrap CSS makes it simpler). Within this we need four title divs, each containing a div class we're calling yum. The yum classes will contain the animations.

```
<div class="intro col-md-3 col-md-push-8">
  <div class="title">
    <div class="yum"></div>
  </div>
  <div class="title">
    <div class="yum"></div>
  </div>
  <div class="title">
    <div class="yum"></div>
  </div>
  <div class="title">
    <div class="yum"></div>
  </div>
</div>
```

3. The initial CSS

The intro div should be given some slight padding at the top. The title div needs to assign an overflow:hidden property. We need the child elements invisible until they reach the confines of their parent. We'll call this 'rise'.

4. Insert the title images

Create four PNG images for the three words and an underline, of equal width, which make up our title, 'Um

Num Num!'. Place one apiece into the four yums. Then give each of div an individual class name, corresponding to the image names.

5. Individual animation times

We'll concentrate on writing the keyframes which will animate our title. First, set animation durations specific to each of our individual divs.

```
.um {
  animation-duration: 2s;
}
.num-one {
  animation-duration: 4s;
}
.num-two {
  animation-duration: 6s;
}
.underline {
  animation-duration: 7s;
}
```

6. Write the keyframes

The keyframes re-creates the effect from our inspiration site causing the four title elements to slide up from the bottom, with a slight bounce on arrival.

```
@keyframes rise {
  0%, 60%, 75%, 90%, 100% {
    transition-timing-function: cubic-bezier(0.215, 0.610, 0.355, 1.000);
  }
  0% {opacity: 0; transform: translate3d(0, 3000px, 0);}
  60% {opacity: 1; transform: translate3d(0, -20px, 0);}
  75% {transform: translate3d(0, 10px, 0);}
  90% {transform: translate3d(0, -5px, 0);}
  100% {transform: translate3d(0, 0, 0);}
}
```

EXPERT ADVICE

Content delivery

Packaging important information into an entertaining, easily consumable format has become one of the primary challenges for websites, especially when delivering information that has not necessarily been sought out. One Dollar Lesson employs just the right mix of interaction and spectacle.

Animate an information card box with CSS

Create an expanding card-like box with changing text on hover



Design sometimes creates a requirement for information to be presented in a restricted amount of space. This

problem has traditionally been solved by reducing the amount of information, which compromises the purpose that the design seeks to serve. With websites and apps enabling interactivity to be integrated within design, there are opportunities to solve this problem so it doesn't compromise design or content.

Information cards are a method for websites and apps to present information on a single screen in a way that can be expanded through interaction. These elements can contain information as well as features such as navigation, forms and diagrams that the default design doesn't provide enough room to present. In addition to the spacing issue, information cards can be used to make a design appear to be less cluttered when there is a lot of information to present.

This tutorial shows how information cards can be used for a navigation component that expands to show a preview of the pages being linked to upon being hovered by the mouse cursor. Animation and interactions are defined entirely through HTML and CSS without any need for complicated JavaScript.

1. Get started

The first step is to declare the main HTML page structure including the head and body section within the HTML page. This stage is only to prepare the foundations for the addition of the page content.

2. Content container

There is a need to contain the page content in some type of container so that we can automatically control the flow of the page element such as for adapting the page to different screen resolutions. We will create this

using a <div> element with an ID called 'container' that can be directly referenced from the CSS.

3. Content overview

With the main page foundations in place, we are now ready to start inserting the main content. The first elements to add inside the container will provide an introduction to the page content - these being the page title page and summary. This will provide the information needed by the reader to identify whether they want to investigate further.

4. Card navigation

The information cards are being used as navigation options, so the next step is to define each card as a container for their content in a way that can open a webpage when clicked. This is achieved by using <a> tags within a <nav> container, hence the cardset being treated as any HTML regular navigation. Place the following code inside the content container:

```
<nav>
<a href="http://www.google.com">
</a>
<a href="http://www.google.com">
</a>
<a href="http://www.google.com">
</a>
<a href="http://www.google.com">
</a>
<a href="http://www.google.com">
</a>
</nav>
```

5. Card content

With each of the <a> tags being used as the card container, we now need to insert the content to be

displayed as part of the cards. This content will contain an image and two types of text - one for the introduction and another more detailed set of text to be shown when the information card is being hovered over.

```

<span>
Sun
</span>
<span>
Intro text...
</span>
<span>
More text to be shown on hover...
</span>
</span>
```

6. Initiate CSS styling

The additional CSS stylesheet resource file is required to add the styling. Create a text file called 'styles.css', making sure that your text editor does not add TXT as a file extension. These also need attaching from the <head> section of HTML document.

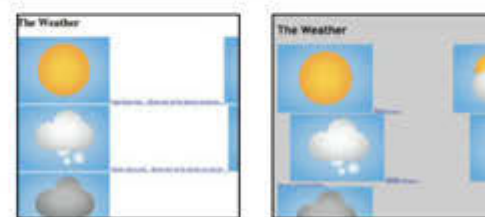
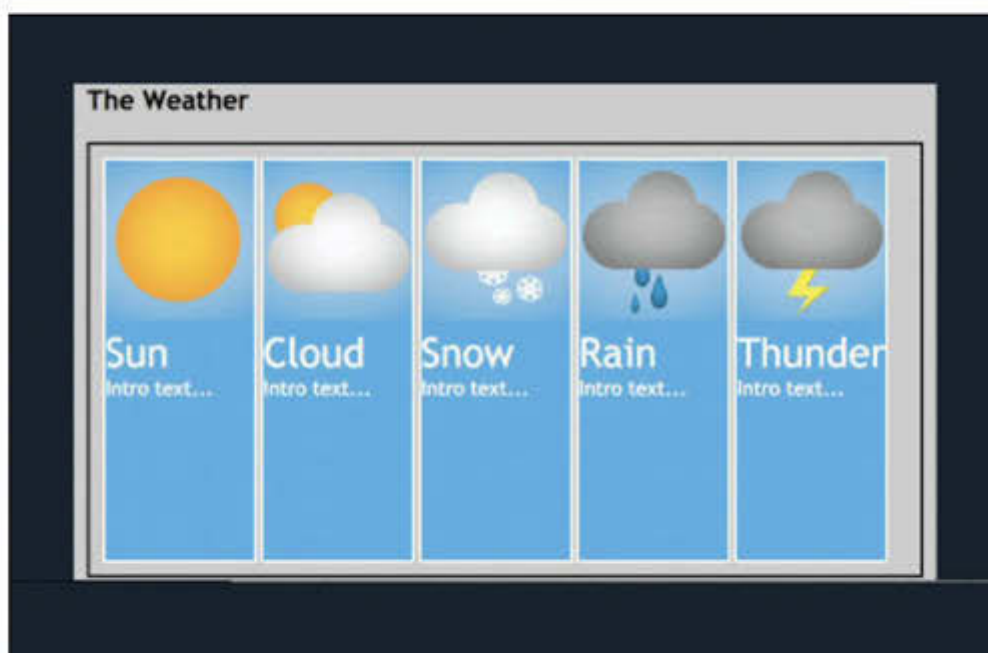
```
<link rel="stylesheet" type="text/css"
href="styles.css" />
```

7. Style HTML body

It is important not to forget that the <html> and <body> elements on a webpage are set to a height of just one

HTML with selectors

There is no need to complicate your HTML and CSS with class names, IDs and properties when you can reference elements using their ordered position, with selectors such as :nth-child().



Left
Default information card styling now positions each of our cards horizontally

Top left
HTML are now elements in place, with content added to article sections, but there is no styling yet

Top right
The cards have been updated with styling to show the page container with the information card content

Tutorials

Animate an information card box with CSS

line by default, hence causing problems if we want to create a container that has a full height of the webpage. We solve this by setting the `<html>` and `<body>` elements to be full height in the CSS.

```
html, body {
  display: block;
  width: 100%;
  height: 100%;
  margin: 0;
  padding: 0;
}
```

8. Page container

The container needs to be styled in a way that it will stand out. This can be done by changing the main page background colour and positioning the container in the centre of the page. To do this, we need to update the `<head>`, `<body>` and container sections.

```
html, body {
  font-family: "Trebuchet MS", Helvetica,
  sans-serif;
  background: rgb(21, 34, 47);
}
#container {
  display: block;
  margin: 0 auto 0 auto;
  padding: 1em;
}
```

```
background: #ccc;
width: 1000px;
height: 100%;
}
```

9. Card container

In addition to the page having a container, we are also using `<nav>` as the container for the information cards. This container should appear to separate the cards from any other page content, as well as to define the height and width used to contain the cards. Displaying the `<nav>` as a block element will automatically make it stretch the full width of the page container.

```
nav {
  position: relative;
  display: block;
  padding: 1em;
  border: 3px solid #000;
  margin: 2em 0 2em 0;
  height: 80%;
  overflow: hidden;
}
```

10. Define cards

The information cards need to be defined in relation to the `<nav>` container they are placed within. This step defines their default width, height and colour that they will be displayed. These elements will be displayed as inline-block elements so that they can be positioned next to each other at a width that fits all five items.

```
nav a {
  display: inline-block;
  padding: 0;
  font-size: 3em;
  text-decoration: none;
  width: 18.5%;
}
```

```
height: 99%;
border: 3px solid #fff;
background: rgb(81, 173, 228);
color: #fff;
overflow: hidden;
}
```

11. Card transition

It is important that the cards have transitions enabled so that they can appear to animate without the need for JavaScript. For the sake of simplicity, we will set all properties of the card containers to have a transition of one second, enabling us to animate all properties from colour to size.

```
nav a {
  -webkit-transition: all 1s;
  transition: all 1s;
}
```

12. Card open animation

Activation of animations is now just a case of defining the properties to animate to. This is achieved by using the `:hover` attribute on the `<a>` tag of the `<nav>` element, which sets the animation to only occur when the information card is being hovered over. The main animation property will be setting the information card to cover half the width of the container.

```
nav a: hover {
  color: #333;
  width: 50%;
  margin-top: 0;
}
```

13. Adapt other cards

The problem with opening the current hovered card to 50 per cent of the screen is that it will push other cards

Foregoing pages

Making use of information cards instead of making the user switch between pages is helpful as it simplifies code - it avoids the need of complicated JavaScript to save temporary data.



Top left

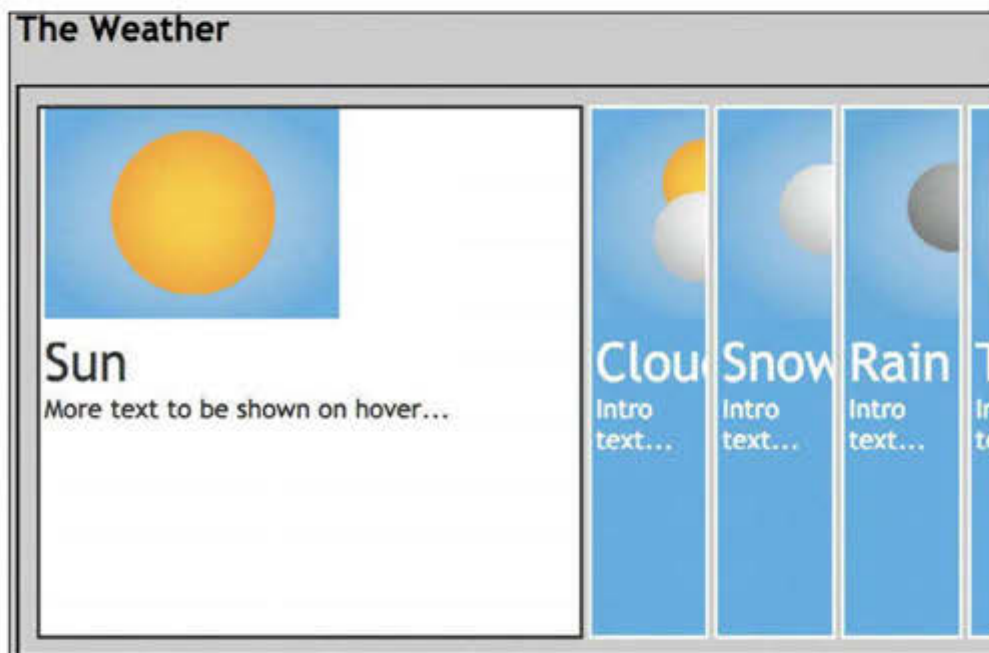
Here we've reduced the width of the information cards when their nav container is hovered

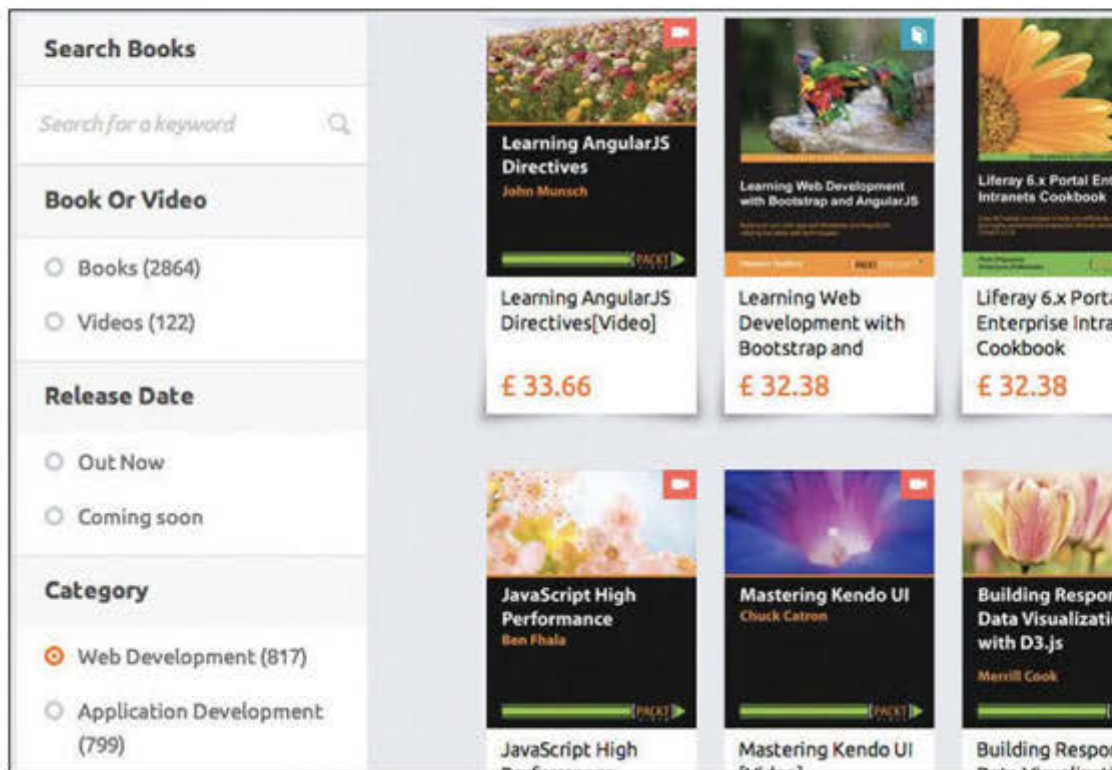
Top right

The text size of the inner `` elements is now half the size of the standard heading text

Right

Adding a white background to the hovered information card makes it stand out more, but the picture icon looks out of place





Use cards for apps and eCommerce

You can utilise information cards to extend product listings for an eCommerce website or as part of an online information brochure. The ability to present more information without the need to reload the webpage will provide more convenience to website visitors and will result in them being more willing to look at more information on the page. Better visitor engagement will enable your website to produce better commercial results, whether it be increased sales conversions, enquiries or other actions that your content is designed to produce. Aside from website-oriented projects, information cards can be used for web-based apps, enabling users to access different features without having to switch pages.

out of view, if not itself as well. This can be solved by animating the other cards to have a smaller width when the <nav> container is hovered so it's activated at the same time as the hovered information card.

14. Content sections

The elements inside of the information cards are used instead of <div> elements to keep the HTML markup compliant with W3C standards. We can use CSS to make these elements act in the same way as <div> elements by setting the display property to 'block', enabling them to have their size and padding set in addition to placing each element on a new line.

```
nav a span{
  display: block;
}
```

15. Preview content

The structure of the HTML uses a tag to contain the card information, with two more tags inside this to contain the preview and full content. We can use CSS to define consistent properties for all inner tags that are placed in the information card content area.

```
nav a span span{
  position: relative;
  font-size: 50%;
}
```

16. Content transition

It is the content areas of the information cards that will need to change between showing their preview and full content when hovered, so we want to apply the ability to

show transition animations. We use the same CSS reference and the transition property to set opacity and display properties and show transitions of one second.

```
nav a span span{
  -webkit-transition: opacity 1s, display 1s;
  transition: opacity 1s, display 1s;
}
```

17. Ensure preview visibility

The HTML is set so that the first inner element is used for preview content, hence you need to hide the second inner element until the information card is hovered over. This is done with :nth-child() CSS selector to hide the second inner element by default.

```
nav a span span:nth-child(2){
  opacity: 0;
}
```

18. Show main content

The information card needs to transition to show the main content it holds when it is hovered. This is just a case of using the :hover and nth-child(2) selectors to target the content container so that full opacity and display as block can be applied.

```
nav a: hover span span:nth-child(2){
  opacity: 1;
  display: block;
}
```

19. Hide preview content

There is no need to display the preview content when the main content is visible, so we use the same strategy

to reference the first element when the information card is being hovered over. This sets the opacity and display so that preview content isn't visible.

20. Highlight current card

Although the information card currently being hovered over expands, it still may be difficult for some people to distinguish the content of the current information card with its neighbours. This can be solved by making the background colour change - in this case, we will make the background white to ensure it stands out from the other information cards.

21. Image tidying

The example images used in this tutorial have a blue background that looks out of place when the information card background changes colour to white. This can be resolved without overcomplicating the HTML or CSS by adding a border to the right side of the image in the same shade of blue to fill in the missing space.

```
nav a img{
  border-right: 500px solid rgb(81, 173, 228);
}
```

22. Cover the padding

The padding added to the information cards exposes the white background towards the left of the image icon. Fix this with a negative margin set to the same size of the padding, moving the icon to cover the gap.

```
nav a img{
  margin-left: -1em;
}
```

web workshop

Make a screen shrink on scroll

As seen on devstars.com

Hidden menu

The regular menu is hidden behind the burger menu and sits over the entire page on the screen when activated.

Resizing sections

Clicking the dots (right) automatically scrolls the page down to the next section, this fits the browser window exactly.

Shrinking images

The graphical content on the screen animates down in size, giving a pleasing shrink effect as the page scrolls upwards.

Reverse in size

Returning back through the navigation dots causes the page to slide back up and the graphic content to expand in size.

Varied navigation

The homepage combines a one-page style site and a slide show. The scrollbar is hidden but navigation dots are still visible.

Make a screen shrink on scroll

↓ **DOWNLOAD TUTORIAL FILES** www.filesilo.co.uk/webdesigner



As a web design and development agency it's very important that you can show off what your skills are; if you can combine samples of your work into an innovative presentation on your site then that's even better. That's exactly what Devstars have done on their site. Devstars.com is a full multipage site but use the one-page style website idea for the homepage to show off exactly what they do. The site's homepage has a number of navigation dots that you immediately associate with a slideshow and these contain the latest work offerings, while giving an overview of the company.

The little animation touches as the user slides down are the neat show-off points in this site. Each browser-sized screen on the homepage is like a different slide in a slide show, but you click the navigation dots on the right-hand side to move down rather than right to left. The scrollbar is hidden away so this nifty navigation is a great way to explore the screens that follow. As you slide down to the next section, the graphic content on the screen animates, shrinking down in size and moving off the top of the page. The real menu is hidden behind the burger icon so what this presents is an unusual slideshow of important content that Devstars wants users to see.



<comment>
What our experts think of the site

Making your work the star

"Bright bold colours make this site stand out with just enough negative space to make the logo and site content stand out. The homepage's featured content has an animated effect that draws attention to the care the team put into the small details of their site."

