

29
PAGES OF
EXPERT TUTORIALS

Web designer

FREE 70 MINUTES
OF JS VIDEOS

HTML5

CSS3

jQuery

JavaScript

@WebDesignerMag

/WebDesignerUK



20 TODAY

HOT NEW TOOLS

Must-know libraries and frameworks

REACTJS

Ten pages on testing and form controls

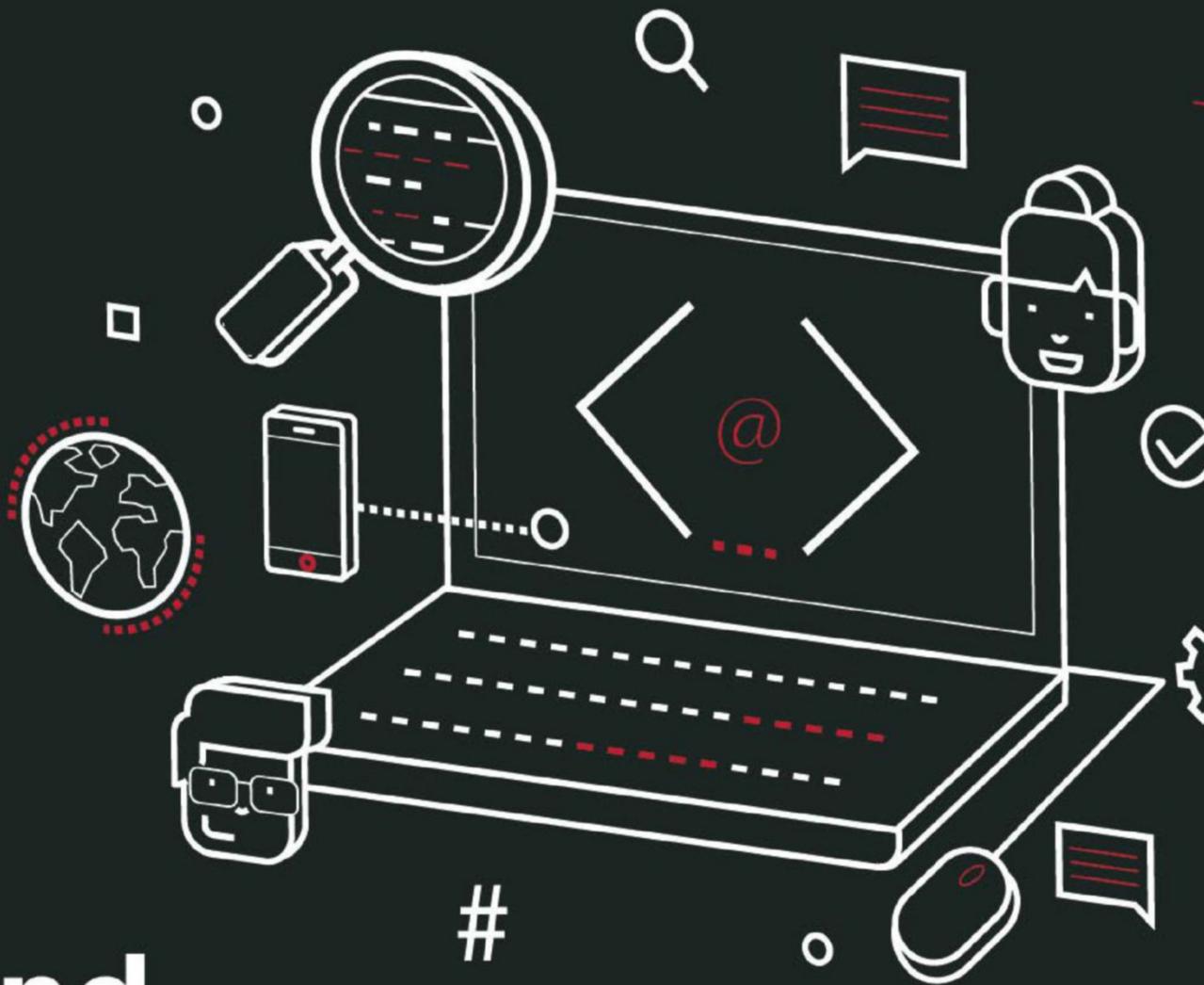
The path to search success in 2018

- How to futureproof your site
- Get to grips with technical SEO
- Optimise for mobile and voice
- 10 essential tools and resources

+ Faster builds with better boilerplates • Get started with Rust • Create custom reusable Sketch components • Build a WebGL racing game • Animated menus



ISSUE 273



<Land Your First Software Developer Job />

Far more than just
a **Coding Bootcamp**.

We'll secure you a career.
www.northcoders.com/web-designer

**NORTH
CODERS**

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



Steven Jenkins
Editor

Follow the path to search success



The ever-changing nature of the Google algorithm is just one reason that you need to keep up-to-date with the latest SEO techniques. So, what else are search engines looking for in 2018? And, what can you do to stay ahead of the game?

In our lead feature (p42) you will discover an essential collection of techniques, tips and tools that will reveal the actions you need to take to help futureproof your site without being penalised. Find out how to get to grips with technical SEO, optimise for your audience and search engine crawlers, enhance and improve the mobile experience, the power

of primary and secondary keywords, the importance of business listings, why content is still king, and an essential collection of tools to help you achieve your goals.

It's not only SEO that is changing, as we all know the web is an ever-evolving animal. Find out how stay up-to-date with our collection of the best fresh tools and technologies, which you might not have heard of, yet. Elsewhere we delve into the world of Rust, a systems programming language that runs blazingly fast and is backed by Mozilla. Find out what it does and how to get started with it. Plus, get yourself 18 pages on ReactJS, Sketch and WebGL. Enjoy the issue.

If there's a problem which has frustrated you, or something you've thought could be made easier, someone will have come up with a solution for it

Highlight



I think that our team could become like a good bottle of wine; the older it gets, the better it is.

Bonhomme focus on engagement, but also deliver minimal beauty across the digital landscape. **Web Designer** finds out more. Page 34

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

Visit our blog for opinion, freebies & more www.creativebloq.com



FREE - exclusive with this issue
34 Designer resources

Video Tuition - 70 minutes of expert JavaScript video guides from Pluralsight (pluralsight.com)

Assets - 13 Greeting card mockups and 10 Wooden picture frame mockups from Sparklestock (sparklestock.com)
- Tutorial files and assets



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...



Paul Betteridge

Paul has 15 years' experience leading top digital organisations and brands across the UK. This issue he reveals 25 tips and a host of tools to help you gain a better understanding of what search engines are looking for in 2018. And, how you can ultimately improve your rankings. **Page 42**

SEO is evolving quickly. With a history of changes, algorithm updates, new technologies and personalisation. You need to adapt to stay ahead ☺

Simon Jones



Simon is always looking for ways to make designing and developing for the web easier. In this issue, he looks at some of the latest and lesser-known tools, libraries and frameworks which you might want to try out.

Page 66

Mark Shufflebottom



Mark is a Professor of Interaction Design at Sheridan College and is also an Adobe Education Leader. In this issue, Mark is finishing off the second and final part of creating an endless flying game in WebGL with ThreeJS. **Page 52**

Tam Hanna



Tam started out wiggling command lines on process computers and is a veteran of many programming languages. This issue he delves into the world of Rust, a systems programming language that runs blazingly fast. **Page 74**

Leon Brown



Leon is a freelance web developer and trainer who helps people to embrace and progress web dev careers. This issue he ventures into the world of ReactJS and shows how to create dynamic form controls. **Page 86**

Steven Roberts



Steven is a creative developer for Asemblr.co. He's also the lead author and developer on the BetterBoilerplate frontend framework. This issue he'll be showing you how to get up and running with BetterBoilerplate. **Page 56**

Luke Harrison



Luke is a web developer from Sheffield, UK who is all about scalable and efficient front-end architecture. In this issue, he explores how to create a highly reusable input symbol in Sketch, leveraging the power of overrides. **Page 60**

Matt Crouch



Matt is a front end developer in London. Every project should have great test coverage, but where do you start? In this issue, Matt introduces Enzyme and the Jest framework to help test for bugs in your components. **Page 80**

David Howell



David is a journalist with over 20 years' experience in publishing and runs his own business, Nexus Publishing. This issue he takes time out to talk to the very talented crew at Bonhomme and discover how they work. **Page 34**

Follow us!

Facebook: [www.facebook.com/
WebDesignerUK](http://www.facebook.com/WebDesignerUK)
Twitter: [https://twitter.com/
webdesignermag](https://twitter.com/webdesignermag)

Future PLC Richmond House, 33 Richmond Hill, Bournemouth, Dorset, BH2 6EZ

Editorial

Editor **Steven Jenkins**
steve.jenkins@futurenet.com
01202 586233

Designer **Harriet Knight**
Editor in Chief **Amy Hennessey**
Senior Art Editor **Will Shum**

Contributors

Brandon Geary, Paul Betteridge, Simon Jones, Tam Hanna, Mark Shufflebottom, Steven Roberts, Luke Harrison, Matt Crouch, Leon Brown, Mark Billen, David Howell, Philip Morris, Rob Mead-Green, Will Shum, Ryan Wells, Laurie Newman

Photography

James Sheppard
All copyrights and trademarks are recognised and respected

Advertising

Media packs are available on request
Commercial Director **Clare Dove**
clare.dove@futurenet.com
Senior Advertising Manager **Mike Pyatt**
michael.pyatt@futurenet.com
01225 687538
Account Director **George Lucas**
george.lucas@futurenet.com
Account Manager **Chris Mitchell**
chris.mitchell@futurenet.com

International

Web Designer is available for licensing.
Contact the International department to discuss partnership opportunities
International Licensing Director **Matt Ellis**
matt.ellis@futurenet.com

Subscriptions

Email enquiries contact@myfavouritemagazines.co.uk
UK orderline & enquiries 0344 848 2852
Overseas order line and enquiries +44 (0) 344 848 2852
Online orders & enquiries www.myfavouritemagazines.co.uk
Head of subscriptions **Sharon Todd**

Circulation

Head of Newtrade **Tim Mathers**

Production

Head of Production **Mark Constance**
Production Project Manager **Clare Scott**
Advertising Production Manager **Joanne Crosby**
Digital Editions Controller **Jason Hudson**
Production Manager **Nola Cokely**

Management

Managing Director **Aaron Asadi**
Commercial Finance Director **Dan Jotcham**
Editorial Director **Paul Newman**
Head of Art & Design **Greg Whittaker**

Printed by William Gibbons, 28 Planetary Road, Willenhall, WV13, 3XT

Distributed by Marketforce, 5 Churchill Place, Canary Wharf, London, E14 5HU www.marketforce.co.uk Tel: 0203 787 9060

ISSN 1745-3534

We are committed to only using magazine paper which is derived from responsibly managed, certified forestry and chlorine-free manufacture. The paper in this magazine was sourced and produced from sustainable managed forests, conforming to strict environmental and socioeconomic standards. The manufacturing mill holds full FSC (Forest Stewardship Council) certification and accreditation.

All contents © 2018 Future Publishing Limited or published under licence. All rights reserved. No part of this magazine may be used, stored, transmitted or reproduced in any way without the prior written permission of the publisher. Future Publishing Limited (company number 2008865) is registered in England and Wales. Registered office: Quay House, The Ambury, Bath BA1 1UA. All information contained in this publication is for information only and is, as far as we are aware, correct at the time of going to press. Future cannot accept any responsibility for errors or inaccuracies in such information. You are advised to contact manufacturers and retailers directly with regard to the price of products/services referred to in this publication. Apps and websites mentioned in this publication are not under our control. We are not responsible for their contents or any other changes or updates to them. This magazine is fully independent and not affiliated in any way with the companies mentioned herein.

If you submit material to us, you warrant that you own the material and/or have the necessary rights/permissions to supply the material and you automatically grant Future and its licensees a licence to publish your submission in whole or in part in any/all issues and/or editions of publications, in any format published worldwide and on associated websites, social media channels and associated products. Any material you submit is sent at your own risk and, although every care is taken, neither Future nor its employees, agents, subcontractors or licensees shall be liable for loss or damage. We assume all unsolicited material is for publication unless otherwise stated, and reserve the right to edit, amend, adapt all submissions.



Future plc is a public company quoted on the London Stock Exchange (symbol: FUTR)
www.futureplc.com

Chief executive **Zillah Byng-Thorne**
Chairman **Richard Huntingford**
Chief financial officer **Penny Ladkin-Brand**
Tel +44 (0)1225 442 244

Introducing **STORM**, a hosting platform that makes your agency and websites run smoother. 80% of our customers have seen a 25% improvement in site speeds.

SUPER CHARGE YOUR HOSTING

STORM

Developed by
 **Nimbus Hosting**

nimbushosting.co.uk

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...

steve.jenkins@futurenet.com • [@WebDesignerMag](https://www.WebDesignerMag) • www.creativebloq.com

08 Digital trends 2018

Are you ready for CX? Are you integrating AI?
Find out what's happening in digital

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 Be your own boss

Brandon Geary, Global Chief Strategy Officer at POSSIBLE reveals what Gen Zers want from their work life

14 Lightbox

A showcase of inspirational sites and the techniques used to create them

26 A sense of identity

Selling the services of a food quality control company is not easy. We find out how Madrid's Twenty Two Degrees managed

34 Design buddies

Bonhomme delivers minimal beauty across the digital landscape. **Web Designer** finds out more

42 SEO Today

Essential tips and tools to send you on the path to search success in 2018

66 20 Hot new tools

A must-have collection of libraries and frameworks to power up your projects

74 Get started with Rust

Rust is an open-source systems programming language brought to you by Mozilla. Discover what it has to offer today's developer

92 Hosting listings

An extensive list of web hosting companies. Pick the perfect host for your needs

94 Course listings

Want to start learning online? Check out what courses are out there with this list

98 Next month

What's in the next issue of **Web Designer**?

Cover focus



42



66
20 Hot new tools
Must-know libraries and frameworks



74
Get started with Rust
Say hello to the open-source language

FileSilo

96 Get the latest must-have resources and videos

- 70 minutes of expert JavaScript video guides from Pluralsight
- 13 Greeting card mockups
- 10 Wooden picture frame mockups



Never miss
an issue
Subscribe
Turn to page 32 now

Save on the cover price and get
the mag delivered to your door



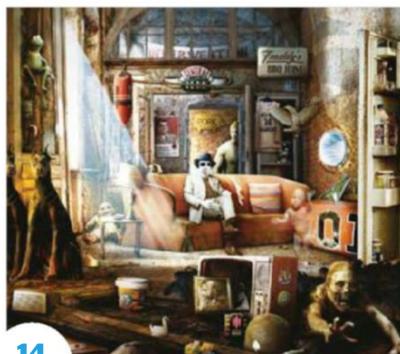
 Kids these days
aren't like they used to
be. If they have an idea,
be ready to listen

COMMENT — Brandon Geary — p11



34

ProFile: Bonhomme
Delivering across the digital landscape



14

Lightbox: PopCorn TV
Good-looking digital TV show quiz

Visit the **WEB DESIGNER** online shop at
myfavourite magazines
myfavouritemagazines.co.uk
for the latest issue, back issues and specials

Tutorials

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



- 52 **Create a 3D WebGL game**
The final part of this two-parter sees the finishing touches added to this endless flying 3D game
- 56 **Faster builds with BetterBoilerplate**
Learn how to build a simple and beautiful page using the new frontend framework BetterBoilerplate
- 60 **Reusable symbols in Sketch**
Discover how to create custom, flexible components that can be used again and again

Web Workshop

- 50 **Create an animated 3D cube**
twotwentytwo.se
Introduce depth and excitement with ThreeJS
- 64 **Staged menu opening animation**
deltstl.com
Add engaging animation to menus

Web Developer

- 74 **Get started with Rust**
Rust is an open-source systems programming language. See what it has to offer today's developer
- 80 **Test React with Jest and Enzyme**
Write powerful unit tests that can notify you of issues with components as you work
- 86 **Code dynamic form controls**
Utilise the power of React to quickly and easily set up an interactive form that controls data and visuals

Header

The tools, trends and news to inspire your web projects

What's big in design and dev in 2018?

Web Designer takes a closer look at Adobe's Digital Trends report and StackOverflow's Developer survey results

Adobe, you may have heard of it, has been fuelling the worlds' creatives for as long as digital and the web has been a thing. Read any of our ProFile interviews (page 34) and the stalwarts from Abode are still very much the number one choice for the many. So when Adobe has something to say it is often worth a listen.

Trends are very much part of the industry and can help give an insight into what's coming very soon, in the future or what's here now, what you need to be taking note of. Here we are going to take a look at Adobe's latest digital trends report and see what we can glean about today and tomorrow. This will only be a snapshot of what the report has to offer. You can read the complete report at adobe.ly/2F9iqI0.

So what are some of the key insights that the report reveals? Companies who are continuing to focus on the customer experience (CX), as well as the content required to facilitate this, are outperforming

their peers. According to the report, we are entering a "design and creativity renaissance", with top-performing companies recognising the importance of these capabilities to complement data and technology excellence. Elsewhere the report states: "Digital skills are vital for a range of marketing tools and platforms.

report. You can view the entire report at bit.ly/2p9sJWB. So what's getting developers excited/worried in 2018? AI is big news. Developers are "optimistic about the possibilities that artificial intelligence offers, but are not in agreement about what the dangers of AI are". Heard of DevOps? Well it and "machine learning are important trends

“Developers are optimistic about the possibilities of AI”

Almost three-quarters (73 per cent) of respondents agree that their companies are 'combining digital marketing skills with technology'. Companies doing this are nearly twice as likely to have surpassed their 2017 business goals by a significant margin (20 per cent vs. 11 per cent).

On the flipside is Stack Overflow's Developer Survey Results for 2018. So what does the latest survey reveal? Again this is only a snapshot of a very comprehensive

in the software industry today. Languages and frameworks associated with these kinds of works are on the rise".

An area that caught our eye was the different priorities female and male developers had when assessing a prospective job. Women were looking for company culture and opportunities for professional development, while men say their highest priorities are compensation and working with specific technologies.

STAT ATTACK

MOBILE OS

Who has control of the worldwide small screen market?

Android



Google's OS stands tall amongst its competitors

iOS



Still very popular in North America

Unknown



A collection of OS' you've never heard of

Windows



Not quite as popular as Google and Apple

Nokia Series 40



The old Nokias are still with us

Source: gs.statcounter.com

CREATIVE BLOQ

creativebloq.com

Creative Bloq is a website dedicated to art and design inspiration. It features In-depth tutorials, expert tips, cutting-edge features, industry interviews, inspiration and opinion. Make sure to get your daily dose of creativity, design and development.

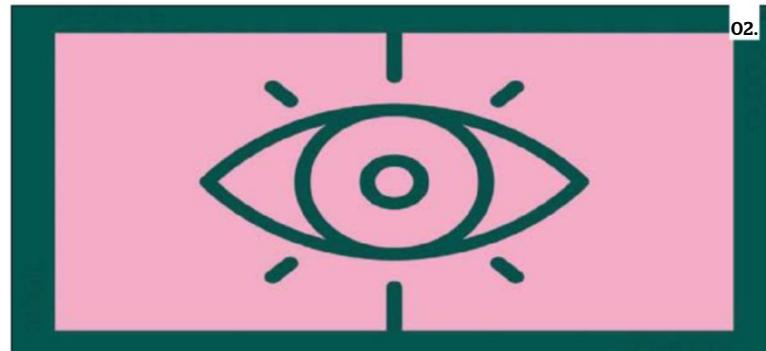
WEB DESIGNER DIGITAL EDITION

Do you want to get your hands on a digital edition of your favourite web design magazine? Head to your preferred app store – Google Play (bit.ly/2wetvGp) or iTunes (apple.co/2igtBYq) – then download, install and purchase the issue of choice from within the app.

Sites of the month



01.



02.



03.



04.

01. SRETKS

sretks.com

A neat 3D animation, which features a choice of touch or scroll interaction.

02. Magic People

magicpeoplevoodoopeople.com

A blinking eye and smart rollover animations engage the user.

03. Alpsko Mleko

alpskomleko.si

Making milk interesting through the use of animation and storytelling.

04. Those Polarizing Patriots

bit.ly/2p8BBuC

Love or hate? You get to choose with this interactive infographic.

Graphics Mutidimensional Genie

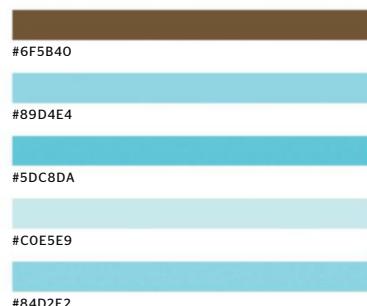
bit.ly/2Ghf9HX

A great example of illustration and character design by Davide Molino.



Colour picker Siesta

bit.ly/2FJSztG



Typesetter Juicy

myfonts.com/fonts/positype/juicy

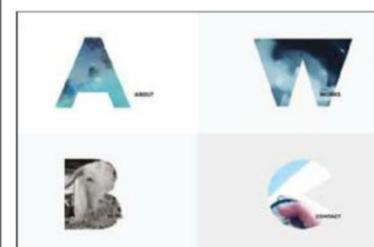
An Art Deco-inspired, high contrast, upright semi-script, layered typeface, which is ideal for branding.

ABCabc
012345

WordPress Aoko

themes.blahlab.com/aoko/

A four-way fullscreen menu with engaging masked background video.



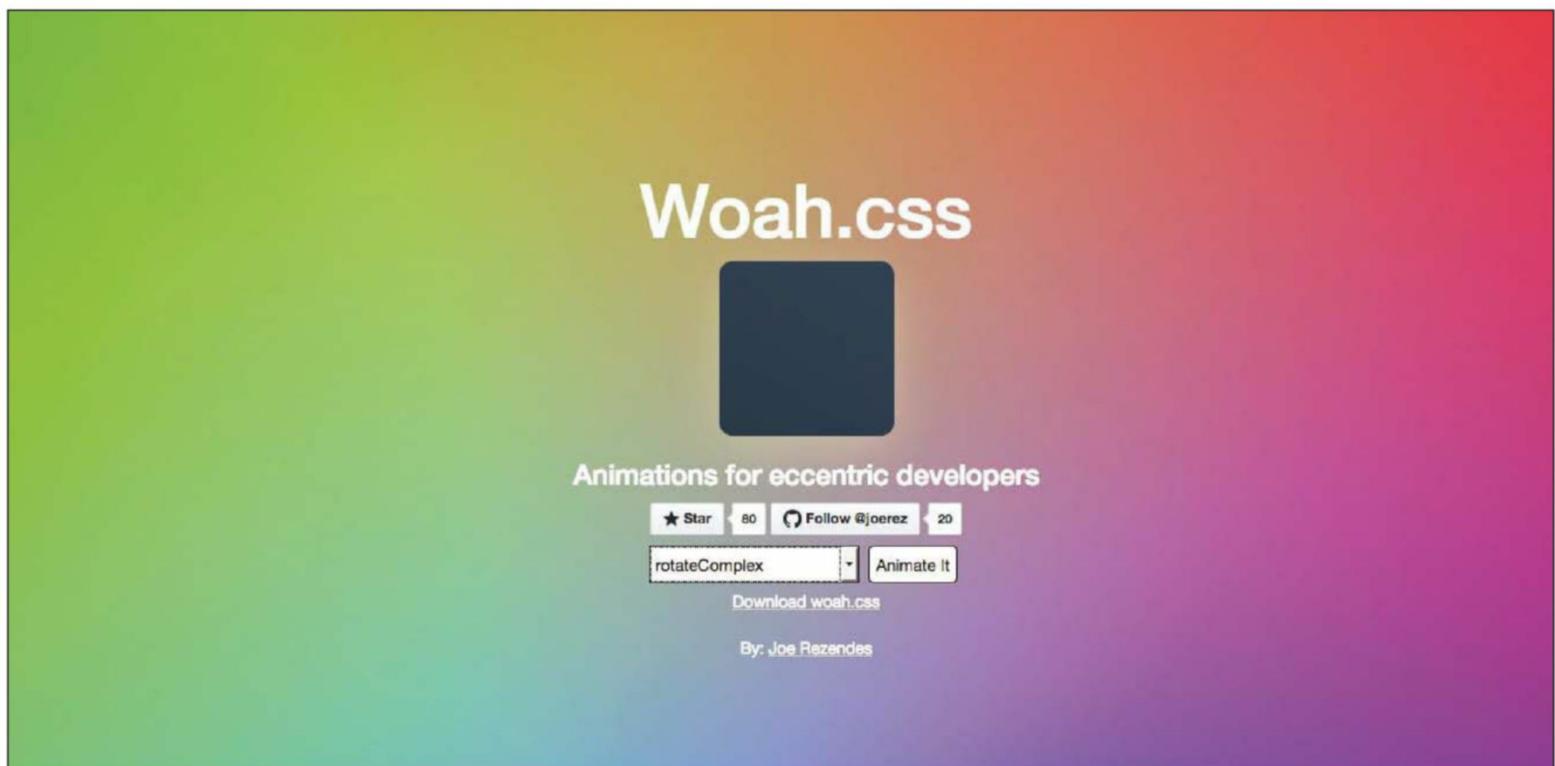
Header

Resources

CONTACT US AT: steve.jenkins@futurenet.com | @WebDesignerMag

webkit

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



Woah.css

joerezdes.com/projects/Woah.css

CSS and animation are well-established friends and the beauty of libraries is that they take the hard work of the creative process. The Woah library provides a host of animations that go just beyond the normal. They describe the library as 'animations for eccentric developers'. Head to the site to check out what's on offer. We liked rotateComplex.



Ghostery

github.com/ghostery/ghostery-extension

Ghostery is a browser extension that helps users get control over ads and tracking technologies to speed up page loads and help protect your data.



Radi.js

radi.js.org

Radi is a tiny (3kb minified & gzipped) JavaScript framework. This means that it is really fast. Also works with browsers that support ES6.



SlimerJS

slimerjs.org

This is 'A scriptable browser for Web developers and currently in beta'. Use it to perform functional tests, network monitoring, screen capture and more.

TOP 5 Web conferences - May 2018

Get yourself a seat at the biggest and best conferences coming your way soon



99U Conference

conference.99u.com

Three days of insightful talks, stimulating workshops, and unexpected connections to help you stay creative.



Awwwards

bit.ly/2BGfikX

Head to San Francisco for this two-day event, which is aimed at designers, web developers and digital dreamers.



ReactEurope

react-europe.org

Bringing you the best and most passionate people, from the very core team to the coolest people from the community you love.



Deconstruct

deconstructconf.com

Check out this two-day single-track, language- and technology-agnostic software development conference.



UX London

2018.uxlondon.com

Celebrate the conference's tenth anniversary with some of the world's leading design experts and practitioners.

Who wants to be your own boss?

How to work with the most entrepreneurial generation yet



Brandon Geary
Global Chief Strategy Officer at POSSIBLE
possible.com

“ Kids these days aren’t like they used to be. If they have an idea, be ready to listen ”

Think of what you wanted to be when you grew up. It was probably a job. Lawyer, firefighter, filmmaker, or rock star. Ask that question to a Gen Zer, and you’ll get a different answer. According to Gallup, 77 per cent of students in grades five through 12 say they want to be their own boss.

Not surprisingly, the ‘40 under 40’ lists that come out in business magazines now have company. Business Week publishes a ‘25 under 25’ list. Not to be outdone, Fortune has an ‘18 under 18’. And plenty of kids are launching their own businesses, including seventh-grader Mikaila Ulmer. So, what’s driving the Gen Z hustle? A few things. First, traditional jobs once reserved for teens are not as prevalent as they once were. The many plug-and-play business services and technologies available today are also playing a role. With cloud solutions from AWS, Google, and Microsoft, it’s never been easier to start a digital venture. In addition, kids have access to a new wave of inexpensive business services: Legal Zoom and RocketLawyer for legal matters, 99Designs for logos, Postmates for delivery, and Square for payment. Nearly every business headache has gone away.

Finally, inspiration for young entrepreneurs abounds. They’ve been watching their Gen Y counterparts create huge followings on YouTube, Instagram, and Snapchat. They’ve been raised on user-generated content, so they know how to build an audience for themselves and their companies.

But brands aren’t taking the new wave of entrepreneurship seriously. Or at least not seriously enough to build it into their marketing and product development budgets. Instead, they’re focusing on influencer marketing, product placement, and regramming. While there’s nothing wrong with co-creation and product placement, we can do better. Here are three new areas of focus for brands:

- **Provide encouragement and support** – While brands look to youth for influencer activity, they traditionally don’t consider them as a source of viable business ideas and investment opportunities. This will change with the right support and encouragement. Eddy Zhong dropped out of high school in 2016 to start Leangap, the first high school entrepreneurship accelerator program in the world. It helps students conceptualise and launch their business in as little as six weeks. Brands can help scale products and services formed in Gen Z-focused accelerators and connect them to a bigger stage in keeping with corporate product development strategies.

- **Take the call** – Steve Jobs’ story of landing a summer job at HP by calling the then CEO Bill Hewlett directly is the stuff of Silicon Valley legend, but it mostly serves as an example of job hunting persistence. Marketers take notice. Kids these days aren’t like they used to be. If they have an idea, be ready to listen. For example, when 14-year-old app developer Soroush Ghodsi cold-called Y Combinator founder Sam Altman, Sam took the call. Not only that, he made Soroush and his partner Stefan Stokic the youngest founders in Y Combinator history. “If they build something good,” Sam said, “it doesn’t matter how old they are.”

- **Test and earn** – While paying influencers to co-create content is part of most marketing playbooks, licensing partnerships are not. And they should be. The NBA entered a licensing partnership with Mozhia “Mo” Bridges, the 15-year-old founder of Mo’s Bows, which manufactures handmade bow ties and neck ties. Mo can now sell ties with the logos of NBA clubs and distribute them through his company’s online store. Bella Weems founded Origami Owl, a custom jewellery company at the age of 14 and quickly built a customer base for her locket designs. It is now a commerce platform for independent jewellery designers and salespeople, and has a licensing agreement with Illumination Entertainment and Universal Studios to give independent designers access to Despicable Me characters.

As media continues to fragment and compete for limited attention, it’s time to start thinking of Gen Z as more than a source of content creation or product placement. Young people can be a source of tangible product and service development for your brands. After all, the future of businesses, not just marketing, depends on them.

Header

Resources

CONTACT US AT: steve.jenkins@futurenet.com | @WebDesignerMag

webkit

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

Flutter is now in Beta! [Learn More.](#)

Build beautiful native apps in record time

Flutter is Google's mobile UI framework for crafting high-quality native interfaces on iOS and Android in record time. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.

GET STARTED

The Flutter website features a clean design with a blue header bar at the top. Below it, there's a large blue section with white text. To the right of the text is a 3D-style illustration of a smartphone and a tablet connected by dashed lines, symbolizing the cross-platform nature of the framework. At the bottom left is a blue 'GET STARTED' button.