Mark Shufflebottom, professor of Interaction Design, Sheridan College

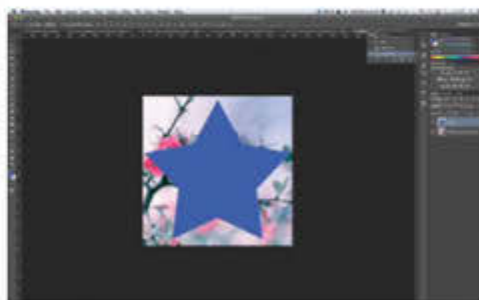
Technique

1. Create the shrinking logo effect

When making this effect work, the logo needs to be placed within a page. As there is no page content here, some div tags are being added, which will have 600 pixels of height added to them. When the image reaches the top of the page, it will be shrunk down.

```
<div class="space"></div>

<div class="space"></div>
<div class="space"></div>
```



2. Add the library

In the head section of the page, the script tag calls the jQuery library so that we can detect the position of the page and make DOM changes based on that. The CSS styling gives the 'space' class a height of 600 pixels, as the comment states this is purely for demonstration only.

```
<script type="text/javascript" src="js/
jquery.js"></script>
<style>
.space{
height:600px;
/*Purely Demonstration Purposes*/
}
```

3. Finish the style

The image with the ID of 'shrink' will be scaled down when it hits the top of the page - this shrink will then be detected through jQuery code later. The image is displayed as a block element to enable the centring of the image horizontally.

```
#shrink{
display: block;
margin: 0 auto;
}
</style>
```

4. Detect the position

Just before the closing body tag, the script tag is added along with any of the code that follows. This script tag will check how far down the page the document has actually scrolled. From there the 'topDistance' variable grabs the position of the image in relation to the top.

```
<script >
$(function () {
$(window).on('scroll', function() {
var scrollTop = $(this).scrollTop();
var topDistance = $('#shrink').offset().top;
```

5. Start the shrink

When the image is within the space of the top of the page, the 'shrink' image is animated down to 0 on the width and height over half a second. Now go ahead and save the page, then you can test the effect in the browser to see the effect in action as you scroll the image to the top of the page.

```
if ( topDistance < scrollTop ) {
$('#shrink').animate({"width": 0, "height":
0}, 500 );
}
});
});
</script>
```

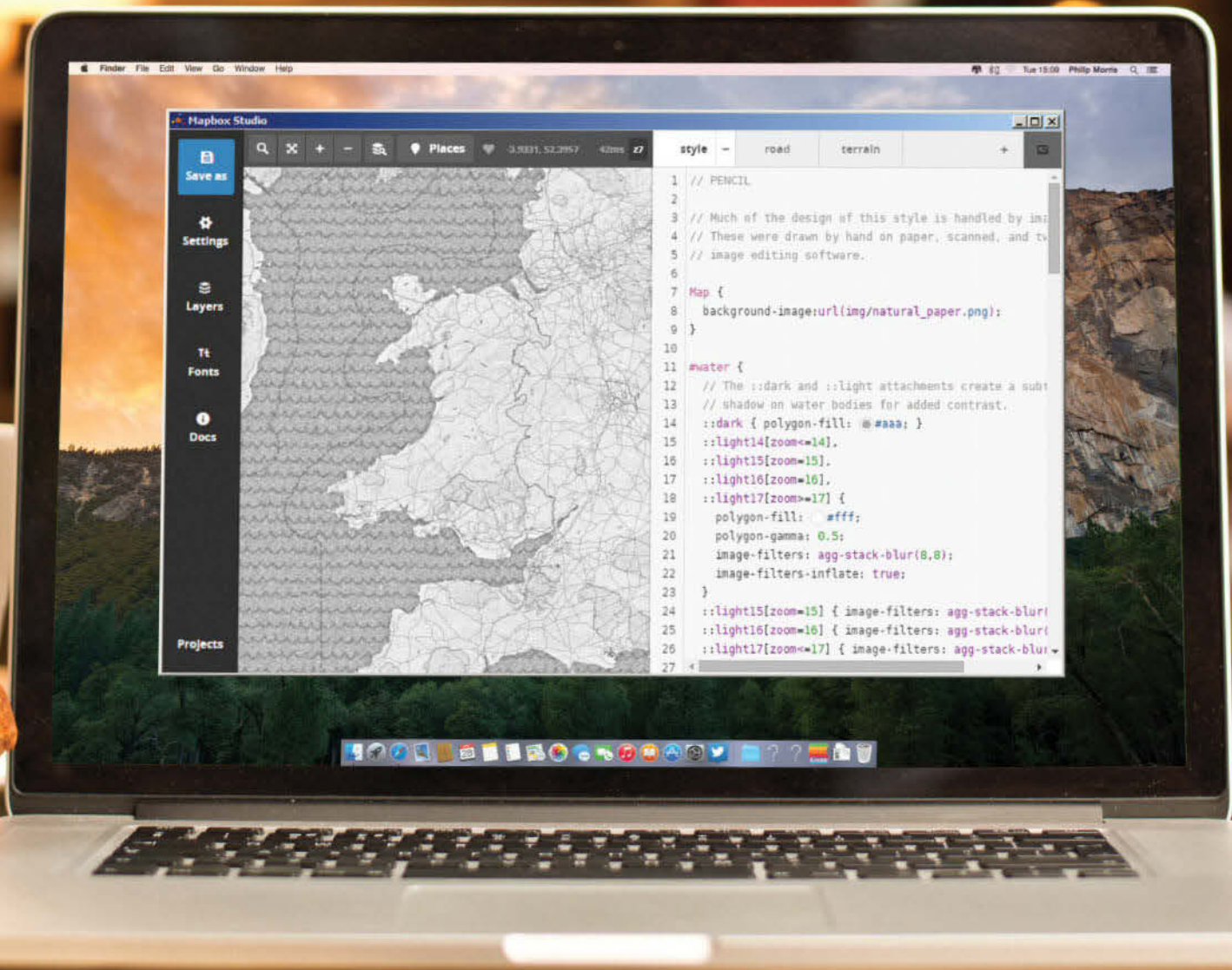
EXPERT ADVICE

Reinforcing the brand

A logo can say a lot about any company and Devstars' logo features a series of stars laid out horizontally. For their showcase content, Devstars uses the shape of the logo as a mask for images of work that they've produced. This not only looks good but also reinforces the brand in the homepage design.

Create desktop applications with Electron

Use pseudo elements HTML, CSS and JavaScript to create cross-platform desktop applications





GitHub have created an open source project called Electron (formerly Atom-Shell) that combines Chromium, on which Google's Chrome is built, and

Node.js, a JavaScript environment designed for building applications. The result enables the building of cross-platform desktop applications using HTML, CSS and JavaScript. Companies such as Microsoft, Facebook, Slack and more have started to make use of Electron.

Electron applications normally start out with a single JavaScript file which is executed initially. The code within it has control over the main process, can create and control new application windows and listen for important events. Each window that is created can be set to render your UI through the use of HTML, CSS and JavaScript. The JavaScript executed within a window can also access special Electron APIs and the Node.js API.

The Chromium part of Electron is also kept up to date with the latest releases, enabling use of the latest HTML/CSS features and emerging JavaScript standards. The Node.js environment is also well maintained, to the point where it's currently using a fork of Node.js called io.js, which is capable of supporting ES6 JavaScript syntaxes and can make use of npm registry modules (npmjs.com).

1. Prebuilt Electron binaries

Compiling Electron from scratch isn't required for building applications. The project makes available precompiled versions of Electron for multiple operating systems, which can be easily installed using Node.js' npm package manager (available via nodejs.org).

```
$ npm install -g electron-prebuilt
```

2. Test Electron

Electron should now be available on your system. This can be confirmed by running the 'electron' command,

which will display a standard message for the application - just ignore this message for now and close the window.

```
$ electron
```

3. Use package.json

Package.json is required for Electron to know which file to execute for the application. It will also store the name of your application and your version, and 'npm' will add the details of any modules that you install and use. In this tutorial, the application will be used for editing Markdown-based files.

```
{
  "name": "wdMarkdown",
  "main": "index.js",
  "version": "0.1.0"
}
```

4. The application and index.js

The index.js file is the core of the application and will create our first application window to display the application. Depending on the type of application being developed, this file will usually contain the core of your application and do all the heavy lifting required. By using the 'app' object, we can find out when Electron is ready, and a browser window (appearing as a normal application window) can then be created.

```
var app = require('app');
var BrowserWindow = require('browser-window');
var window = null;
app.on('ready', function() {
  window = new BrowserWindow({width: 800,
  height: 600});
  window.loadUrl('file://' + __dirname + '/
  root/index.html');
  window.on('closed', function() {
```

```
  window = null;
});
});
```

5. Test the app

To enable us to test the application so far, a HTML file can be created containing some simple HTML for now. After adding the code below, execute the command 'electron .' within the same directory as the package.json file created earlier.

```
<html><body>
<h1>Hello World, from Electron</h1>
</body></html>
```

6. Install Bower

Our application has a few dependencies for the UI that will be constructed. By installing Bower, these dependencies can be easily retrieved along with anything else they may require and Bower could be used to keep them up to date.

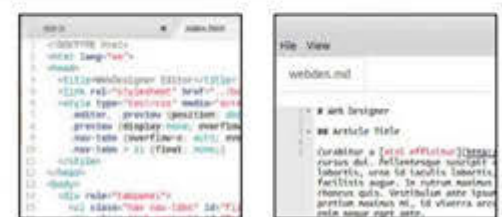
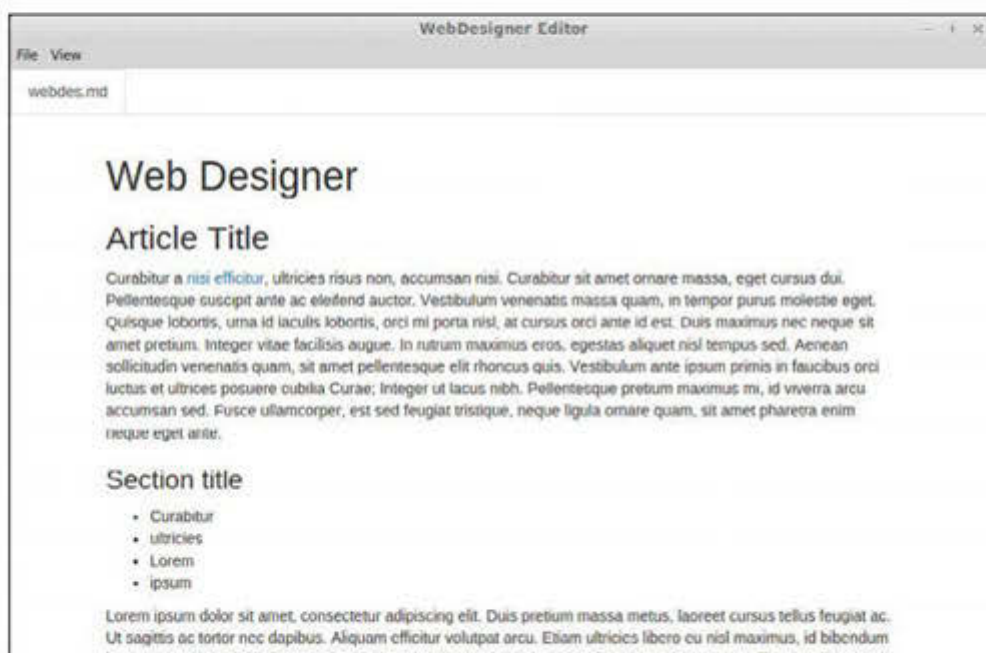
```
$ npm install -g bower
```

7. Bootstrap framework

In order to have something to display, a UI needs to be created using HTML and CSS. As a base theme,

Desktop applications

For fuller and more complex applications, consider using npm modules (more information is available at npmjs.com) for databases, network communications and more, plus a framework for your UI from the likes of React, AngularJS, Backbone.



Left

By using the view menu we can have the editor switch between viewing the original Markdown content and a rendered HTML version of the content as a preview

Top left

Here is the final UI HTML page. Its primary purpose is to include all the JavaScript files needed to construct UI, Bootstrap for the appearance and our main JavaScript file

Top right

Here is what the application will look like when editing a Markdown document. The tab shows the title of the file, and the editor contains the raw Markdown content

Tutorials

Create desktop applications with Electron

Bootstrap can be useful as it provides many useful components and styles. More advanced applications may take advantage of AngularJS or React as well. By using Bower to retrieve a copy of Bootstrap we'll also have a copy of jQuery included as it's required.

```
$ bower install bootstrap
```

8. Markdown rendering

The app will be used for editing Markdown files.

Markdown is a widely used markup language used for writing formatted text in a plain text file. It can then be processed by a library such as 'Marked' into HTML.

```
$ bower install marked
```

9. Edit Markdown with Ace

The Ace editor project is a feature-packed code editor that's written in JavaScript and can be used within web pages. For this tutorial, we will be utilising Ace editor for editing the Markdown files as it can provide us with things like line numbering, syntax highlighting and other very useful features.

```
$ bower install ace-builds
```

10. Include Bootstrap and JS

Next, make use of Bootstrap, Ace editor and Marked by including them in our index.html file. Make note of the alternative way jQuery is included, which is due to an

issue with jQuery being unsure what kind of environment it is operating within when inside Electron.

11. Styling

The Markdown editor being constructed will provide an edit area for altering the content and a preview of the HTML generated. As the app will be kept simple, the editor and preview areas will fill the application, minus an area for tabs to switch between files.

```
.editor, .preview {position: absolute
!important; top: 58px; right: 0; bottom: 0;
left: 0; z-index:5;}
.preview {display:none; overflow:auto;
z-index:10; background-color: white;}
```

12. Make the tabs

Bootstrap provides the styling and JavaScript required to provide the tabbed interface used for the application. Each tab will represent a file and clicking between them will display a related editor or preview. For now, we only need very basic HTML as we'll build the tabs and related panes using JavaScript.

```
<div role="tabpanel">
<ul class="nav nav-tabs" id="fileTabs"></ul>
<div class="tab-content" id="filePanes"></div>
</div>
```

13. root/app.js

This file will hold our main UI code. Using the ipc module, it can subscribe and publish events for communication with the main JavaScript process, that Electron executes (/index.js). We'll need to identify files edited and store additional information, plus references to the tab and pane areas of the UI. Also, we will need a small function to grab the ID of the currently active tab.

```
var ipc = require('ipc');
var files = {}, rollingId = 0;
var tabs = $('#fileTabs'), panes =
$('#filePanes');
var ActiveTab = function() {
return id = tabs.find('.active a').
attr('href').slice(1);
};
```

14. FileNew function

To create a file or handle opening an existing one, our UI needs to create a new tab and pane, as well as set a relevant title and content. We will use FileNew for this.

15. Use Ace

With the tab, pane and content now ready, it's time to fire Ace editor into action. Then we can store some information for easier referencing and switch to the newly created tab.

16. Menu events

Now we execute the FileNew function once, present a new file tab on opening, and then listen for menu events sent from the main application process. The menu events, which we'll create soon, are for creating a new file and opening an existing one.

```
FileNew();
ipc.on('file-new', FileNew); ipc.on('file-
open', FileNew);
```

17. Menus

At this point, our Electron app should start up and present a fresh new tab to edit content in. However, we can't open any more or save anything we create. So in this case, a menu is required and it'll be built within the main /index.js file. The 'fs' and 'path' modules from Node.

Debugging Electron

To debug any window you have created using Electron, call the .openDevTools() method on the instance of that window. This will open a developer tools window.



Top left

Microsoft have recently released their own editor, built on top of Electron, called Visual Studio Code

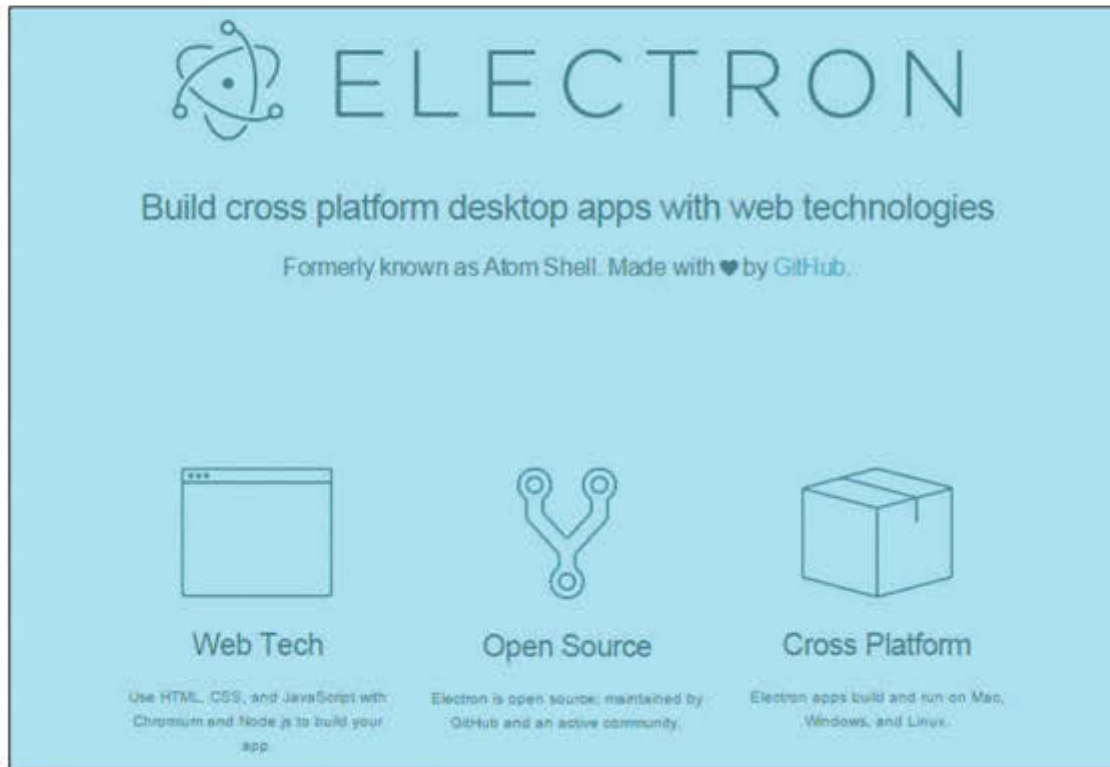
Top right

Atom is GitHub's programming editor and is where Electron started out as part of the project

Right

The final version of the editor, showing the multiple tabs and some Markdown formatted content





Packaging applications

Through Electron's 'crash-reporter' module, you can configure your application to let users submit crashes to a website under your control. Your application code can also be archived into a more protective ASAR file, before being distributed. This requires use of the 'asar' command installed through npm; 'npm install -g asar'. Electron currently only supports autoupdating on OS X through its 'auto-updater' module. But there's no reason why you couldn't implement a check against a URL for newer versions of your app and then notify the user. GitHub provides prebuilt versions of Electron at github.com/atom/electron/releases. Your app code can combine with these releases and redistributed.

js will be used for handling files. 'Dialog' and 'menu' provide opening/saving dialog boxes and an application menu bar.

```

/* Below 'window = null' */
var fs = require('fs'), path =
require('path');
var ipc = require('ipc');
var dialog = require('dialog'), Menu =
require('menu');
/* Within "app.on('ready')"

```

```

];
Menu.setApplicationMenu(Menu.
buildFromTemplate(template));

```

18. Open a file

Within the ./index.js file, we can now add a function (already referred to in our menu) to show a dialog (provided by Electron), open the file and pass the data back to our UI for display. At this point, executing our app will provide us with a menu where we can open a new file and have it displayed in our editor.

19. Pass file data

Saving a file requires a few more steps. When the save/save as menu item is clicked, we send an event to the UI. The UI holds the data that we're going to save, so it must listen for those events and send the data back to the main process for us to display a dialog and create/update the related file.

20. Write a file

Our UI is sending a file-save event back to our main process (./index.js), providing the reference ID, content and path (new files don't have one). We'll need to listen for this event, saving the file directly if a path is provided or prompting for a location (save-as) first before creating or overwriting a file with the content.

```

ipc.on('file-save', function(evet, data, id,
type, filepath) {
if (filepath) {
return fs.writeFile(filepath, data,
function(err) {

```

```

if (err) return console.error(err);
});
}
dialog.showSaveDialog(window, {
filters: [{
name: 'Markdown', extensions: ['.md',
'markdown']
}]
}, function(filepath) {
if (filepath && filepath.length > 0) {
fs.writeFile(filepath, data, function(err) {
if (err) return console.error(err);
window.webContents.send('file-saved', id,
path.basename(filepath), filepath); });
} });
});

```

21. Rendering

Markdown is great for quickly producing text and indicating how it's formatted, but it's more useful when you can see how it looks when it's rendered. Our menu contains an option to switch to a rendered view so our UI needs to listen for it, grab the editor content, pass it through 'marked' for conversion and output it for display.

22. Closure

Now we need to close a tab. Listen for the menu event within the UI, and after that it's just a matter of having the editor clean itself up, and remove the tab and pane elements. Finally, switch to another tab if one exists. For the full code in this tutorial, make sure you go to filesilo.co.uk/webdesigner.

web designer SPECIAL SUBSCRIPTION OFFER

TRY
FIVE
ISSUES FOR

£5*



OVERSEAS
READER
SPECIAL
6 ISSUE
OFFER



CONTEMPORARY FEATURES
WRITTEN BY EXPERTS



BEHIND THE SCENES WITH THE
WORLD'S BIGGEST AGENCIES

Order hotline
0844 848 8413

Calls will cost 7p per minute plus your telephone company's access charge

Online at
www.imaginesubs.co.uk/wed

Every issue packed with...

- Expert tutorials on HTML, CSS, jQuery and leading libraries and frameworks
- In-depth interviews with the world's leading creative agencies
- A close-up look at the latest web development technologies
- Designer resources and pro video guides

Why you should subscribe...

- Try 5 issues for £5*
- Save 30% off future issues - paying £25.15 every 6 issues
- FREE delivery direct to your door
- Never miss an issue

Order by

31ST AUGUST

Use code **SUMMER151** for this offer.

BY POST

Send your completed form to:
Web Designer Subscriptions, 800 Guillat Avenue,
Kent Science Park, Sittingbourne, Kent ME9 8GU

YOUR DETAILS

Title _____ First name _____
Surname _____
Address _____

Postcode _____ Country _____
Telephone number _____
Mobile number _____
Email address _____

DIRECT DEBIT PAYMENT

UK Direct Debit Payment

I will receive my first 5 issues for £5, then I will pay only £25.15 every 6 issues thereafter

ip		Instruction to your Bank or Building Society to pay by Direct Debit		DIRECT Debit	
<small>Please fill in the form and send it to: Imagine Publishing Limited, 800 Guillat Avenue, Kent Science Park, Sittingbourne, Kent, ME9 8GU</small>					
<small>Name and full postal address of your Bank or Building Society</small>				<small>Originator's Identification Number</small>	
<small>To: The Manager</small>		<small>Bank/Building Society</small>		5 0 1 8 8 4	
Address _____				<small>Reference Number</small>	
Postcode _____				_____	
<small>Name(s) of account holder(s)</small>				<small>Instructions to your Bank or Building Society</small>	
_____				<small>Please pay Imagine Publishing Limited Direct Debits from the account detailed in this instruction subject to the safeguards assured by the Direct Debit guarantee. I understand that this instruction may remain with Imagine Publishing Limited and, if so, details will be passed on electronically to my Bank/Building Society</small>	
<small>Branch sort code</small>				<small>Signature(s)</small>	
_____				_____	
<small>Bank/Building Society account number</small>				<small>Date</small>	
_____				_____	
<small>Banks and Building Societies may not accept Direct Debit instructions for some types of account</small>					
<small>A6 instruction form</small>					

PAYMENT DETAILS

YOUR EXCLUSIVE READER PRICE, 6 ISSUES

EUR – £32 ROW – £37 USA – £37

Cheque

I enclose a cheque for £ _____
(made payable to Imagine Publishing Ltd)

Credit/Debit Card

Visa Mastercard Amex Maestro

Card number _____ Expiry date _____

Issue number _____ (if Maestro) _____

Signed _____

Date _____

Please tick if you do not wish to receive any promotional material from Imagine Publishing Ltd by post
by telephone via email

Please tick if you do not wish to receive any promotional material from other companies by post
by telephone Please tick if you DO wish to receive such information via email

TERMS & CONDITIONS

*This offer entitles new UK Direct Debit subscribers to receive their first 5 issues for £5. After these issues, subscribers will pay £25.15 every 6 issues. Offer code SUMMER151 must be quoted to receive this special subscription price. New subscriptions will start with the next available issue. Details of the Direct Debit Guarantee are available on request. This offer expires 31 August 2015.