Flutter

flutter.io

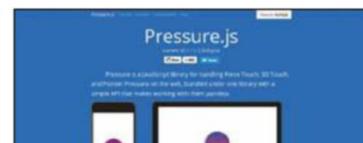
The option to build faster is one that most developers are happy to take onboard. And this is what Flutter is here to do. The framework is 'Google's mobile UI framework for crafting high-quality native interfaces on iOS and Android in record time'. It works with existing code and will help developers create beautiful user interfaces.



Pagedraw

pagedraw.io

Want to turn your mockups into functional working code without the fuss? Pagedraw generates code that will work correctly across all browsers.



Pressure.js

pressurejs.com

This is a library for handling Force Touch, 3D Touch, and Pointer Pressure on the web. A simple API lets users press and hold to expand chosen elements.



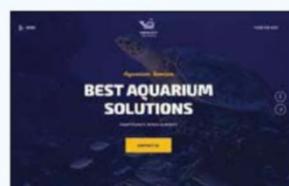
GitLab

about.gitlab.com

GitLab streamlines collaborative workflows. It lets you visualise, prioritise, co-ordinate, and track progress with its flexible project management tools.

TOP 5 WordPress themes for May 2018

Need a good-looking and practical site to promote your business? Then check out our top picks



Aqualots

aqualots.themerex.net

A full-screen image slider takes centre stage, with blue overtones across the site reinforcing the message.



Rosebud

bit.ly/2Dn6WPz

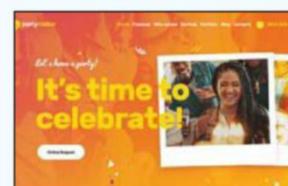
Flowers are a riot of colour and this theme makes great use of imagery to sell the product to the customer.



Jharon

bit.ly/2p6jR4a

Cleaning is not the most glamorous subject, but this theme uses imagery and message to sell its services.



PartyMaker

bit.ly/2Gobf05

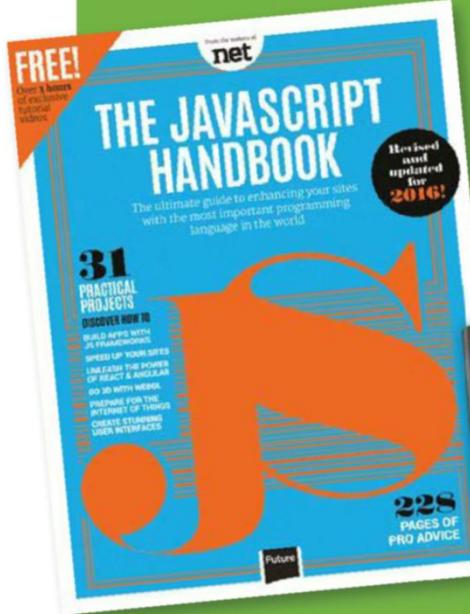
If you are a party planner then this theme provides the ideal online presence. Offers neat options to reveal services.



Haustiere

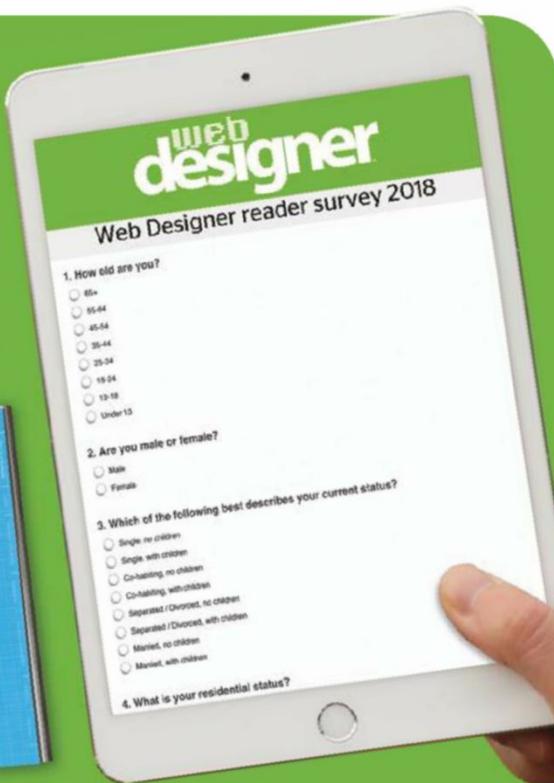
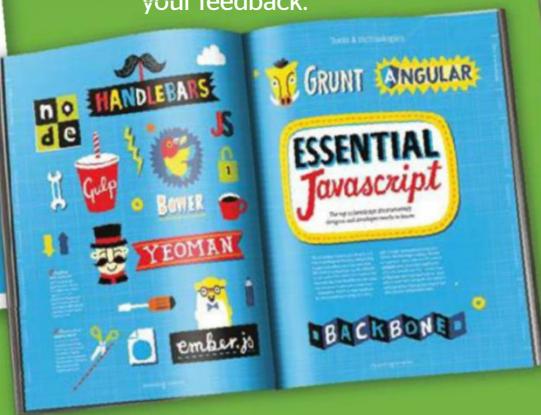
bit.ly/2HwMxtR

Fullscreen imagery of cute dogs immediately offers impact. Nested design elements bring the site together nicely.



FREE The JavaScript Handbook

After completing the survey,
you'll receive a digital copy of
The JavaScript Handbook
as a thank you for
your feedback.



FREE DIGITAL EDITION!

Complete our reader survey at
<http://bit.ly/2BtaLmR>



Each and every issue, we do our best to create a magazine that you love to read, and we're always looking for ways to make it even better. Once again, we'd like your help in improving Web Designer magazine, and all you need to do is answer some very quick questions about yourself - we're really looking forward to hearing from you.

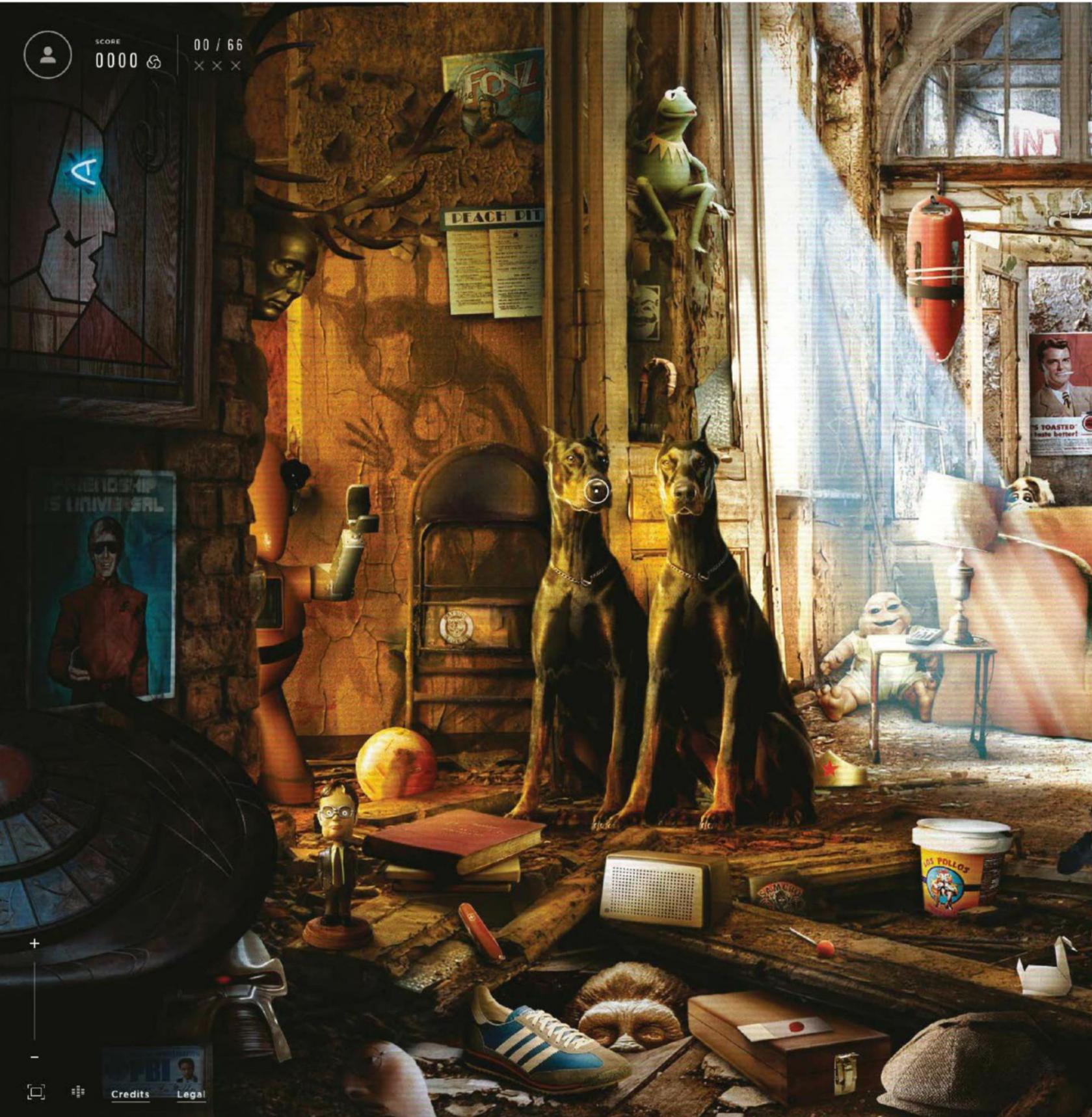
Steven Jenkins
Editor



PopCorn TV

www.popcorntv.fr

Designer:
PopCorn 66 www.popcorn66.fr



SCORE
0000 ♂

00 / 66
x x x

[] Credits Legal

“French agency PopCorn 66 combine digital and pop culture to awesome effect with this discovery game for iconic TV shows”



Colours

#CD7300	#A8BB7F
#4693A9	#4B2C10

Tools

jQuery, Angular JS,
Three.js, GSAP

Fonts

abcABC
1234567890

Pathway Gothic One by Eduardo Tunni, a narrow grotesque sans typeface, is used on score labels.

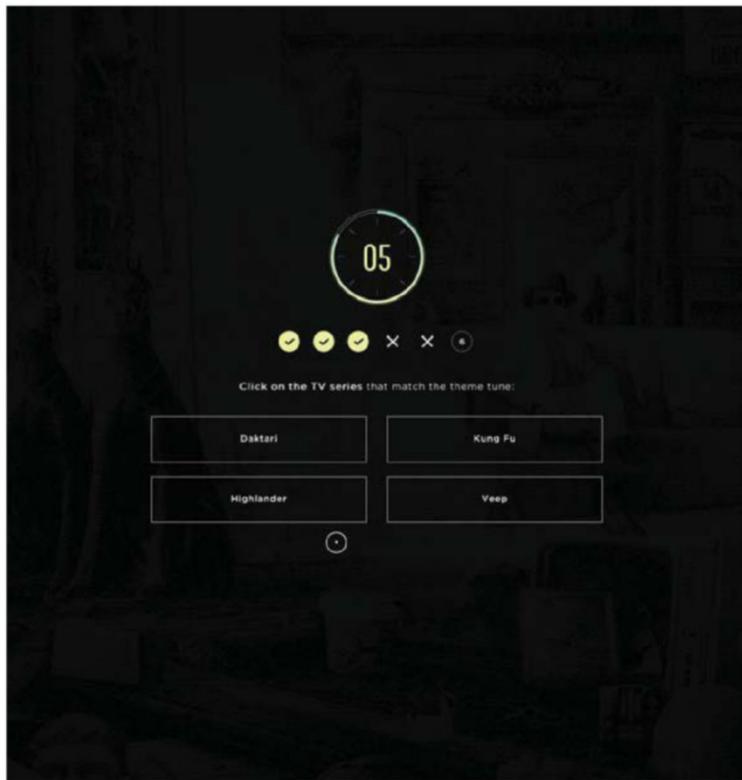
abcABC
1234567890

Gothic Medium is the secondary font used across the various other UI elements, buttons and links.



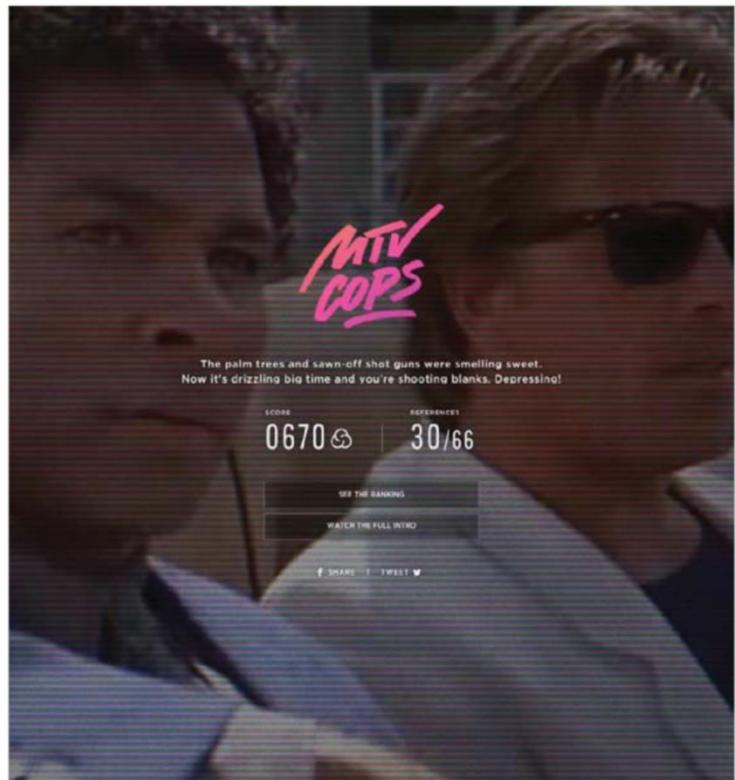
Above

The experience opens to an evocative full screen video intro accompanied by pumping 80s synth music by Eric 'Elvis' Simonet



Above

Extra bonus points are available with a timed multiple-choice quiz for naming iconic theme tunes from yesteryear



Above

Enter too many incorrect guesses and the game is over. Players receive a retro-flavoured rating on performance

Create an interactive TV noise background effect

Make use of the 'switch on' button to present content with a TV background noise effect

1. Initiate HTML document

The first step is to initiate the structure of the HTML document. This consists of the document's HTML container that stores a head and body section. The head section's main responsibility is to load the external JavaScript and CSS stylesheet, while the body section stores the visible content defined in step 2.

```
<!DOCTYPE html>
<html>
<head>
<title>Background Noise on Click</title>
<link rel="stylesheet" type="text/css"
href="styles.css" />
<script src="code.js"></script>
</head>
<body>
*** STEP 2 HERE
</body>
</html>
```

2. HTML content

The HTML content consists of a button followed by the content container - an article element with a 'data-noiseBox' attribute applied so that it can be found by JavaScript and CSS. The article container consists of an inner 'section' container, which will be used for an overlay effect. Visible content text is placed inside this inner section.

```
<button>Click Me</button>
<article data-noiseBox>
<section>
<h1>Hello There!</h1>
</section>
</article>
```

3. JavaScript button listener

Create a new file called 'code.js' to store the required JavaScript. This step's code waits for the page to load, upon which the 'document.querySelector' is used to find the HTML button created in step 2. Change the 'button' parameter to another CSS reference such as using an ID for your real project.

```
window.addEventListener("load", function()
{
  var button = document.
querySelector("button");
  button.addEventListener("click",
function(){
  *** STEP 4 HERE
});});
```

4. Trigger function

After finding the button, an event listener is applied to it to trigger a function when clicked. This function uses 'nextElementSibling' to find the article placed immediately after the button; It reverses the value of its 'data-noiseBox' attribute value between 'active' and blank depending on its current value. The switching of the 'data-noiseBox' attribute will be used by CSS to trigger presentation changes.

```
button.addEventListener("click",
function(){
  var nextNode = this.nextElementSibling;
  if(nextNode.getAttribute("data-
noiseBox") == "active"){
    nextNode.setAttribute("data-noiseBox",
"");
  }else{
    nextNode.setAttribute("data-noiseBox",
"active");
  }
});
```

5. CSS initiation

Create a new file called 'styles.css'. The first rules in this file set the presentation of the HTML document and the body content containers. They are set to display with a black background using white text. The padding and margin attributes are set to zero to guarantee no visible border spacing.

```
html, body{
background: #000;
padding: 0;
margin: 0;
color: #fff;
}
```

6. Noise box

Containers that use the effect are identified with the 'data-noiseBox' attribute. This step sets these containers to cover the full screen, but with opacity set to invisible by default, a transition sets changes to opacity to animate over one second. A black gradient that fades in and out is also applied to the first level inner container of the 'data-noiseBox' element.

```
[data-noiseBox]{
position: relative;
width: 100vw;
height: 100vh;
opacity: 0;
transition: opacity 1s;
}
```

7. Active noise box

Visibility of the main container is activated by the 'data-noiseBox' attribute being set to 'active'. The main background has a 'noise' image applied, with an associated 'noiseMove' animation that quickly loops over a duration of 0.25 seconds and repeats infinitely. The first level inner container is set to cover the static background with a black gradient that fades in and out with a transparency.

```
[data-noiseBox="active"]{
background: url(img/noise.png);
background-size: 25% 25%;
animation: noiseMove .25s infinite
linear;
opacity: 1;
}
[data-noiseBox="active"] > *{
display: block;
width: 100%;
height: 100%;
background: linear-
gradient(rgba(0,0,0,1), rgba(0,0,0,0),
rgba(0,0,0,1));
```

8. Noise animation

The background noise effect is made from rapidly changing the position of the background texture applied to the container in step 7. While the previous step deals with the rapid speed of the 'noiseMove' animation, this step sets the points for the background position to animate between each key frame. Percentage is used for the co-ordinates to allow the effect to adapt to screen or container size.

```
@keyframes noiseMove {
0%{ background-position: 45% 50%; }
20%{ background-position: 65% 40%; }
50%{ background-position: 85% 70%; }
80%{ background-position: 65% 80%; }
100%{ background-position: 45% 50%; }
```



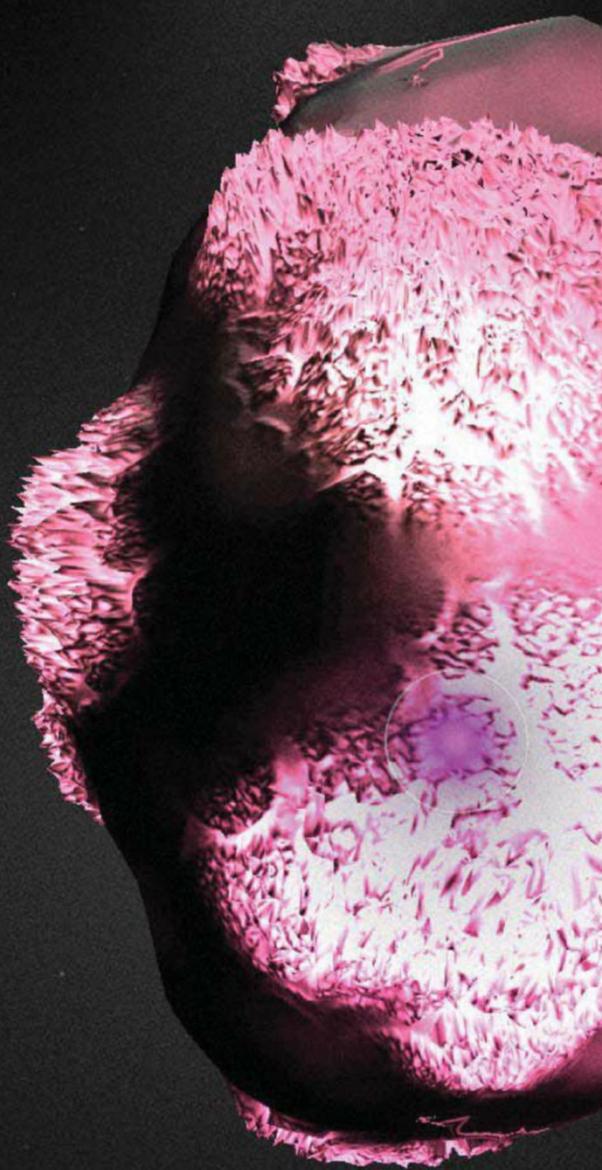
Love Lost

<https://lovelost.jam3.com>

Designer:

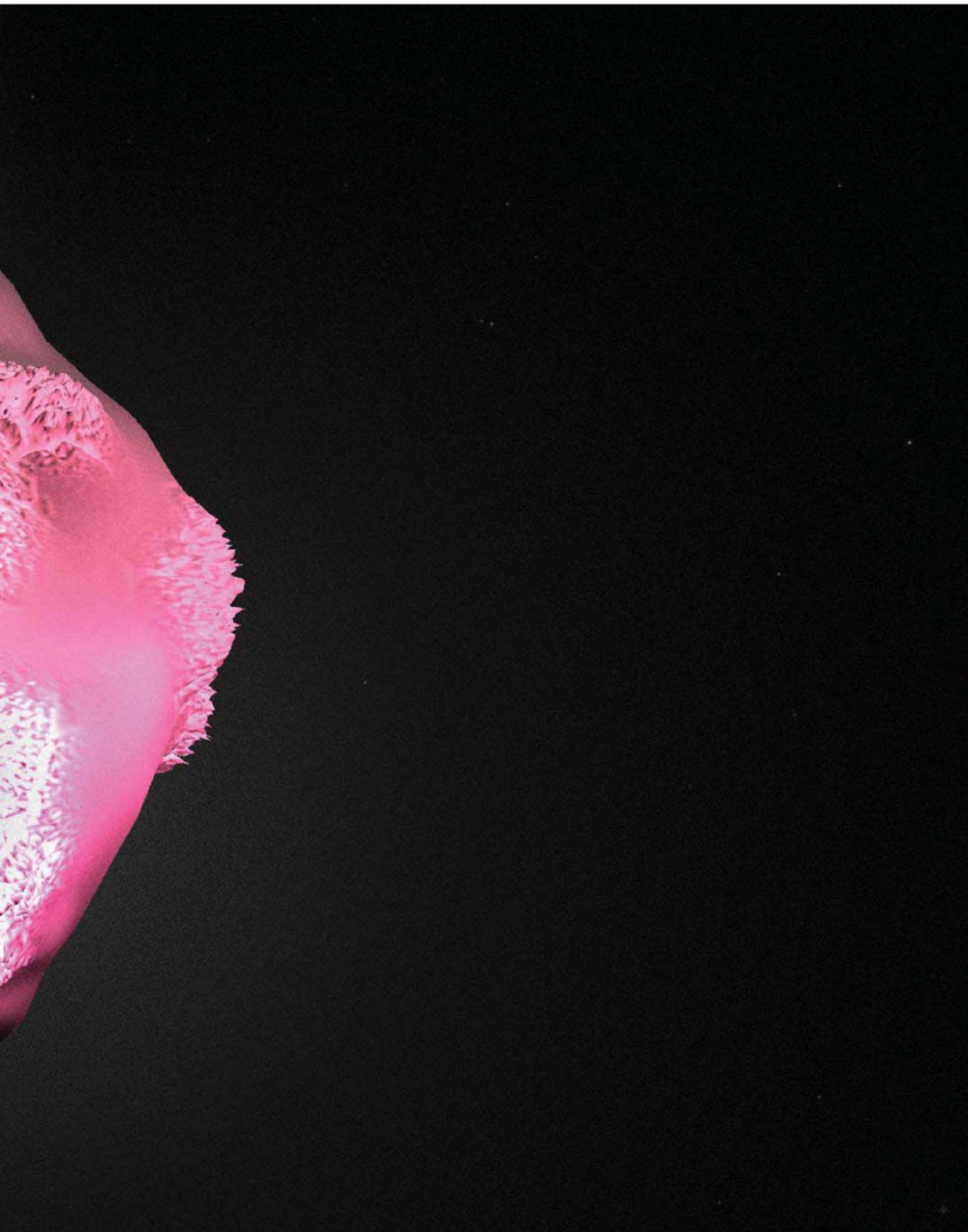
Jam3 www.jam3.com

H E A R T



WITH LOVE FROM 

“Canada’s Jam3 present a beautifully dark, mobile-ready interactive poem with real heart about the enduring feelings around lost love”



Colours



Tools

GSAP, Modernizr, Apple Mobile App Capable, HTML5

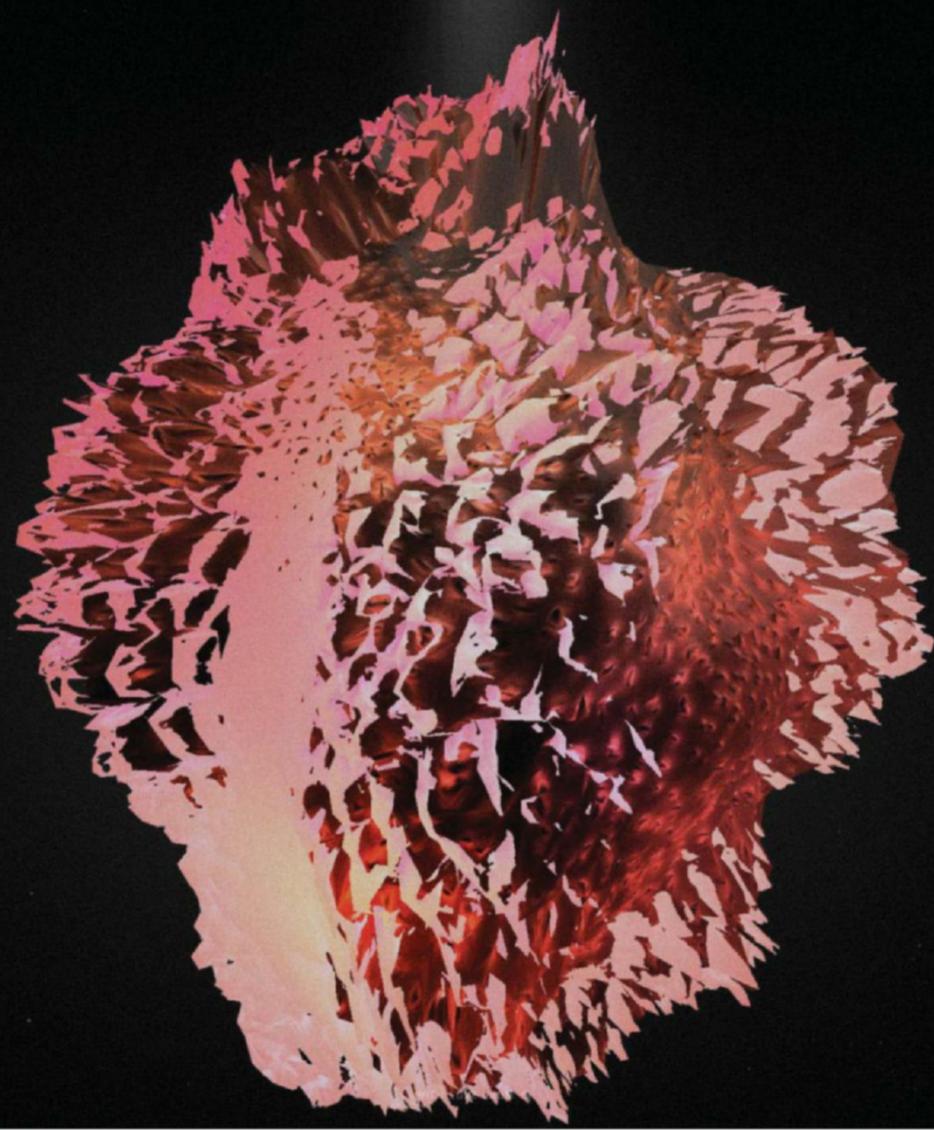
Fonts

abcABC
1234567890

Rajdhani by Indian Type Foundry is used in Medium weight on the ending share text and replay link.

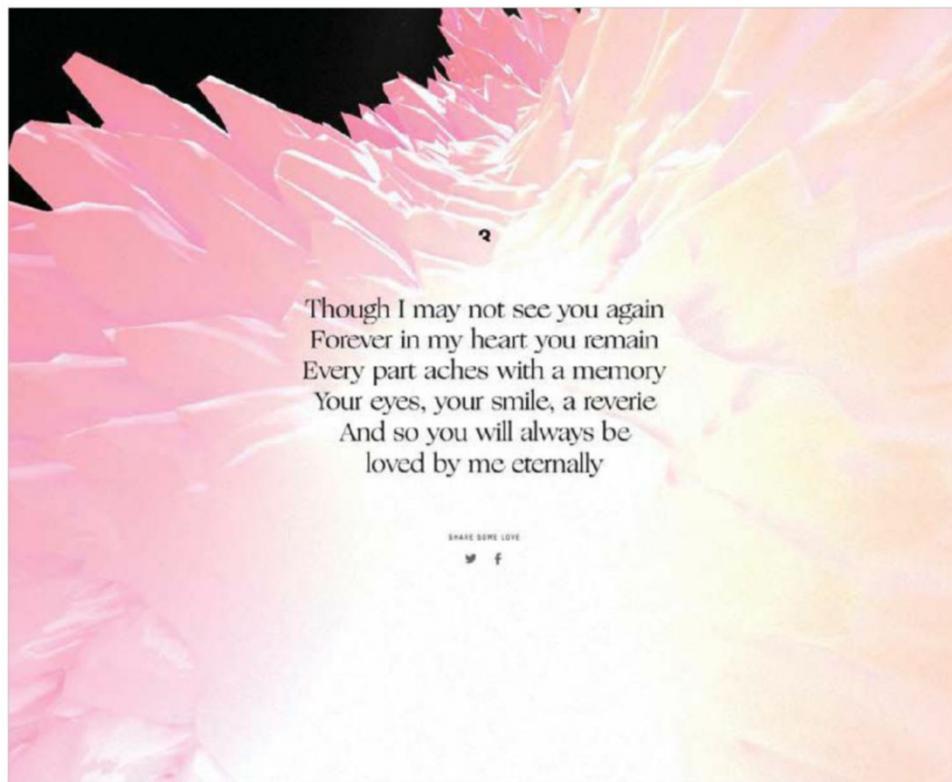
abcABC
1234567890

The Ogg font by Sharp Type appears in Roman typeface to style the poem text shown at the end.



Above

Pulsing 3D heart-like rocks can be rotated and pressed to make them swell and burst out lines from the poem



Above

Each line of the poem is combined at the end with social links to 'Share Some Love' with friends



Above

The mobile experience is wholly consistent, delivering the very same 3D animation sequences and atmospheric music

Create a visually animated 3D text intro

An animated introduction giving the illusion of 3D depth of space and specified text content

1. Initiate the HTML document

The first step is to define the structure of the HTML document. This includes the HTML container that initiates the document, which contains the head and body sections. While the head section is primarily used to load the external CSS file, the body section will store the visible page content created in step 2.

```
<!DOCTYPE html>
<html>
<head>
<title>3D Text Movement</title>
<link rel="stylesheet" type="text/css"
      href="styles.css" />
</head>
<body>
*** STEP 2 HERE
</body>
</html>
```

2. Visible HTML content

The visible HTML content consists of an article container that contains the visible text. The important part of the article container is the 'data-animate' attribute, which will be referenced by the CSS to apply the visual effects. The text inside the article is made from a h1 tag to indicate that the content is the page's main title.

```
<article data-animate="move in">
  <h1>Hello There!</h1>
</article>
```

3. CSS initiation

Create a new file called 'styles.css'. The first set of instructions set the page's HTML container and body to have a black background, as well as no visible border spacing. White is also set as the default text colour for all child elements on the page to inherit; avoiding the default black colour of text making content appear to be invisible.

```
html, body{
  background: #000;
  padding: 0;
  margin: 0;
  color: #fff;
}
```

4. Animation container

The content container referenced with the 'data-animate' attribute has specific styles applied. Its size is set to match the full size of the screen using vw and vh measurement units, as well as being slightly rotated. An animation called 'moveIn' is applied, which will last for a duration of 20

seconds and will repeat infinitely.

```
[data-animate="move in"]{
  position: relative;
  width: 100vw;
  height: 100vh;
  opacity: 1;
  animation: moveIn 20s infinite;
  text-align: center;
  transform: rotate(20deg);
}
```

```
10%
{
  opacity: 1;
  text-shadow:
    .2em -.2em 4px #aaa,
    .4em -.4em 4px #777,
    .6em -.6em 4px #444,
    .8em -.8em 4px #111;
}
*** STEP 7 HERE
```

5. Animation initiation

The 'moveIn' animation referenced in the previous step requires a definition using @keyframes. The first frame starting at 0% of the animation sets the default font size, text positioning and visible shadow. Additionally, the item is set with zero opacity so that it is initially invisible - ie displayed out of view.

```
@keyframes moveIn {
  0%{
    font-size: 1em;
    left: 0;
    opacity: 0;
    text-shadow: none;
  }
  *** STEP 6 HERE
}
```

7. Finalising the animation

The final frames occur at 80% and at the very end of the animation. These are responsible for increasing the font size and moving the element towards the left. New settings are also applied for the text shadow to animate towards, while also fading the text out of view from frames 80% to 100%.

```
80%{
  font-size: 8em;
  left: -8em;
  opacity: 1;
}
100%{
  font-size: 10em;
  left: -10em;
  opacity: 0;
  text-shadow:
    .02em -.02em 4px #aaa,
    .04em -.04em 4px #777,
    .06em -.06em 4px #444,
    .08em -.08em 4px #111;
}
***
```

6. Animate into view

The next frame is placed at 10% through the animation. This frame sets the opacity to fully visible, resulting in the text gradually being animated into view. Additionally, multiple shadows are added with blue and decreasing colour values to give the illusions of 3D depth to the text.



Hula Hoop

www.hula-hoop.fr

Designer:

Hula Hoop www.hula-hoop.fr

hula hoop®

...



“This clean portfolio site for French agency Hula Hoop puts the team up front with a clever click through photo”

MY HULA HOOP —

Colours

#FF0000	#E8E8E8
#7E9BB1	#E9B395

Tools
HTML5 History API, PHP

Fonts

abcABC
1234567890

Pistilli by Claude Pelletier is the distinctive curly font used predominantly on headings and lead paragraphs.

abcABC
1234567890
abcABC
1234567890
abcABC
1234567890

Rational by Rene Bieder is used in Book, Medium and Semibold varieties across subsequent text and links.

Create an interactive multi-layered pop-out selector

Allow content items to become the focus of attention upon their selection by the user

1. Initiate the HTML document

The first step is to define the structure of the HTML document. This consists of the main HTML container, which stores a head and body section. While the head section's main responsibility is to load the external CSS and JavaScript resources, the body section will be used to store the visible content created in step 2.

```
<!DOCTYPE html>
<html>
<head>
<title>Pop Out Select</title>
<link rel="stylesheet" type="text/css" href="styles.css" />
<script src="code.js"></script>
</head>
<body>
*** STEP 2 HERE
</body>
</html>
```

2. HTML content

The HTML content consists of a container for the selectable items - to be referenced by the 'data-picture' attribute. Each child element of this container will display as an interactive element that will animate to become the focus of attention when clicked. You can place any type of HTML content within these elements.

```
<article data-picture>
<div>1</div>
<div>2</div>
<div>3</div>
<div>4</div>
<div>5</div>
</article>
```

3. JavaScript initiation

Create a new file called 'code.js'. This step waits for the page to complete loading, then will search for all of the first level children of items inside children of containers using the 'data-picture' attribute. Each of these children will have a click event listener applied to them - a function that is executed when the item is clicked.

```
window.addEventListener("load", function(){
{
  var nodes = document.querySelectorAll("[data-picture] > *");
  for(var i=0; i<nodes.length; i++){
    nodes[i].addEventListener("click",
    function(){
      *** STEP 4 HERE
    });
  }
});
```

```
}
```

4. Event code

The code placed inside the event listener function searches for all the first level elements inside the parent 'data-picture' container so that their 'data-status' attribute can be reset to blank. After this reset, the selected item's 'data-status' attribute is reversed between blank or active depending on its current value.

```
var search = this.parentNode.
querySelectorAll("[data-picture] > *");
for(var n=0; n<search.length; n++){
  if(this != search[n])search[n].
setAttribute("data-status","");
  if(this.getAttribute("data-status") ==
"active"){
    this.setAttribute("data-status","");
  }else{
    this.setAttribute("data-status",
"active");
  }
}
```

5. Picture styling

Create a new file called 'styles.css'. This step first defines containers using the 'data-picture' attribute to cover the full size of the screen. Each of the first level children of containers using the 'data-picture' attribute are set to display with a default size and background colour. Absolute positioning and transition rules are set to allow unique positioning and animation effects to occur when selected.

```
[data-picture]{
  position: relative;
  width: 100vw;
  height: 100vh; }
[data-picture] > *{
  position: absolute;
  display: block;
  background: rgba(255,0,0,.25);
  border: 2px solid #000;
  width: 10vw;
  height: 20vw;
  transition: width 1s, height 1s, z-index
1s, top 1s, left 1s; }
```

6. Child styling

Each child within the 'data-picture' containers need a unique vertical position, horizontal position and z-index for depth. This example sets specific positions for each child item based on their order within the HTML using the 'nth-child' selector. In this example, we reference child

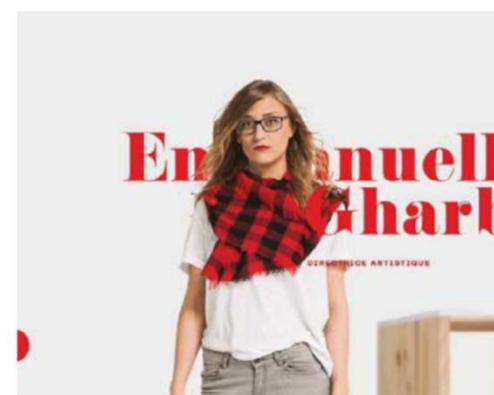
items 1 to 5 using this nth-child selector.

```
[data-picture] > *:nth-child(1){
  left: 20vw;
  top: 20vw;
  z-index: 5; }
[data-picture] > *:nth-child(2){
  left: 25vw;
  top: 10vw;
  z-index: 4; }
[data-picture] > *:nth-child(3){
  left: 30vw;
  top: 20vw;
  z-index: 5; }
[data-picture] > *:nth-child(4){
  left: 35vw;
  top: 10vw;
  z-index: 4; }
[data-picture] > *:nth-child(5){
  left: 40vw;
  top: 20vw;
  z-index: 5; }
```

7. Active children

Any item that has been selected by the user will have had a 'data-status' attribute applied that has been set to active by the JavaScript applied in steps 3 and 4. Active items are set to appear at a bigger size at a specific location that appears above all other elements. Their changes will appear animated due to the transition defined in step 5.

```
[data-picture] > *[data-status="active"]{
  z-index: 10;
  background: red;
  width: 20vw;
  height: 40vw;
  top: 10vw;
  left: 20vw; }
```



generate (mt)

The conference for web designers

NEW YORK | 25-27 APRIL 2018



TICKETS ON SALE NOW

www.generateconf.com | #generateconf



This is possible thanks to members with their **commitment, honesty, enthusiasm, discipline, responsibility and personal ethics** in creating a safe, healthy, human and fair environment.

A SENSE OF IDENTITY

SELLING THE SERVICES OF A FOOD QUALITY CONTROL COMPANY MIGHT CAUSE MOST AGENCIES TO BOIL OVER, BUT FOR MADRID'S TWENTY-TWO DEGREES THE CREATIVE KITCHEN NEVER PROVED TOO HOT

*to our staff
non sense,
possibility,
inal integrity in a
ment.*

ANALYTICA PROJECTS

analyticaprojects.com

BY TWENTY-TWO DEGREES

veintidosgrados.com

@twentytwodgres



RUBÉN SÁNCHEZ
DESIGNER



VICENTE LUCENDO
DEVELOPER

There's an interesting thought to be had about the modern web experience, in that it's increasingly, almost exclusively, an eCommerce thing. Why? Because everyone is seemingly selling something, even if they aren't actually well, selling anything. The presence or absence of a shopping cart defines nothing in an age where identity is such a currency and the marketing of a memorable online image so coveted. This month's web design project is all about that pursuit of promotion and the exciting bringing together of an agency and client who share an outlook. Madrid-based digital creative studio Twenty-Two Degrees prides itself, like most, on an ambitious approach where every minute detail of a project is obsessed over. Striving for their own perception of perfection around design and

development, each new brief is an opportunity for innovation. Crafting wholly custom-made results, the duo of co-founders Rubén Sánchez and Vicente Lucendo place real emphasis projecting the personality of their clients. Vicente Lucendo takes up the reins for the rest of the interview, "Our market niche is geared mostly towards those who want a modern and technological image for themselves. Those who are not afraid of change, those willing to get away from current trends and feel comfortable with creative and sometimes risky proposals. In short, those clients who want to stand out from the competition." In this instance, it just so happens that Analytica Projects would indeed be one of those clients looking for something different. As leaders in food quality control, they had a story to tell about the work they do without giving visitors the hard sell. "They wanted to get away from that conventional over-structured design that most companies have where the main focus is to give lots of information to users. They didn't want to sell anything though the website, instead they wanted to use it to convey their online image, one which had to be different, funny and creative." So began the task of taking a brief so blissfully open to possibilities when it came to selling Analytica's identity by the dozen. >



WORKING WISELY

> As is often the case, this was a working relationship with a successful history. Having worked with the client in the past, everything was immediately quite straightforward in spite of a slight delay in scheduling. The Twenty-Two Degrees guys couldn't pick the project up for a few months and Analytica were prepared to wait given its satisfaction with previous work. This confidence was fundamental in the vision they had for presenting itself within parent company Analytica Alimentaria. "For Analytica Projects the humane factor is crucial because all their processes involve and require the in-field experience of their team members and they wanted to reflect that on their online image too. So because of this we wanted to convey both the

process and the human factor. They also liked the idea of explaining their process without getting into too much detail, in a simple way without many interactions, using a path to guide the users while highlighting those parts of their procedures where the action of their team is crucial." From that early understanding, the guys requested visits to the Analytica offices, to interview key staff and capture photos or videos useful to getting a clearer handle on the client's work. While a very large multi-national company, Analytica Alimentaria very much started as a family concern driven by two owners Udo and Mar. With the pair still involved in every area of its running, they would provide the most influential point of contact despite being understandably busy and tight on time.
"Because of this we tried to bring to every one of

our meetings visual proposals that explained what we were going to do next so they could have a better idea of it and could give us better feedback. In total we met with them in person or over Skype 5 or 6 times, which is not a lot compared to other clients. Because we had worked with them in the past we already knew their preferences on some topics so we could make better choices during the process and long meetings weren't required this time."

LESS IS MORE

Clearly this mutual confidence was useful, with both parties happy to talk and the client especially comfortable to flag up when their goals weren't aligned. "From the first moment



INNOVATION
AND ANALYSIS,
WHICH
MAKE
FOOD SAFE.

What makes our processes robust? We act before the food enters the market. All our methods are scientifically justified. Their quality is based on systems which are continually being revised and improved, because our customers must take clear decisions based upon our services.

GERMANY
KÄFERWEG 3/14 44261 KÖLN
TEL. +49 221 620 6000
INFO@GERMANIAALIMENTARIAPROJECTS.COM

SPAIN
POLIGONO INDUSTRIAL SECTOR 29/3 CARRIL ALMERIA
TEL. +34 965 022 000
INFO@SPAINALIMENTARIAPROJECTS.COM

NETHERLANDS
WILHELMINA 10 4405 BN MECHELEN
TEL. +31 492 50 20 00
INFO@NETHERLANDSALIMENTARIAPROJECTS.COM

LOCATION



WHERE
FOOD

Video, hidden behind a mask, reveals
small snippets of the product

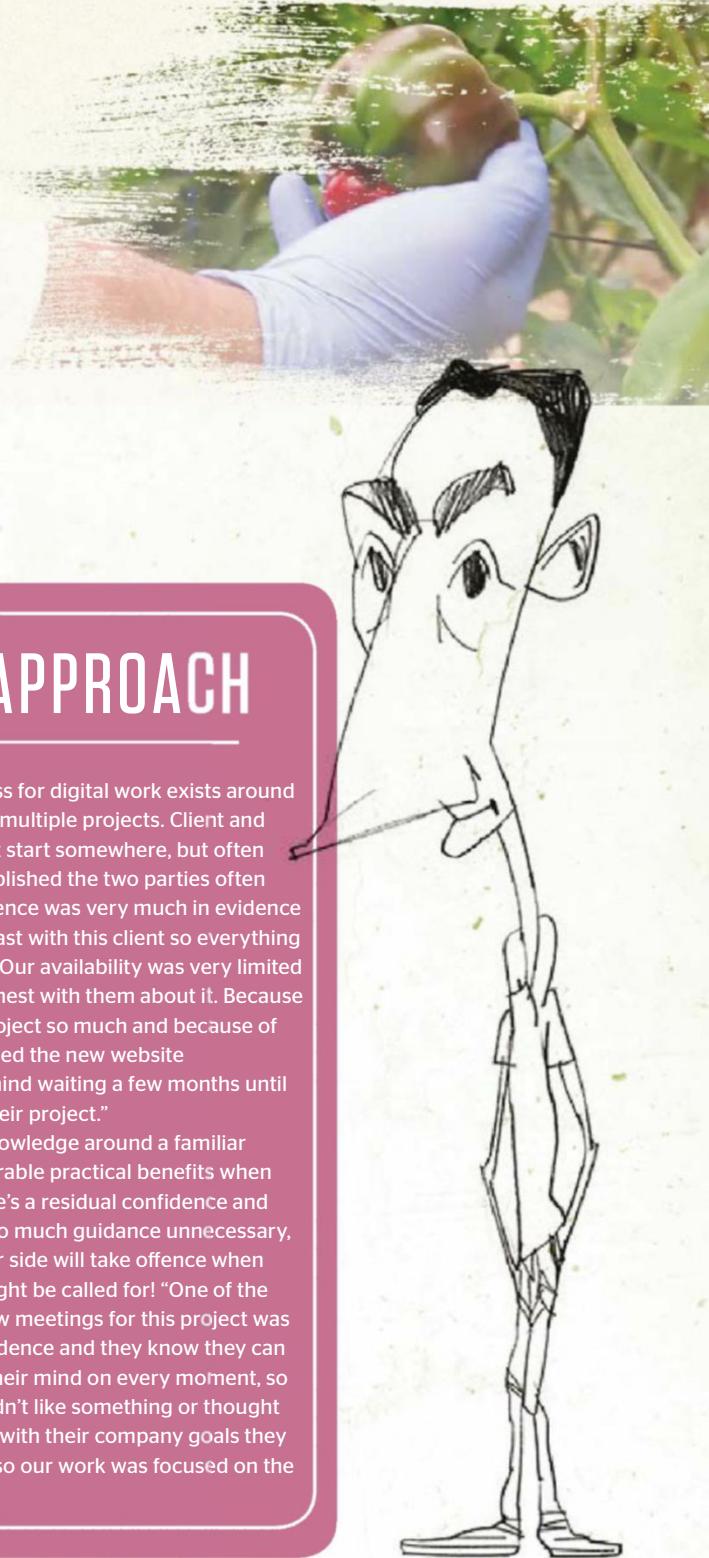
"WITH LIMITED CONTENT, WE THOUGHT ABOUT A STRUCTURE CLOSER TO A LANDING PAGE"

they made it clear to us that they believe in the less is more philosophy, so they wanted a very visual website with a few sentences that gave visitors a general idea of what they do without getting into too many details. That invited us to discard traditional content structures while giving us the opportunity to make the kind of project we really like, clean and very visual, without tedious walls of text nobody really reads. With limited content, we thought about a structure closer to a landing page, where everything is accessible by just scrolling, without the need to make multiple sections and pages that otherwise would have ended up pretty much empty. This simplicity of structure and content was also conducive to bypassing lots of sketching and wireframing to move swiftly into the design process, proposing tangible concepts much faster. One of the main challenges we face when communicating our ideas to clients is when we show them the still designs. We have a different perception that includes context and interactions so we have to be extra careful in explaining to them how everything will work from animations, transitions, movement of elements etc. Most of the time they trust us because they have seen the work we've done in the past. Our main idea for this project was to describe what they do in a single sentence that would be used as the background of the landing site and users could read it while scrolling. Every part of the sentence had to make >

HONEST APPROACH

So much of repeat business for digital work exists around a partnership forged over multiple projects. Client and agency relationships must start somewhere, but often once they have been established the two parties often stick together. This preference was very much in evidence here. "We worked in the past with this client so everything was very straightforward. Our availability was very limited at the time so we were honest with them about it. Because they liked our previous project so much and because of the fact that they didn't need the new website immediately, they didn't mind waiting a few months until we were able to pick up their project."

This enthusiasm and knowledge around a familiar commission adds considerable practical benefits when project time is short. There's a residual confidence and trust that often renders too much guidance unnecessary, while knowing that neither side will take offence when some honest feedback might be called for! "One of the reasons why we had so few meetings for this project was that we have mutual confidence and they know they can tell us directly what's on their mind on every moment, so if for some reason they didn't like something or thought something wasn't aligned with their company goals they could tell us immediately so our work was focused on the right direction."



A SENSE OF IDENTITY

> sense by its own and had to have a specific colour that would differentiate the multiple sections of the site. They loved it so we continued working on this direction."

VISUAL CHARACTER

Photoshop proved a crucial tool for realising the visual design process, presenting quite advanced concepts quite quickly. Although lots of revision here was never necessary, a troublesome element was in translating the site's background sentences and subsequent elements to fit three different languages. "We proposed them to make cartoons of the team for their section and for it we had the help of the very talented Spanish cartoonist Joaquín Aldeguer, who has a very personal style which we thought would match very well with the aesthetics of the site. We sent him photos of some of the team members of the company and after only a few days we had the finished cartoons back with us."

The visual quirkiness these ideas brought were further lifted by a commitment to animate, all the while being mindful of keeping things functional. "One of the parts that took us several days to complete but look insignificant compared to others, were the icons inside the circles in the path. At first they were static images but with the website already finished we decided to animate them in After Effects so they looked more alive. We then tried to integrate them as spritesheets but that caused performance problems. After that we tried to use small videos but that caused even more problems because different browsers render video colours in different ways. Surprisingly what worked out for us in the end was to make simple GIFs with the animations."

PROTOTYPE PERFECT

With the back-end work tied so closely to front-end development and making those animated elements work, any coding challenges really revolved around that. Typically asked to build their sites for WordPress, the guys adopted the same approach here while utilising the Rest API to control preloads, language changes and data requests. "For the front-end we had to use WebGL to achieve some of the effects and to make the site run as smooth as possible. We used the 2D framework Pixi.js and made custom shaders for various elements that couldn't have been made in any other way such as the path, the video mask, the background noise and of course the "barrel effect" that is noticeable when you quickly scroll the page. This last effect was added at the end of the project and turned out to be one of the most special aspects." Of course being a



SITE HIGHLIGHT

WE ASKED CO-FOUNDERS VICENTE AND RUBÉN TO PICK STANDOUT ASPECTS OF THE ANALYTICA PROJECTS WEBSITE THAT BRING THEM THE MOST PRIDE.

"From a creative perspective, it's probably how we use the background sentence to guide the user through the sections of the site, telling them also the different processes the company implements. From an interactive perspective what really stands out is how all the background elements of the site are deformed depending how fast the user scrolls or drags the navigation circle placed on the top right of the user interface."

Talented Spanish cartoonist Joaquin Aldeguer gave a very personal style to the team

This is possible thanks to our **staff** members with their **common sense, honesty, enthusiasm, disposability, responsibility and personal integrity** in a man and fair environment.



HANOVER FIRST

As an agency that really values the details during a project's development, it's probably not terribly surprising that Twenty-Two Degrees would be just as conscientious beyond completion. With "aftercare" becoming increasingly important to defining an agency's service commitment, they take pride in acknowledging that life goes on for the delivered product. "When we give the completed project to a client we also give them all the source files from the design and programming. We think this is crucial because the project now belongs to them so if in the future they want to make some changes to it through another agency or freelance they have everything just like we left it. We also use to advise them about the hosting needs of the website given that our projects tend to have an important amount of photos and video. We try to recommend hiring a quality hosting service which will make the content to be served fast and won't cause excessive waiting or preload times to the visitors." Those recommendations on hosting after handover are a valid point, especially when bad servers might impact on the performance of a site so lovingly crafted. Plus, in addition to the general guarantees Twenty-Two Degrees give each project, clients have the option of buying extra consultation or design work time for the stuff they can't handle themselves. "All our websites have a back-end where clients can change all the elements like menus, sections, galleries and languages etc and so we give them training to use it and edit the content themselves. So unless they want to make important modifications that will require custom programming they usually don't ask us to change anything."

single page or 'landing site' in structure, that scrolling would prove a crucial interaction in delivering the content.

The guys therefore ran an early prototype for detecting every frame the scroll position of the user was in, before hiding everything that wasn't on screen so the browser only had to load what was necessary. "Without this prototype helping us to see if our idea was technically possible we couldn't have optimised every element, leaving a really poor user experience due to the evident frame drops. Similarly, for the section that talks about the processes of the company, we integrated a physics engine so the user could interact with the circles that represent those processes and make them collide with other circles, the screen edges or the background sentence letters."

DELAYED REACTIONS

From the outset, Twenty-Two Degrees knew the time they had before launch was flexible. So the final phases of the project was afforded some last-minute changes with Analytica's blessing. "For example, the issue we mentioned previously about the icons was one of those changes that we made when the development was already finished but we really thought would improve the result in a very positive way. We thought it would only take us a couple of days to complete but ended up taking us almost a week. Another last minute change was a modification of the background sentence in two languages, which had us making changes to many of the elements that interacted with it

such as cartoons, bouncing spheres etc."

The guys admit that while a project of this type might normally require around 10 weeks of production time, the initial availability wait and some subsequent delays caused the launch to slide slightly. "We were afraid of seeing the ideas we had for the site being done by other agencies or studios before we were able to publish it and that would cause it to lose its 'wow' factor.

When you see your own project so many times you lose that first impression feeling and wonder how will people react the first time they visit it. Luckily for us, once published, we started receiving very good feedback over social media by some very well respected professional peers, so we submitted it to some of the most relevant industry awards and won all three - Awwwards, FWA & CSS Design Awards."



Free
WordPress
guide
worth
£12.99



Subscribe today and receive your free gift!

Every issue, delivered straight to your door



Never miss an issue

13 issues a year, and you'll be sure to get every single one



Delivered to your home

Free delivery of every issue, direct to your doorstep



Get the biggest savings

Get your favourite magazine for less by ordering direct

What our readers are saying about us...

"I've been a reader of Web Designer Magazine since the early days."
@sixrevisions via Twitter

"Love the latest magazine that I purchased today"
@navigation_web via Twitter

"My favourite magazine!!!"
@eduardomurillo via Twitter

Pick the subscription that's right for you



Subscribe and save 20%

- ✓ Automatic renewal – never miss an issue
- ✓ Pay by Direct Debit

Recurring payment of £34 every six months,
saving 20% on the retail price



Instruction to your Bank
or Building Society to pay
by Direct Debit

Originator's reference
7 6 8 1 9 5

Name of bank

Address of bank

Account Name

Postcode

Sort Code

Account no.

Please pay Future Publishing Ltd 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 Future Publishing Ltd and, if so, details will be passed on electronically to my Bank/Building Society. Banks & Building Societies may not accept Direct Debit instructions for some types of account

Signature

Date



One year subscription

- ✓ Great offers, available world-wide
- ✓ One payment, by card or cheque

A simple one-off payment ensures you never miss an issue for one full year. That's 13 issues, direct to your doorstep

UK £67.60 (saving 20% on the retail price)

Europe €81.55 USA \$103.20

Rest of the world \$103.20

Pay by card or cheque

Pay by Credit or Debit card



Visa



Mastercard



Amex

Card number

Expiry date

Pay by Cheque

I enclose a cheque for £

Made payable to
Future Publishing LTD

Signature

Date

Your information

Name

Address

Telephone number

Mobile number

Email address

Postcode

Please tick if you want to receive any communications
from Future and its group companies containing news,
special offers and product information.

Please post this form to
Future Publishing Ltd, 3 Queensbridge, The Lakes, Northampton, NN4
7BF, United Kingdom

Order securely online to receive your free gift
myfavouritemagazines.co.uk/WBDPS18G



Speak to one of our friendly
customer service team

Call **0344 848 2852**

This offer will expire on
10 June 2018

*Terms and conditions: This offer entitles new UK Direct Debit subscribers to pay just £34 every 6 months plus receive a bookazine worth £12.99. Gift is only available for new UK subscribers. Gift is subject to availability. Please allow up to 10 days for the delivery of your gift. In the event of stocks being exhausted we reserve the right to replace with items of similar value. Prices and savings quoted are compared to buying full-priced print issues. You will receive 13 issues in a year. Your subscription is for the minimum term specified and will expire at the end of the current term. You can write to us or call us to cancel your subscription within 14 days of purchase. Payment is non-refundable after the 14 day cancellation period unless exceptional circumstances apply. Your statutory rights are not affected. Prices correct at point of print and subject to change... UK calls will cost the same as other standard fixed line numbers (starting 01 or 02) or are included as part of any inclusive or free minutes allowances (if offered by your phone tariff). For full terms and conditions please visit: www.bit.ly/magterms. Offer ends 10.06.18

DESIGN BUDDIES

Describing itself as a digital branding studio illustrates the breadth of insight Bonhomme bring to its clients. The aesthetic embedded within the studio ensures a focus on engagement, yet also delivers minimal beauty across the digital landscape



Morgane Urbain and Emmanuel Cruellas, co-founded Bonhomme in 2013 with the aim of producing creative projects, accessible to everyone across the digital space. With their extensive experiences in the applied arts, they wanted to see how they could apply their skills to projects that pushed the boundaries of the digital aesthetic.

Graduating in applied arts, Morgane began her career as a stylist and quickly became responsible for the collection produced by a Parisian firm. In 2013, she made the decision that to fulfil her creative drive, she would need to take more control of not only the work she was doing, but

also the direction of her life. Co-founding Bonhomme helped her to realise that desire.

Emmanuel graduated in graphic and digital arts, first developing his career as a freelancer in Paris. Over five years he expanded his client roster to include agencies such as Unit 9, Fred & Farid and 84.Paris, and worked for prestigious brands including Audi, Dior and Chanel. He co-founded Bonhomme with the ambition to invent a creative structure that helps to modernise the relationship between advertisers and digital.

To stand out in what is now a crowded space, Morgane and Emmanuel needed to name their studio. Emmanuel explains their approach: "As far

WHO
Bonhomme

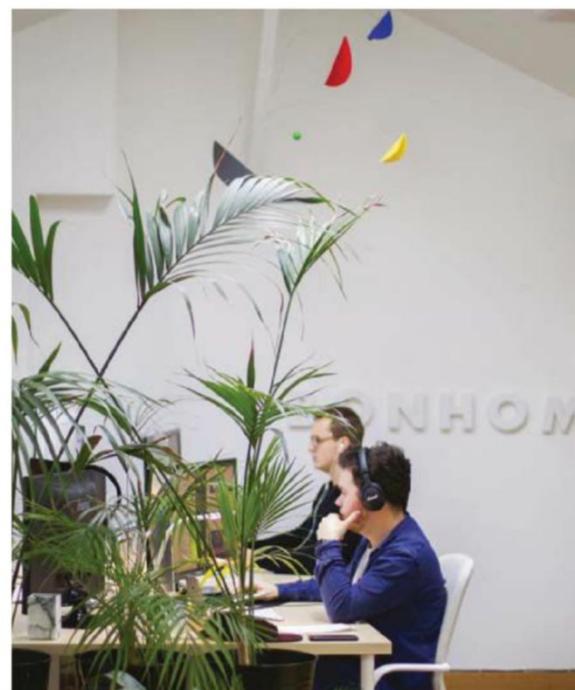
WHAT
Art Direction, Front end Development, Creative Direction, Interactive and Motion Design

WHERE
10 rue Vicq d'Azir 75010 Paris, France

WEB
bonhommeparis.com

KEY CLIENTS
Sonia Rykiel

Festival de Cannes
Audi
Tag Heuer
Dom Pérignon



as we are concerned, the name existed before the studio. Bonhomme is a kind word in French that means "buddy" or "big man". It's often used when adults talk to children. Morgane and I have known each other for a long time. She often used bonhomme as a sort of nickname. We were really surprised to see people's reaction. Many of them thought it was funny that she calls me that. When we started to seriously think about our own studio, that name seemed the obvious choice.

"Bonhomme.com did belong to another company, and at the time, we thought that using the name of the city we are working in, was a way to explain to our potential clients that we were

Bonhomme is a kind word in French that means "buddy" or "big man". It's often used when adults talk to children

easy to meet. Obviously, a lot of people think that the name of the studio is "Bonhomme Paris", but we don't really care. Whatever your clients think your name is, the more important thing is that

they want to work with you."

The web presence that Bonhomme has developed speaks directly to its design aesthetic. White space, and bold monochrome typography gives a clean feeling to the site. "Having an up-to-date website should be a priority," says Morgane. "Our field is changing so fast that it is important to show that you're moving with it. In an ideal world I would love to employ someone to spend all his week designing case studies, and all the things that can show the world the direction we are travelling in, but our resources don't permit this. Our site is about how we can collaborate with our clients and partners to create ►

**MORGANE URBAIN**
Producer and Co-founder

"The only advice I would give to anyone looking to take a step into the industry is to do things over and over again until they see their project come into focus. Just try things out – anything that comes into your mind. Don't hesitate. Don't judge your work. Just beginning and trying your best is enough."

meaningful messages."

As a relatively new studio, the range of clients Bonhomme has been able to work with is testament to its talent as a digital artisan. As Emmanuel comments, it has been fortunate to have a steady flow of clients since Bonhomme opened its doors: "We have been lucky to gain one client after another over the last four years. We try to keep our portfolio up-to-date and to communicate on social networks, hoping that potential clients would come across our work. The main source of traffic comes from award websites, which offers us some visibility. I don't believe that we have a real reputation yet, but we have had enough luck to maintain a certain level of work across our studio."