Is your content king?

Content-led design is the future. Get ready for
your brand to succeed with a 7-step plan



Why do you need content strategy?

STOP! DON'T DESIGN A THING UNTIL YOUR CONTENT STRATEGY IS SOLID

Design is content. Content is design; it's time to get over the old 'designers-don't-do-words' story. It's time to get intimate with content, and embrace it as the first fundamental part of any design journey.

But before we begin, let's start by defining what a content strategy is. There are so many versions of this answer, but really every single one boils down to this: a content strategy is a road map.

To make better design decisions, you need to know what content is needed for the business to fulfil its promise. For example, if your client wants to be the best online pet store, you'll need to know what content the customers will be expecting and demanding from such a proposition first. Otherwise, anything you produce will fall on deaf ears - or worse, it'll bug them so much, they'll start spouting bad press about your brand. It really does happen, for something so seemingly small.

So why do you need content strategy? It grounds you. It saves time. It saves money. It saves your sanity. When you start talking to your client about their business vision and the content they need to fulfil that promise authentically, you can start to have a meaningful conversation about the experience you're creating, not simply whether the logo is in the right place or the right size.

Imagine, a client/designer relationship where you're both focused on the nuts and bolts of the content first, where the client doesn't simply say they love it, like it, dislike it or hate it - they can articulate why it does or doesn't fit.

It's time for some new thinking. And if you're someone who's serious about solving problems with your design, this is really, truly the new way to work.



Natasha Spice
Designer at Folk
@natashalindblum

“ [Starting a project off with a content strategy] means I can solve problems for real people. Designs can look and feel fantastic, whilst leaving the client and the end-user feeling empowered.”

Five masters of content cartography

THESE ORIENTEERS REALLY KNOW THEIR WAY AROUND THEIR BRAND VISIONS

airbnb airbnb.co.uk

Airbnb's content courses through the site. Belonging is everything, so customers feel at home anywhere in the world.

Warby Parker warbyparker.com

Everything feels like part of a whole. Consumer empowerment, the cool but not over-the-top previews. Shame they're stateside though.

Hatchet + Bear hatchetandbear.co.uk

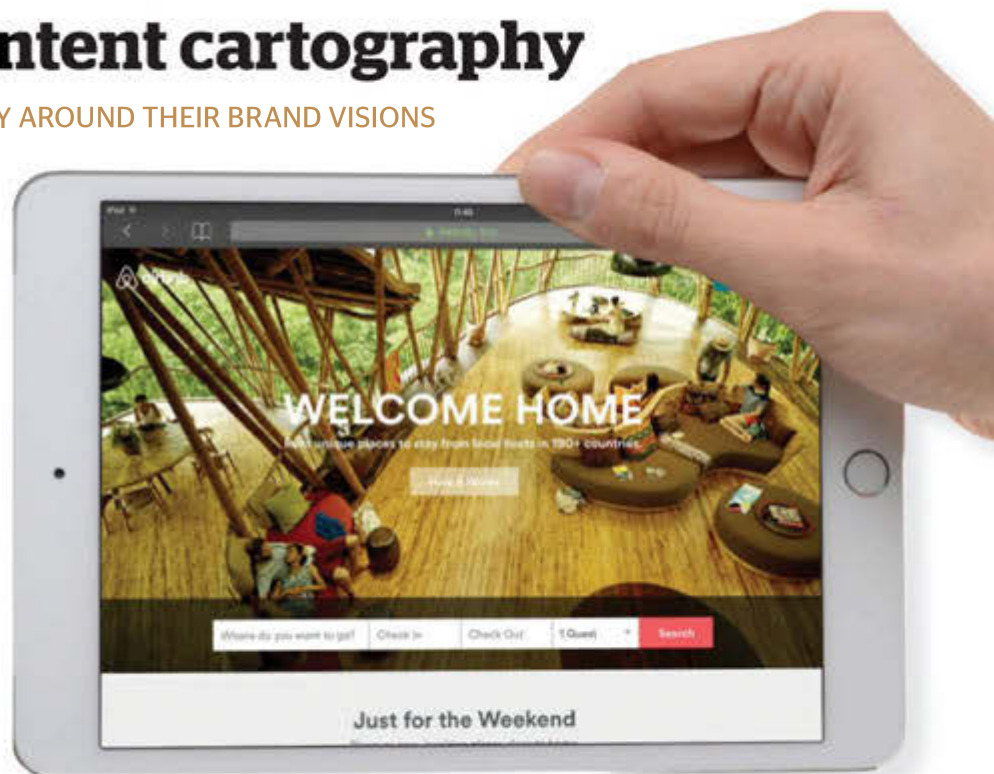
Homely, rustic, unique - Hatchet + Bear know themselves. They understand consistency of personality, which is why they're always out of stock.

Go Pro gopro.com

Go Pro's genius hinges on cleverly showcasing its users' footage throughout the site experience - it's marketing that makes itself.

General Electric ge.com

By creating content linked to vision, the story General Electric tells goes from boring to brilliant, fostering genuine community and action around a greater cause.





Getting started

THE BASICS OF A KILLER CONTENT STRATEGY, IN LINE WITH YOUR BRAND VISION

So, you're about to design a website. Hold it. First, we need to hail back to the old school and draw up a scenario to really think about the process.

Let's say Mr and Mrs Customer saunter up to you and say, "Hi, I'd like to buy a car."

And you say, "Sure, what kind of car do you want?"

"Any will be fine," they reply. "Four wheels and a chassis." You nod. You head over to the garage, and after a few minutes, you pull up in a shiny new hatchback.

They look plain-faced for a second.

"That's not going to work," Mrs Customer says. "Where will our five children sit?"

Don't clench your fists in fury just yet. You could have predicted this.

The user journey

What does a typical user want to do? What are their goals in life, their gains and their pains? Put yourself in their shoes for a second. They might want to spend more time with the kids. They might want to succeed in their job, or attain the perfect work-life balance. Whatever it is, you need to find this goal out, and articulate exactly how you can help them achieve it with your website. If content is king, then the customer is the supreme overlord of the universe. You want them to provide the clicks.

But how do you get the kind of information to make this specificity happen? Everyone knows how annoying it can be to have some anonymous organisation cram a survey down their throats for the chance to win x or y. The problem here lies in the approach. Hell, we're already kicking it old school, so let's just hang around for a while; the best way to find this information is to physically talk to people, face to face. Talk to your friends, neighbours, family members, clients, and anyone you happen to be chatting to through sheer serendipity.

Ask them what makes them happy, sad; ask them what would be their proudest moment if they had the opportunity. Ask them what grinds their gears, and what they'd like to have done as a kid. Everything you can

Case study ASOS asos.com

Most people are at least aware of ASOS in some capacity, even if they don't fall into their 20-something demographic. It's an online fashion superstore, fundamentally. But they are so incredibly good at seamlessly incorporating their content into design, product and messaging that it would be rude to leave them out of this feature. Flawless user experience and targeting means that their customers have literally no reason not to buy again, again and again. Everything is spot on, and laser-focused.

They're great at weaving stories through their products and collections, but notice that it's never ever contrived; think the Festival Chic theme that takes control of their Women's section. This is, once again, something that their customers genuinely want; it's a style that meets their needs and ambitions at the current time in context with what's happening in their customer's worlds. It's also a theme that is bound to be a big product seller for the company too.

They're exceptional at picking partnership opportunities which feel 'on-brand' like Britain's Next Top Model for example, and a variety of festivals - right now ASOS are working on connecting clientele with Glastonbury. Like their content, this is down to good old-fashioned research. The time, the place; the connection. These partnerships are tailored to the consumer's needs to a hair's width.

Every page is a landing page

People won't just be landing on your homepage. Think about the multitude of ways a customer can come to the site. Each page should be self-contained, self-explanatory and unconfusing.

think of. This is bigger than business.

The answers here will be more honest, and more valuable than a dozen surveys. Sure - they might not be as representative as a 10,000 strong body of consumer data, but it's more specific. It's real.

All you need is a trend of five or six, and you can start to get a feel for a customer type and then following on from this you can start to sense the kinds of content that might help them to achieve what the clients actually want to achieve. Another benefit that can be gained here is that you can gain useful, precise feedback on ideas roaming uncharted territory.

It's a good idea to evaluate any existing content once you've done this. This includes anything you or your client may have already, but focuses more specifically on your competitors.

Get a big list - brands whose products are similar, whose audience is similar, whose mantras are similar -



and pick them apart. You'll usually find a hair's width angle in the market that's untapped by anyone. Learn it, own it, master it. This is yours. Write the user and competitor analyses down. Stick them on the wall. They're your bible now.

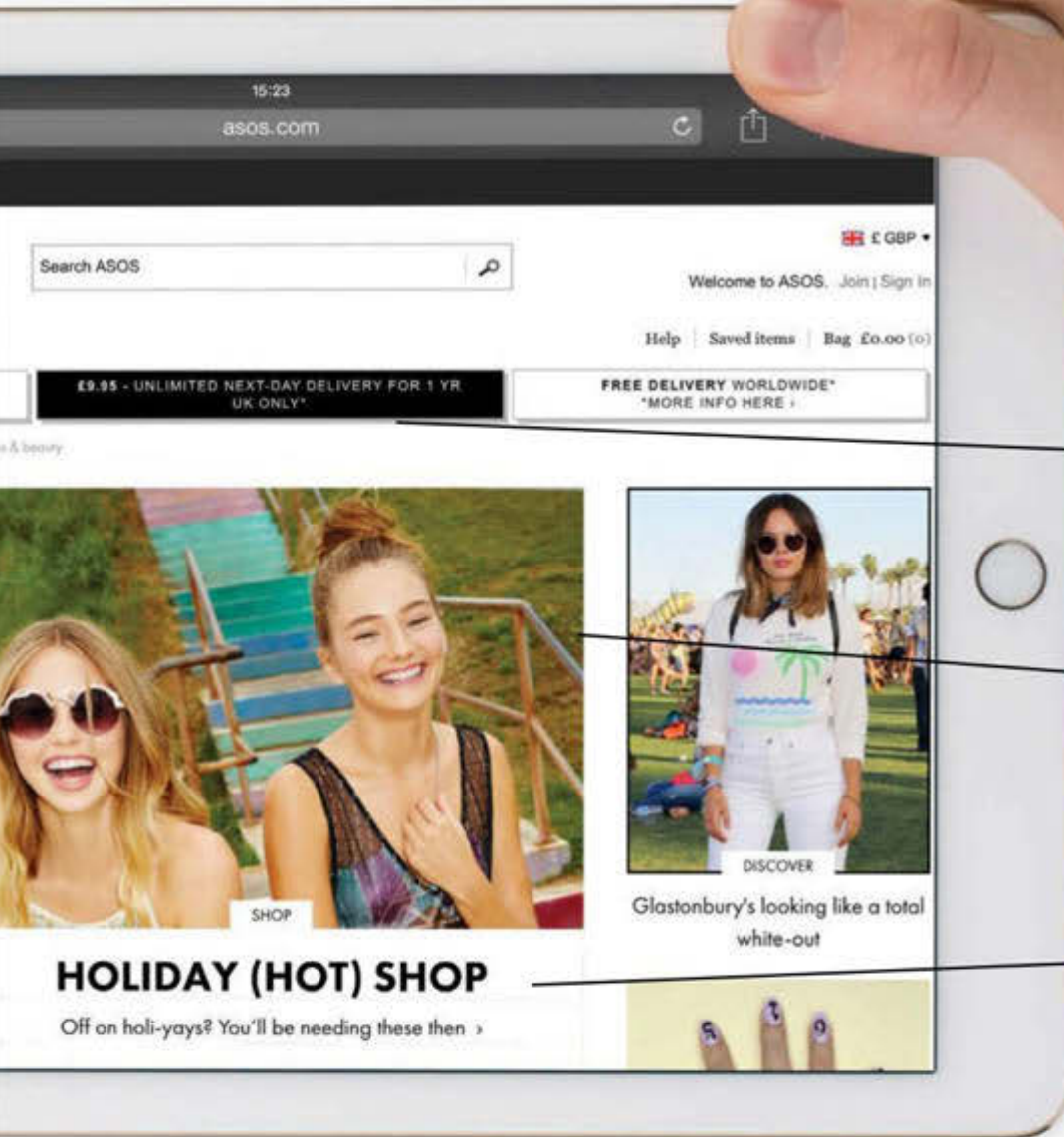
Sitemapping

Before you start writing any content, or drawing up any designs, you'll need a comprehensive map of the pages you need, an idea of what they'll contain, and how they'll interact with each other.

A logical sitemap enables you to maximise your user experience; to see the bigger picture ahead. It also makes it infinitely easier to delegate content generation tasks to freelancers when you've got every item of detail listed out. Moreover, the impact of call to actions can be clearly distinguished for an excellent site journey with admirable click-rates. This same concept is true when it comes to actually designing each page and page



Is your content king?



Mobile and desktop are one thing, but when you're ASOS, you've got a dragon's wealth of content to deliver. And people want to read it. That's why they span multiple channels - digital and analogue - offering things like the ASOS magazine, which you get for free with ASOS Premium.

ASOS are adept in knowing their customer. This knowledge allows them to create content that is always relevant, useful, interesting and truly resonates culturally with their customers. This makes users want to offer things like their phone number as they know it'll be used to enhance their experience.

ASOS live and breathe omnichannel. The symbiotic relationship between content and design is wholly consistent everywhere - from the website to email messaging, to social media, and even their postal slips. This means that you always know where you are, and this builds excellent trust and usability.

component when the content's in place. Here's the slightly tricky part though. How can you design a website that melds seamlessly with its content if you don't have the content itself, using just lorem ipsum to fill in the spaces? And how can you write excellent content when you're unsure of its design context? The answer is simple: mock up the content yourself.

It might not seem like it's in your remit, and as far as finished products go, it's not. But lorem ipsum is the mortal enemy of a content-first approach. If it's an events page you're drawing up, throw in the date, a title and a description. It doesn't have to be good content; it doesn't even have to be grammatically correct but it's better

than using a placeholder when you have no idea what the finished product will actually look like.

Leave the writing of any of the fancy stuff to your copywriter. This extra distance will make it easier for clients to see the whole picture, and get a better feel for the site in the lead-up to sign-off. It makes it a hell of a lot easier for your content writer to populate too. And finally, it makes it so much easier for you to design a site whose essence is there already.

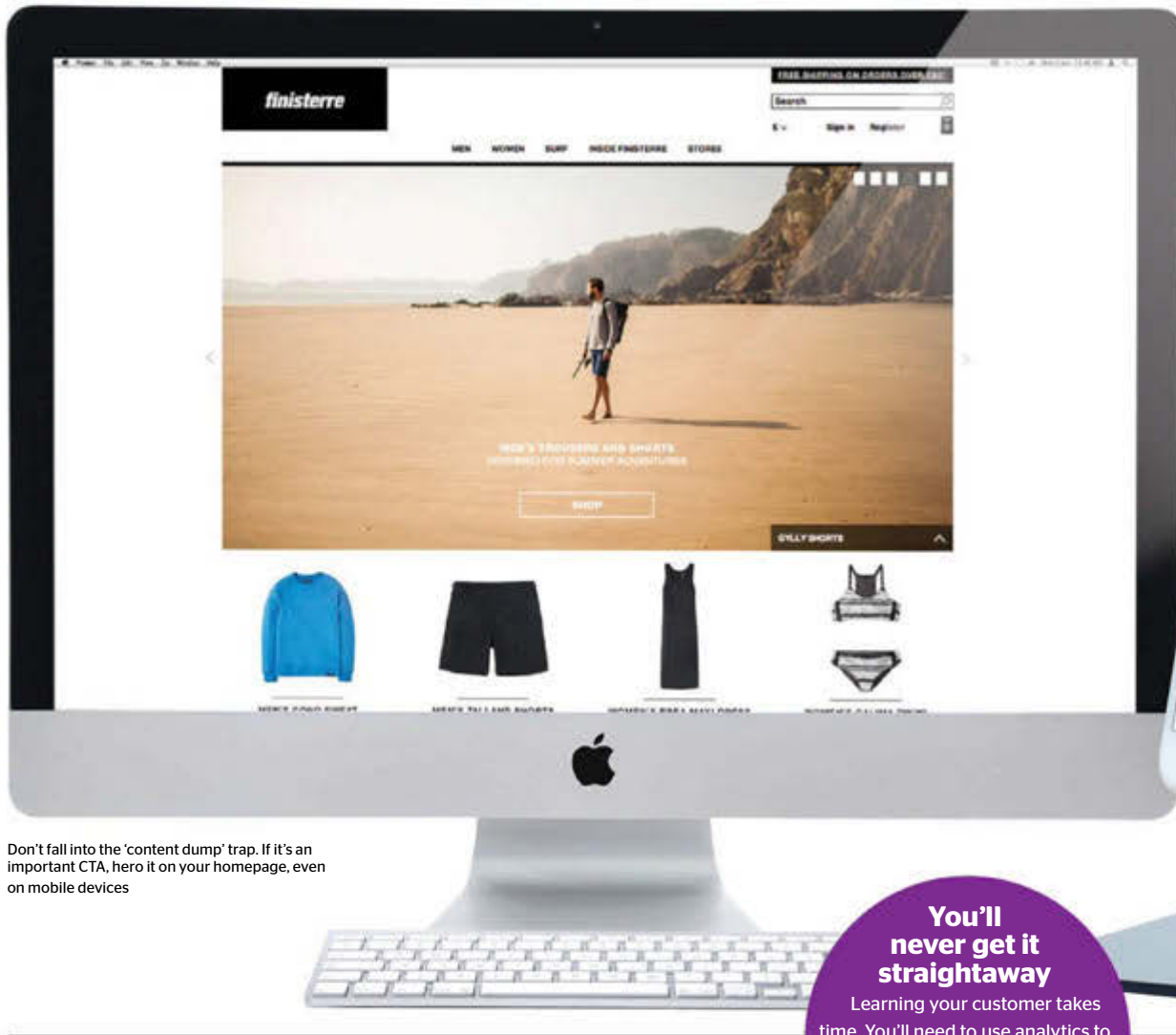
Another point is an important 'don't'. Don't ever, under any circumstances allow the 'luxuriousness' of a brand to influence the readability of its content. Here's an example. Let's say your site's a piece of high-shooting

“A logical sitemap enables you to maximise your user experience; to see the bigger picture ahead”

Plot an effective analysis

You need a solid understanding of a number of things in your competitor analysis. This ranges from homepage meta descriptions and tone of voice, to content themes, objectives, and how effective these appear to be. When you're confident in this, you can plot it and put this to use. You'll need to select two scales, set against axes, eg in a cross-section of cosmetics brands, sustainability/technology vs. fun/professional. Put an X for each brand studied according to how they fare on these scales. Once you've plotted eight to 12 Xs, you'll notice a significant gap in this spectrum. That's the flavour your content needs to have to secure a space in the market.

Is your content king?



Finisterreuk.com is pretty minimal. Notice that only the boldest CTAs are present on mobile

Don't fall into the 'content dump' trap. If it's an important CTA, hero it on your homepage, even on mobile devices

You'll never get it straightaway
Learning your customer takes time. You'll need to use analytics to check how they're interacting with content. You need to take this information and constantly tweak and improve.

contemporary art. It sells high-price-point goods to high-net-worth customers. The first temptation for many is to put on airs with the content. "But if it's a luxury product, surely it's got to sound luxury, right?"

Wrong. Kind of. If your product or service is the kind of incredible masterful, once-in-a-lifetime piece that your customers are looking for, don't put them off with verbosity. Often, you'll lose traction in your attempts to be descriptive. Keep sentences simple, and try to include just one concept in each. Use the active voice, and write as if you were having a conversation with your customer.

Start as close to the end as possible, and don't waste your reader's time

The legendary American writer, Kurt Vonnegut, said to remember these two points. Ultimately, it boils down to a

few things. Research your customer extensively. The more you know about the customer, the greater clarity you'll have on the direction, themes and purpose of your content.

Know their desires, their fears, their ambitions, their habits, their favourite brands, and more. Research these brands - and other competitors - extensively. This includes any content you've done before. The more you know about them, the more likely it is that your content will find a home in the marketplace.

Looking at your own past content will assist you in identifying why things may not have worked so well previously, and might highlight ways you can specifically improve. This research stage is crucial, as it'll define the parameters for your content to sit within.

With this information to hand, next you'll need a sitemap. As mentioned earlier, this gives you a pragmatic knowledge of every pathway, every step, every call to action, and every goal within your site. It naturally gives you a comprehensive list of every page too, so it'll keep you and any content delegates sane and ordered when it comes to designing and populating the site. Remember that dropping in some rough working copy gives the work some vibrancy.

With your themes and content/design relationship bubbling along nicely, don't neglect to check your readability. You can do test this out by using a number of online sources - just look up Flesch-Kincaid, and aim for a score of 65 in its Reading Ease results. Note that the higher the score, the easier it can be understood and vice versa. Texts like *Time* magazine for example have a score of about 52 whereas the BBC News website comes up as 67.4 (bit.ly/1HqOpmd). Remember: even the most elite, technical or hard-to-find products need to be accessible to an online reader. Okay. Now you can move past content strategy and start designing again.

“With your themes and content/design relationship bubbling along nicely, don't neglect to check your readability”



Desktop vs mobile

SHOULD CONTENT DIFFER FROM DESKTOP TO MOBILE? YES. ABSOLUTELY

If your customer is looking for off-the-cuff, pass-the-time content - say they're on the Tube - then it's unlikely they'll be viewing it on their PC. Say they're conducting some in-depth research for a paper: they won't be doing it on their smartphone. This kind of distinction is especially true for features like store finders.