The factors that a studio uses to decide which clients to work with can be many. For Bonhomme, the time and resources it has available are often the determining factor, as Morgane explains: "We try to stay open-minded at the studio. Every

project has the potential to become interesting. Because we are small we are quickly overbooked. This is the reason why, most of the time, the schedule defines whether we can or can't work on a project. The only projects we choose not to work on are the ones that have political or religious intentions. For many reasons, we refuse to contribute to those kinds of projects."

What defines a studio? Its design direction and its overall ethos can be difficult to define. For Bonhomme, getting to the underlying meaning of a project and how this will be expressed is how it always approaches new work. "We are particularly proud of our last project," says Emmanuel. "Duroc is a subsidiary of Delassus Group, specialising in small tomatoes produced in Morocco. In order to be recognised by its clients as the most creative business in this field, the company asked us to design an outstanding website."

Emmanuel continues: "While we were asked to think about how the site could be made

Driven by a digital DNA, we have had the chance at Bonhomme to move from websites to brand identity, design logotypes, take pictures and make movies

interactive, our first concern was the content. We knew that talking about our client's firm would be really challenging, so we tried to find a way to make this brand stand out. Colours and minimalist shapes were used to build an identity that's really unexpected for a tomato producer.

"The main idea was to produce assets that can work separately as well as together. Fifteen scenes were imagined that together tell a story, that comes to life in a movie. The most challenging aspect of the project was to find a way to deliver corporate information while keeping an entertaining tone of voice. This is actually what drove us to design such modern and colourful scenes.

"Driven by a digital DNA, we have had the chance at Bonhomme to move from websites to brand identity, design logotypes, take pictures and make movies. Being identified as a partner that can give life to a brand, regardless of its market, is really fulfilling for us."

What about trends across the design landscape? Do these influence Bonhomme? ▶

BONHOMME

AGENCY BREAKDOWN

2 CO-FOUNDERS

3 ART DIRECTORS / MOTION DESIGNERS

2 FRONT-END DEVELOPERS

1 PROJECT MANAGER



TOP Visitors to events can capture their emotions and memories using special interactive kiosks

MIDDLE Memories converge to create a digital landscape that Bonhomme curated to deliver an emotional connection within the immersive space

BOTTOM Compiled into a wave graphic that connects to the Coca-Cola brand, Bonhomme takes graphics and images to create a new engaging space

COCA-COLA

As an official partner of the new AccorHotels Arena, Coca-Cola aims to highlight the emotions experienced during the various shows. Bonhomme implemented a unifying and immersive concept to help Coca-Cola support its key goals.

The project started with the idea that Coca-Cola is the partner of your best memories. AccorHotels Arena is the place you can see your favourite singer as well as the best sports competitions. We thought of a digital wall on which people share the pleasure they feel during the event. The huge screens that people interact with eventually take their memories and using the idea of the wave that symbolises Coca-Cola, they can see all their shared memories.

People discover all the memories shared by the visitors through this wall. Thanks to two interactive kiosks positioned around the wall, people can take several pictures, gathered into a gif, dedicated to their memory. All of these memories are placed onto a website, which enables visitors to the site to live the same experience.

Making this interactive experience was really didactic. While we are used to watching Google Analytics to understand users' experiences, we had the chance for once to see live experience and adapt the interface during a beta testing period. Working for real people, for the real world is an idea that we love at Bonhomme.

DESIGN BUDDIES

DUROC

duroc.ma/en

Duroc is a subsidiary of Delassus Group, specialising in tomatoes produced in Morocco. It is the tomato market leader in England. Bonhomme helps Duroc in the creation of digital content, as well as the creation of its digital support for its customers and partners.

Our approach was to ensure that all of the digital assets we created were modular in that they could be used in isolation and also together. Over all we created fifteen scenes that can be used together to tell a story.

The website was the best place to use those assets. We believe that there are two kinds of people on a website: Firstly, the ones that are looking for information. Secondly, the other group are much more connected to emotion, instinct, and they love entertainment. This is the reason why the site relies on a mosaic that gives an impression of infinity. People can play with this grid and discover the content that appeals to them. For those who're looking for answers, a traditional navigation enables them to go straight to the information they are looking for.

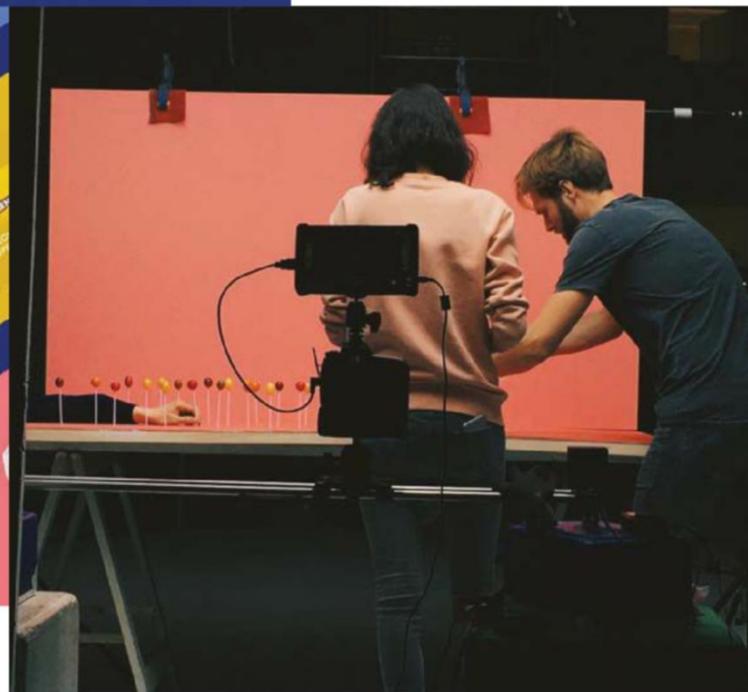
Making a project for a huge sport company that delivers amazing photography or videos is not that challenging. But being able to keep visitors interest in a website, talking about small tomatoes with such a poor identity is a real challenge. Everyone at Bonhomme had a real enthusiasm for this project, in spite of the fact that it wasn't the most appealing brand we had to work with. We are really proud to be identified by brands that share our vision, and that offer us the chance to help them.



TOP AND LEFT Using interconnected static and motion graphics gives this site the dynamism Bonhomme had conceived for Duroc

BOTTOM LEFT Clearly it's the striking colour palette that Bonhomme chose for this site that makes each of the assets highly engaging

BOTTOM Filming tomatoes is not a typical day at the office for Bonhomme, but one that it relished as part of the creative process



Emmanuel continues: "We are following certain trends as our clients are looking for modern graphic design for their campaigns. Web design is a newborn, the graphic codes and the media of diffusion are numerous and change very quickly. This is the reason why we have to keep an eye on trends. However, we try to add something to the things we like, to be part of their evolution. We ask ourselves: what could make our work a unique production? Above all, we try to break down the barriers between the applied arts, to infuse in our work the influences of all disciplines including architecture, fashion, object design and graphics."

In addition, Emmanuel explains how he believes that being based in Europe has an impact on Bonhomme's design sense: "France has its own graphic culture. Well-known in music, fashion and the film industry, it has a huge impact on design. But what has been a real benefit to French design is the influence from Eastern Europe. For years, graphic design has followed trends and best practices spread by a lot of countries including Germany, Switzerland and Sweden. Typography such as the Helvetica font had a huge impact on design, as did grid systems taught by Joseph Muller Brockmann. It has completely revolutionised designers' vision of posters and typography. I believe this Eastern European influence gives Bonhomme a different approach to design. In Europe, we think of colours, typography and grids."

Bonhomme takes a similar approach to the diverse range of projects that it has worked on, as Emmanuel explains: "Every project that comes to the studio is handled the same way. First, Morgane and I work on the perimeter. Some of our clients contact us with a specific idea without understanding what they really need. So, to begin with, we design a proposal. During this phase, we start by submerging ourselves in our client's field and try to identify how they want to communicate with their clients or business partners.

"Then, we define the creative direction and one of our Art Directors starts the design. They are in charge of the whole project. From the UX and UI, our creatives are multi-disciplinary. We expect them to handle UX schematics as well as brand identity assets, such as logotypes, motion-design videos or websites. We try to break the walls that are supposed to separate a Project Manager from an Art Director or a Developer. Everyone is free to give his opinion and his point of view on the project.

"Our designers create user journeys, motion-design videos that show how users would interact with the website. Our developers are specialised in frontend development and are the guarantor of the quality of our work. Because technologies evolve every day, they have to reinvent their job

Our developers try to avoid frameworks and existing libraries. This obviously comes from the fact that we try to ensure every project is unique and show we have pushed our skills

all year long to stay up-to-date. Development complexity is what defines most of the timeline of a project. Depending on the needs of a client, a project can jump from a month to six months, and can use between one and three developers."

Today with such a wide range of available tools, Bonhomme still favours some of the market leaders. Morgane explains the studio's favourites: "Designers at Bonhomme work with Adobe's tools, particularly with Photoshop and After Effects. Those are really flexible. Photoshop can be used for web design, as well as picture editing. So far, it's the best software I know. Our developers try to avoid frameworks and existing libraries. This obviously comes from the fact that we try to ensure every project is unique and show we have pushed our skills."

Current technologies that are now the basis of the web have been evolving for over a decade. Bonhomme take a different approach when considering how these technologies might develop, as Emmanuel comments: "I began this job at a time where Flash was the only thing to learn. Then Apple decided to kill this technology, and it took about five years to do in HTML what we were able to do in Flash. The funny thing is that I know a lot of people that are still programming in Flash for installation or VR experiences. What matters to us is knowing what UX will be like in the next few years. Will websites exist? Will the internet we are used to interacting with be the same? Even more, will desktops still exist?

"Most of our clients know that a huge proportion of their visitors come from mobile devices. However, all of them spend much more time working on the desktop version of their sites or assets instead of thinking about what the future will be like. And to be honest, people generally get bored when we present the mobile version.

"As far as we are concerned, we believe that mobile should be the starting point of a project, for clients have a big part of their traffic on mobile. Then, when the UX is completed, you can start to ►

TIMELINE

2013

Bonhomme launches its first project for Volkswagen, an interactive website base on a 3D movie.

Employees: 2

2014

Brice Darmon's website received five awards including Website of the Year on CSSDA.

Employees: 3

2015

Bonhomme leads Coca-Cola's project at the AccordHôtel Arena, a huge interactive wall receiving more than 2,000 visitors per day.

Employees: 5

2016

Bonhomme ends its collaboration with Le Festival de Cannes after a year of work to launch five websites and a mobile application.

Employees: 6

2017

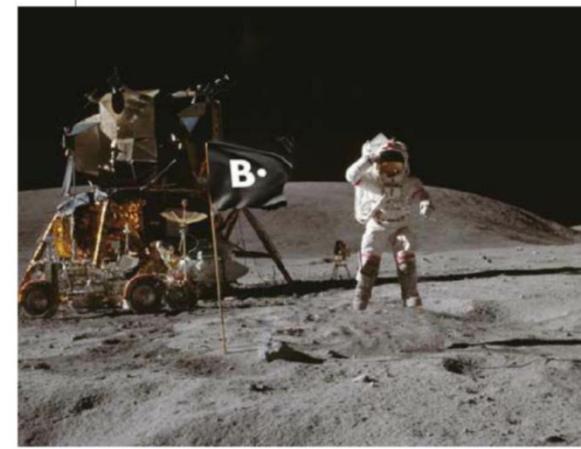
After four years of hard work, Bonhomme has received over 50 digital awards, making it one of the more awarded studios in Paris.

Employees: 8

2018

Bonhomme hopes to be the first Branding Digital Studio to put its footprints on the moon ;)

Employees: 8



FESTIVAL DE CANNES

festival-cannes.com/fr

On the eve of its 70th anniversary, the Festival de Cannes wanted to overhaul its digital ecosystem. We helped them to create a digital identity that was usable on all the festival's interfaces including its corporate website, site events, WebTV, and a mobile application. We produced a working design, UX and UI to make those interfaces as accessible as possible.

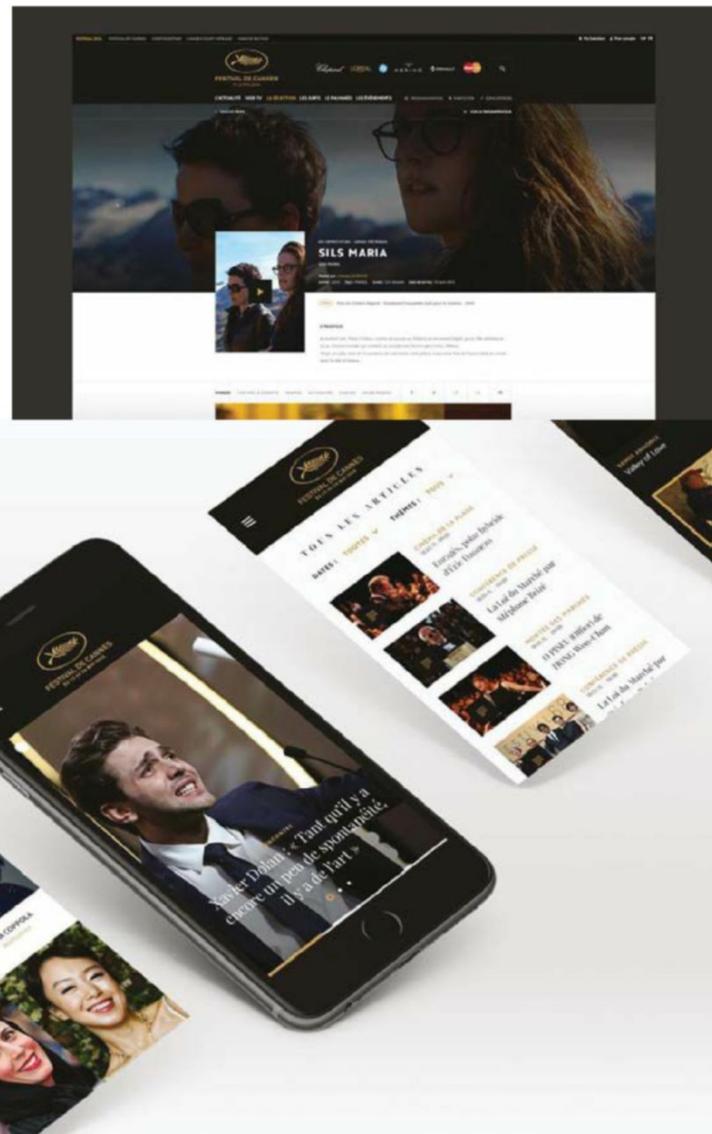
One of the challenges was to deliver specific UX and UI for each dedicated channel and its specific users. An important component of this was the work we did on the user experience. We needed to ensure that people – depending of their needs – could easily find what they were looking for. Journalists, festival-goers and film-makers use the website for different reasons. Each of them were studied to design a polymorphic interface that supported everyone.

TOP The design for the Festival de Cannes had to ensure it was engaging on several platforms and have a lasting legacy after the event took place

RIGHT Developing a content strategy to showcase the rich content available on mobile devices illustrates Bonhomme's design prowess

The second thing that had a real importance in our collaboration is that the Festival isn't just the events itself. It was vital that the site supported the event as it happened, but also that the site and its assets, had a legacy throughout the year. We adapted its UX to ensure the site could be navigated no matter the time of year, who was visiting it, or using whichever device.

We planned a multitude of workshops to structure the website architecture and its content. At Bonhomme, we don't have clients; we try to be part of the brand's team to collaborate closer with them.



EMMANUEL CRUELLAS Creative Director and Co-founder

"The thing that excites me the most is the idea that in five years Bonhomme will have the same people working in the studio," concludes Emmanuel. "I really see this structure as a rock band. The more we play together, the better we get. It's really, really hard to find people that love to work together, and that have a true esteem for each other. And I think that our team could become like a good bottle of wine; the older it gets, the better it is."





BONHOMME

<https://bonhommeparis.com>

FOUNDERS

Morgane Urbain and Emmanuel Cruellas

YEAR FOUNDED

2013

CURRENT EMPLOYEES

8

LOCATION

Paris, France

SERVICES

Ideation

Identity Systems and Guidelines

UX, Motion and Visual Design

Web Design

Full-stack Development

think how the experience should be different on other devices. Users have different habits on mobile and desktop. This is the reason why I think we should design different experiences.

"This technique is at the heart of our production. Taken individually, it is neither a problem nor an obstacle. What looks like a time-consuming factor today, is actually a multiplicity of technologies and communication media.

"Making a website is more complex today than it was when Flash was more dominant. Today, this means creating code compatible with multiple browsers that do not tolerate the same technologies in the same way. And of course, when you add in mobile, this just makes the landscape even more complex to manage."

What the future of design could look like is open to debate. However, as technologies develop and the social impact of technology and communication also evolve, what does Bonhomme think the future looks like? "We love the work Google has been doing for years, and the development of Progressive Web Apps," says Emmanuel. "Google's ticket-booking module, for example, or city tour, is very rich and very ergonomic. I think the question should really extend to the web in general. In a few years, will brands still need a website? Will their presence not be reduced to a Facebook or Instagram account, and an excellent SEO on Google? Let's talk about it in five years."

The team at Bonhomme remains small. When it comes to adding a new team member, Morgane outlines how the studio's approaches recruitment: "I guess I expect people to be as passionate as I am. We are fascinated by this industry at Bonhomme. Every pixel on a screen, any colour, or any movement in a motion design, can provide us



We believe that perfection lies behind the details

much more fulfilment than anything else. We believe that perfection lies behind the details. So, I ask any prospective employee if they truly love digital. Our days are filled with understanding and then expressing our clients' project needs, something which we work on again and again. You need to love digital to work at Bonhomme or your days will be very long."

"The only advice I would give to anyone looking to take a step into the industry is to do things over and over again until they see their project come into focus. Just try things out - anything that

comes into your mind. Don't hesitate. Don't judge your work. Just beginning and trying your best is enough."

With an ambition to redefine what the digital space is, with innovative designs that push the communications envelope, Bonhomme is breaking the boundaries of what digital design means today. "The thing that excites me the most is the idea that in five years Bonhomme will have the same people working in the studio," concludes Emmanuel. "I really see this structure as a rock band. The more we play together, the better we get. It's really, really hard to find people that love to work together, and that have a true esteem for each other. And I think that our team could become like a good bottle of wine; the older it gets, the better it is." ■



SEO TODAY



"SEO is evolving quickly and becoming even more competitive. With a history of changes, algorithm updates, new technologies and personalisation, you need to adapt quickly to stay ahead."

Paul Betteridge
Head of Digital Marketing at Hampshire SEO
www.hampshireseo.co.uk

AN ESSENTIAL COLLECTION OF THE LATEST TIPS, TECHNIQUES AND TOOLS TO SEND YOU ON THE PATH TO SEARCH SUCCESS IN 2018

With the ever-changing nature of the Google algorithm and the need to keep up to date and informed with the latest SEO techniques, the following tips, techniques and tools will help you have a better understanding of what the search engines are looking for in 2018, and how you can ultimately improve your rankings.

This essential collection will align your focus on what actions are needed to achieve those gains, effectively future proofing your website without being penalised.

Highlights include how to optimise for both your audience and the search engine crawlers; getting to grips with technical SEO; website

speed; mobile optimisation; and making it easier for the search engine crawlers to understand who you are and what you do.

We'll look at the power of on-page SEO, plus the use of primary and secondary keywords and how utilising them with content relevancy can help you grow your audience. Truly understanding who your key audience is should be at the forefront of your decision making. We'll help dig that bit deeper for you.

Other opportunities include schema data, featured snippets, voice search, visual content, social signals and links. All play a part in increasing your search success. Read on and stay one step ahead.

01

SET YOUR SEO FOUNDATIONS

For your website to rank successfully in 2018, On-Page SEO is an absolute must. Focus on optimising your metas such as your titles, descriptions, headers and image alts with your targeted keywords and phrases. Combine this with a good URL structure and content relevance to the targeted search term or phrase.

It's also the perfect opportunity to include location, product and service terms within these metas. This is a compelling reason to choose you over the competition (the meta title and description being key to encourage the click).

03

HELP THE SEARCH ENGINE SPIDERS

Search engines will not rank your site unless they can find it, so it's therefore extremely important to make sure search engines are able to discover and crawl your website's content quickly and easily. Without it, nothing else really matters - we want search engine spiders to successfully crawl and access our URLs, with the intention to crawl and parse our website content.

Keeping a Logical Site Structure, resolving 404 errors and utilising 'rel=next' and 'rel=prev', will help with this greatly. Make it easy for the search engines to add your pages into their web index.

05

SET PRIMARY AND SECONDARY KEYWORDS

Ranking for the right keywords can make or break your website. By researching your market's keyword demand you can not only learn which terms and phrases to target with SEO, but also learn more about your customers as a whole. Select a primary keyword and a set of related secondary keywords that share your searcher's intent (understand what their motivation is). The intent behind these keyword terms and phrases should be the same, so the same content can ultimately serve it. Employ these primary, secondary and related keywords in the page's content, metas and links.

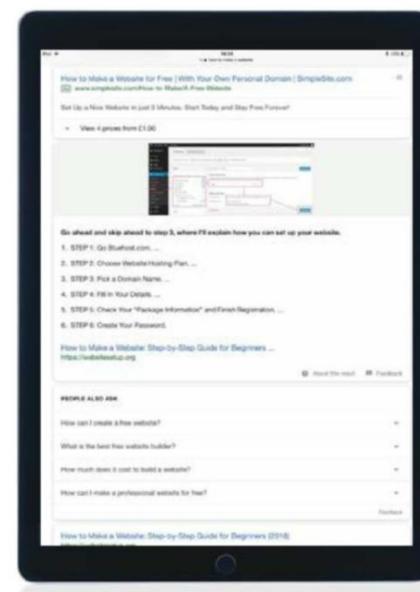
02

STAND OUT MORE WITH SCHEMA

Future proof your website SEO and stand out more in 2018 with Schema data markup. Schema markup is used to tag entities in your pages and content; this includes products you sell and services you offer. It makes it easier for search engines to understand who you are and what you do, but more than that it



enhances your potential visibility and search engine real-estate. Add this to your HTML, via the Data Highlighter in Search Console and improve the way your pages are represented in SERPs. You can find out much, much more at www.google.com/webmasters/tools/home.



04

LEVERAGE RICH ANSWERS

Rich answers can provide your website with a new chance of page one visibility. A rich answer is a snippet that contains a brief answer to a search query; it appears above other organic search results and thus enjoys more exposure. In addition to the featured snippet, the rich answer box will provide a direct link to your page from where the answer came. Identify questions you might answer on your website (a good example is FAQs). Make sure that you create answers that are informative and engaging and then amplify its reach to increase your chances of appearing.

06

HAVE EMPATHY FOR OUR AUDIENCE

One of the most important elements in building an SEO marketing strategy in 2018 is empathy for your audience. Once you grasp what your target market is looking for, you can effectively reach out and keep those users. Therefore the main focus of your content should be aimed at the audience group that contributes to the success of your business.

Reach the right customers with the right keywords.

Get keyword ideas to help build your campaigns with the AdWords Keyword Planner.

[START USING KEYWORD PLANNER](#)

Develop your content and focus, by outlining what your target customer is thinking during their buyer journey. This includes their Awareness stage (inspiration-pain points), Consideration stage (research-comparison) and Decision (purchase-advocacy).

READ MORE: <http://bit.ly/2D7oRtI>



Don't get left behind in 2018 – our online interactions have evolved to become far more visual. There is a general public desire to engage with more images and videos, which has led to a surge in visual online interactions. Your audience will expect you to offer video content, so show them your personality as

you connect with information that meets their needs. This will ultimately have a positive impact on reach, engagement and conversions.

YouTube is the obvious choice, so create and optimise content specifically for this channel and you will ultimately own more Google real estate.

07

VIDEO ENGAGEMENT AND EXPECTANCY

08

TAKE ADVANTAGE OF SERP FEATURES

Increasingly, SERP features such as local packs and featured snippets are taking away searchers attention and clicks from organic listings. With the evolution of SERP features and a more personalised approach to search results, you cannot solely rely on a page 1 ranking to get you as much traffic as possible. Investigate the SERP and review what Google believes is the most relevant to your keyword search. Can you position yourself in the local pack? Can you get a featured snippet for this query? Can you add any structured data?

READ MORE: <http://bit.ly/2GhSHx5>

09

UX METRICS AND PERSONALISATION

UX-related metrics have made their way into Google's ranking algorithm. For example, website speed, mobile-friendliness and the HTTPS protocol. Your website audience will also demand a more personalised experience; provide this to them with content-centred experiences and personalisation (such as chatbots and push notifications).

Improve your audience's experiences and embrace the future of voice-activated interfaces and augmented reality. Having the right mix of content and technology should be a big part of your website design and user experiences (UX) in 2018.



10

VOICE SEARCH OPTIMISATION

Voice Search is too good to miss, a recent Google report highlighted 55% of teens and 40% of adults use voice search daily. This is an exciting opportunity for businesses and marketers to increase conversion and its actively shaping the future of local SEO. As it's rapidly becoming the way customers will find your business, and it recognises 'near me' searches, ensure your Google My Business listing is up to date (as this populates the results). Also ensure your website is mobile friendly and optimise for natural language queries (long-tail keywords, full sentences and questions).

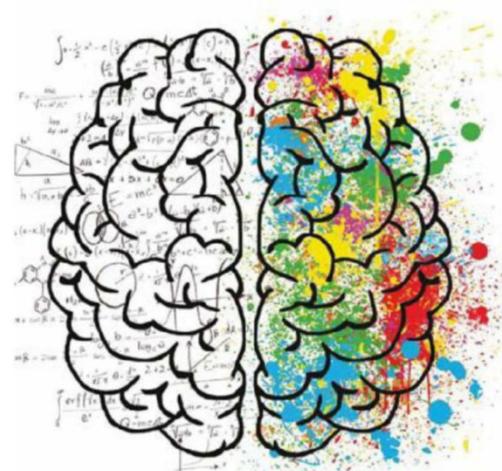
11

SETTING A NEW PRECEDENT FOR SEO

RANKBRAIN USES AI TO FIND PHRASES AND TERMS THAT ARE SIMILAR TO WHAT A USER HAS TYPED IN ORDER TO BETTER UNDERSTAND WHAT THE INTENT IS, RETURNING RELATED RESULTS

As Google refines its algorithm, RankBrain is becoming even more important in how a website is ranked in 2018. RankBrain uses AI to find phrases and terms that are similar to what a user has typed in order to better understand what the intent is, returning related results. RankBrain is constantly teaching itself by paying attention to certain metrics such as bounce rates, time on page and CTR. Make the most of these metrics by increasing your average dwell time and CTR with engaging and unique content.

READ MORE: <http://bit.ly/2hWVYaA>





12

CONTENT IS STILL KING

In today's content driven world of digital marketing, it's a marketer's goal to persuade a content saturated audience to consume their content, share it with their network, or make a purchase. Your website copy and language will steer your audience to take action, to click a button to buy, download or move onto the next step.

Content is therefore paramount, although it's going nowhere unless people can search and find it. It should be written with an understanding of factors such as SEO, latent semantic indexing and aimed at the right audience groups.

Create useful, expert-level unique content and present it in the most engaging form possible.

Credibility and relevance is key for your audience and its success in search, making amplification, link building and social sharing more likely.

When writing your content serve the searcher's goal (keyword research on terminology) and our own goals (contact/conversion/advocacy), giving them a reason compelling enough to take action by adding value and relevancy.

You also need to be aware of the pitfalls, such as Google's Panda ranking algorithm update, which sifts out pages with thin, non-authentic, low-quality content. De-duplication of content (canonicalisation) will also help, as well as preventing cannibalisation and making better use of your crawl budget.

ESSENTIAL TOOLS

TAKE THE MOBILE TEST

<http://bit.ly/2rQpFRI>

Test how easily a visitor can use your page on a mobile device by using Google's mobile-friendly test. Simply enter your page URL to see how your page scores.

TEST YOUR PAGE SPEEDS

<http://bit.ly/1RBjMsy>

Enter a webpage URL into Google's PageSpeed Insights tool for it to carefully analyse the content of your web page. It will then generate suggestions of how to make that page faster.

ADD SCHEMA DATA

<http://bit.ly/1UjXtMX>

Use this guide to provide explicit clues about the meaning of your website pages to Google. It will help you to include structured data on your pages.

KEYWORD RESEARCH

<http://bit.ly/2gt3tmS>

Reach the right customers for your website with the right keywords. Get ideas to help build your campaigns with the AdWord's Keyword Planner.

MAINTAIN YOUR GOOGLE-FRIENDLY WEBSITE

<http://bit.ly/17WOXzv>

Get the data, tools and diagnostics needed to create and maintain Google-friendly websites with Google Search Console. Analyse clicks, get alerts on errors and test if Google understands your content.