Chances are, people are more likely to use a smartphone for viewing content when they're on the go - so bringing it to the forefront for mobile is important. If you want to take it one step further, great integration with map software means your customers will know you've thought about their journey.

Next, and unsurprisingly, mobile should be as minimal as possible. It's pretty obvious that when you're designing for 320 x 480, you can't include half the stuff you would for 1920 x 1080 without sacrificing a scary chunk of usability. This is when your user journey from before comes in handy. Think back to their pains and gains. What content do they want? Boil every last speck of your site content down to the essence that drives your customer, and minimalism won't be a problem.

On a related note, there's a lot less floating hypertext content on a mobile site, but that's not because the links aren't there. You'll typically see a greater number of buttons and bars which serve this linking function, to accommodate users' stubby fingers. Similarly, but more to do with software and download speed restrictions,

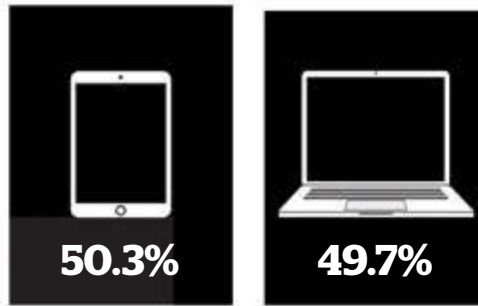
there are far fewer large graphics on a mobile site. It's kind of like the distinction between a corner shop and a supermarket; one's a little more expensive, with a smaller range of products available, but it's supereasy and requires much less effort. For the other, the opposite is true. Both play an important role in our lives

Don't forget buttons
Ban 'Submit' from your dictionary. Use CTAs which grab user attention and are relevant to what you want them to do. If you're offering something for free, use 'get'. If you want to inspire a purchase, say 'purchase'.

Credibility or bust

People online are skeptical about everything. Make sure to include trust indicators in your designs. And while you don't have to state them in your copy, just keep your trust promises front of mind. Place any trust indicators somewhere that follows the eye so they feel natural and unobtrusive.

Where does site traffic come from?



Source - Shopify, August 2014

NINE TO FOLLOW

THE ALL-IMPORTANT ACCOUNTS

Alistapart
@alistapart

For people who make websites - some fantastic resources and authors.

Ian Lurie
@portentint

Ian provides great tips for optimising everything to do with web and design.

Jonathon Colman
@jcolman

The content strategist at Facebook shares his thoughts and insights about everything you do.

Kristina Halvorson
@halvorson

Author of one of the most useful books on content strategy, *Content Strategy for the Web*, Christina knows her stuff.

The Content Marketing Institute
@CMICContent

As the name suggests, this Twitter account is more focused on content marketing, but still has some great resources and insights.

Melissa Rach
@melissarach

The co-author of *Content Strategy for the Web*, regularly posts about the latest content case studies. Melissa describes herself as a content nerd.

Contents magazine
@contents

A digital magazine dedicated to the art of content strategy & online publishing. It hasn't been updated since 2013 but still divulges some must-know topics.

We Live Content
@welvecontent

This account shares timely and useful resources for content marketers and content strategy professionals. Something new appears daily.

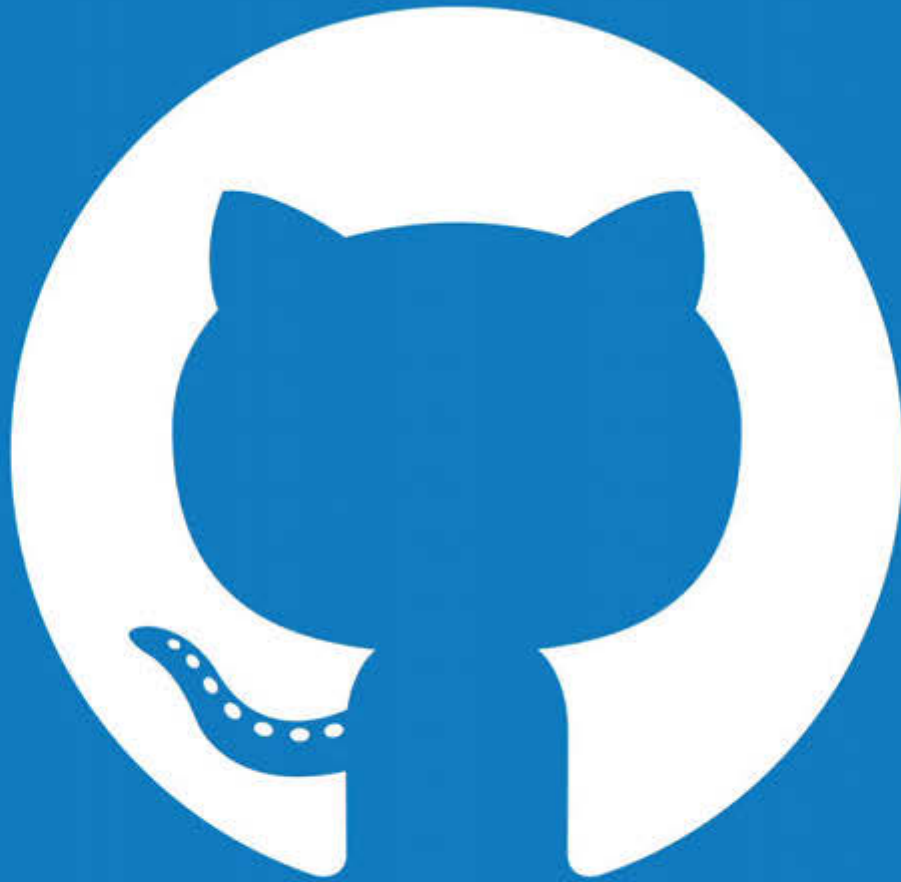
Karen McGrane
@karenmcgrane

Karen is an experienced content strategist who on a good day makes the web more awesome. She is also the author of *Content Strategy for Mobile*.

Content Strategy for the Web
contentstrategy.com
@BrainTraffic

Written by Kristina Halvorson and Melissa Rach, this is one of the best books about content strategy available today. It gives so much information and many tips to make what could be a complicated process very simple, and it gives some totally solid reasons to share with clients about why it's best to get the content parts sorted first above all else.

It's a manual that provides you with the knowledge to deliver meaningful and effective content when and where it's most relevant to your clientele. It also helps you understand processes, teaches you to make better decisions and plan for the long term.



20 BEST GITHUB PROJECTS

GitHub acts as a harbour for impressive code and this selection of extraordinarily cool projects will certainly enrich your coding



GET TO GRIPS WITH FULLPAGE.JS

fullPage.js

github.com/alvarotrigo/fullPage.js

Use for: designing slideshow-like webpages
Generations of badly designed PowerPoint slides have accustomed users to full-screen presentations. This JavaScript framework provides a quick and convenient tool which transforms websites into full-screen PowerPoint-like presentations. Since its initial release, both large and small companies enthusiastically embraced the framework: it was even used on [apple.com](https://www.apple.com) for some time.

As with most other bits of JavaScript code, the usage of the fullPage.js framework will require the inclusion of the jQuery library along with some helper files. Our example here uses the bare minimum of files - advanced scrolling effects will require the inclusion of additional helper libraries:

```
<link rel="stylesheet" type="text/css"
href="jquery.fullPage.css" />
<script src="http://ajax.googleapis.com/
ajax/libs/jquery/1.11.1/jquery.min.js"></
script>
<script type="text/javascript" src="jquery.
fullPage.js"></script>
```

In the next step, the actual content needs to be defined. This is accomplished by adding the following <div> structure to your website:

```
<div id="fullpage">
<div class="section">Some section</div>
<div class="section">Some section</div>
<div class="section">Some section</div>
<div class="section">Some section</div>
<div class="section active">Some section</div>
```

```
div>
</div>
```

By default, fullPage.js displays the section at the top of the DOM tree. You can modify this behaviour via the section active attribute shown below - our snippet would start out by displaying the last item from the list. The individual sections can contain multiple slides which are displayed in a horizontal fashion:

```
<div class="section">
<div class="slide"> Slide 1 </div>
<div class="slide"> Slide 2 </div>
</div>
```

Even though the individual slides can be formatted via CSS, the mandatory bring-up of the framework can also be used in order to provide additional context such as background colours:

```
$(document).ready(function() {
$('#fullpage').fullpage({
sectionsColour: ['#1bbc9b', '#4BBFC3',
'#7BAABE', 'whitesmoke', '#ccddff'],
anchors: ['firstPage', 'secondPage',
'3rdPage', '4thpage', 'lastPage'],
});
});
```

In this snippet, both the background colours and a set of anchors are provided. The latter simplifies the creation of internal links, which permit you to approach individual pages easily.

Describing all parameters of the JSON object passed to the initialisation function would require an entire issue of [Web Designer](https://www.webdesigner.com) to explain. Because of that, please consult the readme file at github.com/alvarotrigo/fullPage.js in order to learn more.

TOP TOOLS

Justice.js

github.com/okor/justice

Use for: analysing web performance

Figuring out more about website performance can be difficult. Justice displays a helpful footer with key metrics.

DeckOfCards

github.com/crobertsbmw/deckofcards

Use for: simulating a card effect with an API

This library re-creates a realistic deck of cards. It is ideally suited to all kinds of card or poker games.

SigmaJS

[sigmajs.org](https://github.com/sigmajs)

Use for: drawing graphs

Displaying graph data can be difficult. This framework takes care of rendering and even manages interactivity.

Glyphs

webhostinghub.com/glyphs

Use for: treating icons as fonts

WebHostingHub provides this custom font containing a load of nicely designed symbols.

Bourbon

bourbon.io

Use for: enhancing CSS

Sass simplifies long CSS declarations. Bourbon adds mix-in support in order to make your CSS even shorter.

VideoJS

videojs.com

Use for: showing off video in a widget

This is a svelte-looking widget dedicated to HTML5 video.

slick

github.com/kenwheeler/slick

Use for: making HTML5 carousels

Slick delivers what its name promises: a highly efficient carousel for highlighting website content and products.

AwesomePlete

github.com/LeaVerou/awesomeplete

Use for: autocompleting code

Users hate typing. This slick AutoComplete text box does its magic without needing bulky external frameworks.

RandomColour

github.com/davidmerfield/randomColor

Use for: generating colours

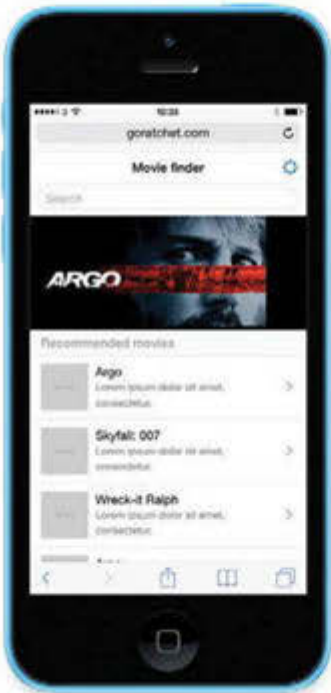
Generating sensible random colours is an act of art. This helper library is dedicated to doing one task well.

App Launch Guide

github.com/adamwulf/app-launch-guide

Use for: learning how to launch an app

This repository is a guide containing a list of hints that makes sure that your app gets attention!



BUILD SIMPLE MOBILE APPS

Ratchet

goratchet.com

Use for: making apps with simple components
Creating good-looking user interfaces for mobile applications constitutes an art of its own. Ratchet is managed by the team behind the well-known Bootstrap framework: it aims to simplify this endless task. After downloading Ratchet, prepare yourself for a complete rearrangement of your application. The framework requires you to adhere to a strict sequence of controls - firstly, all 'bar' items must be right below the <body> tag of the individual pages:

```
<header class="bar bar-nav">
<button class="btn pull-left">
Left
</button>
<button class="btn pull-right">
Right
</button>
<h1 class="title">Title</h1>
</header>
```

As for the actual application, a large selection of controls is provided. For example, tables can be spruced up with various useful gadgets conveying extra information. Apps are made up of forms, each of which is to be contained in an HTML file of its own. They are connected to one another via the Push.js framework - by default, all links are processed by it.

Fortunately, designating external links is as easy as adding the data-ignore property to their declaration. This is necessary for all references which point outside of your app - the following Google link would be a classic example of such a task:

```
<a href="http://www.google.com" data-
ignore="push">Google<a>
```

Ratchet differs from classic GUI stacks such as jQuery UI/ Mobile due to the availability of two stylesheets which seek to mimic the design of the host operating system's controls. Sadly, Ratchet's platform support is limited to recent versions of Android and iOS. The developers have not yet decided whether they want to embrace Windows Phone - as for BlackBerry 10, you don't even need to ask.



CODE CUSTOM BUTTONS

Buttons

github.com/unicorn-ui/Buttons

Use for: implementing fully customisable buttons

Buttons are the epitome of touch-screen user interface design: where there is a touchscreen, expect large and small, red and green, round and square knobs en masse. The ubiquity of this common control has motivated a group of developers to start working on a GUI framework dedicated solely to the display of buttons. This insular approach is beneficiary in that external dependencies are minimised. Getting started with Buttons is as easy as including the following files to your web project - additional resources are needed only for drop-down menus and symbols:

```
<!-- Buttons core css -->
<link rel="stylesheet" href="css/buttons.css">
```

Button's developer team took great care to cover all possible approaches to development. You can create buttons both via <a> and <button> tags:

```
<a href="" class="button button-pill button-tiny">Go</a>
<button class="button button-square button-tiny"><i class="fa fa-plus"></i></button>
```

Buttons is able to implement a few hundred knob types by default. Since most apps are unlikely to require the entire palette, a customisation utility is provided. It enables you to strip out unneeded button types, resulting in the preparation of a custom set of resources which take up less server space and/or bandwidth.



EMBED A FONT ICON SET

mfglabs iconset

mfglabs.github.io/mfglabs-iconset

Use for: embedding icons using a web font or CSS

According to a well-known proverb a figure can be worth a thousand words and these icons will surely say what you want without the text. Most desktop products provide their users with a symbol bar providing quick access to commonly used functions. Mfglabs' iconset will differ from normal icon sets in that it is implemented via a custom font mapped into the Unicode codespace. This is beneficial to us because fonts contain vector information. Your icons will thus look great at all display resolutions and the pixelation that is commonly found in bitmap icon sets will not occur.

Deploy iconset by copying all relevant resources to a folder of your web application. Then proceed to adding the following snippet in order to load the stylesheet and deploy an icon:project - additional resources are needed only for any drop-down menus and symbols:

```
<link rel="stylesheet" href="css/mfglabs_iconset.css">
<i class="icon-paperplane"></i>
```

User interface designers will be quick to point out that misuse of symbols is among the most common mistakes found to impact usability severely. A symbol should be used only when its meaning is 100 per cent clear to the target audience, so absolutely no confusion or second-guessing on their part. Ensuring this becomes especially difficult once products get internationalised.

ANGULAR ALERTS



ng-notify

matowens.github.io/ng-notify

Use for: notifying the user with error messages

A small German startup introduced slide-in notifications in a long-since-forgotten iPad competitor. But ever since Windows Phone 7 brought this concept into wider usage, slide-in alerts have been in ubiquitous usage all over the mobile market.

Ng-notify is a truly tiny add-on for AngularJS apps. It provides developers with a selection of different notification styles which can be deployed with a single function call.

Ng-notify is to be deployed like any other AngularJS module - the code shown in the snippet should be familiar to every AngularJS head:

```
var app = angular.module('demo', ['ngNotify']);
```

Dispatching actual messages can then be accomplished by invoking the set method found in the ngNotify object:

```
ngNotify.set('Your error message goes here!', 'error');
```

In addition to that, Ng-notify permits you to create custom notification styles: customise colours, slide directions and other properties.

Please note that dialogs and alerts spawned by Ng-notify are not modal. This means that they are displayed only while your app is on the screen: if the product is in the background, the user will not see the information.

Row ID	Scroll progress %
2343	46.80%
2344	46.87%
2345	46.89%
2346	46.91%
2347	46.93%
2348	46.95%
2349	46.97%
2350	46.99%
2351	47.01%
2352	47.03%
2353	47.05%

DISPLAY DATA SETS

Clusterize.js

nexts.github.io/Clusterize.js

Use for: improving the frontend

Displaying large amounts of information in a list is challenging and having to handle thousands of DOM elements can overwhelm even the fastest of browsers. Clusterize.js helps us solve this problem of data by recycling the display widgets in a creative way: the framework holds a small amount of templates, which then gets populated with data as soon as the user starts to scroll. Scrollbars are then fooled into accurate positioning via the use of dummy elements. Websites that are working with the Clusterize plugin will tend to work significantly faster. It's simply a must-have if you are using big data sets on your site.

NO-CODE FAKE REST

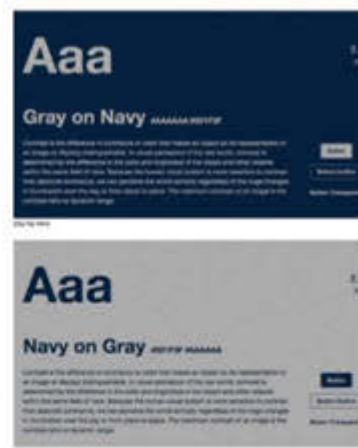
json-server

github.com/typicode/json-server

Use for: optimising development

The initial stages of a client-server project tend to resemble a classic chicken-and-egg problem. Front-end development work cannot commence as long as the back-end functions have not been set in stone. Json-server addresses this problem by providing a surrogate. You specify the desired responses, and treat your json-server instance as if it were the actual production server. Json-server automatically takes care of returning data as required.

Using the product is sensible not only from a front-end perspective.



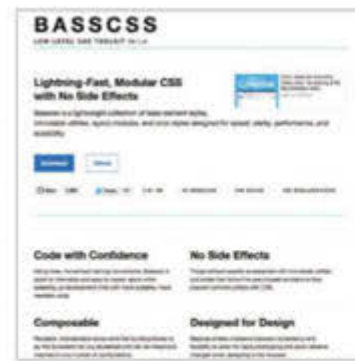
LOW-LEVEL CSS TOOLS

BassCSS

basscss.com

Use for: lightweight CSS styles

Getting colours just right is a balancing act between readability and design: great looks and great readability are two kettles of fish. BassCSS is a collection of interesting, core CSS elements which can be integrated into your application with minimal effort. In addition to that, a selection of 96 readable and good-looking colour schemes, layout and typographic utilities, and reusable layout modules are provided for you. It claims to be responsive by default as it's lightweight and flexible enough to work on any device. You can use one or more of them in your app by simply copying its style declarations into your CSS file.



FIX COMMANDS

The Fuck

github.com/nvbn/thefuck

Use for: correcting commands

Get one character wrong with the Unix command line and the whole command is refused. The Fuck is a workaround that analyses wrongly entered commands; it suggests fixes like this Python example:

```
→ apt-get install vim
E: Could not open lock file /var/lib/dpkg/lock - open (13: Permission denied)
E: Unable to lock the administration directory (/var/lib/dpkg/), are you root?
→ fuck
sudo apt-get install vim
```

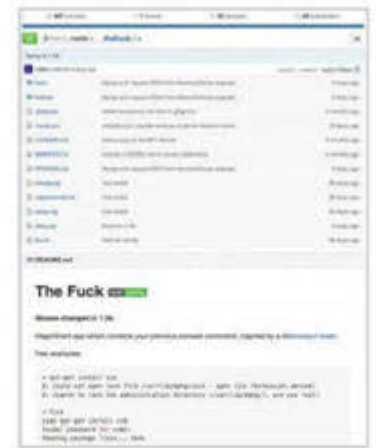




IMAGE ZOOM JQUERY PLUGIN

Zoom.js

github.com/fat/zoom.js

Use for: creating a CSS image dialog

Displaying images inevitably becomes a balancing act between visible detail and screen real-estate consumption. Photographers and infographic designers obviously want their creation placed in the limelight - UI designers tend to focus their attention on reading flow.

Zoom.js solves this problem by transforming small images into large clickable galleries. Clicking an image opens a pop-up with the image in its full glory; this is ideally suited for device reviewers wanting to provide their readers with an optional, larger view of interesting pictorial material.

Embedding the Zoom.js plugin can be accomplished by adding the following three files to your web

application. Transition.js is not a part of the main framework, but is instead made available as part of the bootstrap framework:

```
<link href="css/zoom.css" rel="stylesheet">
<script src="js/zoom.js"></script>
<script src="js/transition.js"></script>
```

Any individual tags must also then be enhanced with a bit of markup in order to display the images in a larger fashion:

```

```

Keep in mind that Zoom.js is no solution for the bandwidth demands of large images. Loading a large image takes some time even if it is displayed in the scaled-down version: as of writing, Zoom.js is not able to load a different resource as images get clicked.

FIVE TO FOLLOW

GitHub @github

Do you use a GitHub service? Then you better follow GitHub's official Twitter account, which provides a never-ending source of all kinds of information which are interesting for the Giterati.

GitHub Status @githubstatus

Like most other computer systems, GitHub will go down from time to time. Following githubstatus is an easy way to get notified of planned maintenance and unplanned server outages.

Tom Preston-Werner @mojombo

Having found himself run out from his company by a - somewhat dubious - scandal, Tom deserves credit and attention for being the coder who had the idea to give GitHub to the world.

Chris Wanstrath @defunkt

The current co-CEO of GitHub takes to Twitter from time to time. Following him might be interesting if you are into adding a bunch of prominent people to your Follow list.

Linus Torvalds @linus__torvalds

A feature on GitHub wouldn't be complete without a mention of Linus Thorvalds. The world-famous programmer created Git in order to maintain the source code of the Linux kernel.

RESOURCES & GUIDES

The GitHub docs

help.github.com

Learning about version control systems used to be a daunting task. Fortunately, the team at GitHub strives to make its product as accessible as possible. Visiting the help pages provides a torrent of information about both simple and advanced features of their various free and premium offerings.

Search syntax

help.github.com/articles/searching-github

Find code quickly by using advanced searching syntax in the Search box

Go Pro!

enterprise.github.com/home

GitHub's free service is not for everyone but you can pay a bit extra and demand the best by going Pro!

The GitHub blog

github.com/blog

Gossip, updates and the latest news from the house of GitHub can be found in the official blog.

Git Manual

git-scm.com

GitHub is a hosted instance of Git. More information on Git looms in the documentation on the website.



FOR THE GNU GENERATION

www.linuxuser.co.uk



Linux User & Developer

Available from all good newsagents & supermarkets today

ON SALE NOW:

» Build better servers » Code a Raspberry Pi synth » Make a browser



BUY YOUR ISSUE TODAY

Print edition available at www.imagineshop.co.uk

Digital edition available at www.greatdigitalmags.com

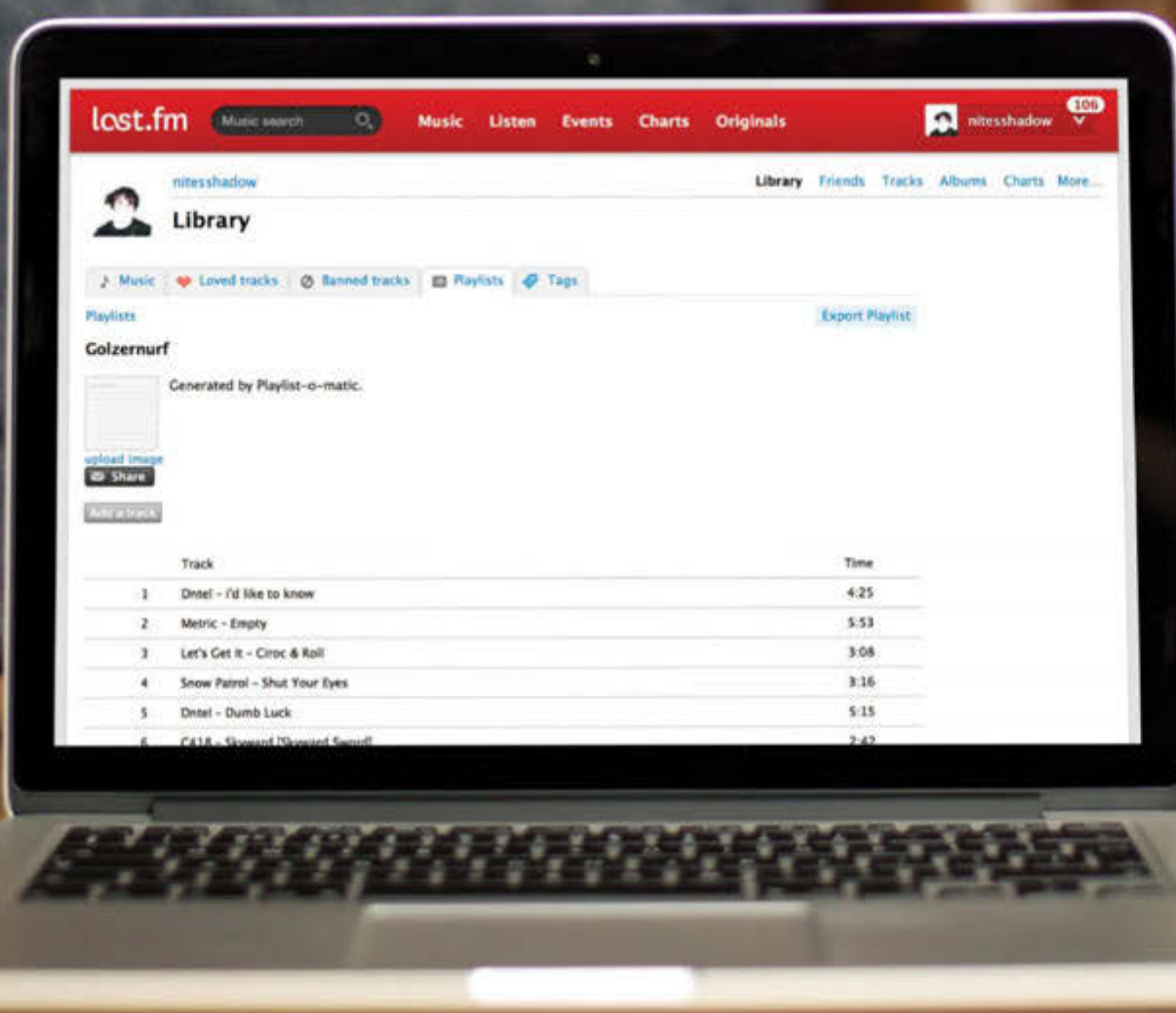
Available on the following platforms



[facebook.com/LinuxUserUK](https://www.facebook.com/LinuxUserUK) twitter.com/LinuxUserMag

Make playlists with your friends using the Last.fm API

Master Fetch and Last.fm's API by making a playlist from you and your friends' mutual tastes





Music is both universal and incredibly subjective. What one person considers the epitome of human creative achievement another may dismiss as bad taste. In this tutorial we're going to attempt to bridge the gap by using data from Last.fm to create playlists of songs you and a friend have in common.

Last.fm was founded in 2002 by two university students and has since been bought out by CBS Interactive. It offers a place to store what you've listened to and has a number of API services that enable developers to dig into this rich mine of data.

To help us achieve this we're going to use a new method for making requests called Fetch (more information at fetch.spec.whatwg.org). Fetch is currently supported by Chrome 42, Firefox 39 and Opera 29. Fetch aims to replace the old and somewhat hack-y XMLHttpRequest introduced by Microsoft in 1999. The web has come a long way since then and Fetch is designed around streams and promises.

The front-end portion of this tutorial is relatively straightforward but previous experience with Angular will aid understanding. The server-side portion is covered briefly but the full source code with comments can be found on FileSilo.

1. Install Express

The server for our simple app will be Node.js using Express. This will then enable the scaffolding process of an Express app for you and start to serve it up on port 3000. We'll be editing the app.js file at the root of the project later on. We'll also use a wrapper for the Last.fm API on the server to help with authentication.

2. Install client-side packages

Code that is served to the user is stored under the 'public' folder. Navigate to this folder and run the following Bower command. Alternatively you can install Angular 1.3, angular-query-string (npmjs.com/package/angular-query-string) and Underscore manually, and Meteor will automatically restart.

3. Capture username

Now we'll start writing our Angular app. We want to capture a Last.fm username so that we can start the process of finding friends on Last.fm. Create index.html at the root of the project and a text input which updates the 'user' model when focus moves from it.

```
<body data-ng-app="app" data-ng-controller="FriendController" data-ng-cloak>
<input class="username-input" type="text"
placeholder="Your Last.fm username"
data-ng-model="user" data-ng-model-options="{ updateOn: 'blur' }">
<!-- script tags -->
</body>
```

4. Define module

Back in the public folder make a JavaScript file called 'playlist-app.js'. The module 'app' is what the ng-app attribute in the previous step looks for; likewise with the controller and its definition. We're also injecting a LastFmService which we'll create later on. Note the dependency on angular-query-string as well which provides UrlQueryString.

```
angular.module('app', ['angular-query-string'])
.controller('FriendController', ['$scope', '$window', 'LastFmService', 'UrlQueryString', function ($scope, $window, Lastfm, UrlQueryString) {
}]);
```

5. Get friends

When the user enters a username we want to make a request to the Last.fm API to get a list of their friends. There are many ways we could do this, one alternative would be to add a function to the input which updates on blur, the same as the model, but we're going to watch the user property for changes.

```
$scope.$watch('user', function (username) {
if (username) {
lastfm.getFriends(username).then(function
```

```
(data) {
$scope.friends = data.friends.user;
$scope.$apply();
});
}
});
```

6. Create Angular service

You may have noticed that we called lastfm.getFriends but currently 'lastfm' doesn't exist in our app! We'll rectify that by adding a service to the 'app' module. You'll need to register for a Last.fm API key by going over to last.fm/api/account/create.

```
.service('LastFmService', function () {
var url = 'http://ws.audioscrobbler.com/2.0/?api_key=YOUR_API_KEY&format=json';
return {
/* next step */
};
});
```

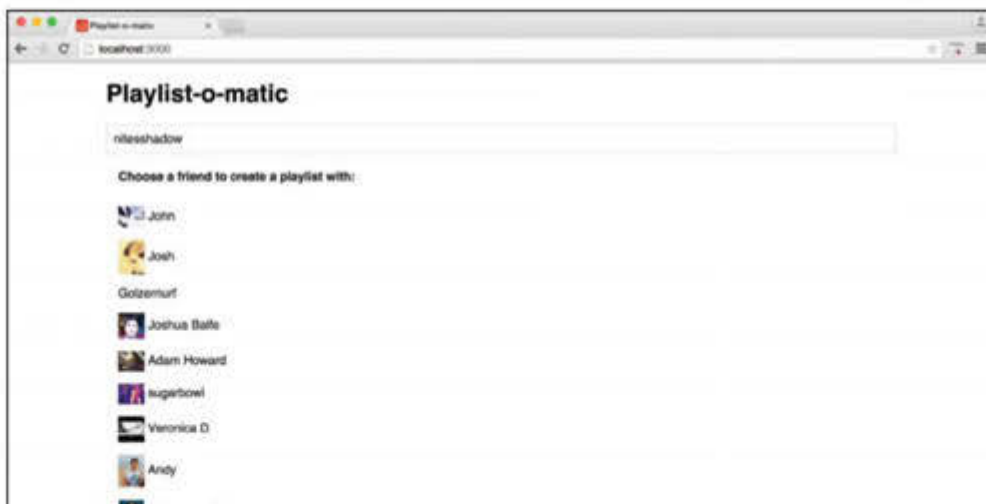
7. Last.fm API call

We're using the experimental Fetch API to make a request to user.getFriends (last.fm/api/show/user.getFriends). Fetch works with streams and promises which are a little exotic. The Promise API means that we can chain multiple asynchronous functions with 'then'. Response.json() returns a stream so that the reading of large files can be progressively rendered.

```
getFriends: function getFriends (username) {
return fetch(url + '&method=user.getfriends&recenttracks=1&user=' + username)
```

Network red herring

If you're using the Network tab in your Web Inspector to debug requests made by Fetch, they won't appear under the XHR tab as they're not XMLHttpRequests!



Left
When a username is entered we can use it to get a list of their friends and display it

Top left
By using Bower to manage our packages we can include multiple projects at once and increase the project's maintainability as well

Top right
This input captures the username input and is tied to the 'user' model. Updating on blur keeps requests low

Developer tutorials

Make playlists with your friends using the Last.fm API

```
.then(function (response) {  
  return response.json();  
});  
},
```

8. Display friends

Fetch is making the API call and we've already written a handler for it by populating `$scope.friends` with the result. To finish showing our user's Last.fm friends we just need to write a simple repeater which will show a list of the friends. We're outputting a small image and prefer the 'real name' over the username.

```
<section class="friend-container" data-ng-  
show="friends.length">  
<h4>Choose a friend to create a playlist  
with:  
</h4>  
<ul>  
<li class="user" data-ng-repeat="friend in  
friends" data-ng-  
click="chooseFriend(friend)">  
{{friend.realname || friend.  
name}}  
</li>  
</ul>  
</section>
```

9. Choose friend

You should now have a list of friends appearing and when the user clicks one of these friends we want to start the comparison process. Create a playlist array which will contain the track objects to display and call 'compare' on the 'lastfm' service. Then pass it the username of who we are and the friend's name.

```
$scope.chooseFriend = function (friend) {  
  $scope.playlist = [];  
  $scope.chosenFriend = friend;  
  lastfm.compare($scope.user, friend.name)  
  .then(function (data) {  
    var artists = data.comparison.result.  
    artists.artist;  
    return artists;  
  }).then(function (artists) {  
    /* next step */  
  });  
};
```

10. Get tracks

Anticipating the response from that, we'll loop over each mutual artist that's returned from Last.fm. For each artist we call another service which looks for tracks by a given artist that our friend has listened to and select a random two from the list returned. For those two tracks we call another method we haven't written yet called 'addTrack':

```
artists.forEach(function (artist) {  
  return lastfm.getTracksOfArtist($scope.  
  chosenFriend.name, artist.name)  
  .then(function (tracks) {  
    _ .sample(tracks.artisttracks.track, 2).  
    forEach(addTrack);  
  });  
});
```

11. Last.fm API calls

The 'compare' and `getTracksOfArtist` methods follow a similar format to the previous API call with Fetch. In both cases we parse the response as JSON and then pass it onto the next function so that we can access the response in the controller. The comparison is done via Last.fm's tasteometer, which also returns a 'compatibility' rating between 0 and 1.

```
compare: function (user1, user2) {  
  return fetch(url + '&method=tasteometer.comp  
are&limit=30&type1=user&type2=user&value1='  
+ user1 + ' &value2=' + user2).  
  then(function (response) {  
    return response.json();  
  });  
},  
getTracksOfArtist: function (friend, artist)  
{  
  return fetch(url + '&method=user.  
getartisttracks&user=' + friend + '&artist='  
+ artist).then(function (response) {  
    return response.json();  
  });  
},
```

12. Add track

The logic for adding a track to the playlist is reused, so to keep things in line with DRY (the don't repeat yourself principle) we've made it a separate function. It ensures the track has a name and loops through the array to make sure the Last.fm ID isn't already in there. If it is, it adds a track and shuffles the playlist to mitigate the same artist being played sequentially.

```
var addTrack = function (track) {  
  if (track && track.name) {  
    var inPlaylist = $scope.playlist.
```

An extensive API

Last.fm has 133 documented services for you to use, ranging from user information to geographical events. You can read about in more detail at last.fm/api.



Top left

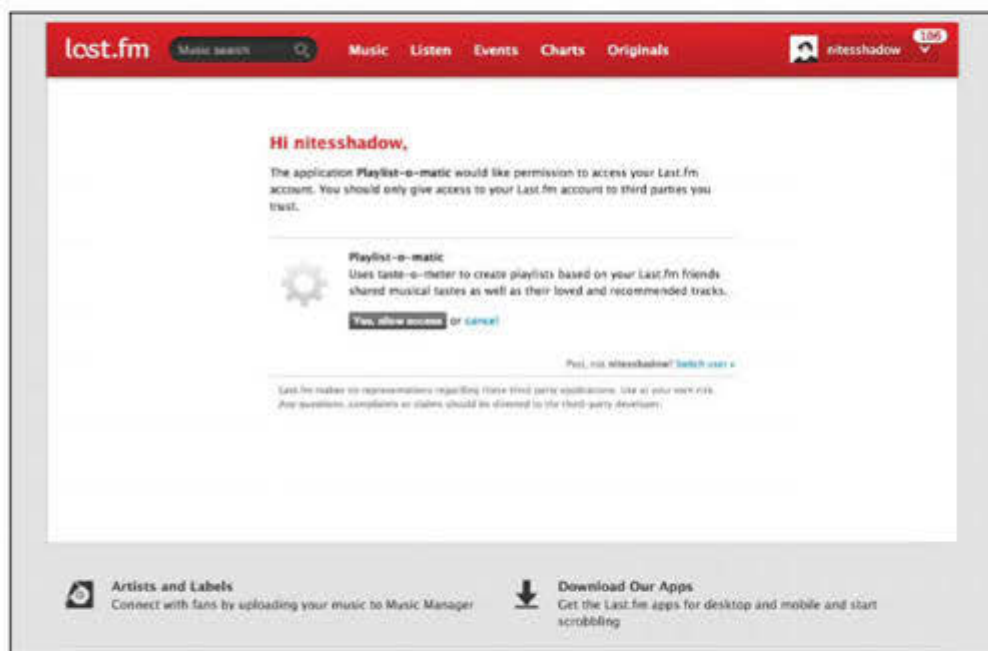
A request made by the Fetch API contains all of the same information as the trusty XHR request

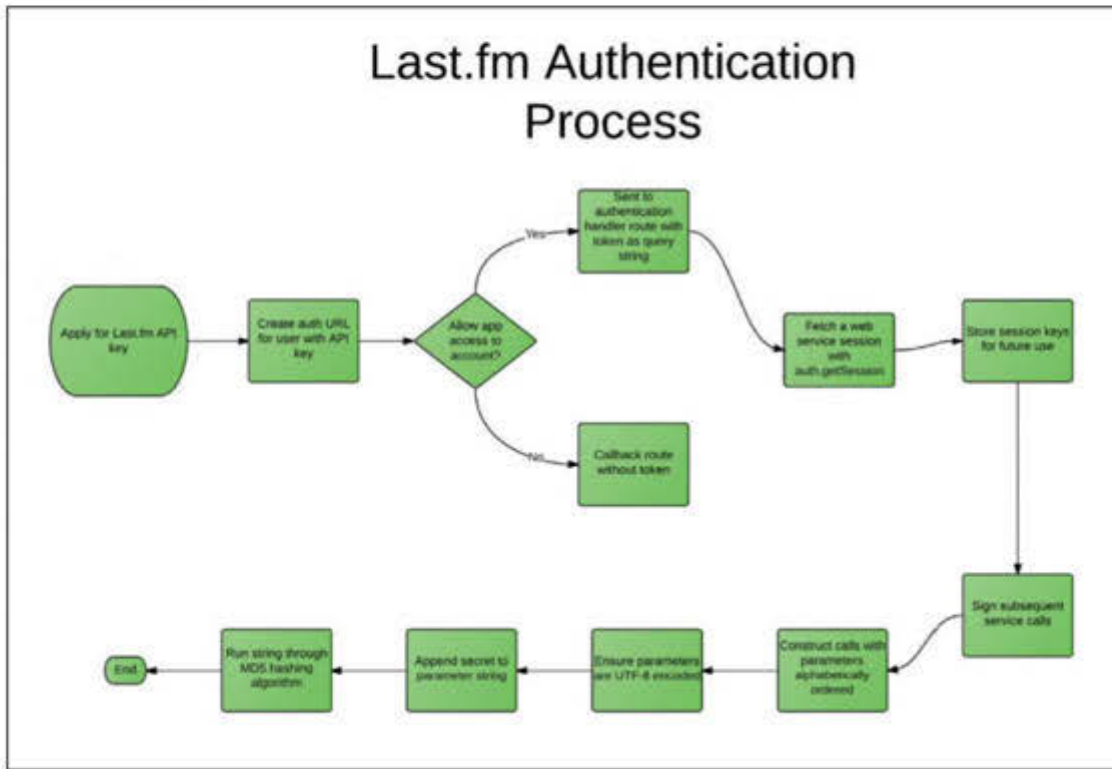
Top right

We will need to hand the user a link to see their newly created playlist and cement their friendship

Right

Taking the user to the Last.fm site provides a handy breakdown of what they're enabling the app to do





Last.fm authentication

We've used a library to deal with the authentication process for us. You make a request with your API key and in return (after the user has granted permission) Last.fm gives you a token that can be saved to make subsequent requests without having to reauthenticate. The trickiest part of this process is signing your calls to API methods that require authentication. They require parameters to be supplied alphabetically and UTF-8 encoded, then hashing it with the MD5 algorithm. Any language can be used and Last.fm has a list of recommended wrappers including Python and .NET. It could even be done client-side but this would mean exposing your app's secret key, which is not recommended.

```

some(function (t) {
  return t.mbid.length && t.mbid === track.mbid;
});
if (!inPlaylist) {
  $scope.playlist.push(track);
  $scope.playlist = (_.shuffle($scope.playlist));
  $scope.$apply();
}
}
});

```

13. Get loved tracks

You could use this project to play some songs when you're both in the same physical space. Being a good host, you'll probably add some songs for your guest so we'll add a couple of their loved tracks to the playlist.

```

.then(function () {
  lastfm.getLovedTracks($scope.chosenFriend.name).then(function (data) {
    _.sample(data.lovedtracks.track, 2).forEach(addTrack);
  });
});

```

14. Loved tracks request

The actual API call is similar to the last few. Most of the Last.fm services return a subset of the total results and has a pagination system so you could loop through incrementing the &page value. You can also limit the results if you know you want less than 50 results.

```

getLovedTracks: function getLovedTracks (user) {
  return fetch(url + '&method=user.getlovedtracks&user=' + user).then(function (response) {
    return response.json();
  });
},

```

15. Display playlist

We've got some tracks coming back from the API so let's display them! Create an ordered list and display the track name, artist and album with a hyphen between them. Different services will either return .name or #text.

```

<section class="playlist-container" data-ng-show="playlist.length">
  <h1>Your playlist with {{chosenFriend.realname || chosenFriend.name}}</h1>
  <ol>
    <li class="track" data-ng-repeat="track in playlist">
      {{track.name}} <span data-ng-show="track.artist.name || track.artist['#text']"></span> {{track.artist.name || track.artist['#text']}} <span data-ng-show="track.album.name || track.album['#text']"></span> {{track.album.name || track.album['#text']}}
    </li>
  </ol>
  <button data-ng-click="savePlaylist()">Save playlist</button>
</section>

```

16. Save playlist listener

We added a button to save the playlist. To create a playlist, authenticate the user on the server. Now send over the name of our friend and the playlist tracks. When we receive a response it will send us to authenticate on the Last.fm site.

```

$scope.savePlaylist = function () {
  var data = JSON.stringify({
    name: $scope.chosenFriend.name,
    tracks: $scope.playlist,
  });
  lastfm.savePlaylist(data).then(function (url) {
    if (~url.indexOf('last.fm')) {
      $window.location = url;
    }
  });
};

```

17. Fetch POST request

POSTing data with the Fetch API is slightly more involved than the previous GET requests we've made. Instead of encoding the data in the URL, this time the data is sent with the POST request body. It's also essential that we set the Content-type header to application/json for the server to identify it.

```

savePlaylist: function savePlaylist (data) {
  return fetch('http://localhost:3000/save-playlist', {
    method: 'post',
    headers: {
      'Content-type': 'application/json;

```

Developer tutorials

Make playlists with your friends using the Last.fm API

```
charset=UTF-8'  
},  
body: data  
}).then(function (response) {  
return response.text();  
});  
}
```