13

BUILD CONFIDENCE WITH SOCIAL PROOFING

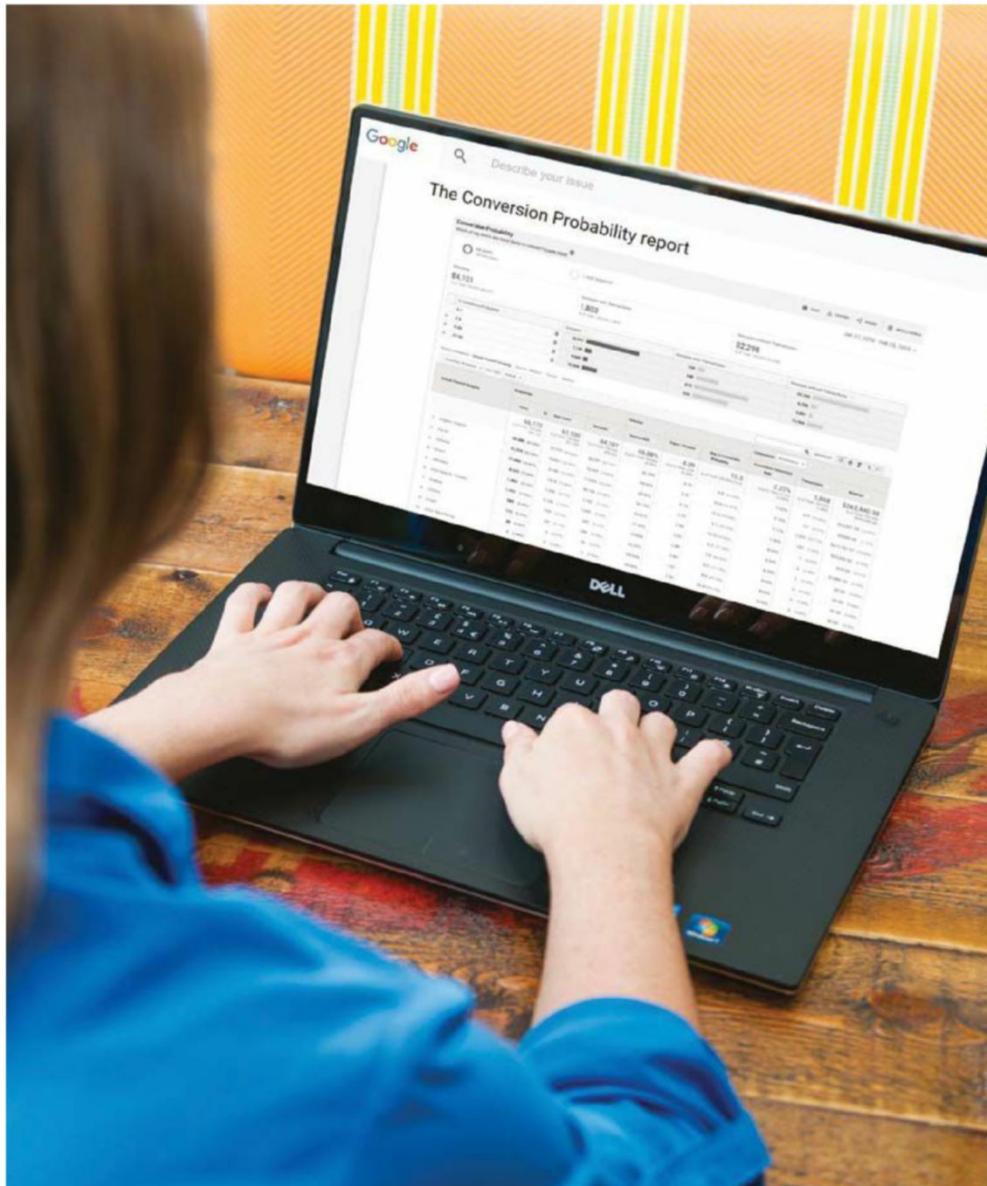
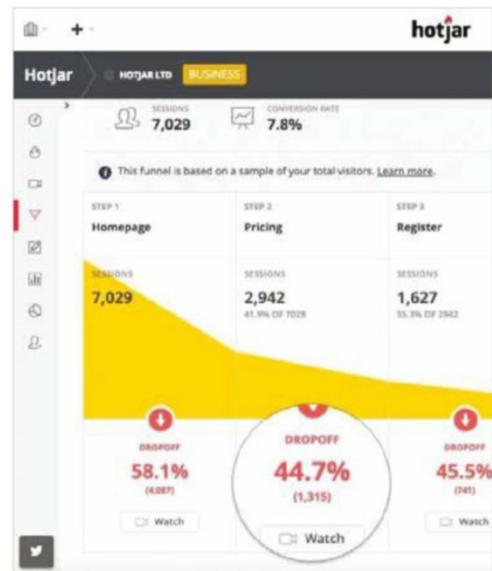
Social proofing is another trust signal that builds confidence with your website visitors. It highlights how other like-minded individuals have purchased your products or services and are happy with their decision to do so. Consumers are actively looking for social proof statements, and they are actively impacting their decision-making process. Focus on eliminating any fears and overcoming objections. Add testimonials, reviews, number of items sold, the number of customers purchased in the last hour/day/week on your site. All will have a positive impact on sales and conversions.

14

CONVERSION RATE OPTIMISATION (CRO)

Convert more of existing traffic by focusing on what you already have. Analyse your user's experiences and identify any obvious barriers for conversion. Then ease the journey of your audience so they can quickly and confidently reach a converting decision.

By increasing the percentage of website visitors who take your desired action, we increase both micro and macro-conversions. Micro-conversions are your smaller conversions, such as filling out a form or signing up to an email. And macro-conversions are your primary goals, such as your users purchasing a product or service.



15

NEW ADVANCEMENTS IN GOOGLE ANALYTICS HELP YOU UNDERSTAND YOUR USER'S JOURNEY BETTER

There is more of a focus on user behaviour in standard reports, and Google will continue to improve the accuracy of user-level data. Take advantage of the default option to analyse users alongside sessions. Google has also added a new tool to help us analyse visitors individually, enabling you to measure lifetime metrics and dimensions for individual users. Providing more accurate data than before, you can see a variety of data on past (historical), present and predicted future behaviour. This helps you to deliver a better experience for your users, tailoring your marketing activities and messaging accordingly.

Audience Reporting is another new feature that enables you to connect your data more clearly, allowing you to see trends and opportunities. It uses Audience as its primary dimension and permits you to compare performance across different segments. Use these audience lists in your CRO tests.

Conversion Probability is a beta feature that uses machine learning to provide more analytic intelligence on users who are most likely to convert, which you can then use to create audiences for analysis and remarketing. Take advantage by creating re-marketing lists that target prime users that have a higher probability of conversion. Focus on reaching out to these audiences via AdWords. Find out more at <http://bit.ly/2DOOA6F>

16

LOCATION, LOCATION, LOCATION

One out of every three searches on mobile is related to a location. Given the sheer dominance of mobile it's imperative to capitalise on these audiences locally. Good marketing does not attempt to reach everyone; its targeted to connect with a few specific and defined audiences - 'local' being one of them.

Google My Business (previously known as Google Places - find at www.google.co.uk/business) capitalises on this local buying intent and enables you to boost your visibility. Optimise your Google My Business listing and create incentives to get more positive reviews and citations across the web.

18

REVIEW AND ANALYSE YOUR COMPETITORS

Competitive analysis is a critical part of any 2018 marketing plan. Insight into what your competitors are doing well can help you capitalise on it. Add yourself to their email newsletters, social platforms and set up Google Alerts. Review the top performing websites around your targeted keywords; you can take advantage of what works and replicate it.

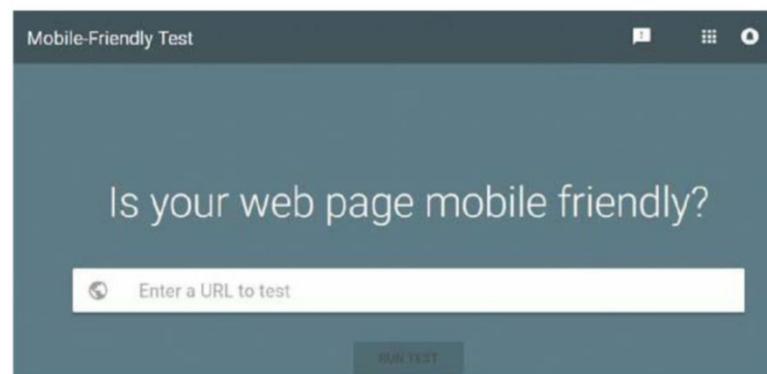
Understanding the competition's focus on content and keyword terminology will also help to close the gap between those terms you are not taking advantage of and ultimately increasing traffic to your own domain.

DON'T BE IGNORED START BY OPTIMISING YOUR LANDING PAGES, ITS CONTENT, METAS (TO INCLUDE TITLES AND MORE IMPORTANTLY META DESCRIPTIONS) AND BY REACHING OUT TO YOUR NETWORK

17

MOBILE FIRST IN 2018

You cannot ignore the importance of mobile SEO. The statistics already show that over half of Google searches come from a mobile device - and this is only going to increase. If your site pages are not optimised for mobile, they are likely to be discarded from mobile search results. With the rise of voice



search, the impending mobile-first index and mobile-friendliness being a ranking factor, you need to focus on improving the responsiveness of your pages for all devices. Your content should also be consistent across desktop and mobile, as mobile will be favoured.

19

IMPROVE AND OPTIMISE YOUR SITE SPEED

Google has officially confirmed that it uses page speed in its ranking algorithm. Not only is it a ranking signal, it's also a major UX factor. In turn this impacts your website rankings and usability.

Focus on optimising your pages to load faster (take Google's Page Speed Test, which you can find here - <http://bit.ly/18oQqhi> - to determine what can

PageSpeed Insights

be improved). Opportunities can include optimising images and reducing server response time.

You can also consider implementing AMP (Accelerated Mobile Pages), a slightly-controversial Google initiative to help build a more user friendly mobile web. Get an in-depth look at www.ampproject.org.

20

IMPROVE CLICK THROUGH RATES (CTR)

Striving for greater engagement and higher click-through rates will bring your website better rankings as well as indirect SEO results in the form of attracted links, shares and mentions. A focus for 2018 should be on encouraging engagement - your results need to stand out from the crowd.

Don't be ignored - start by optimising your landing pages, its content, metas (to include titles and more importantly meta descriptions) and by reaching out to your network. Establish connections with influencers - anything you can do to raise interest among all the noise and encourage the click.



ESSENTIAL TOOLS

CONVERT MORE OF YOUR EXISTING TRAFFIC

<https://www.hotjar.com>

Utilise heat-mapping tools to review the reading and scanning habits of your users. By understanding how they interact with your pages you can start the conversion rate optimisation process better.

LOCAL MAP LISTINGS IN THE SERPs

www.google.com/business

Attract new customers with your free Google listing. Claim it and then optimise. Without it you won't appear on the local map listings in the SERPs.

REVIEW AND BUILD ON YOUR LINK PROFILE

<http://bit.ly/2IOEy9f>

A powerful website backlink checker that shows all links pointing to your domain and their quality, as well as helping you to discover everything about your competitors' backlinks.

GOOGLE ALERTS

<https://www.google.co.uk/alerts>

Monitor the web for interesting new content and mentions of your business and that of your competitors. Simply create an alert, and have it emailed to you directly.

GOOGLE ANALYTICS

<https://www.google.com/analytics>

Measure your website, app, digital and offline data to gain valuable customer insights into the impact of your marketing activities.

The screenshot shows a Google My Business listing for 'Travelodge London'. At the top, there's a photo of the interior lobby with a large 'Google' mural on the wall. To the right, there's a map pin indicating the location at 'Dominion Theatre' on 'Tottenham Court Road'. Below the photo, there are two buttons: 'See photos' and 'See outside'. The main card displays a 4.6-star rating from 820 reviews. It includes the address '1-13 St Giles High St, London WC2H 8AG', operating hours ('Closed · Opens 8AM Mon'), and a phone number ('020 7031 3000'). There are also 'Suggest an edit' and 'Know this place? Answer quick questions' buttons.

21

ONLINE REVIEWS: INCREASE TRUST

Consumers are actively using these review scores more than ever in their decision-making process. Focus on actively gaining, monitoring and responding to reviews on social platforms, in particular 'Google My Business' as these are shown in Google results when searching for your business.

Encourage users to leave a review, as well as commenting on your blog - this will build an active community for your brand and website. It will also help you to have a better understanding of your user's experiences, take what works and expand on it further.

22

BE CONSISTENT

Clean and own your business information and data across the web, including mentions of your business's name, address and phone number (NAP) on other webpages. Being consistent is imperative, across these citations and all other touch-points such as: natural landing pages, Google Maps, schema data and paid ads. These all rely on the accuracy of the information they are being fed, and if not done correctly, can have a negative impact on your brand and awareness. Review placements on localised and industry directories, affiliates, referrals, blogs, exhibitions, shows, awards and so on.

23

INCREASE YOUR SEARCH ENGINE REAL ESTATE

Bully the page and push the competition down in 2018 by supporting your natural SEO efforts with a paid campaign across Google and Bing. Attract the right customers immediately, with the products and services they are actively searching for.

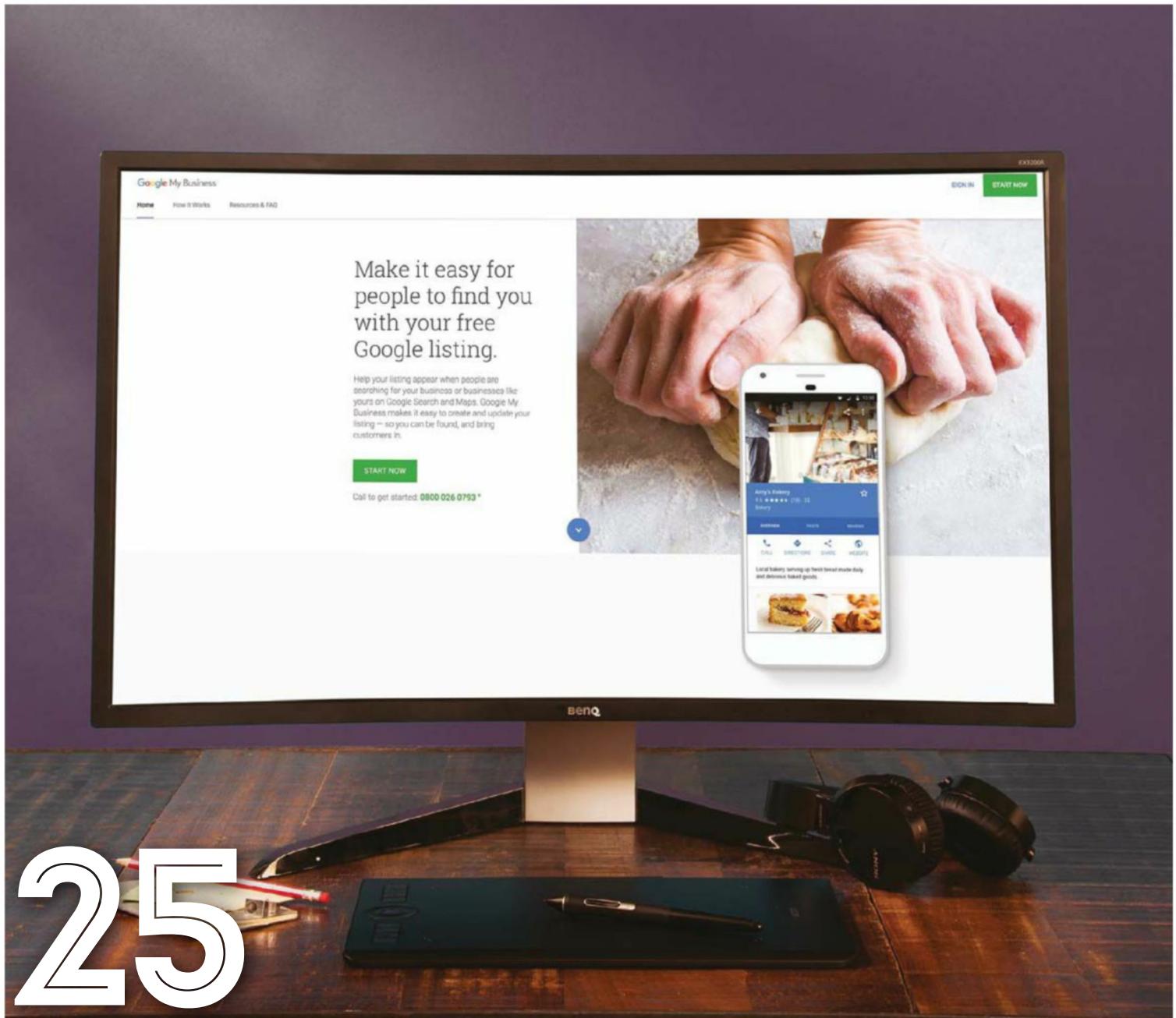
Keywords are still the primary, but voice search will start to impact terminology as well as a real focus on audience insights. Keep an eye on the next 12 months for further exciting opportunities, in particular how Google Ads will connect us to more and more businesses. **READ MORE:** <https://support.google.com/adwords>

24

EARN SOCIAL SIGNALS

Keep on top of your wider web presence and drive extra traffic and search engine real estate with the aid of social. For example, your Google+ posts or tweets can make it to Google's organic search results, which is a great opportunity to drive extra engagement and more page placements in search. Attract social links and shares with viral content that is easy to share. Add social buttons to your pages and use Open Graph metas. And don't forget to engage - this is a primary opportunity to build relationships and nurture buyers and advocates.

Tag	Description
og:url	The canonical URL for your page. This should be the undecorated URL, without session variables, user identifying parameters, or counters. Likes and Shares for this URL will aggregate at this URL. For example, mobile domain URLs should point to the desktop version of the URL as the canonical URL to aggregate Likes and Shares across different versions of the page.
og:title	The title of your article without any branding such as your site name.
og:description	A brief description of the content, usually between 2 and 4 sentences. This will be displayed below the title of the post on Facebook.
og:image	The URL of the image that appears when someone shares the content to Facebook. See below for more info, and check out our best practices guide to learn how to specify a high quality preview image.
fb:app_id	In order to use Facebook Insights you must add the app ID to your page. Insights lets you view analytics for traffic to your site from Facebook. Find the app ID in your App Dashboard.



25

CONTENT AND LINKS GO HAND IN HAND AND WILL BE AT THE HEART OF SEO IN 2018

The number of backlinks and linking domains is still a major ranking signal for 2018, and will have a massive impact on your ranking potential. The more high-authority backlinks and quality citations a website has across the web, the better positioned it is to gain higher search engine rankings and expand its overall reach.

Before reaching out, be mindful of the Google Penguin ranking algorithm by ensuring your backlink profile looks natural. Authoritative links should be earned through others quoting, referring to or sharing your content. It's also prudent to run regular link audits to spot any dangerous links early on, and have them removed in time.

Start by claiming your local business listings, and adding yourself to the local directories. Content then follows; without amazing content, you will never get the links you want. And without the right promotion, content fails. Understand where your audience lives and let them know you exist, target influencers that can share your content and increase your outreach to already engaged communities.

You can also cut corners by looking at your competitor's profile and trying to win some of their links for your own website. It's also worth tracking mentions of your brand, as well as focusing on PR, reputation management, brand awareness and online reviews.

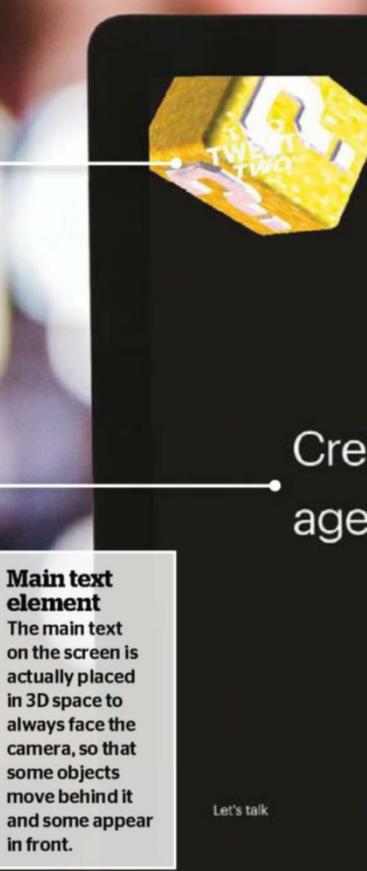
WITHOUT AMAZING CONTENT, YOU WILL NEVER GET THE LINKS YOU WANT. AND WITHOUT THE RIGHT PROMOTION, CONTENT FAILS. UNDERSTAND WHERE YOUR AUDIENCE LIVES AND LET THEM KNOW YOU EXIST

web workshop

Create a ThreeJS animated 3D cube scene

Inspired by twotwentytwo.se/

Site heading
The site heading is in the top left-hand corner and as the mouse moves over this the rollover animation pops out.



Mouse responsive
As the user moves their mouse over the screen, the 3D objects respond to the speed the mouse moves over them and rotate with the force of its speed.

Site menu
In the top right of the screen is the menu, enabling users to navigate to other areas of the site.



Animating cubes
As the site loads a number of cubes, pizzas and other objects animate onto the screen from the top and bottom.

EXPERT ADVICE

Use the right tool for the job

If you are going to add 3D elements onto a webpage, it's possible to fake that kind of thing up using 3D images and the DOM. If you want it to work properly though, it's best to rely on ThreeJS that takes the stress out of creating 3D WebGL scenes, leaving you to make the content.



<comment>

What our experts think of the site

Spinning pop culture

Our design connects minimalism, elegance, and pop culture. Visitors to our site enter a galaxy of flying pizzas, sunglasses, lollipops, and pieces of toast – comfort items. We want people to feel at home! Dive into our pizza and game night by spinning the orbit, and explore our interactive experience

Jesper Landberg, Owner / Creative director

Technique

1. Intro animation

To recreate the animation on the homepage of Twotwentytwo, a ThreeJS scene must be created. Here the scene, camera and project are added to the page. An ambient light is added to illuminate everything.

```
var scene, camera, renderer, object, light,
material;
function init() {
if (!Detector.webgl) Detector.
addGetWebGLMessage();
scene = new THREE.Scene();
camera = new THREE.PerspectiveCamera(75,
window.innerWidth / window.innerHeight, 0.1,
2000);
renderer = new THREE.WebGLRenderer({antialias: true});
renderer.setSize(window.innerWidth, window.
innerHeight);
document.body.appendChild(renderer.
domElement);
light = new THREE.AmbientLight(0xede155);
scene.add(light);
```

2. Setting the material

Another light is added that will act as the main light source. A box geometry is created to make cubes, then a material is created, which gives the box a yellow colour.

```
light = new THREE.DirectionalLight(0xffffffff,
0.5);
light.position.set(0, 1, 0);
scene.add(light);
var geometry = new THREE.BoxGeometry(5, 5,
5);
var material = new THREE.
MeshLambertMaterial({
color: 0xe3db14
});
```

3. Random cubes

Twenty random cubes are created and these are positioned from the top of the screen, and an animation is added to each so that they move down towards the centre of the screen.

```
for (var i = 0; i < 20; i++) {
let cube = new THREE.Mesh(geometry,
material);
cube.position.x = 200 * Math.random() - 100;
cube.position.y = 100;
cube.position.z = 200 * Math.random() - 100;
cube.rotation.x = Math.random() * Math.PI;
```

```
cube.rotation.y = Math.random() * Math.PI;
new TWEEN.Tween(cube.position).to({ y: 30 *
Math.random() + 20}, 1000 + Math.random() *
3000).easing(TWEEN.Easing.Quadratic.Out).
start();
scene.add(cube); }
```

4. More cubes

The next loop creates another twenty cubes. This time they appear off the bottom of the screen and randomly animate up the screen, so that now there are cubes animating from the top and bottom.

```
for (var i = 0; i < 20; i++) {
let cube = new THREE.Mesh(geometry,
material);
cube.position.x = 200 * Math.random() - 100;
cube.position.y = -100;
cube.position.z = 200 * Math.random() - 100;
cube.rotation.x = Math.random() * Math.PI;
cube.rotation.y = Math.random() * Math.PI;
new TWEEN.Tween(cube.position).to({y: 30 *
Math.random()}, 1000 + Math.random() * 3000).
easing(TWEEN.Easing.Quadratic.Out).start();
scene.add(cube); }
```

5. Finishing the init

The camera is positioned at a height of 20 units on the y axis and the render function is called. The 'init' function is closed and the render function is started. This calls itself to keep running, which rotates the camera and updates the Tween engine.

```
camera.position.set(0, 20, 0);
render(); }
function render() {
requestAnimationFrame(render);
camera.rotation.y -= 0.001;
TWEEN.update();
```

6. Finishing the project

The camera and scene are rendered and displayed on the screen. The render function is closed and the 'init' function is called. Save the page and run it in the browser to see the cube animation running.

```
renderer.render(scene, camera); }
init();
```



Build a 3D WebGL racer - part two

Last issue the groundwork was laid for this endless flying game. Here we add the finishing touches, ready for desktop and mobile.



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

Tn this tutorial the game that was started last issue is going to be resumed and completed, allowing for a stunning game that will work across mobile devices, tablets and desktop browsers that have support for WebGL. All the groundwork has been started, including the visual look of the game. The model is loaded and it's all ready for the game logic to be put in.

This is going to consist of making object classes for each of the four coloured track panels that will control their movement and the enemies within those panels, such as spinning saw blades, obstacles and sliding blockades. The 3D models will be linked up with those classes when they are instantiated as game objects. The next part will be to add the logic to control the build of the panels, so that they keep coming at the player and get faster. The panels also need to be updated so that they run their respective functions each frame and when they've passed the user the game needs to remove them, adding new panels in the distance.

The final part will be collisions to ensure the player has a challenge. The game will be controlled by moving the mouse on desktop, and on mobile devices the ship will be controlled by dragging back and forth on the screen. At the end you will need to upload it to a server to test the game, since it needs a server to load the models in.

1. Getting started

Open up the 'start' folder in your code editor and open the file 'game_pt1.js', which is where the code was finished at the end of part one of the tutorial. Scroll to the bottom of the document and start to add the code as shown here. This section of code contains the classes for the track panels that control their behavior.

```
function Cyan(obj) {
  this.obj = obj;
  this.enemies = [];
}

Cyan.prototype.reset = function () {
  //enemies refers to cubes blocking the
```

```
way
for (var i = 0; i < this.enemies.length;
i++) {
  let rpos = Math.floor(Math.random() * 6);
  this.enemies[i].position.x = (rpos * 9)
- 18;
}
};
```

2. Coloured panels

Each panel is a different colour, which was created like that for debugging, but stuck as a concept. Each coloured panel has a series of code that resets it and updates it every frame. For the Cyan panel it just randomises blocks in the way of the ship for it to swerve around.

```
Cyan.prototype.update = function () {
  this.obj.position.z += (SPEED * delta);
}

function Mag(obj) {
  this.obj = obj;
  this.enemies = [];
  this.l_enemies = []; //bars on each side
  this.r_enemies = [];
}
```

3. Magenta panel

The Magenta panel has blockades down each side that shoot out just before the player gets there. These are randomised from each side - left and right - and placed into an array called 'enemies', ready for when the player flies past them.

```
Mag.prototype.reset = function () {
  //set each side back to orginal position
  for (var i = 0; i < m.l_enemies.length;
i++) {
  this.l_enemies[i].position.x = -30;
  this.r_enemies[i].position.x = 30;
}
//reset enemy selection from last time
this.enemies = [];
};
```

```
//choose a new random side for the 4
spots
for (var i = 0; i < 4; i++) {
let rnd = Math.floor(Math.random() * 2);
if (rnd == 1) {
  this.enemies.push(this.l_enemies[i]);
} else {
  this.enemies.push(this.r_enemies[i]);
}
}
};
```

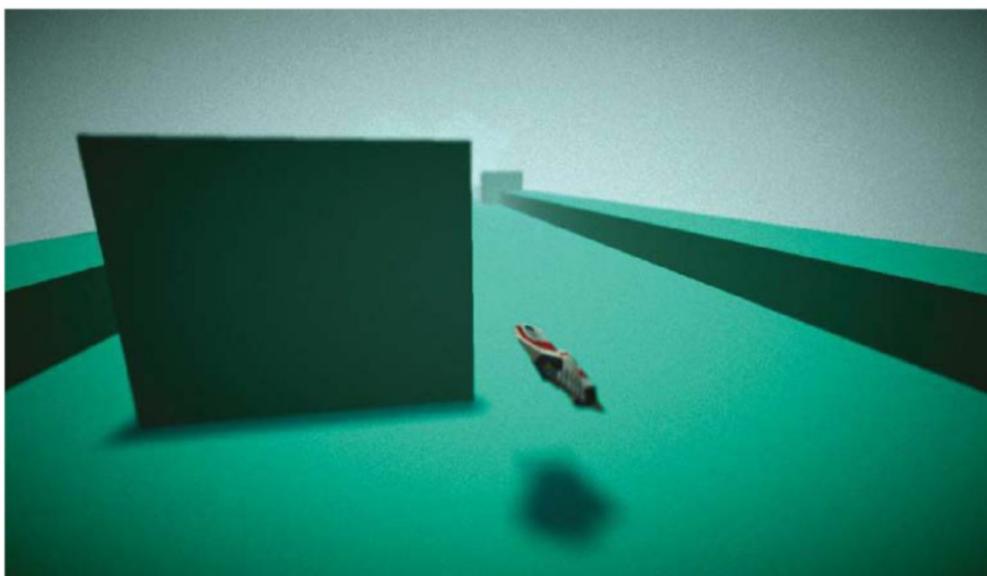
4. Updating the blockades

As the player gets within a certain distance of the panels, the blockades shoot out from each side, so that the player has to react and dodge the panels as they fly down the Magenta panel on the screen. The random nature means that the player isn't sure whether they're coming from the left or the right.

```
Mag.prototype.update = function () {
  this.obj.position.z += (SPEED * delta);
  if (this.obj.position.z > -10) {
    this.enemies[0].update();
  }
  if (this.obj.position.z > 80) {
    this.enemies[1].update();
  }
  if (this.obj.position.z > 170) {
    this.enemies[2].update();
  }
  if (this.obj.position.z > 280) {
    this.enemies[3].update(); }}
```

Arrays

Arrays are pretty much like a grocery list - you can add items to it and anytime you don't need an object, you can remove it from the list. They're handy for keeping track of several different things at once.



Left

The Cyan panel is set up with a number of enemy blocks, which are randomly positioned on the track panel for the player to avoid

Top

The Orange panel has circular saws that swing up out of the ground at the player. This section is the trickiest when the speed increases

Tutorials

Build a 3D WebGL racer - part two

5. Setting up the Orange panel

The Orange panel has circular saw blades on pendulum arms that fly out of the floor. Here these are all reset so that the arms are facing straight down. Later as the player approaches, the arms will swing around and the player will have to avoid the saw on the end of this arm.

```
function Obj(obj) {
    this.obj = obj;
    this.enemies = [];
}

Obj.prototype.reset = function () {
    for (var i = 0; i < this.enemies.length;
i++) {
    let rpos = Math.floor(Math.random() * 4);
    this.enemies[i].position.x = (rpos * 10)
    - 25;
    this.enemies[i].rotation.z = Math.PI;
    //180 facing down
    this.enemies[i].add(this.snd);
}
};


```

6. Updating the arm

The update for the Orange panel checks the position of the panel and, if it's close to the player, then the arm is set to rotate. All animation in the panels is controlled by a 'delta' so that if the frame rate slows down or speeds up, then the movement stays consistent since it's based on time not frames.

```
Obj.prototype.update = function () {
    this.obj.position.z += (SPEED * delta);
    if (this.obj.position.z > -10) {
        this.enemies[0].rotation.z -= 0.05;
        this.enemies[0].children[1].rotation.z +=
        delta * (Math.PI * 2);
    }
    if (this.obj.position.z > 85) {
```

Raycasting

Raycasting sends an invisible ray in a direction that our code states and tells us of all the models in the line of that ray. This is used for detecting collisions.



Top

The game runs fine on a 2-year-old Nexus 6P, showing that it works well on desktop, mobile and tablet

Right

When the player collides with an object it triggers 'game over' which causes the game to render in greyscale and freeze at the point of contact, leaving just the ship

```
this.enemies[1].rotation.z -= 0.05;
this.enemies[1].children[1].rotation.z +=
delta * (Math.PI * 2);
}
if (this.obj.position.z > 175) {
    this.enemies[2].rotation.z -= 0.05;
    this.enemies[2].children[1].rotation.z +=
delta * (Math.PI * 2);
}
```

7. A slice of lime

The Lime panel is set up now and this is one of the simplest track panels since there is nothing to hinder the player on this. The 'reset' function is still present because every panel that gets called onto the game gets the 'reset' function called, so even though it's empty it stops the code from breaking.

```
function Lime(obj) {
    this.obj = obj;
    this.tubes;
    this.init();
}
Lime.prototype.init = function () {
    this.tubes = this.obj.
    getObjectByName("Tubes", true);
};
Lime.prototype.reset = function () {}
```

8. Juicing the lime

The Lime panels update really just controls the animation of the tubes rotating to give the scene some movement as the player flies through this panel. This is also always the starting panel for every game that will be played, and so has a grandstand feel to it.

```
Lime.prototype.update = function () {
    this.obj.position.z += (SPEED * delta);
    this.tubes.rotation.z += 0.01;
}
//returns random number within a range
function getRand(minVal, maxVal) {
    return minVal + (Math.random() * (maxVal -
minVal)); }
```

9. Linking up the classes

Now it's time to link up the classes with the actual models. In the 'init' function is a section that loads the panels, where you will find a comment 'PANEL SETUP TO DO'.