18. Authenticate on server

Now we're going to jump to app.js at the root of the project. We're going to work around the code that's already there so you might want to append the following to the end of the file or copy the contents of the included resources from this tutorial on FileSilo. Const is a type of variable introduced in ES6 which cannot be altered.

```
var LastfmAPI = require('lastfmapi');  
const LASTFM_API_KEY = 'YOUR_API_KEY';  
const LASTFM_API_SECRET = 'YOUR_API_SECRET';  
var lfm = new LastfmAPI({  
  api_key: LASTFM_API_KEY,  
  secret: LASTFM_API_SECRET  
});
```

19. Express routes

To ensure everything's running in the same place, serve the index.html file at the root level. The /save-playlist route is what we POSTed the playlist data to; this is stored in a variable that we can access elsewhere for now (this should be done via sessions in a real project).

```
app.get('/', function (req, res) {  
  res.sendFile(path.join(__dirname + '/index.  
html'));
```

```
});  
app.post('/save-playlist', function (req,  
res) {  
  data = req.body;  
  var authUrl = lfm.getAuthenticationUrl({  
    'cb' : 'http://localhost:3000/auth' });  
  res.send(authUrl);  
});
```

20. Last.fm authentication

In the last step we told Last.fm to send the user back to <http://localhost:3000/auth> once they've provided the app access to their account. This process returns us a unique token which we can use to then create a session.

```
app.get('/auth', function (req, res) {  
  var token = url.parse(req.url, true).query.  
  token;  
  lfm.authenticate(token, function (err,  
  session) {  
    lfm.setSessionCredentials(session.username,  
    session.key);  
    // next step  
  });  
});
```

21. Add tracks

To create the playlist we need to give it a title (the friend's name) and a description. Then we use a callback function to add each track to the playlist. We've set a timeout of 500 milliseconds between each request or else, only some of the tracks will be added. Once created we redirect with parameters indicating the playlist's URL.

```
lfm.playlist.create({  
  title: data.name,  
  description: 'Generated by Playlist-o-  
  matic.'  
}, function(err, playlists) {  
  data.tracks.forEach(function (track, index)  
  {  
    setTimeout(function () {  
      lfm.playlist.addTrack(playlists.playlist.id,  
      track.artist['#text'] || track.artist.name,  
      track.name);  
    }, 500 * index);  
  });  
  res.redirect('/?playlist=' + playlists.  
  playlist.url + '&friend=' + data.name);  
});
```

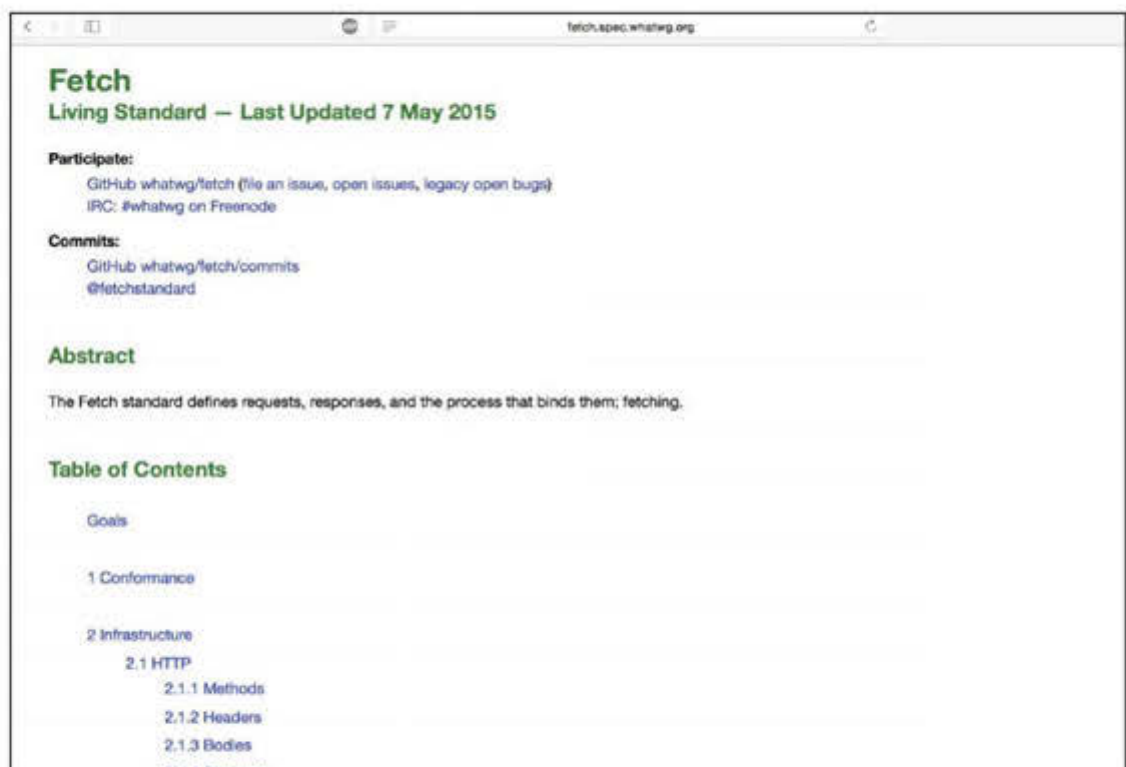
22. Share playlist URL

When the user is redirected to where they began we'll display a message to say the playlist was created.

```
// client///playlist-app.js  
if (UrlQueryString.friend) {  
  $scope.created = {  
    friend: UrlQueryString.friend,  
    url: UrlQueryString.playlist  
  };  
}  
//index.html  
<p data-ng-show="created">Your playlist with  
{{created.friend}} <a href="{{created.url}}"  
target="_blank">has been created!</a> <a  
href="/">Create another?</a></p>
```

Fetch: gotchas and how to use it today

A few 'gotchas' to look out for is that you can only read a Fetch's stream once. If you write 'response.json(); response.text()'; the second will throw an error because the stream has already been read. If you want to do this then write 'response.clone().text()'. Also, there is no way to cancel a request but this may be fixed in the future. This is due to the Promise-based nature of Fetch. Fetch doesn't have comprehensive support and to tackle this, GitHub developers wrote a polyfill which wraps XHR with the Fetch API (see github.com/github/fetch). Fetch can also be used in service workers, unlike XHR requests which can't. This means that your asynchronous processes can spawn more asynchronous processes!



Special offer for readers in **North America**



2 issues FREE

When you subscribe*

FREE
resource
downloads
in every
issue



“ *The only magazine you need to design and develop stunning websites* ”



Order hotline **+44 (0)1795 592862**

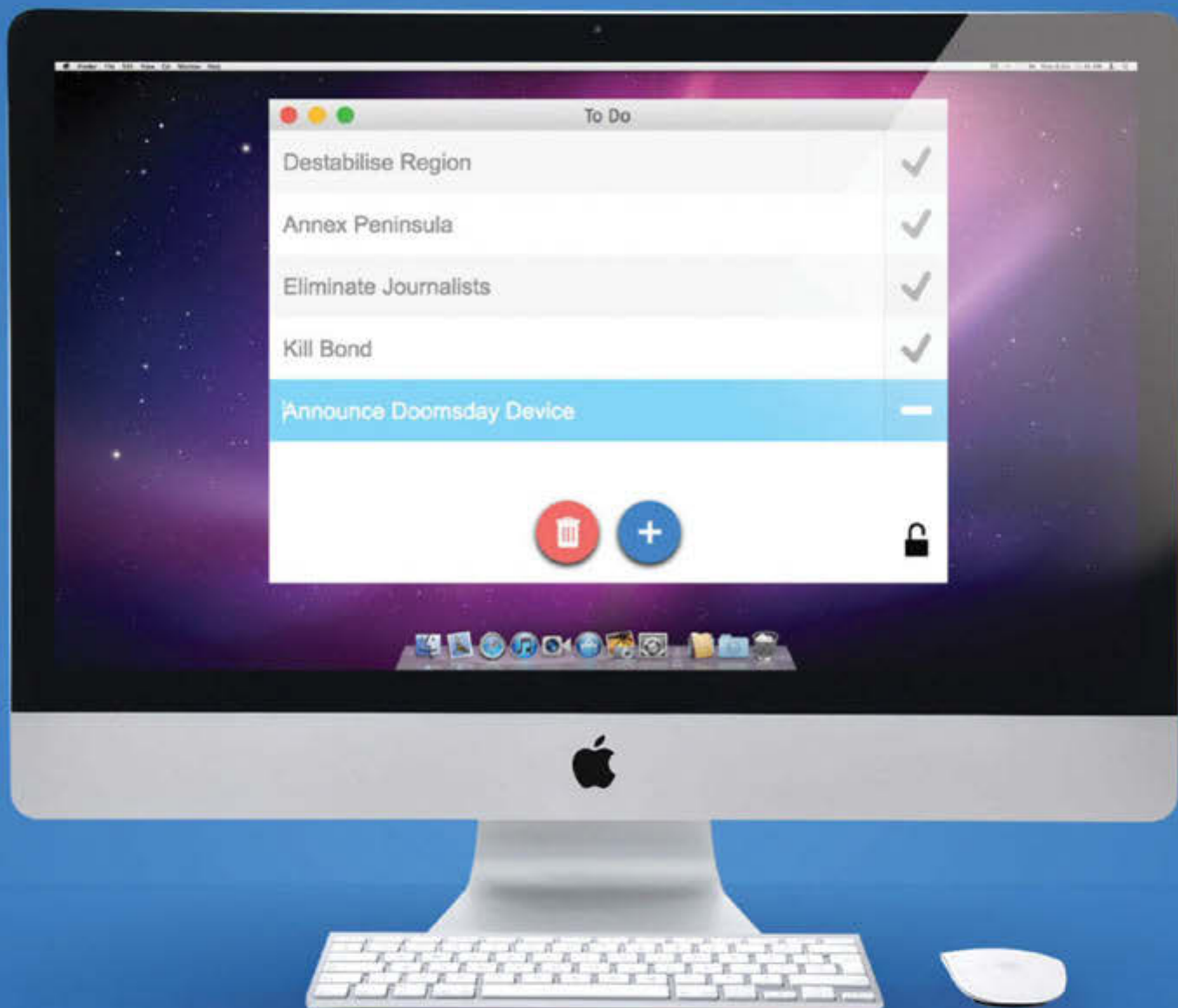
Online at **www.imaginesubs.co.uk/wed**

*Terms and conditions: This is a US subscription offer. You will actually be charged £37 sterling for an annual subscription. This is equivalent to \$57 at the time of writing, exchange rate may vary. 2 free issues refers to the USA newsstand price of \$14.99 for 6 issues being \$89.94, compared with \$57 for a subscription. Your subscription starts from the next available issue and will run for 6 issues. This offer expires 30 September 2015

Quote
summer151
for this
exclusive
offer!

Build desktop apps with NW.js and JavaScript

Use NW.js to create JavaScript-powered desktop apps that can be run on OS X, Linux and Windows





ver the last 25 years, the browser has opened new windows (see what we did there) on content and code distribution. Whole ecosystems have

sprung up to support the creation and deployment of apps that can be accessed with a simple URL. Despite how great the browser has become at integrating itself into operating systems, we still sometimes find that we want an app that we can install on our system and run at any time with just that extra little bit of power.

What we need is something with the low-level access of Node with the ease of development of browsers and a way to package it all up as an app that can be run on our traditional OS like any regular, native app. That's what NW.js (formally known as Node-WebKit) is. NW.js bundles io.js and the WebKit browser engine together in a way that lets us access the full power of Node through the DOM. This then enables us to write web apps and also to run them in self-contained windows that look almost exactly like OS native apps.

In this tutorial, we're going to set up NW.js and create a simple to-do app to show how simple it is to write native apps with JavaScript and NW.js. We will be doing this for Mac OS X.

1. Get NW.js

Installing NW.js is pretty simple, all we need to do is head on over to GitHub (github.com/nwjs/nw.js) and grab the build for your OS. If we were so inclined we could build NW.js from scratch, and that may be advisable if you want to be certain of stability, but for learning, the prebuilt binaries will do. For this tutorial the 32-bit release of NW.js 0.12.1 was used and tested on OS X 10.10.3.

2. Relocating the NW.js executable

Unzip the package we just downloaded. This folder contains the prebuilt NW.js executable that we can use to run our apps. When we get to testing and deploying our apps, we're going to use the nwjs app bundle included in this folder, so move the entire folder to a safe, memorable place - like Documents.

3. Create a NW.js project

NW.js apps are supersimple with few prerequisite dependencies, so there's no need for a project creation tool like those found in, say, Cordova. Create a new folder somewhere on your system and create a HTML file called 'index.html' and insert the code below. This will make up the UI of our app and load the NW.js code to handle the creating of our window.

```
<!DOCTYPE html>
<html>
<head>
<meta content="text/html; charset=utf-8"
http-equiv="Content-Type">
<title>To Do</title>
<link rel="shortcut icon" href="favicon.
ico">
<link rel="stylesheet" href="styles.css"
type="text/css" />
<meta name="viewport" content="initial-
scale=1.0, user-scalable=no" />
</head>
<body>
<h1>Hello World</h1>
</body>
</html>
```

4. Create the manifest

Just like an io.js app, NW.js uses a JSON manifest to get the wheels rolling with our app. Create a file called 'package.json' in the same folder as your index.html. This is the root of your NW.js, all URLs are relative to the package.json file.

5. A simple manifest

Open up your new manifest file. The simplest manifest requires two properties, a main and a title. The main is the file that our app will launch when it's first run, so in our case it's index.html. There are other properties that can affect how our app runs, but for now add this:

```
{
  "title" : "To Do Today",
```

```
  "main" : "index.html"
}
```

6. Point the app

When we're testing out our app, we shouldn't have to package and repack it every time we correct code. We can use the NW.js executable to run our app by pointing it to the root of our project. Open up a terminal and navigate to the folder downloaded from GitHub.

7. Launch our app

The NW.js executable is buried in the NW.js app in our folder. The executable is the program where the io.js and NW.js code lives. With NW.js, we don't change the app, we create projects that run with it. To run the app, enter:

```
$ nwjs.app/Contents/MacOS/nwjs /path/to/
your/project/root.
```

8. First impressions

A new window will have opened and a NW.js icon will have appeared in your dock. We're running our first NW.js app, and we can see our HTML file - but it still looks like a browser window. To make our app look a little more like it belongs, we can remove certain elements with our package.json manifest.

9. Enhance our manifest

We've got some system chrome we need to get rid of now! Just after the 'main' property of your manifest, add a 'window' object and add the following properties to it:

```
{
  "title" : "To Do Today",
```

Compiling code

There is no compiling per se in NW.js, which means that you may want to keep your source code from prying eyes. There are ways to make it quite hard to get at your source, documented on NW.js' GitHub wiki (github.com/nwjs/nw.js/wiki).



Left

We don't want to have to bundle things every time we want to test our app. Instead, we can point the NW.js binary to our project and run it in terminal

Top left

When we first run our app, it looks a lot like a browser. We will need to tweak the manifest file

Top right

Once we've adjusted our manifest properly, our app starts to look a lot more like it belongs on our system

Developer tutorials

Build desktop apps with NW.js and JavaScript

```
"main" : "index.html"
"window" : {
  "toolbar" : false,
  "frame" : true,
  "width" : 480,
  "height" : 320,
  "show" : true,
  "resizable" : true
}
```

The window object affects how our window will behave once it's been created. We can remove the frame, define the height, make it persistent across multiple workspaces and much, much more, but for now, these properties will set us up.

10. Create our GUI

Now if you repeat Step 7 when our app launches, you will see that we no longer have the URL bar and that our window is the width and height we defined. All that's left of the system chrome are the close, minimise, maximise and title. We can remove these too, but they're handy for UX purposes, so we'll leave them in.

11. Add scripts

It's time to start putting the component parts of our GUI together. Open up index.html and amend it like this:

```
<!DOCTYPE html>
<html>
<head>
<meta content="text/html; charset=utf-8"
http-equiv="Content-Type">
<title>To Do</title>
<link rel="shortcut icon" href="favicon.
ico">
<link rel="stylesheet" href="styles.css"
type="text/css" />
<meta name="viewport" content="initial-
scale=1.0, user-scalable=no" />
</head>
<body>
<ol id="list">
</ol>
<div id="UI">
<div id="delete" class="button">
</div>
<div id="add" class="button">
</div>
</div>
<script src="scripts/core.js"></script>
<script src="scripts/window.js"></script>
</body>
</html>
```

This is the markup for our list. The app logic isn't what we're focusing on, so we won't write core.js or styles.css from scratch, you can download them from FileSilo.

12. Access NW.js globals

We will now write window.js to get the hang of NW.js coding. Create a folder called 'scripts' in your root folder and create a file called 'window.js' in it. Here, we're going to affect the app window with events and listeners to give our app that native feel.

13. NW.js GUI and windows

Just like a Node app, we can use require() to access node modules and NW.js globals. In order to manipulate the properties of the window and listen to events, we need to require the GUI module from NW.js and abstract the window away from it. Insert the following into window.js:

```
var gui = require('nw.gui'),
win = gui.Window.get();
```

14. The menu bar

NW.js is platform agnostic, but there are small OS tweaks we can make for each platform that make the app look more at home. OS X for example has the menu bar with the About and Edit dialogs that are currently missing. Adding the following to window.js will adjust our to-do apps menu bar with more relevant information.

```
var nativeMenuBar = new gui.Menu({ type:
"menubar" });
nativeMenuBar.createMacBuiltin("ToDo", {
// hideEdit: true,
hideWindow: true
});
```

15. Hide our window

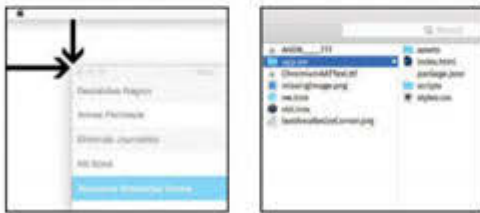
When our app loads, we see a quick flash of white before the content of our page is loaded and initialised. In package.json change the value of 'show' to false instead of true. In window.js add win.show(); to the end of the file. Now when our app loads, the scripts will be initialised and will only show once scripts have finished loading.

16. Position the window

When the app is ready we can position the window. When we open a native app on OS X it's generally at the same x and y coordinates as when you closed it. We can reproduce this, just insert the following before win.show();

Node-webkit

NW.js is, in essence, Node WebKit. The name change accompanied the migration of the project from Node.js to io.js. Rumour is, all the cool kids call it n-dubz for short - enn-double-you-dot-jay-ess is such a mouthful.



Top left

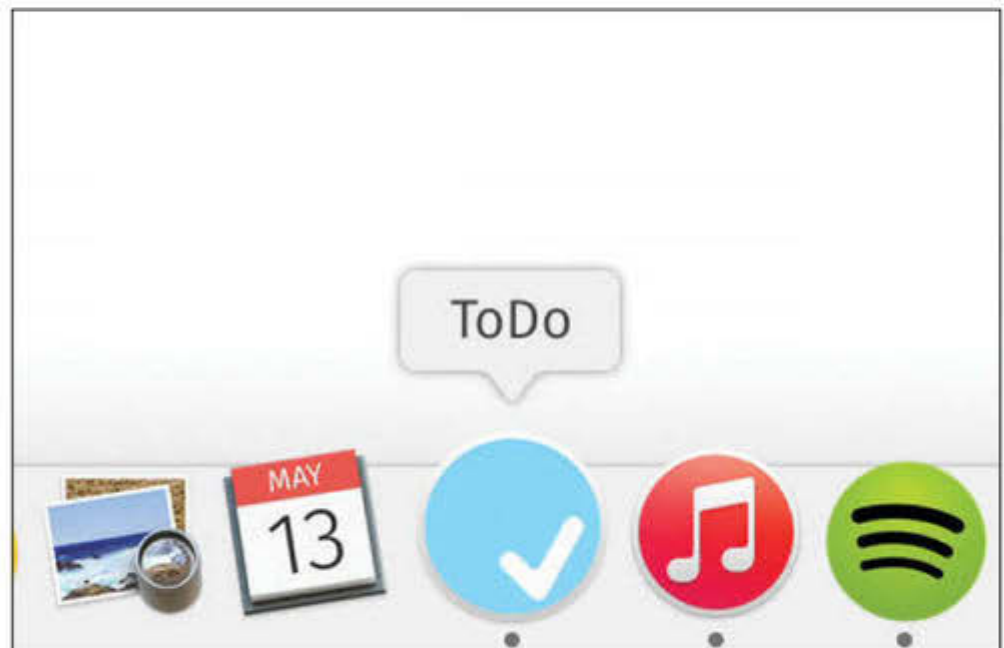
When we position our window with code, the top-left corner is positioned relative to the top and left

Top right

We need to navigate the hierarchy to package our NW.js app properly

Right

Making ICNS files is a bit of a pain, fortunately Image2Icon has us covered, but only if you distribute your app on OS X





Node modules

NW.js is, in part, io.js. That means that you can use modules! Well, at least in principle you can anyway. NW.js can run on any of the major platforms, but just like a native app, your packaging process will of course vary depending on the operating system that you're actually targeting. You can't run an EXE on a Mac, just like you can't mount a DMG on Windows for example. If you want to make use of io.js modules that use a language other than common JavaScript, then it's very likely that you will have to compile your package on the system that you're targeting. That being said though, if you're only using Node modules written in common JavaScript (like the fantastic moment.js module, for example), they should work straight out of the box.

```
var initialPosition = {
  top : 50,
  left : 50
};
var savedPosition = JSON.
  parse(localStorage.
    getItem('windowposition'));
win.on('move', function(){
  localStorage.setItem('windowposition',
    JSON.stringify({left : win.x, top:
  win.y}));
});
if(savedPosition !== null){
  win.moveTo(savedPosition.left,
  savedPosition.top);
} else {
  win.moveTo(initialPosition.left,
  initialPosition.top);
}
```

On our win object, which is the reference to the NW.js window, we've added an event listener. Every time our user moves their window, the 'move' event is dispatched to win. When this happens, our callback is called and we can save the coordinates that the window has been moved to in localStorage. Now, whenever we move our window, its coordinates will be saved in localStorage and the window will be restored to its previous position.

17. Negative numbers

If our user happens to have multiple screens and our app is positioned on one of them, when the screen is

unplugged our app could end up being stuck where we can't see it! If our stored coordinates are less than 0, we're going to override the coordinates and place our window back on the main screen. Just after if(savedPosition !== null){, add the following:

```
if(savedPosition !== null){
  if(savedPosition.left < 0){
    savedPosition.left = 50;
  }
  if(savedPosition.top < 0){
    savedPosition.top = 50;
  }
}
```

18. Resize the window

Just as our window is always in the same place that we left it, it should also be the same size right? Well let's do this now. We're going to repeat what we've done with the move event and localStorage, but this time we're going to listen for the resize event instead. When our window is resized, we'll save the dimensions to localStorage. When we restart our app, we'll set the dimensions of our app to the dimensions stored.

```
var savedPosition = JSON.
  parse(localStorage.
    getItem('windowposition')),
  savedSize = JSON.parse(localStorage.
    getItem('windowsize'));
win.on('move', function(){
  localStorage.setItem('windowposition',
    JSON.stringify( {left : win.x, top: win.y
  } ) ) );
```

```
});
win.on('resize', function(width, height){
  localStorage.setItem('windowsize', JSON.
  stringify( {width : width, height :
  height} ) ) );
});
if(savedPosition !== null){
  if(savedPosition.left < 0){
    savedPosition.left = 50;
  }
  if(savedPosition.top < 0){
    savedPosition.top = 50;
  }
  win.moveTo(savedPosition.left,
  savedPosition.top);
} else {
  win.moveTo(initialPosition.
  left, initialPosition.top);
}
if(savedSize !== null){
  win.resizeTo(savedSize.width, savedSize.
  height);
}
```

19. Click to lock

Right at the start of this tutorial, you should have downloaded core.js from FileSilo and then included it in our scripts folder. Now, we're going to add a little bit of code to it. In our app, you should see an unlocked padlock in the bottom-right corner of our window, it doesn't do anything at all right now, but when we're finished, it will be a toggle for locking our ToDo app on

Developer tutorials

Build desktop apps with NW.js and JavaScript

top of all other windows so that it is visible across all of our workspaces.

20. Create the ICNS file

Open up `core.js` for editing and scroll down until you find the `addEvents` function and just before the end of that function, add the following:

```
document.getElementById('lock').  
addEventListener('click', function(){  
  var isLocked = this.getAttribute('data-  
window-is-locked');  
  if(isLocked === "false"){  
    win.setAlwaysOnTop(true);  
    win.setVisibleOnAllWorkspaces(true);  
    this.setAttribute('data-window-is-  
locked', "true");  } else {  
    win.setAlwaysOnTop(false);  
    win.setVisibleOnAllWorkspaces(false);  
    this.setAttribute('data-window-is-locked',  
"false");  
  }  
}, false);
```

Now, if we restart our app and click the padlock icon, it will lock and if you click another window or move to another workspace, `ToDo` will still be on top for us to see. If we click the padlock again, our window will stay on the workspace it's currently on and will disappear behind any other window that has focus.

21. What did we just do?

Our `package.json` file sets up how our window looks and behaves when we first load it, but as we've seen with the

moving and resizing code we wrote a little earlier, we can override these properties with JavaScript if we want to, even after the window has been shown. Our `win` variable is in the global scope, so it is still possible for us to access it from our app and use DOM events to change its properties - even though it's technically Node.js property. Then when we click on our padlock, we will work on checking the `window-is-locked` data attribute on the DOM element to get the window's current state. If our window isn't locked, then we can now use `'win.setAlwaysOnTop(true)'` and `'win.setVisibleOnAllWorkspaces(true)'` to lock it and vice versa.

22. Package our app

If you run the app now, you'll have a fully functioning, persistent to-do list. You can minimise it, close it and resize it. Now we need to package it. Go to the folder you downloaded earlier that was relocated to GitHub. Copy the `NWJS.app` file into the root of your project folder and right-click it. Select 'Show Package Contents' and navigate to `Contents>Resources`.

23. Copy files

Open another Finder window and copy all of the files we've created (`index.html`, `styles.css`, `scripts` folder, `package.json` and all the others) into the newly created `app.nw` folder. Now if you got back to your project root and double-click `NWJS`, your app will be run by default.

24. Edit PLIST

Our app still looks like a NW.js app. To affect some changes like the name and the icon, we need to change

the app's PLIST file. Right-click `NWJS` again and select 'Show Package Contents' and then navigate back to `Contents`. Open the PLIST file. If you have Xcode, it will open its editor, otherwise you can modify it with any text editor. Change the 'Bundle name' and 'bundle display name' to 'ToDo' and save.

25. Create ICNS

The last little touch to make any app look at home is the icon that it uses. You can either create your own using something like `Image2icon` or you can use the ICNS file included in our files - these are available for download from `FileSilo`. Just simply copy this into `Content>Resources`.

26. Affect the changes

The icon and name changes may not take effect straight away. You can relaunch Finder or you can change the name of the `.app` in your project root to anything and then back to 'ToDo'. Then the icon and bundle name should update.

27. Distribute our app

We now have a fully functioning, JavaScript-powered OS X app. You can now move the `ToDo.app` file anywhere on yours or anyone else's system and fire it up, just like a regular app. When running your app on other people's system, they may need to tweak their security settings to enable your app to run. To do so, simply open `System Preferences>Security and Privacy` and enable apps to be run from the Mac App Store with the toggle. Now, move `ToDo` into your applications folder and give it a whirl.

Node.js and io.js, What's going on?

Node is backed by a company called Joyent - this has been great for many reasons, but a lot of people have felt that the speed at which new JavaScript features and optimisations are being integrated into Node is too slow. These feelings led to `io.js` - a fork of the Node source code, but maintained entirely by the community. `io.js` started to implement new features (like ES6 syntax) straight away, and with the benefits of a rapid-release cycle being well known to the developer community, many projects, like `NW.js`, migrated to `io.js`. Now, Joyent and the guys in charge of `io.js` have agreed to merge all of the changes in `io.js` back into `Node.js` and release and develop Node under the newly created Node Foundation.

The screenshot shows the io.js website homepage. At the top, there are navigation links for 'io.js', 'Roadmap', 'FAQ', 'ES6', and 'API Docs'. The main heading is 'JavaScript I/O' with the tagline 'Bringing ES6 to the Node Community!'. Below this, it states 'io.js is an npm compatible platform originally based on Node.js™'. A prominent box displays 'Version 2.2.1' with download links for 'Linux', 'Win32', 'Win64', 'Mac & Others', and a 'Changelog' link. A news section features a 'Weekly Update - May 15th: io.js decides to join the Node Foundation' with a link to 'News Archive on Medium'. At the bottom, it mentions 'Nightly releases are available for testing.' and a 'Frequently Asked Questions' link.

We don't keep secrets



Tips & Tricks

Learn the truth about iPhone, iPad, Android, Photoshop and more with the Tips & Tricks series' expert advice and tutorials

BUY YOUR COPY TODAY

Print edition available at www.imagineshop.co.uk

Digital edition available at www.greatdigitalmags.com

Available on the following platforms



 [facebook.com/ImagineBookazines](https://www.facebook.com/ImagineBookazines)

 twitter.com/Books_Imagine

To advertise here contact Luke
 luke.biddiscombe@imagine-publishing.co.uk
 +44 (0)1202586431

Get your listing **highlighted!** Contact Luke
 luke.biddiscombe@imagine-publishing.co.uk
 +44 (0)1202 586431

Hosting listings

Got a deal you think we should list?

Whether you're a hosting firm keen to promote your products or a happy customer who wants a favourite provider to be listed, drop us a line with the details!

webdesigner@imagine-publishing.co.uk

Keep an eye on the latest packages and deals with our comprehensive list of service providers

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
111WebHost <small>111webhost.com</small>	Budget Web Hosting Pack	N/A	£10	1GB	1GB	5	✓	✗	Add-on	N/A	✗	✓	✓	✗
	WordPress Web Hosting Pack	N/A	£20	5GB	2GB	100	✓	✓	Add-on	N/A	✗	✓	✓	✗
	Startup Web Hosting Pack	N/A	£25	5GB	Unlimited	100	✓	✓	Add-on	N/A	✗	✓	✓	✗
	Unlimited Web Hosting Pack	N/A	£50	Unlimited	Unlimited	Unlimited	✓	✓	Add-on	N/A	✗	✓	✓	✗
	Unlimited Multi-site Web Hosting Pack	N/A	£300	Unlimited	Unlimited	Unlimited	✓	✓	Add-on	N/A	✗	✓	✓	✗
123-reg (www.123-reg.co.uk)	Starter	0845 859 0018	£29.88	1GB	5GB	20	✗	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Plus	0845 859 0018	£59.88	5GB	50GB	500	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Pro	0845 859 0018	£107.88	10GB	100GB	750	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Bus Pro	0845 859 0018	£179.88	20GB	Unlimited	1,000	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Plus (MS)	0845 859 0018	£59.88	2GB	25GB	100	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Pro (MS)	0845 859 0018	£107.88	5GB	50GB	500	✓	✗	✗	✓	✓	✓	✓	✗
123-reg (www.123-reg.co.uk)	Bus Pro (MS)	0845 859 0018	£179.88	10GB	150GB	1,000	✓	✗	✗	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	Light User	0870 321 2020	£45	20MB	1GB	3	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	Everyday	0870 321 2020	£100	200MB	10GB	15	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	Business/Pro	0870 321 2020	£275	500MB	20GB	50	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	JAVA Tomcat	0870 321 2020	£300	100MB	3GB	15	✓	✓	✓	✓	✓	✓	✓	✗
2020Media (www.2020media.com)	ASP.Net	0870 321 2020	£275	100MB	3GB	15	✓	✓	✓	✓	✓	✓	✓	✗
4D Hosting (www.4dhosting.com)	Big Web Hosting	0207 183 0602	£60	5GB	Unlimited	20	✓	✓	✓	✓	✓	✓	✓	✗
4D Hosting (www.4dhosting.com)	Bigger Web Hosting	0207 183 0602	£120	10GB	Unlimited	75	✓	✓	✓	✓	✓	✓	✓	✗
4D Hosting (www.4dhosting.com)	Biggest Web Hosting	0207 183 0602	£180	20GB	Unlimited	100	✓	✓	✓	✓	✓	✓	✓	✗
4D Hosting (www.4dhosting.com)	Big VPS	0207 183 0602	£204	75GB	500GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
4D Hosting (www.4dhosting.com)	Bigger VPS	0207 183 0602	£360	100MB	1,000GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
4D Hosting (www.4dhosting.com)	Biggest VPS	0207 183 0602	£720	200GB	1,500GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	Home	N/A	£40	500MB	5GB	5	✓	✓	✓	✓	✗	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	Business	N/A	£50	1GB	20GB	100	✓	✓	✓	✓	✗	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	eCommerce	N/A	£100	2GB	40GB	200	✓	✓	✓	✓	✗	✓	✓	✗
Blackfoot Hosting Ltd (www.blackfoot.co.uk)	Professional	N/A	£150	3GB	60GB	300	✓	✓	✓	✓	✗	✓	✓	✗
Blacknight (www.blacknight.com)	Minimus	35359 918 3072	£45	10GB	200GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Blacknight (www.blacknight.com)	Medius	35359 918 3072	£79	20GB	400GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Blacknight (www.blacknight.com)	Maximus	35359 918 3072	£45	30GB	600GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Bravo14 (http://bravo14.co.uk)	Starter Linux	N/A	£130	2,000MB	2,000MB	10	✓	✓	✓	✓	✗	✓	✓	✗
Bravo14 (http://bravo14.co.uk)	Starter Windows	N/A	£20	2,000MB	2,000MB	10	✓	✓	✓	✓	✗	✓	✓	✗
Bravo14 (http://bravo14.co.uk)	Business Linux	N/A	£45	4,000MB	4,000MB	4,000	✓	✓	✓	✓	✗	✓	✓	✗
Bravo14 (http://bravo14.co.uk)	Business Windows	N/A	£45	4,000MB	4,000MB	4,000	✓	✓	✓	✓	✗	✓	✓	✗
Bravo14 (http://bravo14.co.uk)	Ultimate Linux	N/A	£60	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✗	✓	✓	✗
Bravo14 (http://bravo14.co.uk)	Ultimate Windows	N/A	£60	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✗	✓	✓	✗
CityCM (www.citycm.couk)	City250	0330 223 0120	£10.99	250MB	5GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	City500	0330 223 0120	£14.99	500MB	10GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	City2000	0330 223 0120	£29.90	2GB	50GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	City10000	0330 223 0120	£49.90	10GB	100GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	CityR10	0330 223 0120	£120	10GB	50GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	CityR20	0330 223 0120	£180	20GB	100GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	CityR30	0330 223 0120	£240	30GB	150GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	City VPS15	0330 223 0120	£180	15GB SSD	400GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	City VPS30	0330 223 0120	£300	30GB SSD	600GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
CityCM (www.citycm.couk)	City VPS75	0330 223 0120	£420	75GB SSD	600GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
Designwasp (http://designwasp.com)	Starter	0844 372 9848	£30	3GB	20GB	1,000	✓	✓	✗	✓	✓	✓	✓	✗
Designwasp (http://designwasp.com)	Home	0844 372 9848	£40	10GB	50GB	10,000	✓	✓	✗	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	Business	0844 372 9848	£60	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	Windows Unlimited	0844 372 9848	£60	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	Windows Home	0844 372 9848	£40	10GB	50GB	1,000	✓	✓	✗	✓	✗	✓	✓	✗
Designwasp (http://designwasp.com)	CheapHost	0844 372 9848	£10	300MB	100MB	5	✗	✗	✗	✓	✗	✗	✓	✗
Digital Gibbon Ltd (http://digitalgibbon.com)	Personal	01865 589 990	£18	1GB	Unlimited	10	✓	✗	✗	✓	✓	✓	✓	✗
Digital Gibbon Ltd (http://digitalgibbon.com)	Personal Plus	01865 589 990	£48	Unlimited	Unlimited	50	✓	✗	✗	✓	✓	✓	✓	✗
Digital Gibbon Ltd (http://digitalgibbon.com)	Business	01865 589 990	£108	5GB	Unlimited	1000	✓	✓	✗	✓	✓	✓	✓	✗

Featured host of the month: **111WebHost** 111webhost.com



Getting more out of one of our quality listed hosting providers

111WebHost is a provider committed to making website hosting easy to use while offering the best possible value. It provides solutions for small- and medium-sized businesses as well as individuals – and web designers and developers are high on their list. 111WebHost offers a range of top-value packages starting at

the ridiculously low price of £1 per month. For this, users get 1GB of webspace and 1GB of monthly traffic. For those looking for more, there is 5GB of webspace and unlimited traffic for just £2.50 a month. Unlimited webspace and traffic is available for just £5 a month. They also provide specialist hosting for WordPress, Joomla and Drupal.

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
Digital Gibbon Ltd (http://digitalgibbon.com)	Business Professional	01865 589 990	£132	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Donhost (www.donhost.co.uk)	Reseller Unix	0845 226 5566	£399.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Donhost (www.donhost.co.uk)	Reseller Windows	0845 226 5566	£499.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Donhost (www.donhost.co.uk)	Enterprise	0845 226 5566	£89.99	1GB	2GB	250	✓	✓	✓	✓	✓	✓	✓	✓
Donhost (www.donhost.co.uk)	Commerce	0845 226 5566	£179.99	2GB	5GB	500	✓	✓	✓	✓	✓	✓	✓	✓
Donhost (www.donhost.co.uk)	Designer	0845 226 5566	£119.99	1GB	2GB	250	✓	✓	✓	✓	✓	✓	✓	✓
Donhost (www.donhost.co.uk)	Developer	0845 226 5566	£259.99	2GB	5GB	500	✓	✓	✓	✓	✓	✓	✓	✓
eHosting (www.ehosting.com)	Starter	0844 999 4100	£23.88	1GB	25GB	10	✓	✓	✓	✓	✓	✓	✓	✓
eHosting (www.ehosting.com)	Personal	0844 999 4100	£59.88	2.5GB	Unlimited	50	✓	✓	✓	✓	✓	✓	✓	✓
eHosting (www.ehosting.com)	Expert	0844 999 4100	£95.88	5GB	Unlimited	250	✓	✓	✓	✓	✓	✓	✓	✓
eHosting (www.ehosting.com)	Virtual	0844 999 4100	£227.88	50GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Equipphase (www.equipphase.net)	Bronze	0121 314 4865	£30	200MB	2GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Equipphase (www.equipphase.net)	Silver	0121 314 4865	£42	400MB	5GB	20	✓	✓	✓	✓	✓	✓	✓	✓
Equipphase (www.equipphase.net)	Gold	0121 314 4865	£72	800MB	10GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Equipphase (www.equipphase.net)	Platinum	0121 314 4865	£114	1,200MB	40GB	200	✓	✓	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Email Only	02380 249 823	£40	1GB	2GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Essential	02380 249 823	£75	2GB	5GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Superior	02380 249 823	£140	5GB	10GB	25	✓	✓	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Premium	02380 249 823	£250	10GB	25GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Starter	N/A	£29.99	500MB	1GB	3	✓	✓	✓	✓	✓	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Home	N/A	£54.99	2.5GB	30GB	50	✓	✓	✓	✓	✓	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Business	N/A	£79.99	6.5GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Evohosting (www.evohosting.co.uk)	eCommerce	N/A	£159.99	30GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Fasthosts (www.fasthosts.co.uk)	Personal	0808 168 6777	£32.87	5GB	Unlimited	50	✓	✓	✓	✓	✓	✓	✓	✓
Fasthosts (www.fasthosts.co.uk)	Business Bronze	0808 168 6777	£58.38	50GB	Unlimited	500	✓	Option	✓	✓	✓	✓	✓	✓
Fasthosts (www.fasthosts.co.uk)	Business Silver	0808 168 6777	£76.39	Unlimited	Unlimited	Unlimited	✓	Option	✓	✓	✓	✓	✓	✓
Fasthosts (www.fasthosts.co.uk)	Business Gold	0808 168 6777	£101.89	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Fasthosts (www.fasthosts.co.uk)	WD Starter	0808 168 6777	£149.99	20GB	Unlimited	Unlimited	✓	Option	Option	✓	✓	✓	✓	✓
Fasthosts (www.fasthosts.co.uk)	WD Advanced	0808 168 6777	£199.99	Unlimited	Unlimited	Unlimited	✓	Option	Option	✓	✓	✓	✓	✓
Giacom (www.giacom.com)	Business Pro	0800 542 7500	£199	100MB	2GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Starter Professional	0845 644 7750	£29.80	2.5GB	10GB	1,000	✓	✓	✓	✓	✓	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Home Professional	0845 644 7750	£89.99	Unlimited	Unlimited	10,000	✓	✓	✓	✓	✓	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Business Professional	0845 644 7750	£129.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Reseller Professional	0845 644 7750	£349.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Hostway (www.hostway.co.uk)	Silver	0808 180 1880	£79.50	150MB	3GB	5	✓	Option	✓	✓	✓	✓	✓	✓
Hostway (www.hostway.co.uk)	Gold	0808 180 1880	£139.50	300MB	5GB	10	✓	Option	✓	✓	✓	✓	✓	✓
Hostway (www.hostway.co.uk)	Gold Plus	0808 180 1880	£189.50	450MB	10GB	30	✓	Option	✓	✓	✓	✓	✓	✓
Hostway (www.hostway.co.uk)	Platinum	0808 180 1880	£359.50	600MB	20GB	50	✓	Option	✓	✓	✓	✓	✓	✓
Hostway (www.hostway.co.uk)	Platinum Plus	0808 180 1880	£599.50	1.2GB	40GB	10	✓	Option	✓	✓	✓	✓	✓	✓
Hostway (www.hostway.co.uk)	Email Plus	0808 180 1880	£49.95	50MB	N/A	5	N/A	N/A	N/A	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Professional	0845 009 9175	£30	250MB	1GB	50	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Advanced	0845 009 9175	£50	2GB	2.5GB	150	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Enterprise	0845 009 9175	£80	2GB	500MB	Unlimited	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Professional Plus	0845 009 9175	£90	500MB	5GB	100	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Premium Plus	0845 009 9175	£150	1GB	12.5GB	500	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Enterprise Plus	0845 009 9175	£300	2GB	20GB	Unlimited	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Reseller Windows	0845 009 9175	£200	Unlimited	Unlimited	Unlimited	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Reseller Linux	0845 009 9175	£200	Unlimited	Unlimited	Unlimited	✓	✓	option	✓	✓	✓	✓	✓
ICUK www.icukhosting.co.uk	Reseller Enterprise	0845 009 9175	£500	Unlimited	Unlimited	Unlimited	✓	✓	option	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Mail - M105	0800 043 0153	£15.17	256MB	3GB	50	✓	✓	✓	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Home - H25	0800 043 0153	£24.47	1GB	8GB	50	✓	✓	✓	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Professional - P55	0800 043 0153	£45	2GB	15GB	75	✓	✓	✓	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Reseller - R25	0800 043 0153	£97.88	2GB	30GB	250	✓	✓	✓	✓	✓	✓	✓	✓
JAB Web Hosting (www.jabwebhosting.com)	Reseller - R105	0800 043 0153	£271.60	4GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Blog	01438 342 490	£20	100MB	1GB	5	✓	✓	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Starter	01438 342 490	£30	1GB	1GB	10	✓	✓	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Starter	01438 342 490	£30	1GB	1GB	10	✓	✓	✓	✓	✓	✓	✓	✓

To advertise here contact Luke
 luke.biddiscombe@imagine-publishing.co.uk
 +44 (0)1202586431

Get your listing **highlighted!** Contact Luke
 luke.biddiscombe@imagine-publishing.co.uk
 +44 (0)1202 586431


Hosting listings

Keep an eye on the latest packages and deals with our comprehensive list of service providers

Discover what an **enhanced** listing can do for your business. Contact Luke on:

+44 (0)1202 586431

Continued...