The next code goes here. Remove the line 'scene.add(dae);' since this is no longer needed.

```
var x = dae.getObjectByName("Cyan", true);
c = new Cyan(x);
//push the enemy blocks into Cyan's
enemies array - doesn't seem to work from
inside the Cyan object!
x.traverse(function (child) {
    if (child instanceof THREE.Mesh && child.
parent.name == "enemy") {
        c.enemies.push(child.parent);
    }
});
inactive.push(c);
```

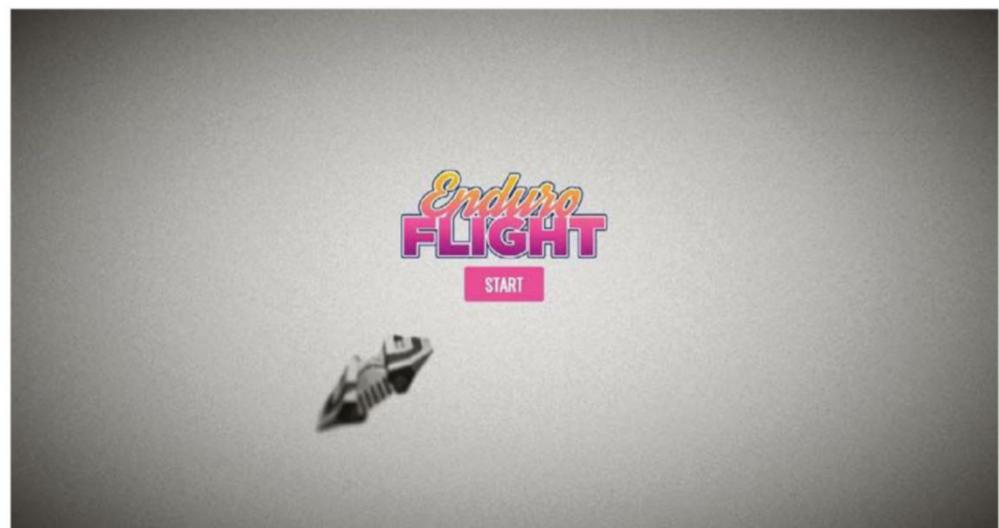
10. Magenta models

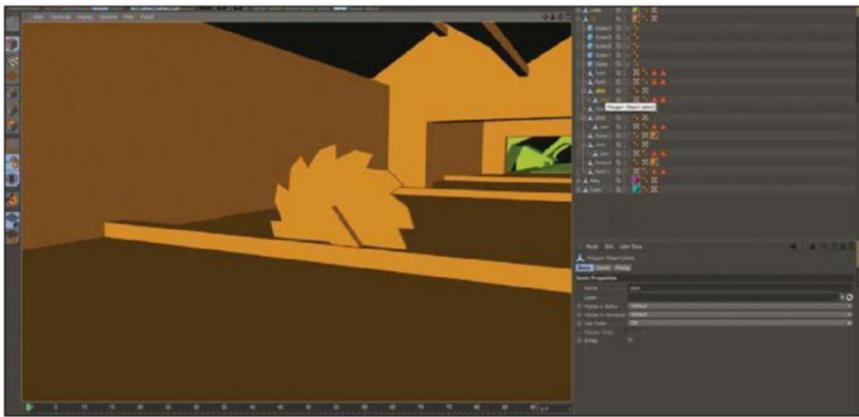
Next up is the Magenta model. Again the class is instantiated and each of the models on the left-hand side that will slide out are placed in an array name 'l_enemies' for the left side. Each enemy is given its own update function to animate it at the appropriate time.

```
x = dae.getObjectByName("Mag", true);
m = new Mag(x);
x.traverse(function (child) {
    if (child instanceof THREE.Mesh && child.
parent.name == "enemyL") {
        m.l_enemies.push(child.parent);
        child.parent.update = function () {
            if (this.position.x < 5) {
                this.position.x += (120 * delta);
            }
        }
}}
```

11. To the right

The same as the left-hand side of the Magenta panel, all of the enemies on the right are stored in their own array and given an update function. All of the main coloured panels are stored in an array called 'inactive' when they are placed on screen they are taken out of here, that way





Getting access to models

All of the panels are loaded in one file which is the 'panels.dae' file. This is a Collada model also known as Digital Asset Exchange, hence the file extension of 'dae'. Each individual model that needs to be controlled in the game has been named specifically in the 3D software so that it can be accessed in the game. The easiest way to think of this is in giving a div an ID. In the ThreeJS

code, we can control these models by using the 'getObjectByName' command very similar to 'getElementById':
 var x = dae.
 getObjectByName
 ("Lime", true);
 With the code here the browser will search the loaded model for the object named 'Lime' and recursively search through all children until it's found it. The model is then stored in the variable 'x'.

it's easy to keep track of the panels in the game and those waiting to be placed.

```
if (child instanceof THREE.Mesh && child.parent.name == "enemyR") {
  parent.name = "enemyR";
  m.r_enemies.push(child.parent);
  child.parent.update = function () {
    if (this.position.x > -5) {
      this.position.x -= (120 * delta);
    }
  };
  inactive.push(m);
}
```

12. Orange segments

The Orange panel is set up with the enemies being located in here. Those models are all named 'stick' as it's the arm that the saws swing on. These are passed into a new instantiated object so that they can be used in the game as a complete panel.

```
x = dae.getObjectByName("0j", true);
o = new Oj(x);
o.enemies = [];
x.traverse(function (child) {
  if (child instanceof THREE.Mesh && child.parent.name == "stick") {
    o.enemies.push(child.parent);
  }
});
inactive.push(o);
```

13. Last panel in the limelight

The final panel to be set up is the Lime panel and the new instantiated object is created. All panels are now stored in the 'inactive' array. This means that panels in this array can be selected and placed in the scene, but resources are not wasted on animating their parts.

14. Placing panels

To put panels into the scene a new function called 'buildPanels' will be created. Place this code after the closing bracket of the 'render' function. If the 'active' array is empty then the Lime panel is added as the first panel. This gets reset and positioned in the scene.

```
function buildPanels() {
  //first build - add the lime panel
  if (active.length == 0) {
    for (i = 0; i < inactive.length; i++) {
      if (inactive[i].obj.name == "Lime") {
```

```
var pln = inactive[i];
active.push(pln);
scene.add(pln.obj);
pln.reset();
pln.obj.position.z = 10;
```

15. Random panels

Once the first panel is placed in the scene, other panels can be placed into the scene. These are randomly selected, so they are added here using a random number to select one of the remaining panels from the 'inactive' array.

16. Upping the speed

The new panel is reset and positioned directly behind the first panel added in the scene. It's placed into the 'active' array and the game counter is increased. This is used to increase the speed in the game. For every two panels that get added the speed goes up slightly.

```
mdl.reset();
mdl.obj.position.z = active[0].obj.position.z - 500;
scene.add(mdl.obj);
active.push(mdl);
inactive.splice(rndm, 1);
pCounter += 1;
if (pCounter % 2) {
  if (SPEED < 3) {
    SPEED += 5;
  }
}
```

17. At the start of the game

Find the 'begin game' function and you will see a comment 'BUILD PANELS HERE'. This is the start of the game and where new panels should be added into the game. All we need to do here is call the 'build panels' function, so add that line in.

```
buildPanels();
```

18. Updating the panels

Find a place to add this code outside of all other functions. Here the panels are moved every frame with their own update function. If the player has passed over them, the panels are removed from the game, placed in the 'inactive' array, ready to be randomly called back in.

```
function updatePanels() {
  for (i = active.length - 1; i >= 0; i--)
```

```
{
  var mdl = active[i];
  mdl.update();
  if (mdl.obj.position.z >= 520) {
    active.splice(i, 1);
    scene.remove(mdl.obj);
    inactive.push(mdl);
    buildPanels();
  }
}
```

19. Detecting collisions

A game needs to be able to detect if the player hits something. This code fires an invisible ray out of the ship, bringing back an array and distance to all models in front of the player. This can be used to detect a collision with another object.

```
function collisionTest() {
  // collision detection - firing 2 rays
  for each side of the ship
    for (var i = 0; i < 2; i++) {
      var originPoint = ship.position.clone();
      originPoint.x += (i * 2.6) - 1.3;
      raycaster.ray.origin.copy(originPoint);
      let intersections = raycaster.intersectObjects(scene.children, true);
      if (intersections.length > 0) {
        var distance = intersections[0].distance;
```

20. Hitting the spot

The last part of the collision detection is to check if the player is less than 3.5 units to another object, if they are this registers as a hit. The variable 'dead' is changed to 'true' to start processing the clean up on the screen at the end of the game.

21. The final step

In the 'render' function there are two comments to 'UPDATE PANELS HERE' and 'COLLISION TEST HERE'. Just call the functions as in the code below and that will tie all the elements together in the game so that it works. Make sure you run the game from a server to load the game models over XHR.

```
////////// UPDATE PANELS HERE //////////
updatePanels();
////////// COLLISION TEST HERE //////////
collisionTest();
```

Faster builds with BetterBoilerplate

Learn how to build a simple and beautiful page using the frontend framework BetterBoilerplate





The BetterBoilerplate is a new frontend framework aimed at developers, freelancers and agencies but can easily be adapted and used in any sized project. It is meant to be used as a starting point for a project, from which you're encouraged to update, change and add to the files. The BetterBoilerplate is made up of many elements, and will handle many aspects of the project build. The features include file structure; code compilation and minification (SASS/SCSS, JavaScript); SASS mixins and functions; image optimisation; SVG sprite compilation; and a pattern library containing common reusable components.

Frameworks like Bootstrap and Foundation have many pre-made and pre-styled components, while this can be useful for developers not familiar with CSS or for prototyping, you can spent a lot of time overriding the styles provided by these frameworks.

The BetterBoilerplate doesn't include many pre-built components and shouldn't be included as an external stylesheet, instead the BetterBoilerplate is where you will write your project files. The SASS framework allows for brand elements (such as colour and spacing) to be easily programmed and used throughout.

In this tutorial we'll look at a few of these features and build a simple page.

1. Create the project folder

We'll start by creating a new project folder, downloading the BetterBoilerplate and adding it to our project folder. You can download it from GitHub (<http://srt.lt/TxCn5>) or you can clone the project:

```
$ git clone https://github.com/BetterBrandAgency/betterboilerplate.git .
```

2. Install the dependencies

The BetterBoilerplate has a few prerequisites. You will need to install NPM, SASS and Gulp before you can use the framework. Once they're installed we can install the dependencies from the boilerplate. To do this, navigate to your project folder in command line, and run the 'npm install' command.

3. Create our first build

Now we have everything installed we can create our first build by running the 'gulp' command inside the 'better-boilerplate' folder. When we run this command a new folder named 'dist' will appear in the root of your project. This folder will contain the compiled CSS and JavaScript as well as the generated SVG sprite. This command will also listen for changes to any files in the project and automatically recompile.

4. Create our index file

Next we'll create our index file and include these compiled files in our website. In the root of the project create the 'index.html' file and include the compiled CSS.

```
<!doctype html>
<html lang="en">
  <head>
    ...
    <link rel="stylesheet"
      href="dist/css/main.css">
  </head>
  <body></body>
</html>
```

5. Gather project assets files

Distribute them into the correct folders. If we now run the 'gulp' command inside our 'better-boilerplate' folder the 'dist' folder will be updated with our new files.

6. Set our brand colours

Next we'll dig into the CSS framework and set our brand properties. We'll start with the brand colours. Let's jump into the 'src/styles/01-settings/_colours.scss' file and update the phonetic colour variables to reflect our brand.

```
$color-alpha: #FF7C1D;
$color-bravo: #00A2B2;
$color-charlie: #707070;
```

7. Set our brand type settings

Now we've change the colours we'll also update the fonts. To do this we'll move into the '_fonts-and-text.scss' file (also in the '01-settings' directory) and update the

relevant variables.

```
$font-family-alpha:
  'PT Sans', sans-serif;
$font-family-bravo:
  'Nanum Pen Script', cursive;
$base-font-size:
  18px !default;
```

8. Markup the hero area

Let's jump back into our HTML and create the code for a simple hero area. We'll use the BEM methodology for naming the classes and create an overlay, logo, headline and icon inside.

```
<div class="hero">
  <div class="hero__overlay"></div>
  <div class="hero__content"></div>
</div>
```

9. Markup the hero content

We can copy and paste the SVG code from the 'patter-library/html-patterns' folder and change the relevant parts to use the SVGs we added to the project. We simply replace the 'SVG-NAME' with the file name of the SVG. We'll do this for the logo and smiley icon, as well as adding a headline.

```
<svg class="hero__logo svg" viewBox="0 0 331 230">
  <title>Emailify Logo</title>
  <use href="dist/images/sprite.svg/#logo"></use>
</svg>
<h1>...</h1>
<svg class="hero__smiley svg" viewBox="0 0 61 64">
  <use href="dist/images/sprite.

```

CSS file structure

The BetterBoilerplate uses a system very similar to ITCSS in order to manage specificity, whereby the further down the file structure the more specific the styles.

The screenshot shows a website with a header 'Better.' and a navigation bar with links to About, Services, Work, Blog, and Contact. The phone numbers 01642 989 158 and 020 3856 8166 are listed under 'Teeside' and 'London' respectively. Below the header is a hero section featuring a logo (a green checkmark inside a circle), a headline 'Helping to build a Better web – Introducing the BetterBoilerplate', and a smiley icon. The footer contains social media icons for GitHub, LinkedIn, and Twitter, along with a copyright notice: 'Friday 2nd February 2018'.

Tutorials

Faster builds with BetterBoilerplate

```
svg#smiley"></use>  
</svg>
```

10. Create our own SCSS file

In order to style our hero area we'll create a new file for all our hero specific styles. We'll do this inside the 'src\styles\06-components' folder and add the new file to the list in 'src\styles\main.scss' in order to include it in the outputted CSS file.

```
...  
// Components - Complete Chunks of UI  
...  
@import "06-components/hero_";  
...
```

11. Style the hero

Now we've created the file to contain our styles we can go ahead and start writing our styles. While we're writing our CSS we need to be running the gulp command so changes will be compiled every time we save a file.

```
.hero {  
  position: relative;  
  display: flex;  
  align-items: center;  
  justify-content: center;  
  color: white;  
  background: {  
    image: url(..../images/hero-image.jpg);  
    size: cover; }  
http://srt.lt/L5Wc7
```

12. Style the hero overlay

Next we'll style the hero overlay. We want this to sit over the background image to darken it. We can utilise the BEM nesting ('&__') built into SASS to nest our elements

Documentation

The documentation for the boilerplate covers everything on offer and in plenty of detail too, from installation to explanation. It's definitely worth checking it out.



inside the block.

```
.hero { ...  
&__overlay {  
  position: absolute;  
  top: 0; left: 0;  
  width: 100%; height: 100%;  
  background-color: $color-charlie;  
  mix-blend-mode: multiply;  
  opacity: 0.8;  
}
```

max-width: \$base-spacing-unit * 30;

14. Style the SVGs

We'll again utilise the '\$base-spacing-unit' to perform calculations used for widths and margins. Since these are SVG's we can add a max-width and then they will scale with the browser but never above the maximum width.

```
&__logo {  
  max-width: $base-spacing-unit * 12;  
  margin-bottom: $base-spacing-unit;  
}  
&__smiley {  
  max-width: $base-spacing-unit * 2;
```

15. Style the hero headline

The BetterBoilerplate includes a number of mixins. One of the most useful is the ability to create responsive typography entirely in CSS.

```
h1 {  
  @include fluid-type(30px, 48px,
```



Top

We've added some styles to the hero area in order to apply a background image

Right

We've styled the overlay by applying one of our brand colours as the background





```
$responsive-start,
$responsive-end
);
line-height: 1;
margin-bottom: $half-spacing-unit; }
```

16. Hero styling finishing touches

That's the hero area pretty much complete now, but we can add a couple of finishing touches. We'll add a minimum height to the hero and turn on the font smoothing option in the settings.

```
// _hero.scss:
.hero {
  min-height: 100vh;
}
// _base-options.scss:
$use-font-smoothing: true;
```

17. Altering the settings

The BetterBoilerplate has a number of settings, all of which are turned off by default. By turning on some of these settings we get access to some classes to handle a number of different things in the project. We'll turn on the colour, text, padding and margin classes.

```
$use-color-classes: true;
$use-helper-text-styles: true;
$use-helper-padding-styles: true;
$use-helper-margin-styles: true;
```

18. Markup a new section

Let's markup a new section using just these classes. We'll start by creating a section tag and adding the classes to change the text and background colours and add some padding to the top and bottom of the element.

```
<section class=""
background--bravo
color--white
padding-double--ends
">
```

19. Markup the content container

Using the 'content-container' class we add a maximum width to the container and centre it in the page. We can use the 'text-center' class and the 'padding-double' class to add more spacing and centre the text.

```
<div class=""
content-container    text--center
padding-double
">
```

The pattern library

Included in the downloaded files is the 'pattern-library' folder. This folder contains a number of customizable elements, objects and components.

Some, like the SVG pattern in the HTML folder are simply there as a reference to copy and paste into your project and modify where needed. Other components, such as the accordion and overlay, require elements from the HTML and JS folders,

as well as turning on the default styles for these elements in the '01-settings/_modules-and-widgets.scss' folder.

Almost everything in the BetterBoilerplate is styled using the variables found in the '01-settings' folder. These properties are inherited and used to provide the default styles which can be overwritten where needed to match the brand of the project you're working on.

">

20. Markup the content

Next we'll add a headline and some text, utilising the margin classes to control the flow of content. The fonts, size and spacing are all inherited from the setting we changed earlier.

```
<h1 class="no-margin">Get Started</h1>
<p>Aenean condimentum...</p>
```

21. Creating additional colour classes

You'll notice our 'color-white' class didn't work. That's because it's not included as a default colour. To add this colour to the outputted CSS we'll open up the '_color-classes.scss' file located in the 'src/styles/05-objects/' folder and add our new colour to the '\$color-list' array.

```
$color-list: (
  alpha $color-alpha,
  bravo $color-bravo,
  ...
  white #ffffff,
);
```

Far Left

We've styled the headline using the BetterBoilerplate's fluid typography mixin

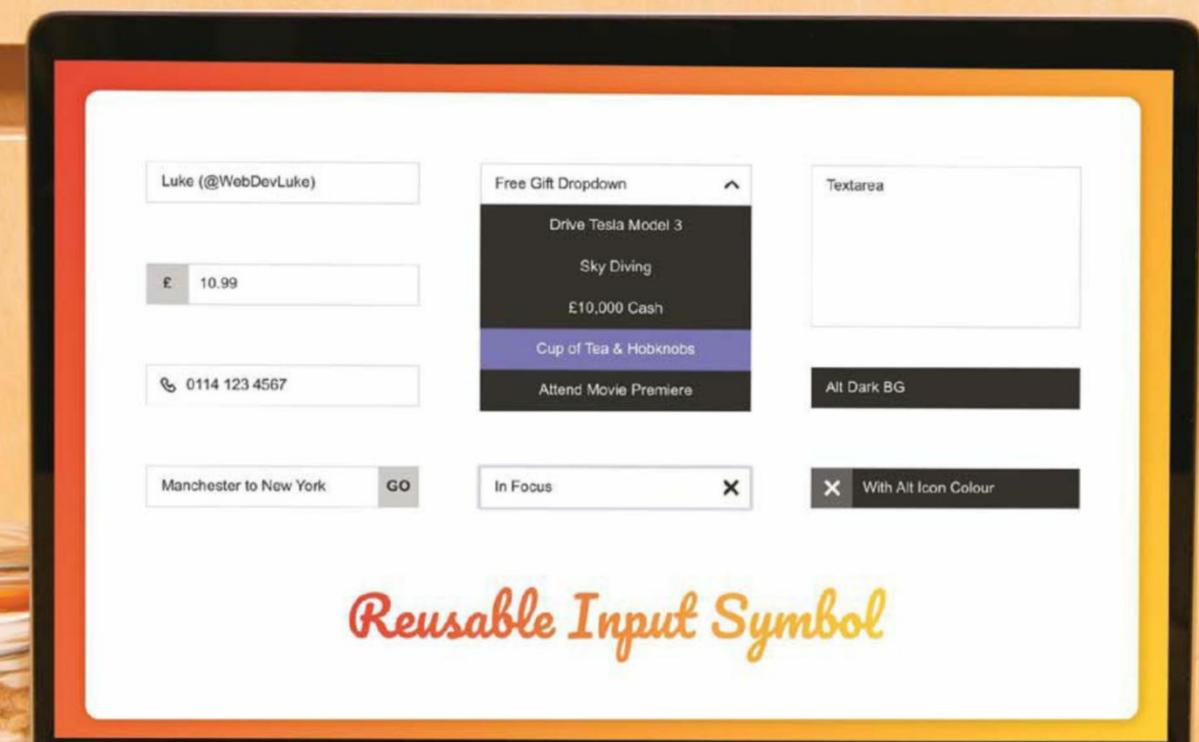
Left

We've used some of the built-in classes and added our own in order to markup our Get Started section



Create a reusable input symbol

In Sketch, discover how to leverage the power of overrides to create a custom, flexible component that can be used again and again



In Sketch, one of the key factors which make a symbol great is a high level of reusability. When building a symbol, the best way to achieve this is by leveraging Sketch's extensive overrides system. This system enables you to create what is, essentially, a rudimentary API for your symbol, making it possible for the end user to edit or change aspects of the symbol on an instance-by-instance basis, without permanently changing the symbol's original design or its functionality.

For example, a button where you can edit the text, an icon with various colour options, or a photo caption where each instance of the symbol can have a different photo or caption. It is this level of customisation which makes a symbol very flexible, enabling it to be used again and again across a variety of different scenarios and projects.

In this tutorial, we're going to be using Sketch's override system to create a highly reusable HTML input symbol. Download the tutorial files from FileSilo and open 'template/design.sketch' in Sketch. This will be the blank canvas where we'll create our input symbol. During this tutorial, 'Layers List' will refer to the left menu, whilst 'Inspector Toolbar' will refer to the right. Let's begin!

1. Draw a transparent rectangle

Use the Rectangle tool (R key) to create one measuring 260 x 55px. With it selected, remove the fill and border via the Inspector Toolbar, leaving a transparent shape. In the Layers List, rename it 'Container' then click the 'Create Symbol' button in the top toolbar. Call the symbol 'Inputs/Main', with 'Inputs' being its category and 'Main' being the symbol itself.

2. Creating input background

In the new 'Symbols' page, find 'Inputs/Main', right-click the 'Container' layer in the Layers List and select 'Mask'. Next, create a 10 x 10px rectangle, give it a #FFFFFF fill, a #C6C6C6 border, then click 'Create Symbol' and call it 'Input/Backgrounds/Light'. Back in 'Inputs/Main', let's

rename the instance of our new layer to 'Colour' (Press CTRL, CMD & SPACE to bring up the emoji keyboard).

3. Our first override

Duplicate the 'Input/Backgrounds/Light' layer, renaming it 'Input/Backgrounds/Dark'. In the duplicate, change the fill colour to #333333, the border to #686868, and then back in the 'Inputs/Main' symbol, stretch the instance of the background symbol to fill the entire rectangle. Back on the Components page, you will see the foundation of our input symbol. Click it and you will see a configurable 'Colour' override.

4. Creating a dropdown add-on

In 'Inputs/Main' create a 260 x 55px rectangle with a fill of #333333 and no border. Convert it to a symbol called 'Input/Dropdown/Placeholder', then in this new symbol, add centered, bold, white, 20px 'Option Value' text (T) and call its layer name '↳ Value'. Next, duplicate the symbol, renaming the new one 'Input/Dropdown/Active' and changing its fill to #7A78BD.

5. Duplicate the options

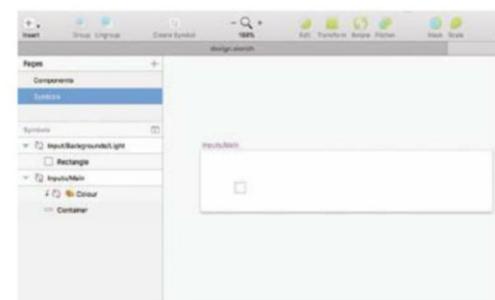
Back in 'Inputs/Main', create six duplicates of the dropdown layer, select them all in the Layers List and convert into a symbol called 'Input/Dropdown/Main'. Rename the instance of your symbol to '↳ Dropdown', then select the 'Input/Dropdown/Main' symbol in the Layers List and extend its height to 440px.

6. Stack options vertically

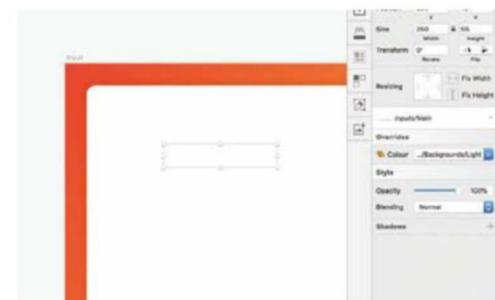
In the 'Input/Dropdown/Main' symbol, stack each of the options vertically, taking care to make sure their order in the Layers List corresponds to their visual order. Rename each layer to '1st', '2nd' etc., as you go. Make sure to leave a 55px gap at the top.

7. Dropdown menu - finishing touches

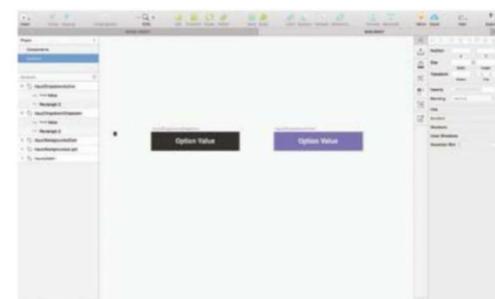
In the main 'Inputs/Main' symbol, the dropdown will look



Step 02



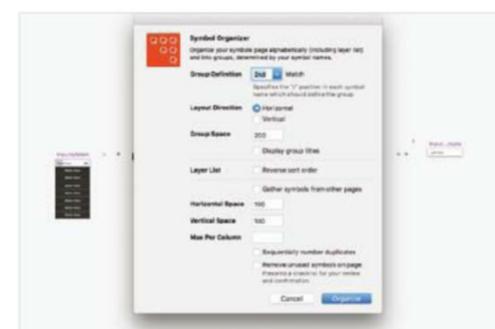
Step 03



Step 04

Emojis in layer names

As the width of the Inspector Toolbar is quite limited, the use of emojis in layer names is a pattern which helps to convey information about an override without taking up too much space.



Left

One thing you may have noticed as you create more and more symbols is that on the symbols page, things can get a little busy

Above

If you download the Symbol Organiser plugin (You can find it through Sketchrunner, Sketchpacks, or Sketch Toolbox), you'll be able to sort your symbols based on a set of criteria

Tutorials

Create a reusable input symbol



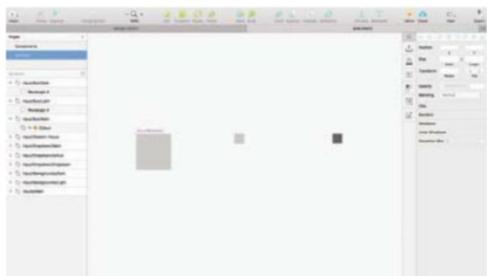
Step 07



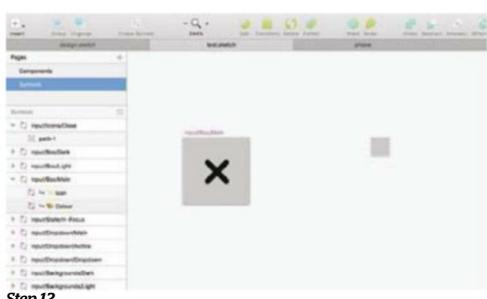
Step 08



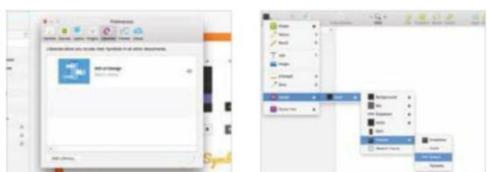
Step 09



Step 10



Step 11



Top left

In Sketch, you can easily share your reusable symbols across projects via a library. Click Sketch > Preferences > Libraries (or alternatively CMD + comma)

Right

To add a new Library, click the 'Add Library' button and then select the file containing your symbols

Top right

You can now use your symbols in other documents. If any of these library symbols are later updated, any Sketch documents using it are notified, enabling you to preview, check and then confirm changes

distorted. Within the layer's list, move its layer below 'Container' and set its height to 440px. Back on the Components page, our input now has a dropdown menu. In its overrides menu in the Inspector Toolbar, we can toggle it on/off and also change any of the option's text, as well as set its state to active.

8. Creating focus state - pt1

Create a 40 x 40px rectangle in the main symbol and convert it into a symbol called 'Input/State/In-Focus'. In the Layers List, rename the symbol's instance in the main symbol to 'Focus', move it so it's below the dropdown layer, and then resize so it's 100% width/height of its container. Now let's turn our focus (wahey!) to our recently created symbol.

9. Creating focus state - pt2

Start editing the 'Input/State/In-Focus' symbol, remove its fill and change its border colour to #7A78BD. In the Inspector Toolbar, click 'Shadows' and change its colour to #7A78BD with an alpha (A) of 50. Then set X to 0, Y to 0, Blur to 4 and set Spread to 5. If you go back to the Components page, our input should now have a purple focus, which can also be toggled in the overrides layer.

10. Create a square

Next, in the main symbol, create a 55 x 55px square and align it to the left of the input. Change its fill to #C6C6C6, remove its border and then convert it into a symbol called 'Input/Box/Main'. In our new symbol, select the square and then click 'Create Symbol' again, naming this one 'Input/Box/Light'.

11. Duplicate and rename

Duplicate 'Input/Box/Light' and rename the new one 'Input/Box/Dark', then change its fill to #686868. Resize both 'Input/Box/Light' and 'Input/Box/Dark' symbols to 15 x 15px, then in 'Input/Box/Main', rename the 'Input/Box/Light' layer to 1. Colour'.

12. Import icons

From the FileSilo package, navigate to the 'support-files'

directory and open all three of the SVG files inside: 'arrow', 'close' and 'phone'. Copy the 'close' icon and paste it into the 'Input/Box/Main' symbol, then convert it into a symbol called 'Input/Icons/Close'. Still in the 'Input/Box/Main' symbol, resize the icon to 20 x 20px, re-centre it, then rename its layer 'Close Icon'.

13. Add a colour mask

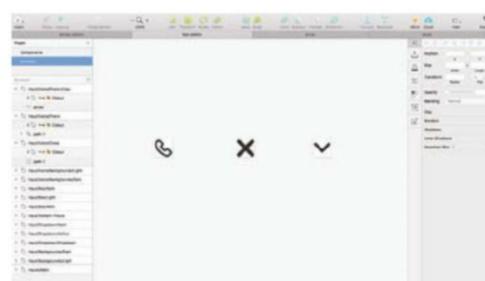
In your new 'Close Icon' symbol, create a rectangle which stretches the full 12 x 12px artboard and give it a fill of #333333 and remove its border. From this rectangle, create a symbol called 'Input/Icons/Backgrounds/Dark'. Next, rename its instance in the icon symbol to 1. Colour', then right-click the icon's layer ('path-1' by default) and select the 'Mask' option. This should make the colour symbol overlay our 'Close' icon.

14. Create alternative icon colour

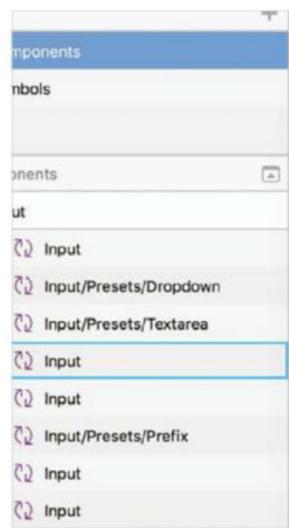
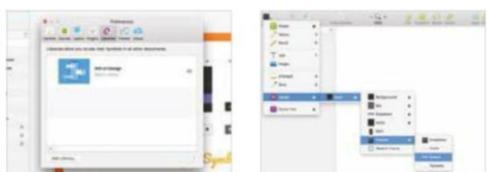
Find the 'Input/Icons/Backgrounds/Dark' symbol, resize it to 20 x 20px, then duplicate it. In this duplicate, change

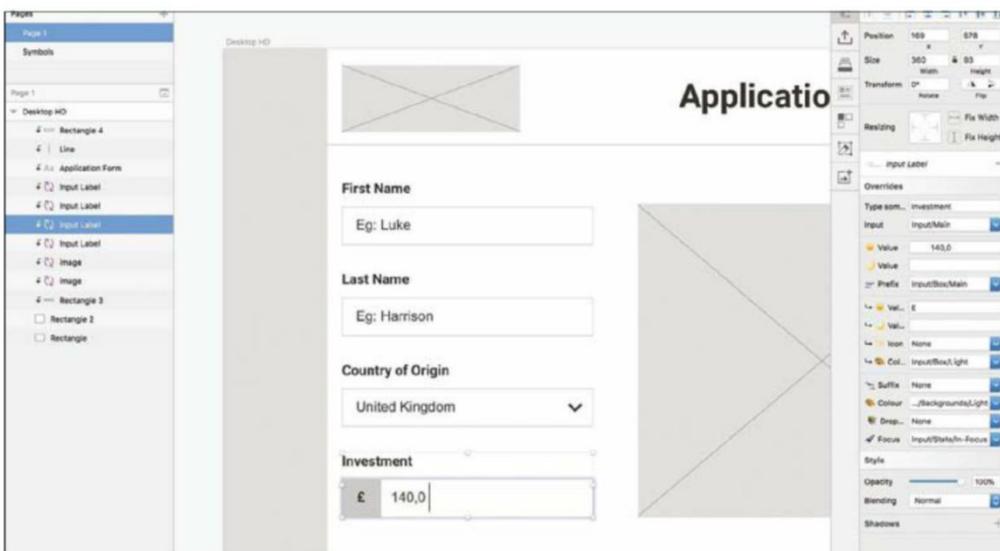


Step 14



Step 15





Reusable symbols

Developing a suite of reusable symbols for you and your team can pay dividends when it comes to web design projects.

For wireframing, you no longer have to waste time reinventing the wheel for every project, speeding up things considerably and lowering costs in the process. For high-fidelity production components, reusable symbols make it less likely that other designers will misinterpret or deviate from brand standards, since if your collection is extensive enough, it's just a matter of them picking the right symbol for the right job.

In addition, by tasking your team's best designers with the creation of these symbols, it enables more junior designers to piggy-back on their experience, producing higher quality work and learning more in the process.

the fill colour of its rectangle to #FFFFFF, and name it 'Input/Icons/Backgrounds/Light'. If you go back to the 'Input/Box/Main' symbol and click its icon layer, you'll notice in the Inspector Toolbar that an override option has appeared, letting you toggle the colour of the icon.

15. Process remaining icons

For each of the remaining icons, duplicate 'Input/Icons/Close' and rename it 'Input/Icons/Arrow' and 'Input/Icons/Phone' respectively. In each symbol, replace the 'Close' icon with the correct icon, taking care to remember to toggle the 'Mask' option for each icon and make sure the artboard keeps the same 12x12px dimensions.

16. Adding text

In 'Input/Box/Main', add two 20px bold text layers containing 'E' in the centre, overlaying the icon and each other. Name one 'Value' (AKA 'Dark Value') and give it a text colour of #FFFFFF and then name the next 'Value' (AKA 'Light Value'), with a text colour of #333333.

17. Rename elements

Go back to your 'Input/Main' symbol and rename the 'Input/Box/Main' instance 'Prefix', then duplicate it, align to the right, and rename this one 'Suffix'. If we click either, we'll see a bunch of options in its overrides menu which enables us to configure a variety of icons, background colours and text layers. To disable either text layer so we can use the other, simply set its value to a space character.

18. Change resize rules

Back in the main Components page, if we resize the input

horizontally, we'll notice that these boxes will distort. To fix this, let's assign some resize rules. Open the 'Input/Main' symbol and click on the 'Prefix' layer. In the Inspector Toolbar, above the list of overrides, there's a 'Resizing' option. Here, toggle the 'Fix Width' option to 'Active', then toggle the top, bottom and left options. For the 'Suffix' layer, follow the same procedure, but toggle the top, bottom and right options instead.

19. Adding text to main symbol

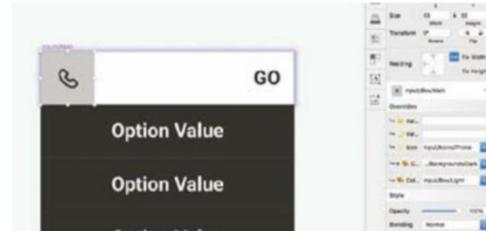
Now let's give the 'Input' symbol some text. In 'Input/Main', create two 20px text layers, overlaying all existing layers, placed 20px from the left of the artboard. For the first, set its colour to #FFFFFF, give it the value of 'Dark Value' and rename its layer to 'Value'. For the second, set its colour to #333333, give it a value of 'Light Value' and rename its layer to 'Value'.

20. Text layer resize rules

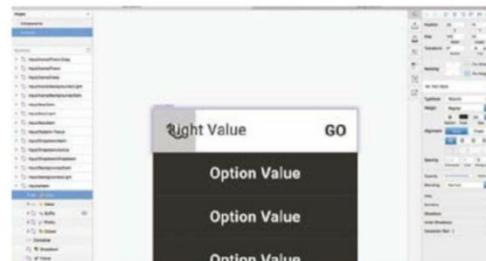
If we go back to the Components page and resize the input horizontally, we'll notice these text layers slide as we resize. Like with the prefix/suffix symbols back in Step 18, we need to assign resize rules to our text layers. For each one, toggle the top and left options, set their alignment to fixed, and also make sure their bounding boxes have a width stretching across most of the input (around 225px). This should stop the sliding issue so when we resize the 'Input' symbol horizontally, the text should stay where it is.

21. Finishing touches

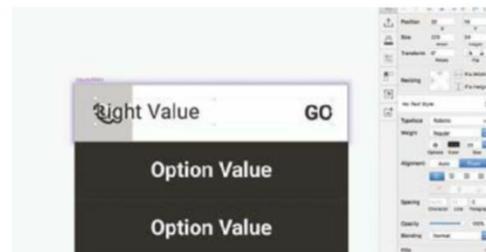
Back in the Components page is a finished 'Input' symbol. It looks a little messy as all the options will be haphazardly enabled. Disable all options by setting their values to 'None' where a dropdown, or empty if a text input, then set 'Colour' to 'Input/Backgrounds/Light'. This gives us a simple input which we can build from. To save it as a preset, with it selected, simply click 'Create Symbol' and name it 'Input/Presets/Simple'. You can create as many presets as you like. See the 'design.sketch' file in the 'heres-one-i-made-earlier' directory for some ideas.



Step 018



Step 019



Step 020



Step 021

Third-party libraries

In addition to creating your own Sketch library, you can also use third-party ones. Sketch comes installed with an iOS symbol library out of the box, allowing for quick prototyping of apps.

web workshop

Created a staged menu opening animation

Inspired by deltstl.com

The home menu logo

Clicking this enables users to return to the website homepage. Allows people to return to a familiar place if they get lost.

Close menu icon

Clicking on this icon will close the menu, returning the user to the main page content that was visible before the menu opened.

Social media links

Links to social media accounts - part of the menu, but making use of space to appear isolated from the main website navigation.

- Homepage
- Web design
- SEO services
- Marketing
- Brand strategy
- Complete branding
- Our portfolio
- Meet the team
- DELT Magazine
- Contact us

The menu container

The menu container is designed to cover the full content space so that only the featured menu content is visible to the user.

Fixed positioning

The menu container also uses fixed positioning to guarantee that it remains fully visible over all content regardless of scrolling.



Created a staged menu opening animation

[DOWNLOAD TUTORIAL FILES](#) www.filesilo.co.uk/webdesigner

EXPERT ADVICE

Space efficiency

This approach to presenting the menu maximises the use of available space without conflicting with the content's presentation. Make sure that users fully understand how to access the menu content to avoid confusing them, since that risks them abandoning the website.



<comment>

What our experts think of the site

Developed to be highly flexible

The technique used in this tutorial has made use of JavaScript to automatically stage the opening of each menu link. While this could be manually created using the nth-child selector or inline styling, this approach allows new items to be added quickly without the need for additional formatting.

Leon Brown, web developer and trainer

Technique

1. Initiate document structure

The first step is to define the HTML document, which consists of the HTML container storing the head and body sections. While the head section is mainly used to load the external CSS and JavaScript resources, the body section is used for the page content created in Step 2.

```
<!DOCTYPE html>
<html>
<head>
<title>Staged Menu Open</title>
<link href="styles.css" rel="stylesheet" type="text/css" />
<script src="code.js"></script>
</head>
<body>
*** STEP 2 HERE
</body>
</html>
```

2. HTML menu content

The menu consists of two parts - a link to open the menu, and a navigation container for the menu links. Take note how the 'href' attribute of the first link is a reference to the ID of the nav container.

```
<a href="#menu">Open</a>
<nav id="menu" class="animated">
<a href="#">&times;</a>
<a href="#">Item</a>
<a href="#">Item</a>
<a href="#">Item</a>
<a href="#">Item</a>
</nav>
```

3. Nav container style

The navigation container is set to be hidden by default. When its ID is targeted by the URL, it will appear as a block element covering half the browser window with a black border. The absence of a height definition will result in the navigation container adapting to the height created by its inner contents.

```
nav{
display: none;
}
nav.target.animated{
display: block;
width: 50vw;
padding: 1em;
border: 1px solid #000;
}
```

4. Navigation elements

Each first-level element inside the targeted navigation container is set with relative positioning and padding for spacing. Initially, these items are set to be invisible through the use of '0' opacity. Each item is animated into view via the applied 'open' animation defined in Step 5.

```
nav.target.animated > *{
display: block;
position: relative;
animation: open 1s forwards;
opacity: 0;
padding: 1em;
text-decoration: none;
color: #777;
}
```

5. Animation definition

The 'open' animation referenced in the previous step animates the menu items vertically from two character elements above where they should be positioned. This is made possible through their relative positioning. Additionally, their transparency are animated from invisible to fully visible.

```
@keyframes open {
0% {top:-2em; opacity:0;}
50% {opacity:1;}
100% {top:0;}}
```

6. Staged animation

The problem at this point is that all of the navigation elements appear at the same time. This step's JavaScript finds each of the navigation items and adds an animation delay - increased by 0.2 seconds for each new item. The result is each item appears after the previous one.

```
window.addEventListener("load", function(){
var time = 0;
var nodes = document.querySelectorAll("nav.animated > *");
for(var i=0; i<nodes.length; i++){
nodes[i].style.animationDelay = time+"s";
time += 0.2;
}});
```

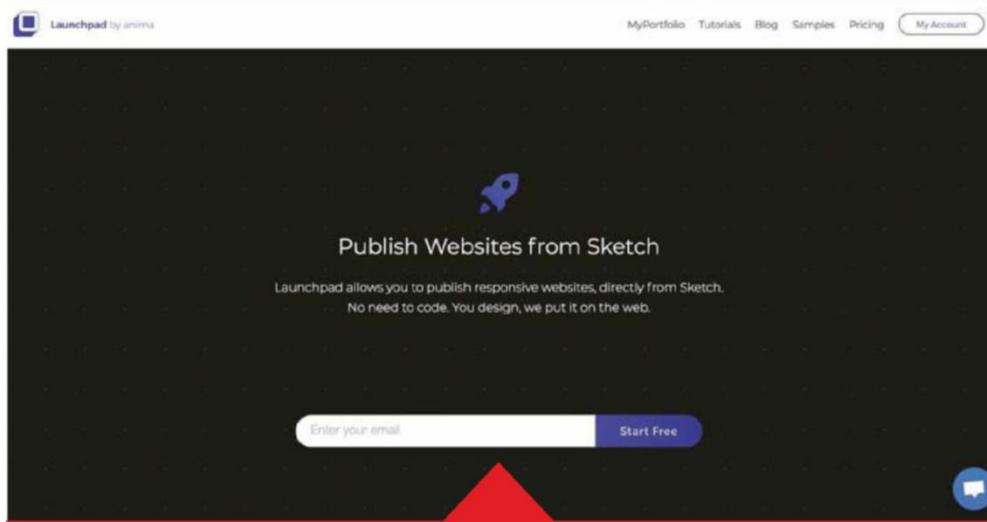


20 HOT TOOLS TO TRY TODAY

Get up to date with some fresh tools and technologies you might not have heard of

The web changes extremely quickly, and you'd be forgiven for not keeping pace with the continuous release of new tools and technologies. Yet it can be very beneficial to follow the latest developments, so you don't have to wait for things to become mainstream before you can take advantage of them. The chances are that if there's a problem which has frustrated you, or something you've thought could be made easier, someone will have come up with a solution for it. You only have to look at the proliferation of front-end frameworks available as evidence of the level of choice you have.

This pace of change and range of choice applies to all areas of the web. We'll look at some of the latest design tools, dev tools, libraries and frameworks you have to choose from. Depending on your use case, you may not want to go too cutting-edge, especially with libraries and frameworks given their occasionally short lifespan. But ultimately the best thing to do is often to go and experiment.



Launchpad by anima

MyPortfolio Tutorials Blog Samples Pricing My Account

Publish Websites from Sketch

Launchpad allows you to publish responsive websites, directly from Sketch. No need to code. You design, we put it on the web.

Enter your email Start Free

A screenshot of the Launchpad website. It features a dark background with a central white section. At the top, there's a navigation bar with links for MyPortfolio, Tutorials, Blog, Samples, Pricing, and My Account. Below the navigation, the main heading is "Publish Websites from Sketch" with a small rocket icon above it. A subtext below the heading reads: "Launchpad allows you to publish responsive websites, directly from Sketch. No need to code. You design, we put it on the web." At the bottom of the main section, there's a form with a placeholder "Enter your email" and a blue "Start Free" button. To the right of the main content, there's a sidebar with a teal arrow pointing upwards.

LAUNCHPAD

Build websites from Sketch without coding

www.animaapp.com

1 There's a good chance you're already using Sketch for web and mobile design, since it's rapidly risen to become one of the most popular tools on the market. But until now, it was purely a design tool and you then had to take your designs and build them as a real site.

Launchpad is a Sketch plugin which enables you to automatically build working HTML and CSS from your designs, and publish it as a live site. This offers an incredibly quick way to get static websites up and running, and while dynamic JavaScript functionality isn't supported, there is the capability to include forms to capture information from your users.

You can get started for free if you're happy to use a Launchpad domain, but there's a commercial offering if you want custom domains and Launchpad's branding hidden.



LISTS

www.lists.design

2 If you're designing pages and tired of placing Lorem Ipsum everywhere, Lists could be just what you need. It's a collection of realistic data of all different types which you can use to populate your mockups. Want a list of British addresses to use on a page, or fake credit card numbers? Both those things are available. Lists covers a wide range of different data types, and for each you can download the data in JSON format to use. This is beneficial as it both makes your designs look more convincing, and can help users understand context within the page when evaluating prototypes. The team behind Lists also invite contributions on GitHub if you have new data types you think should be added.

TRUEUX

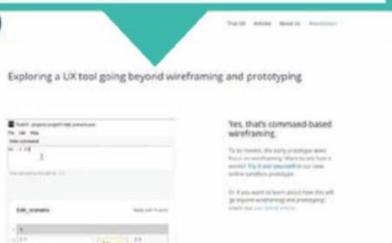
www.trueuxapp.com

3

TrueUX is setting out to provide an alternative to low-fidelity prototyping tools like Balsamiq or OmniGraffle. It's still in its early stages, and exists only as an online sandbox at this point. However, it introduces some unique concepts which provide an interesting spin on traditional wireframing.

The TrueUX sandbox offers a text command interface which allows you to create and modify prototypes by typing brief commands with the keyboard, rather than clicking and dragging elements on the screen. This introduces a small learning curve, but the theory is that once you're familiar with it, you should be able to make changes much more quickly.

If you like the concept of the text command-driven wireframing, it's worth watching further developments around this tool as the author has many other aspirations for it.



FONTPMAP

<http://fontmap.ideo.com>

5

Choosing the right font for a page or app can be tough. Google Fonts is a great source of freely-available fonts you can include in your work, but maybe you've thought that there was something missing in terms of ease of use and discoverability.

FontMap is an interesting experiment which uses AI to group fonts from Google Fonts together, showing you a visualisation. This might seem simple, but it solves the common problem of finding fonts which are similar. The user interface is very straightforward, and you can spend quite some time in the tool examining the fonts it has laid out. Once you've made a selection, Google Fonts is only a click away to go ahead and include the font in your work.

Creativity & Design Marketing & Analytics PDF & E-Signatures Business Solutions Support

xd ADOBE XD CC Features What's New Learn & Support System Requirements Free Trial More Products Buy now

Introducing Adobe XD. Design. Prototype. Experience.

Go from concept to prototype faster with Adobe XD, the all-in-one UX/UI solution for designing websites, mobile apps and more. With smooth, powerful performance, it's easy to deliver experiences that work and feel as good as they look on any screen.

ADobe XD

Adobe's answer to Sketch for UX design
www.adobe.com/uk/products/xd.html

4

Where they once dominated the UX landscape with Photoshop and Illustrator, arguably Adobe have lost ground to Sketch in recent years. Only recently out of beta, Adobe XD is the company's attempt to win back UX designers who may have jumped ship. It offers a familiar interface for users of other Adobe products, and offers interactive prototyping without the need for plugins. Collaboration features, key for many UX designers these days, are said to be coming in the future.

Adobe XD looks to be shaping up as a very good product, and is also likely to be particularly popular with Windows users for whom Sketch is unavailable. Of course, however, it comes with a price tag, and it may take a few more feature additions to convince many designers to switch.

20 HOT TOOLS TO TRY TODAY

REASONREACT

<http://bit.ly/2HwAkW2>

6 You might not be familiar with Reason. Reason is a relatively new language developed by Facebook which offers a JavaScript-like development experience using OCaml. Reason code can be compiled to JavaScript using an engine called BuckleScript, which is essentially an OCaml-to-JavaScript compiler. What this means in practice is that you can write statically-typed, functional code in a language with syntax combining elements of both OCaml and JavaScript, and then compile it to vanilla JavaScript. The closest comparison is probably TypeScript, which itself offers an abstraction on JavaScript with static typing, albeit with significantly different syntax. ReasonReact extends this, allowing you to write React components using Reason code, and is made by the creator of React. If you like React, this is one to take a look at.



WORKBOX

<https://developers.google.com/web/tools/workbox>

8 Service workers are among the most exciting new technologies on the web. They allow modern web apps to store resources locally and provide offline functionality. However, if you've tried working with them, you'll rapidly find that maintaining caches properly is a complex problem. This is where Workbox comes in. It is a set of build tools and libraries designed to make it easier to create service workers and hence easier to build offline apps. In simple terms, you tell it which resources you want to cache and it will generate the service worker code required to cache and update your resources. As you begin to build more complex sites with offline features, this abstraction is likely to make your life a lot easier.

Finance					
API	Description	Auth	HTTPS	CORS	Link
Alpha Vantage	Realtime and historical stock data	apiKey	Yes	Unknown	Get
Barchart OnDemand	Stock, Futures, and Forex Market Data	apiKey	Yes	Unknown	Get
Consumer Financial Protection Bureau	Financial services consumer complaint data	apiKey	Yes	Unknown	Get
IEX	Stocks and Market Data	No	Yes	Unknown	Get
iG	Spreadsheets and CFD Market Data	apiKey	Yes	Unknown	Get
Paid	Connect with users' bank accounts and access transaction data	apiKey	Yes	Unknown	Get
Reazoney iFSC	Indian Financial Systems Code (Bank Branch Codes)	No	Yes	Unknown	Get
RoutingNumbers.info	ACH/NACHA Bank Routing Numbers	No	Yes	Unknown	Get
VAT Rates	A collection of all VAT rates for EU countries	No	Yes	Unknown	Get

Food & Drink					
API	Description	Auth	HTTPS	CORS	Link
BreweryDB	Beer	apiKey	No	Unknown	Get
Edamam	Recipe Search	apiKey	Yes	Unknown	Get
FoodFork	Recipe Search	apiKey	No	Unknown	Get

PUBLIC APIs

<http://bit.ly/1UqyKJi>

7 As you start to put more complex apps together, it's likely that you'll have some use for a variety of APIs. Yet finding appropriate APIs for your use case isn't always easy, and even harder is finding ones which are publicly available and can be used for free.

It is of course a work in progress, but the Public APIs page is an effort to consolidate and document a list of all the publicly available APIs out there. If you want an API to get a weather forecast, for example, it has a list of those available, and you might choose to utilise MetaWeather, which you can immediately see requires no authentication. To be kept alive it needs ongoing community contribution to keep up to date with API availability.



The screenshot shows the Workbox documentation and tool interface. At the top, there's a navigation bar with links for 'Web', 'Fundamentals', 'Tools', 'Search', and a user profile icon. Below that is a header for 'Tools for Web Developers' with tabs for 'HOME', 'CHROME DevTools', 'LIGHTHOUSE', 'PUPPETEER', 'WORKBOX', and 'CHROMIUM'. The main content area is titled 'Workbox' and includes sections for 'Overview', 'Get Started', 'Examples', 'CodeLab', 'Reference Docs', 'Community', 'GitHub', and 'Next Release'. There's also a 'Beta Release' section with a summary of changes and a link to the 'Getting Started Guide'. At the bottom, there's a logo for 'Workbox' and a brief description of what it does.

PRETTYER

Reformat your code

<https://prettier.io>

9 Consistently formatting your code in a readable style can make a huge difference to the maintainability of your work, especially if you're working as part of a team. Fortunately, there are a multitude of tools out there available to help address this problem, enabling you to save time thinking about code style or discussing it in code review. Prettier is one such tool. It is highly opinionated, so takes away the effort of having to think about and define your style guide in favour of its own standards. It can be integrated either with your build pipeline or most popular text editors via plugins, and understands many of the languages you'll want to work with including JavaScript, JSX, TypeScript, Vue, CSS, Less and SCSS.



“THIS IS PROBABLY THE QUICKEST AND MOST COMPREHENSIVE WAY TO TEST YOUR PAGE ACROSS SO MANY SCREEN SIZES”

A screenshot of the Sizzy web application. On the left, there's a sidebar with settings like 'Filter devices by OS' (iPhone 4, iPhone 5, iPad), 'Filter devices by type' (Smartphone, Tablet), and 'Settings' (Hide dimensions, Show keyboard, Reset all settings, Switch to landscape, Switch theme). The main area shows two device preview windows side-by-side. The top window is for an iPhone 4 (320x480) displaying an article from Creative Bloq titled '5 of the best JavaScript frameworks'. The bottom window is for an iPhone 5 (320x568) showing a similar article. At the bottom of the main area, there's a 'Zoom (100%)' slider.

SIZZY

<https://sizzy.co>

10

You want to make sure all the effort you've put into responsive design has paid off, right? Sizzy is designed to help you do just that. It comes in either web or browser plugin form, and will show you, side-by-side, an interactive view of your page rendered on a number of different device screen sizes. In addition to this, you can also show and hide a simulated device keyboard, and switch between portrait and landscape modes.

You can also use the functionality locally with your development server by providing it with a localhost URL, so you don't need to put your site live to have Sizzy test it out. This is probably the quickest and most comprehensive way to test your page across so many screen sizes.

DRAGGABLE

Get impressive interactions
<https://shopify.github.io/draggable>

11

Built by the team at Shopify, Draggable does very much what its name implies: it's a library providing drag-and-drop support, abstracting fairly complex functionality. It supports fast manipulation of the DOM to reposition elements, which means you can swap and sort objects, and will work fine with mouse, touch and force touch events. It also supports all major browsers. Good practical use cases for this might be things like interactive calendars or Kanban boards. The project's website also has some impressive and colourful demos showing what's possible using the library, including 3D interactions. It's currently in beta, which means the API is subject to change, but that shouldn't stop you giving it a try.

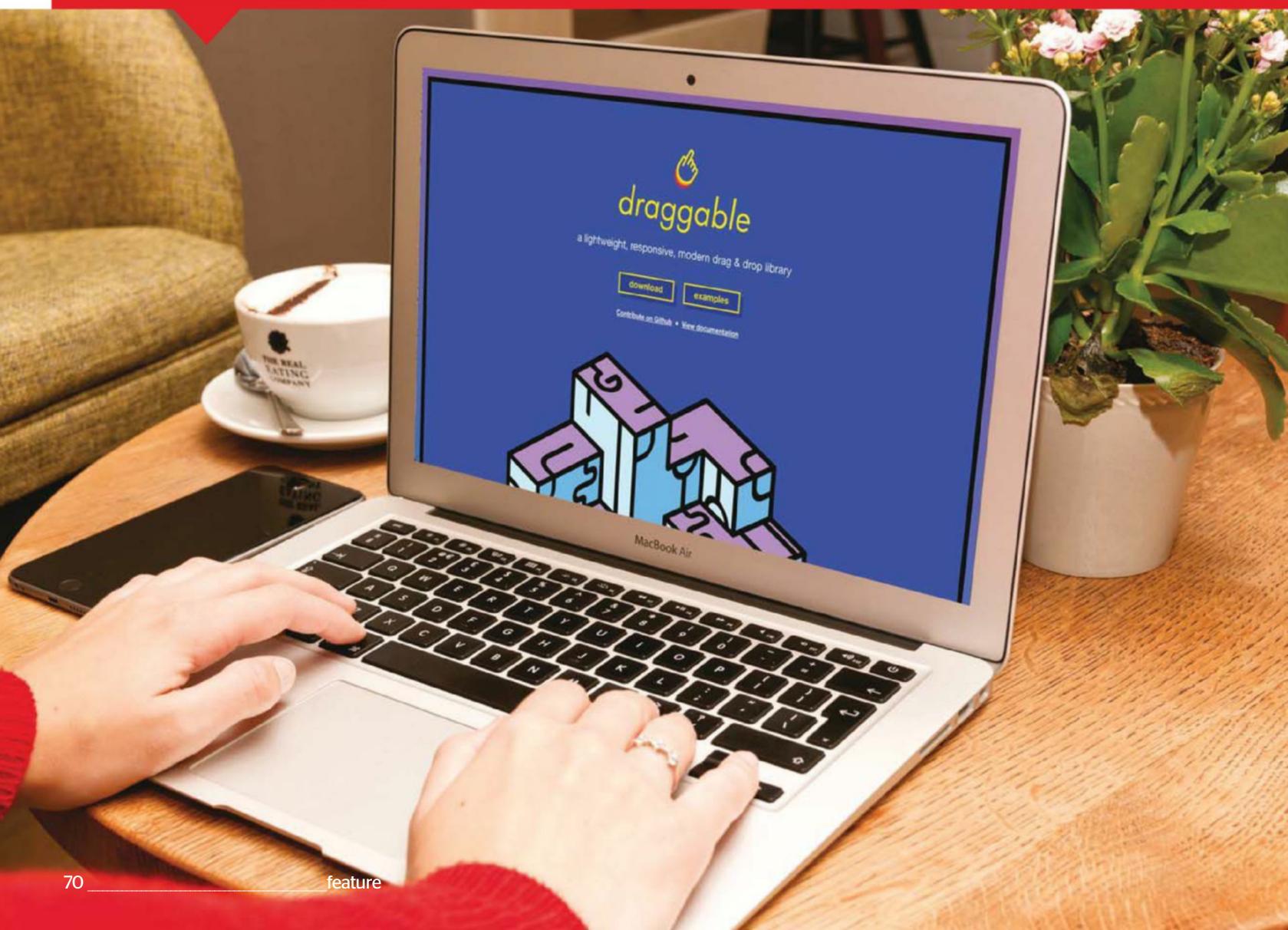
BOTTENDER

<https://bottender.js.org>

12

Conversational interfaces, or chatbots, have exploded in popularity in recent years. You might think that building a bot is a fairly complex undertaking compared to putting together a single-page app, but it doesn't have to be.

Bottender is a JS framework for building cross-platform bots running on Node.js, and comes with support for popular messaging platforms like Facebook Messenger, Telegram and Slack. For testing purposes, it also makes it easy to run your bot in the console. It's very early days yet for Bottender, but this is one to keep an eye on as this type of project opens up many opportunities to build interfaces which would otherwise have been out of reach.

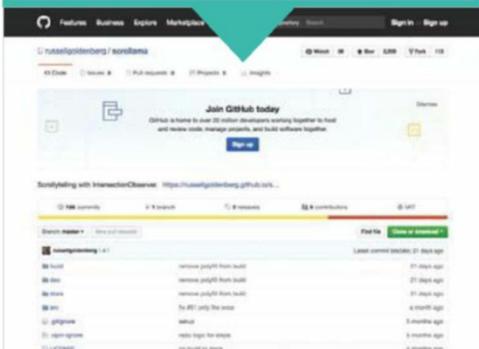


SCROLLAMA

<http://bit.ly/2InIK4v>

13

You've probably seen some impressive examples of scrolling interactive storytelling. If you aren't familiar with this, take a look at The Boat (<http://www.sbs.com.au/theboat>). It's an engaging way to tell a story online, using native browser functionality. Yet it isn't necessarily straightforward to implement, as handling scroll events can be fiddly and introduce performance issues. Scrollama is designed to help. It uses IntersectionObserver, which while relatively new, is more performant than previous solutions to detect whether an object is visible on the page. Scrollama have a range of examples illustrating what you can do, and if you don't quite like their way of doing things, their GitHub page offers alternatives.