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
LCN (www.lcn.com)	Dynamic	01438 342 490	£50	2GB	2GB	20	✓	✓	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Premium	01438 342 490	£80	5GB	5GB	50	✓	✓	✓	✓	✓	✓	✓	✓
LCN (www.lcn.com)	Unlimited	01438 342 490	£120	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
LD Hosts (http://ldhosts.co.uk)	LD Budget (Linux)	07891 235858	£11.88	1GB	1GB	500	✓	✓	✓	✓	✓	✓	✓	✓
LD Hosts (http://ldhosts.co.uk)	LD Home (Linux)	07891 235858	£23.88	10GB	10GB	5,000	✓	✓	✓	✓	✓	✓	✓	✓
LD Hosts (http://ldhosts.co.uk)	LD Pro (Linux)	07891 235858	£41.88	50GB	50GB	20,000	✓	✓	✓	✓	✓	✓	✓	✓
LD Hosts (http://ldhosts.co.uk)	LD Unlimited (Linux)	07891 235858	£60	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
LD Hosts (http://ldhosts.co.uk)	Windows Home	07891 235858	£17.88	2GB	2GB	1,000	✓	✓	✓	✓	✓	✓	✓	✓
LD Hosts (http://ldhosts.co.uk)	Windows Unlimited	07891 235858	£60	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Media Temple (http://mediatemple.net)	Shared-Server Pro	+1 310 841 5500	£64	2GB	1TB	1,000	✓	✓	✓	✓	✓	✓	✓	✓
Media Temple (http://mediatemple.net)	Shared-Server Advanced	+1 310 841 5500	£109	5GB	1.5TB	5,000	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Email Only	01604 212 904	£11.99	2.5GB	15GB	10	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Starter	01604 212 904	£36.99	10GB	150GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Home Pro	01604 212 904	£59.99	25GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Business	01604 212 904	£109.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	StartUp Plus	0845 363 3632	£89.99	500MB	5GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Business	0845 363 3632	£149.99	2,000MB	20GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Business Plus	0845 363 3632	£89.99	3,000MB	30GB	200	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Designer	0845 363 3632	£239.88	1GB	20GB	200	✓	✓	✓	✓	✓	✓	✓	✓
Namesco (www.names.co.uk)	Designer Plus	0845 363 3632	£479.88	Unlimited	30GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	Shared 100	0207 1000 424	£60	100MB	1GB	5	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	VS100	0207 1000 424	£600	5GB	5GB	5	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	VS200	0207 1000 424	£1,000	10GB	10GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	VS300	0207 1000 424	£2,000	15GB	50GB	20	✓	✓	✓	✓	✓	✓	✓	✓
Netplan (www.netplan.co.uk)	Dedicated Servers	0207 1000 424	£3,000+	73GB+	1,500GB	100+	✓	✓	✓	✓	✓	✓	✓	✓
 Netcetera www.netcetera.co.uk	DEVELOPER	0800 808 5450	£32.89	1GB	Unlimited	500	✓	✓	✓	✓	✓	✓	✓	✓
	ONE	0800 808 5450	£109.99	5GB	Unlimited	1000	✓	✓	✓	✓	✓	✓	✓	✓
	RESELLER	0800 808 5450	£274.89	Unlimited	Unlimited	1000	✓	✓	✓	✓	✓	✓	✓	✓
	VM500 Server	0800 808 5450	£300	20GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
	2200DC Server	0800 808 5450	£720	160GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
3000DC Server	0800 808 5450	£1,200	2x500GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓	
2600QC Server	0800 808 5450	£1,800	2x500GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓	
PurplePaw (www.purplepaw.co.uk)	Email	N/A	£25	100MB	1GB	10	✓	✓	✓	✓	✓	✓	✓	✓
PurplePaw (www.purplepaw.co.uk)	Play	N/A	£40	100MB	1GB	10	✓	✓	✓	✓	✓	✓	✓	✓
PurplePaw (www.purplepaw.co.uk)	Plus	N/A	£65	750MB	5GB	25	✓	✓	✓	✓	✓	✓	✓	✓
PurplePaw (www.purplepaw.co.uk)	Power	N/A	£95	2GB	10GB	100	✓	✓	✓	✓	✓	✓	✓	✓
PurplePaw (www.purplepaw.co.uk)	R3 Reseller	N/A	£660	10GB	50GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Starter	0843 289 4625	£18	1GB	Unlimited	10	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Business	0843 289 4625	£59.88	10GB	Unlimited	100	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Premium	0843 289 4625	£107.88	100GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Starter	0843 289 4625	£18	1GB	Unlimited	10	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Business	0843 289 4625	£59.88	10GB	Unlimited	100	✓	✓	✓	✓	✓	✓	✓	✓
Reddex UK (www.reddexuk.com)	Reddex Design Premium	0843 289 4625	£107.88	100GB	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
STRATO Hosting (www.strato-hosting.co.uk)	STRATO BasicWeb	00800 8007 0070	£48	2GB	Unlimited	1,000	✓	✓	✓	✓	✓	✓	✓	✓
STRATO Hosting (www.strato-hosting.co.uk)	STRATO PowerWeb	00800 8007 0070	£72	6GB	Unlimited	2,000	✓	✓	✓	✓	✓	✓	✓	✓
STRATO Hosting (www.strato-hosting.co.uk)	STRATO AdvancedWeb	00800 8007 0070	£120	20GB	Unlimited	4,000	✓	✓	✓	✓	✓	✓	✓	✓
STRATO Hosting (www.strato-hosting.co.uk)	STRATO EnterpriseWeb	00800 8007 0070	£180	50GB	Unlimited	6,000	✓	✓	✓	✓	✓	✓	✓	✓
Streamline.net (www.streamline.net)	Trial (3 month)	0844 941 1000	N/A	10GB	Unlimited	1,000	✓	Option	✓	✓	✓	✓	✓	✓
Streamline.net (www.streamline.net)	Starter	0844 941 1000	£23.88	500MB	Unlimited	20	✓	Option	✓	✓	✓	✓	✓	✓
Streamline.net (www.streamline.net)	Personal	0844 941 1000	£41.88	3GB	Unlimited	500	✓	Option	✓	✓	✓	✓	✓	✓
Streamline.net (www.streamline.net)	Plus	0844 941 1000	£71.88	10GB	Unlimited	1,000	✓	Option	✓	✓	✓	✓	✓	✓
Streamline.net (www.streamline.net)	Multisite	0844 941 1000	£137.88	20GB	Unlimited	Unlimited	✓	Option	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	Email	08445 67 69 71	£18	-	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	Windows Hosting	08445 67 69 71	£66	-	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓

Can you recommend your host?

Tweet us today with your hosting comments and advice @WebDesignerMag

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER YEAR	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
Swish Hosting (www.swishhosting.co.uk)	Linux Hosting	08445 67 69 71	£66	-	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	eCommerce	08445 67 69 71	£90	-	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Swish Hosting (www.swishhosting.co.uk)	SiteBuilder	08445 67 69 71	£12	-	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Switch Media (www.switchmedia.com)	Switch Standard	0151 236 9111	£159	1GB	50GB	500	✓	✓	✓	✓	✓	✓	✓	✓
Switch Media (www.switchmedia.com)	Switch Business	0151 236 9111	£249	10GB	150GB	1500	✓	✓	✓	✓	✓	✓	✓	✓
Switch Media (www.switchmedia.com)	Business Pro	0151 236 9111	£348.96	20GB	300GB	5000	✓	✓	✓	✓	✓	✓	✓	✓
Tidy Web Hosting (www.tidywebhosting.co.uk)	Entry	0844 884 9100	£25	100MB	1GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Tidy Web Hosting (www.tidywebhosting.co.uk)	Home	0844 884 9100	£50	500MB	5GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Tidy Web Hosting (www.tidywebhosting.co.uk)	HomePro	0844 884 9100	£100	1GB	10GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Tidy Web Hosting (www.tidywebhosting.co.uk)	Business	0844 884 9100	£150	2GB	20GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Tidy Web Hosting (www.tidywebhosting.co.uk)	BusinessPro	0844 884 9100	£250	5GB	50GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Basic5S	0845 641 0776	£24	100MB	1,500MB	25	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Standard5S	0845 641 0776	£45	200MB	3,000MB	50	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Business5S	0845 641 0776	£70	500MB	7,500MB	100	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Advanced5S	0845 641 0776	£110	1,000MB	15,000MB	200	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Basic5S	0845 641 0776	£24	100MB	1,500MB	25	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Standard5S	0845 641 0776	£45	200MB	3,000MB	50	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Business5S	0845 641 0776	£70	500MB	7,500MB	100	✓	✓	✓	✓	✓	✓	✓	✓
TwentyHost (www.twentyhost.co.uk)	Advanced5S	0845 641 0776	£110	1,000MB	15,000MB	200	✓	✓	✓	✓	✓	✓	✓	✓
UK2.NET (www.uk2.net)	Starter Hosting	0808 168 2427	£24	500MB	1GB	✓	✓	✓	✓	✓	✓	✓	✓	✓
UK2.NET (www.uk2.net)	Business Hosting	0808 168 2427	£54	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
UK2.NET (www.uk2.net)	Business Cloud	0808 168 2427	£108	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
UK2.NET (www.uk2.net)	Reseller Starter	0808 168 2427	£180	10GB	100GB	10	✓	✓	✓	✓	✓	✓	✓	✓
UK2.NET (www.uk2.net)	Reseller Plus	0808 168 2427	£360	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
UK2.NET (www.uk2.net)	Dedicated Server	0808 168 2427	£704	2x500GB	10TB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
VARIHOST (www.varihost.net)	WordPress Basic	0208 144 7057	£47.88	2GB	10GB	10	✓	✓	✓	✓	✓	✓	✓	✓
VARIHOST (www.varihost.net)	WordPress Plus	0208 144 7057	£71.88	4GB	100GB	100	✓	✓	✓	✓	✓	✓	✓	✓
VARIHOST (www.varihost.net)	WordPress Extra	0208 144 7057	£119.88	unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
VARIHOST (www.varihost.net)	WordPress Basic	0208 144 7057	£47.88	2GB	10GB	10	✓	✓	✓	✓	✓	✓	✓	✓
VARIHOST (www.varihost.net)	WordPress Plus	0208 144 7057	£71.88	4GB	100GB	100	✓	✓	✓	✓	✓	✓	✓	✓
VARIHOST (www.varihost.net)	WordPress Extra	0208 144 7057	£119.88	unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
WebFusion (www.webfusion.co.uk)	Fusion Professional	0845 130 1602	£107.40	5GB	50GB	1,000	✓	✓	✓	✓	✓	✓	✓	✓
WebFusion (www.webfusion.co.uk)	Fusion Business	0845 130 1602	£179.40	10GB	150GB	1,500	✓	✓	✓	✓	✓	✓	✓	✓
WebFusion (www.webfusion.co.uk)	Fusion Developer	0845 130 1602	£227.40	20GB	300GB	5,000	✓	✓	✓	✓	✓	✓	✓	✓
WebFusion (www.webfusion.co.uk)	Fusion Reseller	0845 130 1602	£329.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Z-Host (z-host.co.uk)	Z-100	N/A	£15	100MB	10GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Z-Host (z-host.co.uk)	Z-200	N/A	£21	100MB	10GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Z-Host (z-host.co.uk)	Z-500	N/A	£42	500MB	20GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Z-Host (z-host.co.uk)	Z-1000	N/A	£60	1GB	40GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✓
Zen Internet (www.zen.co.uk)	Bronze (Linux)	0845 058 9000	£47.88	2GB	10GB	10	✓	✓	✓	✓	✓	✓	✓	✓
Zen Internet (www.zen.co.uk)	Silver (Linux)	0845 058 9000	£95.88	5GB	50GB	25	✓	✓	✓	✓	✓	✓	✓	✓
Zen Internet (www.zen.co.uk)	Gold (Linux)	0845 058 9000	£143.88	10GB	100GB	50	✓	✓	✓	✓	✓	✓	✓	✓
Zen Internet (www.zen.co.uk)	Platinum (Linux)	0845 058 9000	£239.88	50GB	250GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Zen Internet (www.zen.co.uk)	Reseller (Linux)	0845 058 9000	£479.88	150GB	500	250	✓	✓	✓	✓	✓	✓	✓	✓
Zen Internet (www.zen.co.uk)	Designer (Windows)	0845 058 9000	£59.88	2GB	20GB	-	✓	✓	✓	✓	✓	✓	✓	✓
Zen Internet (www.zen.co.uk)	Developer (Windows)	0845 058 9000	£179.88	10GB	100GB	-	✓	✓	✓	✓	✓	✓	✓	✓

Golden rules to top hosting We identify and explain the key criteria for success...

1 The best resources for you

Selecting your ideal package is largely determined by the kinds of resources and quantity of features you require from your hosting solution. Key aspects such as web space and monthly bandwidth are important for those services likely to be subjected to heavy amounts of traffic, which is why package solutions for enterprise applications are typically much more expensive. The general rule of thumb is not to buy more than you need or underestimate potential requirements.

2 Competitive and reliable

The hosting market is big business and hosting providers do try to drive prices down in a bid to entice your custom. Use our chart to compare costs, but be sure to visit the vendor websites to keep track of the latest deals, as they do change. Remember that low price should not always be a deciding factor and that paying a premium for a more reliable, trusted and experienced vendor can offer you much better value for money in the long term.

3 Putting you in control

Modern hosting is all about giving customers the power to set up, monitor and maintain their web space with minimal fuss. Most commercial vendors offer access to award-winning and intuitive control panels that enable you to log in remotely and intuitively tweak your account - without the need to relay complicated instructions down the phone. Be sure to find out from your potential host as much as you can about the control panel and request a demo.

4 Fantastic customer support

If all else fails and you need some extra help to get your hosting back online, then a commitment to future customer support is key. Many vendors offer a service-level agreement which outlines what you can expect here, however most will be more explicit about whether phone support is included or email contact is preferred. Think about what you need for your own peace of mind and factor good, comprehensive technical support against the price.

IMAGE IS EVERYTHING

www.advancedphotoshop.co.uk



ADVANCED PHOTOSHOP®

Available from all good newsagents and supermarkets

ON SALE NOW

> Photoshop & 3D > Movie posters > 25 quick fixes > Make watercolour brushes



BUY YOUR ISSUE TODAY

Print edition available at www.imagineshop.co.uk

Digital edition available at www.greatdigitalmags.com

Available on the following platforms



 facebook.com/AdvancedPhotoshop

 twitter.com/advancedpshop



trade secret

noun {c} ~tred si kriht - A secret formula, method, or device that gives one an advantage over competitors.

Low cost address lookup for your website

- Buy online from £15 + vat
- Eliminate spelling mistakes
- Reduced abandoned carts
- 80% reduction of keystrokes entering addresses
- Create a professional image for your clients
- Simple integration with SDK & working examples

For more details visit postcode-software.net or call **0845 83 82 666**



«klikpic»

Websites for photographers

Try the **new** Klikpic for only £40 pa includes ecommerce



- Stunning range of new templates.
- New admin system makes it even easier to use.
- Stylish slideshows and carousels throughout.
- New blog styles with extra options.
- Easy upload and link to your social media.
- And much more besides!

Compatible with mobile phones and tablets

Visit www.klikpic.com for a **FREE 14 day trial**



I spy with my little eye domains ending in...

.black .buzz .care .club

.guru .ink .host .london

.ninja .party .press .rocks

.science .social .website .wiki


Think **Different**

Think **Context**

Think **Available**



www.blacknight.host sales@blacknight.host @blacknight



Domains

.COM just £5.80/pa



Hosting

From £2.99/pm + FREE Domain



Cloud

SSD Server from £5/pm




Servers

Save 50% for 3 months



Datacentre

Rackspace from £35/pm



0800 808 5450
netcetera.co.uk/wdm

YOUR FREE RESOURCES

LOG IN TO WWW.FILESILO.CO.UK/WEBDESIGNER AND DOWNLOAD THE LATEST COLLECTION OF MUST-HAVE VIDEOS AND RESOURCES



**PLUS
120 MINS OF
EXPERT
CSS3 VIDEO
TUITION**

30 HI-RES BLURRED BACKGROUNDS
CREATE ATTENTION-GRABBING PAGE DESIGNS



EXPERT CSS3 VIDEOS
MASTERING MEDIA QUERIES

**TOTAL
RESOURCES 54**



TUTORIAL FILES
THIS ISSUE'S ASSETS

YOUR BONUS RESOURCES



ON OUR BRAND NEW FILESILO THIS ISSUE YOU WILL FIND AN UNMISSABLE COLLECTION OF VIDEO GUIDES AND RESOURCES

- 120 Minutes of expert CSS3 video guides from CartoonSmart
- 30 Blurred backgrounds from SparkleStock (worth \$8)
- 3 Web templates: Thomsoon UI template, Alor responsive template and Auction template from W3 Layouts
- The Nyxali and Octynaz font from Typodermic fonts
- 19 Instant hipster Photoshop actions from SparkleStock



www.filesilo.co.uk/webdesigner

FILESILO – THE HOME OF PRO RESOURCES

DISCOVER YOUR FREE ONLINE ASSETS

- 🔒 A rapidly growing library
- 🔒 Updated continually with cool resources
- 🔒 Lets you keep your downloads organised
- 🔒 Browse and access your content from anywhere
- 🔒 No more torn disc pages to ruin your magazines
- 🔒 No more broken discs
- 🔒 Print subscribers get all the content
- 🔒 Digital magazine owners get all the content too!
- 🔒 Each issue's content is free with your magazine
- 🔒 Secure online access to your free resources



This is the new FileSilo site that replaces your disc. You'll find it by visiting the link on the following page.

The first time you use FileSilo you'll need to register. After that, you can use the email address and password you provided to log in.

The most popular downloads are shown in the carousel here, so check out what your fellow readers are enjoying!

If you're looking for a particular type of content like brushes or fonts, use the filters here to refine your search.

Can't find the resource you're looking for in these filters? Click on More Options to specify exactly what kind of resource you want.

Green open padlocks show the issues you have accessed. Red closed padlocks show the ones you need to buy or unlock.

Top Downloads are listed here, so you can get an instant look at the most popular downloaded content.

Check out the Highest Rated list to see the resources that other readers have voted for as the best!

Find out more about our online stores, and useful FAQs like our cookie and privacy policies and contact details.

Discover our amazing sister magazines and the wealth of content and information that they provide.

HOW TO USE **FileSilo**

EVERYTHING YOU NEED TO KNOW ABOUT ACCESSING YOUR NEW DIGITAL REPOSITORY

To get access to FileSilo, please visit www.filesilo.co.uk/webdesigner

01 Follow the instructions on-screen to create an account with our secure FileSilo system, or log in and unlock the issue by answering a simple

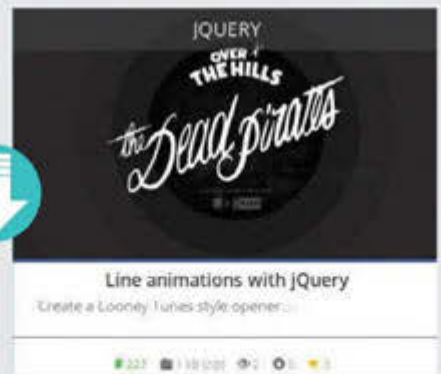
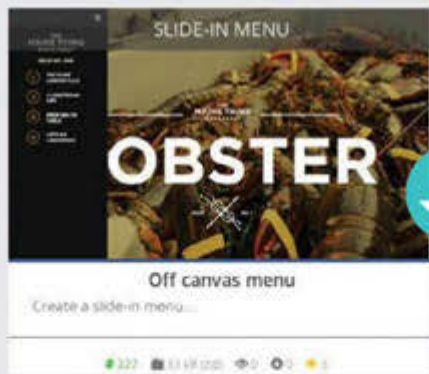


question about the magazine. You can access the content for free with your issue.

02 If you're a print subscriber, you can easily unlock all the content by entering your unique Subscriber ID. You can find this on all correspondence from Imagine Publishing, including the clear plastic envelopes your magazine gets delivered to your door in every month.

03 You can access FileSilo on any desktop, tablet or smartphone device using any popular browser (such as Safari, Firefox or Google Chrome). However, we recommend that you use a desktop to download content, as you may not be able to download files to your phone or tablet.

04 If you have any problems with accessing content on FileSilo, or with the registration process, take a look at the FAQs online or email filesilohelp@imagine-publishing.co.uk.



MORE TUTORIALS AND INSPIRATION?

Web Designer is all about giving web designers and developers what they want. Make sure that you pay a visit to the Web Designer website where you will find a treasure trove of design and development tutorials. All the latest technologies and tools are covered including HTML5, CSS3, jQuery, PHP and responsive design. Plus, delve into the lives of the world's biggest agencies with our exclusive interviews. Find out how they design and develop some of the best websites seen on the web.



www.webdesignermag.co.uk

Issue 238 of **web designer** is on sale **23 July 2015** from GreatDigitalMags.com

UNLOCK THE POWER OF ANDROID

www.littlegreenrobot.co.uk



Android magazine

Available from all good newsagents and supermarkets

ON SALE NOW

> All-new Android Wear > Galaxy S6 secrets > Upgrade Google apps



BUY YOUR ISSUE TODAY

Print edition available at www.imagineshop.co.uk

Digital edition available at www.greatdigitalmags.com

Available on the following platforms





AMAZING WEB ANIMATION

DISCOVER THE BEST TOOLS AND TECHNOLOGIES NEEDED TO
CREATE CLASSY, CONTEMPORARY ONLINE ANIMATION



CODE ANGULAR NGMESSAGES

How to use the ngMessages directive to create context-sensitive messages



INTERACTIVE ANIMATION

Take advantage of the Snap.svg library to add page engagement



RESPONSIVE LAYOUTS

Combine HTML and CSS to create aspect ratio-based site designs



BUILD WITH BETTER CODE

The best testing tools to ensure a faster, more efficient web experience

Visit the **WEB DESIGNER** online shop at

imagineshop.co.uk

for back issues, books and merchandise

ALL IN YOUR NEXT

WEB DESIGNER

Issue 238 on sale

Thursday 23rd July 2015

Discover emerging design

Part 1 24–27 June 2015

Part 2 01–04 July 2015

London N1

Book Tickets

newdesigners.com

#ND30yrs





Protect your business, knock out the competition...



We believe in better business online

in association with



| Telephone: 020 3004 9644 | Email: info@iwdro.org | www.iwdro.org |