MAPTALKS

<https://maptalks.org>

15

There are a whole host of use cases for maps within the browser, and of course Google remains the household name for mapping. But what if you could go one better than Google Maps? Maptalks allows you to display both 2D and 3D maps together, while retaining the simplicity of a 2D mapping library.

In addition to the basic 2D and 3D functionality, it also has a plugin ecosystem which allows you add additional functionality such as heat maps, cluster markers, and origin/destination lines. There's also an experimental plugin to add animated wind. The site is rich with examples, so check it out if you're having trouble visualising what you might be able to do.



VIVALDI

<https://vivaldi.com>

14

Chances are that statistically you're probably using Chrome, Firefox, Safari or Edge. These four browsers dominate the web. Yet others are available, and depending on your preferences it might be time to branch out. One which has gained some attention of late is Vivaldi, which is designed to be fast, minimalistic and powerful. One particularly interesting feature is the ability to annotate pages with notes. Vivaldi uses the Blink rendering engine, as do Chrome and Opera, which gives users the ability to install Chrome extensions, many of which will work. For many, however, the leading feature is the ability to manage tabs in 'stacks', which makes it far easier to work with the large numbers of open tabs many of us frequently work with.

MAPTALKS

Gallery Examples API Docs

For More Map Insights

An open-source javascript library for integrated 2D/3D maps.

Getting Started

Download
version: 0.39.1

★ Star 1,340

20 HOT TOOLS TO TRY TODAY

BOOTSTRAP 4

Major overhaul for the most popular front-end component library

<https://getbootstrap.com>

16

To this day, Bootstrap remains the most popular front-end component library out there, and lets you build great-looking, responsive user interfaces in no time. Bootstrap itself might not be new, but the stable version 4 dropped in January and brings a host of new features and changes.

Bootstrap 4 includes a move to favour Sass instead of LESS, utilise relative units instead of pixels, a new flexbox layout for grids, and much more. It's also worth noting that it has now dropped support for IE8 and IE9, which for most shouldn't be a problem.

You might be using Bootstrap with a framework such as Angular or React, and the good news is that while the stable release of version 4 is fairly new, there are already modules available to integrate it into those ecosystems.



SVELTE & SAPPER

<https://svelte.tecnology>
<https://sapper.svelte.tecnology>

18

A common criticism of the most popular front-end frameworks is the code bloat and initial page load performance penalty. Svelte offers similar abstractions to make building component-driven apps easily, but generates JavaScript at build time, resulting in a lower footprint and faster page render.

Sapper goes a step further than this, and allows you to build universal web apps composed of Svelte components. It will take care of the complexities of both server-side and client-side rendering, offline support, and much more.

A screenshot of the Sapper website. The top navigation bar includes links for 'Build', 'Chat', 'Svelte', and 'GitHub'. The main title 'SAPPER' is in large green letters. Below the title, it says 'Military-grade progressive web apps, powered by Svelte'. There are three bullet points: 'Beautiful development experience, inspired by Next.js.', 'Ridiculous performance.', and 'Modern best practices baked in.' At the bottom is a green 'LEARN SAPPER' button.

NEXT.JS OR NUXT.JS

<https://learnnextjs.com>
<https://nuxtjs.org>

17

React and Vue are two popular front-end frameworks. However, setting them up so you are ready to start building an app can be time-consuming, as you need to install modules, configure your build pipeline and more. The likes of Create React App can make this quicker, but there's still quite a learning curve to get started for things like routing. And if you want server-side rendering, which can be extremely useful, this adds more complexity. Next and Nuxt both set out to solve the same problem, for React and Vue respectively. They allow you to quickly create apps using these frameworks which by default are server-rendered, and automatically handle routing based on your page structure.



"A COMMON CRITICISM OF THE MOST POPULAR FRONT-END FRAMEWORKS IS THE CODE BLOAT AND INITIAL PAGE LOAD PERFORMANCE PENALTY"



STENCIL

<https://stenciljs.com>

19

By now you've probably heard a lot about web components. Google's Polymer project is probably the best known framework for building web components, offering both browser support polyfills and useful abstraction to get started.

Stencil, however, offers an alternative way to create web components. Built by the team behind the Ionic framework, you can use Stencil to quickly create custom HTML elements with all the functionality you need wrapped up in them. Because it uses the web component standards, your creations aren't tied to a particular framework. Stencil may be particularly appealing to those familiar with React, since it takes a lot of influence from it.

stencil

Docs Demos PWAs Resources GitHub



The magical, reusable web component compiler

GET STARTED

LEARN MORE

Watch launch video



Simple

With intentionally small tooling, a tiny API, zero configuration, and TypeScript support, you're set.



Performant

6kb min+gzip runtime, server side rendering, and the raw power of native Web Components.



Future proof

Build versatile apps and components based 100% on web standards. Break free of Framework Churn.

PARCEL

An alternative to Webpack
<https://parceljs.org>

20

Webpack is probably the leading module bundler in the JavaScript environment. If you aren't familiar with it, it essentially takes your assets and module dependencies and bundles them into static, deployable files. Parcel is relatively new, and has the same objective, but takes a different approach. It can be run with no configuration at all, and will examine your code to identify dependencies. It also offers multi-core support; Webpack doesn't, meaning that builds can run significantly faster. Parcel can also handle pre-processing such as with Babel or Sass. One thing, however, is that most information online for popular front-end frameworks will assume you're using Webpack for now. If Parcel takes off, this could well change in future.

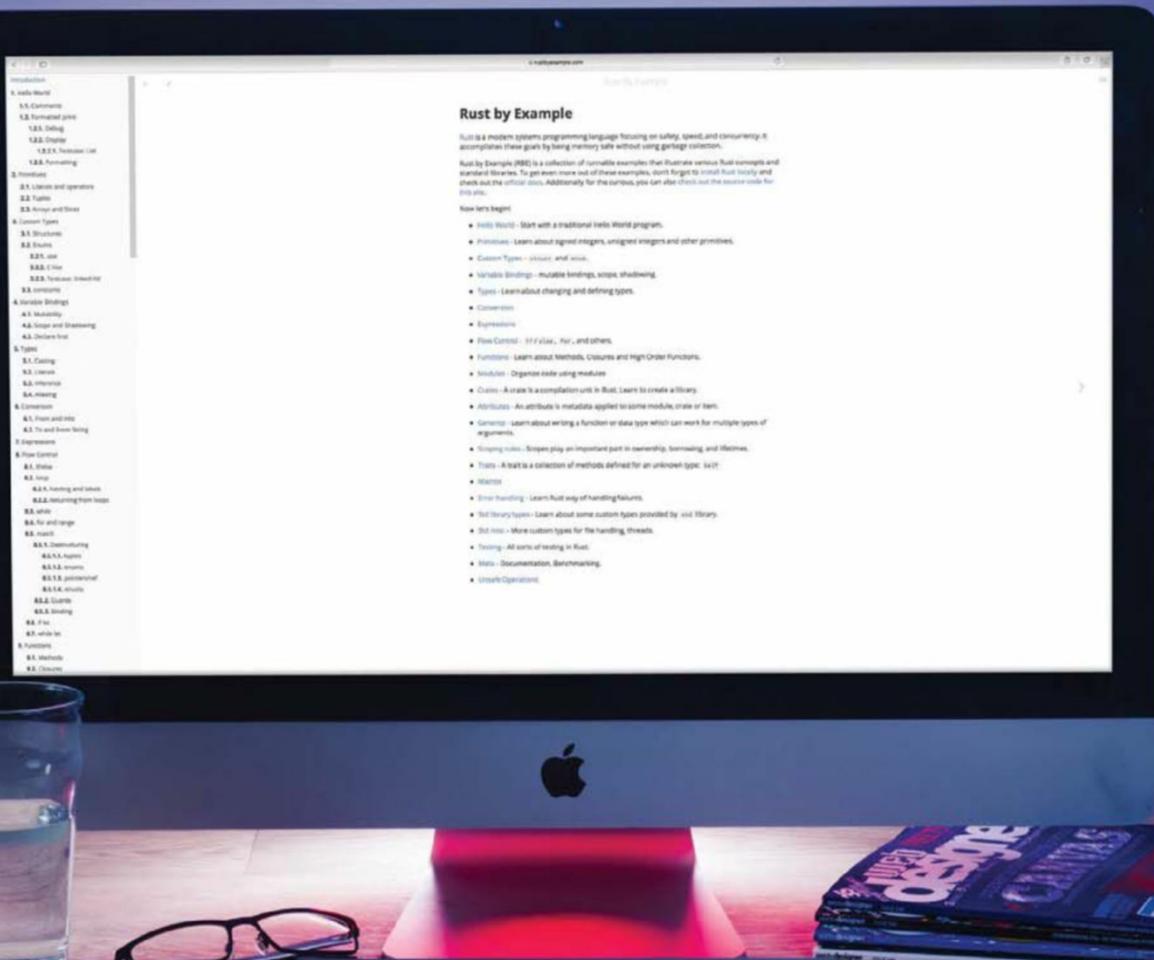


GET STARTED WITH RUST

The C programming language has truly passed the test of time. There are very few environments where it does not thrive. This is mainly due to its high execution performance, which, unfortunately, comes at a price: C does not support many of the features expected in a modern programming language.

Mozilla Research's Rust is an attempt to create a better mousetrap. Its language design remains focused on high performance and being close to hardware. However, its syntax and compiler also take advantage of the various benefits offered by modern programming language research.

Given that Mozilla's mission is the creation of an 'open web', it should not be surprising that the company is working tirelessly to bring its latest brainchild to the web. While Rust can, in theory, run on the client via systems such as Emscripten, its real power lies in the creation of efficient backend services.



HOW TO START USING RUST

While package managers such as 'apt-get' surely made developer's life easier, package list maintainers are, by and large, known to be no friends of excessive speed. Because of that many, if not most, distributions' package caches are heavily outdated.

The Rust team has solved this problem by providing a dedicated installation script, which sets up the system it is run on efficiently. Deployment, then, is a two-step process: first, make sure that the CURL downloader/parser is available:

```
tamhan@tamhan-thinkpad:~$ sudo
apt-get install curl
[sudo] password for tamhan:
Reading package lists... Done...
```

Then, in the second step, download the installer and pass it on to 'sh' using the pipe operator. 'Sh' is a shorthand for the default shell interpreter of your workstation, which will then proceed to run the code at hand:

```
tamhan@tamhan-thinkpad:~$ curl
https://sh.rustup.rs -sSf | sh
info: downloading installer
```

During the installation, the Rust installer will display a variety of prompts similar in appearance to the ones shown in the picture on the right. Simply follow them to achieve a default installation.

In some cases, the Rust deployment will fail with an error which is similar to "error: could not write rcf file file: '/home/tamhan/.bash_profile'". If this happens during installation, use 'sudo -s' to get a root shell, and then rinse and repeat to proceed.

RUN A SAMPLE!

With that out of the way, it is time to run our first small program. Rust files, by default, have the file extension '.rs'. Create a file called 'firsttest.rs', and then provide it with the following piece of code:

```
fn main() {
    println!("Hello World!");
}
```

C and C++ programmers often wonder why invocations of the 'println' method require the use of an exclamation mark. The answer is simple: Rust implements 'println' as a macro, which gets invoked in a different way.

Given that Rust is a compiled language, our example must be processed before it can be run from the command line:

```
tamhan@tamhan-thinkpad:~/rustspace$ rustc firsttest.rs
tamhan@tamhan-thinkpad:~/rustspace$ ./firsttest
Hello World!
```

ADVANCED DECAY!

Describing a complete programming language, such as Rust, in the limited space we have available here is impossible. So let's start by formally

pointing you to Mozilla Research's comprehensive documentation at rustbyexample.com and shown on the first page. One feature which is sure to impress C programmers involves the use of the 'match' command: in addition to direct comparisons, it also enables the use of range operators to greatly simplify the design of advanced programs:

```
fn main() {
    let number = 5;
    println!("Working on {}", number);
    match number {
        1 => println!("One!"),
        2 | 3 | 5 | 7 => println!("Prime"),
        _ => println!("A number"),
    }
}
```

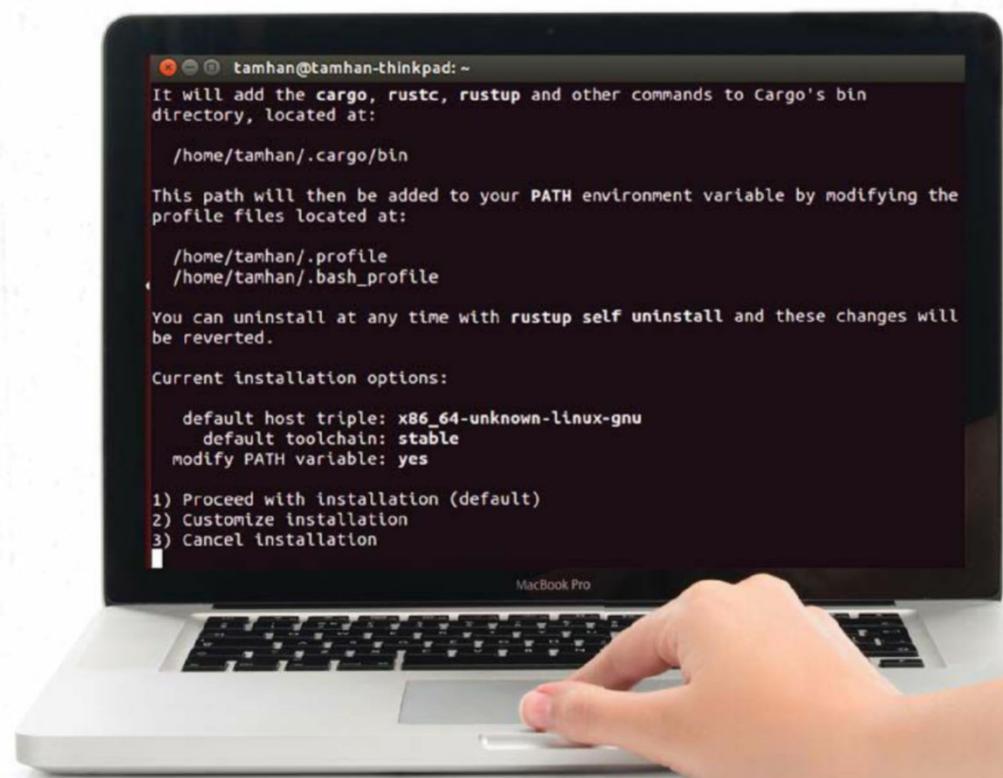
GARBAGE-AT-HAND!

Classic garbage collection has its weaknesses: most implementations bring the program to a screeching halt from time to time. To get around that Rust uses a set of so-called 'zero-cost abstractions' to emulate a similar behaviour in a less annoying fashion.

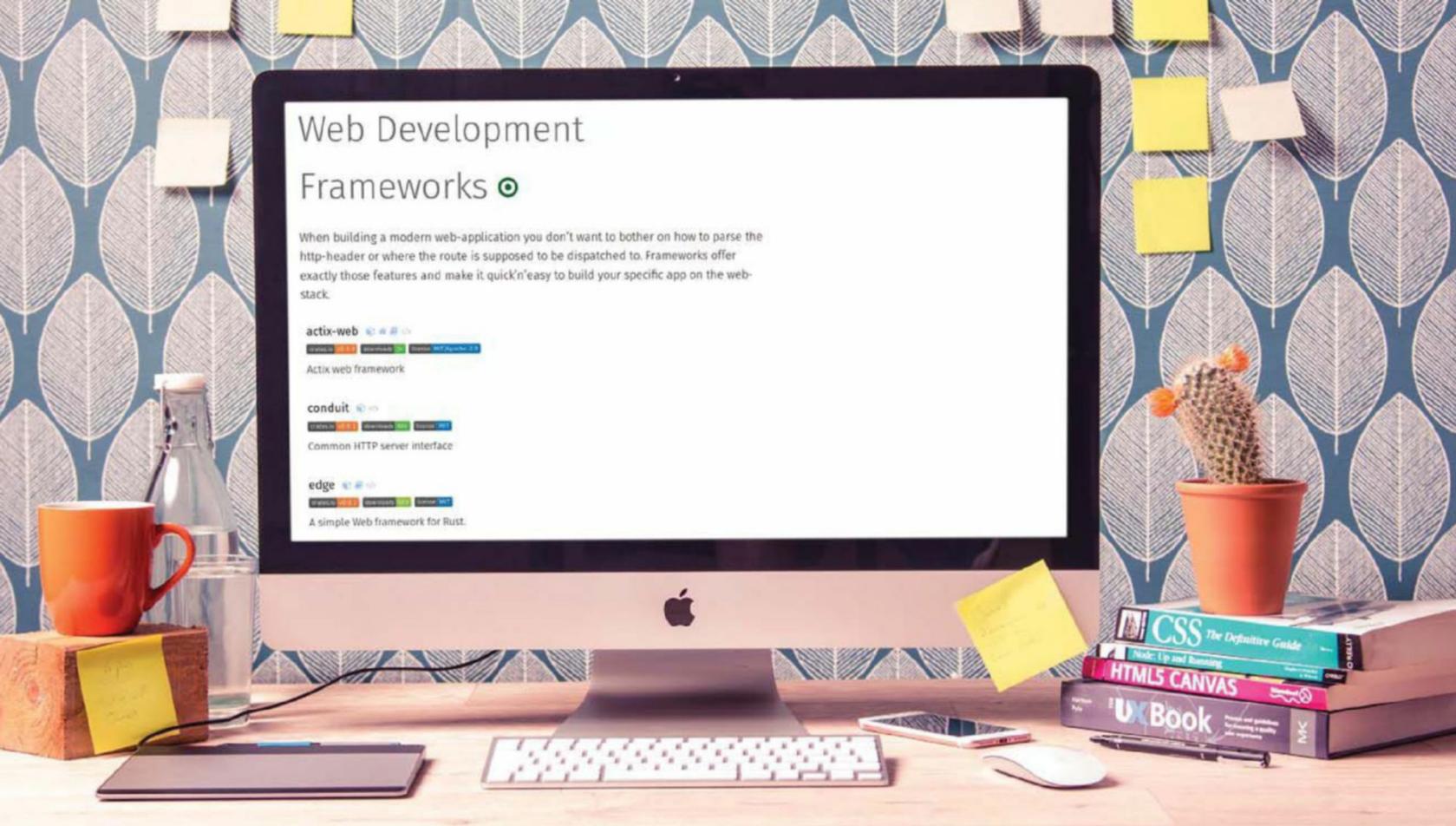
In principle, every resource is created with an owner in a fashion similar to Qt's parent-child subsystem. However, a feature called 'borrowing' enables the programmer to transfer ownership between resources temporarily, thereby passing them around the system.

Unfortunately, a complete discussion of the possibilities of Rust's memory manager would break the limits of this article - let it suffice to declare that Rust provides a very unorthodox, but workable form of memory management.

"A FEATURE CALLED BORROWING ALLOWS THE PROGRAMMER TO TRANSFER OWNERSHIP BETWEEN RESOURCES"



The RUST installer provides a semi-graphic installation environment



CREATE A NEW PROJECT

While C and C++ code can be broken down into libraries, doing so is an annoying and somewhat uncomfortable task. Furthermore, making sure that all the required libraries are where they are needed is something even seasoned developers like to avoid.

The JavaScript community has long solved this problem via products such as NPM. Projects are described via a project structure file, which - among other things - contains references to libraries and other elements needed during the actual compilation run.

In the case of Rust, a package manager called Cargo does a similar job. Creating a new project can be accomplished through the use of the 'cargo new' command in a fashion similar to the following:

```
tamhan@tamhan-thinkpad:~/rustspace$ cargo new futuresample1 --bin
```

```
tamhan@tamhan-thinkpad:~/rustspace/futuresample1$ tree
.
└── Cargo.toml
   └── src
      └── main.rs

1 directory, 2 files
```

Similarities to NPM are purely coincidental...

```
Created binary
(application) 'futuresample1'
project
tamhan@tamhan-thinkpad:~/rustspace$
```

When the creation process is complete, a folder structure similar to the one shown in the picture at the bottom of this page will be generated.

.toml files act as 'controllers' - in the case of our newly-generated project, the file contains the following structure:

```
[package]
name = "futuresample1"
version = "0.1.0"
authors = ["tamhan"]
[dependencies]
```

Similarities to the .ini files of lore are not purely coincidental: in a fashion not dissimilar to NPM, the ancient configuration file format is also used to

describe the configuration files used for Rust projects. The '[dependencies]' block is of special interest for us - it contains a list of all external libraries, which need to be present for the compilation to succeed.

ADD A PACKET!

Creating a web server by hand is an ungrateful and annoying job best left to masochists. We will, instead, opt for a framework. Sadly, finding the right one is not easy - as shown in the picture above, visiting arewewebyet.org/topics/frameworks/ reveals an excessively long list of candidates.

We will pick Rocket, if only because it seems to be quite popular and has seen an update released in January of 2018. Sadly, Rocket's developers tend to take a liking to newly-introduced language features, which is why frequent updating of your Rust installation using the following commands is required:

```
tamhan@tamhan-thinkpad:~/rustspace/futuresample1$ rustup
update && cargo update
```

The next step largely is a question of taste. Most libraries come with pre-provisioned starter projects, which

developers can simply siphon from GitHub. Doing so for your 'prime' library is not necessarily a bad idea - although once more than one library is involved, a manual approach tends to be more fruitful.

Next, open the .toml file, then modify the 'dependencies' section as per the following in order to include a recent version of the Rocket framework:

```
[dependencies]
rocket = "0.3.6"
rocket_codegen = "0.3.6"
```

Rocket is unique in that it requires the inclusion of a total of two packages: in addition to the main framework, a separate code generator file is also required. Either way, our version includes a specific version of the two libraries - Cargo can also accept wildcards, which enable the program to 'pick its poison' without any help from us.

With that out of the way, one problem remains: enter 'cargo run' in the folder containing the .toml file in order to perform an assisted compile, which will - among other things - download the relevant code libraries from the repository and compile the whole enchilada for you.

BEWARE OF SUDDEN LIBRARY DEATH

Being able to 'simply include' libraries from repositories such as NPM or the Crate service tends to make developers believe that anything available in the repo is also fair game for actual products. Be aware that this is not the case.

Most libraries, whether large or small, are supplied by 'community vendors'. If the person(s) in charge lose interest, further development of

the product is likely to be at peril. This is not theoretical: one good example in the Rust community is the Iron framework, which once was considered the leading standard for web frameworks in the newly-developed language.

If you visit the GitHub repository for the Iron framework, you will be greeted with a warning, which categorically states:

NOTE: Iron is not actively maintained at the moment, please consider using a different framework

For a developer currently using Iron, this spells, but does not confirm, trouble. If the library works for you, keep using it. But, always be aware that eventually bugs will not be fixed - unless you are willing to take on maintenance yourself, that is.

"DEVELOPERS BELIEVE THAT ANYTHING AVAILABLE IN THE REPO IS ALSO FAIR GAME"

CREATE SOME SCAFFOLDING...

Simply including and downloading a library is no fun: we want to see Rust in action. To achieve that, a sample program must be written - the scaffolding for which is the topic of the following steps.

Before we can really get coding, however, a small problem must be fixed. Rust's compiler does not allow for the use of advanced language features by default - if your application cannot be compiled due to feature use, you will need to fix the problem via the following command sequence:

```
tamhan@tamhan-thinkpad:~/rustspace/futuresample1$ rustup override set nightly
info: syncing channel updates for 'nightly-x86_64-unknown-linux-gnu'
...

```

Applying the 'set nightly' command in a folder containing a .toml file modifies it to mark its contents to be run using the latest version of rustc - with the flag set, the compile process

should succeed. Next, open 'main.rs' and replace its contents with the following code:

```
#![feature(plugin)]
#![plugin(rocket_codegen)]

extern crate rocket;

#[get("/")]
fn index() -> &'static str {
    "Hello, world!"
}

fn main() {
    rocket::ignite().mount("/", routes![index]).launch();
}
```

Invoke 'cargo run' after saving the changes to see the output shown in the figure below. The package manager isn't limited to loading code, but can act as an advanced build tool.

3, 2, 1 LIFTOFF!

Rocket's developers, obviously, were

inspired by the work of missile teams: make of this what you will. Like most other web frameworks, the actual applications are created as a collection of 'routes', which are assigned to a web server class. In this case, but one route is created - a 'get' call against '/' will yield the returning of the string 'Hello World'.

Incidentally, the main issue faced by developers coming to Rust from other languages is the somewhat odd syntax. Function return types are declared via an arrow following the header:

```
fn is_divisible_by(lhs: u32,
rhs: u32) -> bool {
    if rhs == 0 {
        return false;
    }
}
```

Careful onlookers will determine that the snippet above generates a function returning a Boolean value: Rust knows about a few dozen data types, which must be formally specified at declaration to prevent the passing of invalid types in a fashion similar to TypeScript.

While the 'return' statement is supported by Rust; a special case occurs whenever the last line of a function is an expression. It is considered the 'return' value - a good example for this would look as per the following code:

```
fn is_divisible_by(lhs: u32,
rhs: u32) -> bool {
    ...
}
```

```
lhs % rhs == 0
}
```

With that now out of the way, our next step involves the creation of a brand new route:

```
#[get("/world")]
fn world() -> &'static str {
    "A new route!"
}
```

Rust's language design advocates the use of attributes: the elements inside the '#[' construct are additional properties, which get applied to any element standing nearby.

In our particular case, the affected element is a function going by the name of 'world':

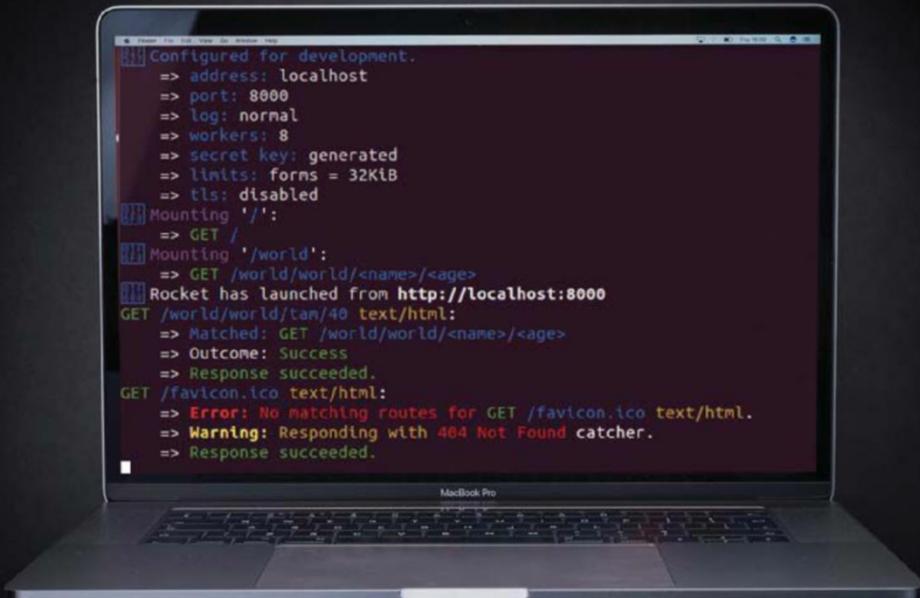
The next problem involves adding the new route to the above-mentioned web server element. This is easily accomplished as per the following:

```
fn main() {
    rocket::ignite().mount("/", routes![index]).mount("/world",
routes![world]).launch();
}
```

This code is interesting mainly because of the use of the code generator: 'mount' takes the 'routes!' macro, which generates code on the fly. With that out of the way, you can now perform another recompile, which will enable you to convince yourself of the correctness of our code - the Rocket handler will now detect a total of two routes.

```
tamhan@tamhan-thinkpad:~/rustspace/futuresample1$ cargo run
Compiling futuresample1 v0.1.0 (file:///home/tamhan/rustspace/futuresample1)
Finished dev [unoptimized + debuginfo] target(s) in 2.50 secs
Running `target/debug/futuresample1'
Configured for development.
=> address: localhost
=> port: 8000
=> log: normal
=> workers: 8
=> secret key: generated
=> limits: forms = 32KiB
=> tls: disabled
Mounting '/';
=> GET /
Rocket has launched from http://localhost:8000
```

Rust's package manager can also run compiled programs



Rocket's Runner is extremely talkative

IN-DEPTH ANALYSIS

Providing resources on request might make for a nice demo, but is lacking in practicability. A more interesting test involves accepting parameters from the client, and using them to modify the system behaviour as a whole.

The first step involves modifying the declaration of the route so it includes one or more parameters. Passing in a numeric and a string variable can be accomplished via a folder structure:

```
#[get("/world/<name>/<age>")]
fn world(name: String, age: u8)
-> String {
    format!("Hello, {} year
old named {}!", age, name)
}
```

During compilation, the program will reveal a folder structure. Prove the correctness of the product by invoking <http://localhost:8000/world/world/tam/40>.

The product also takes care of malformed requests - invoke <http://localhost:8000/world/world/tam> to see a 404 error.

Understanding this behaviour requires a look at the routing infrastructure: like most other web frameworks, Rocket 'throws' incoming requests from route to route until one matches. Developers can also specify route rank via a numeric value:

```
#[get("/user/<id>")]
fn user(id: usize) -> T { ... }
```

```
##[get("/user/<id>", rank = 2)]
fn user_int(id: isize) -> T {
...
}

##[get("/user/<id>", rank = 3)]
fn user_str(id: &RawStr) -> T {
...
}
```

DO THE JSON

Another aspect involves the creation of well-formed JSON. To use it, a set of supporting libraries must be added to the .toml file - a lot of advanced features are not domiciled in Rocket, but in 'rocket_contrib':

```
[dependencies]
...
serde = "1.0"
serde_json = "1.0"
serde_derive = "1.0"

[dependencies.rocket_contrib]
version = "*"
default-features = false
features = ["json"]
```

Using the 'features' array lets us fine-tune the inclusion: you don't need to include all parts of the library. We furthermore load a group of helper libraries, which simplify serialisation.

Now we've edited the .toml file, it is time to return to the main Rust code. The newly-added elements must first be imported into the namespace:

```
#![feature(plugin)]
#![plugin(rocket_codegen)]

extern crate rocket;
#[macro_use] extern crate rocket_contrib;
#[macro_use] extern crate serde_derive;
```

```
use rocket_contrib::Json;
use rocket_contrib::Value;
```

A structure must be declared, which describes the format of the generated JSON object. We will limit ourselves to a numeric and a string value - be sure not to forget the attribute by mistake:

```
#[derive(Serialize,
Deserialize)]
```

```
struct Message {
    id: u8,
    contents: String
}
```

One problem remains: a JSON object must be built and returned in response to an incoming query:

```
##[get("/world/<name>/<age>")]
fn world(name: String, age: u8)
-> Json<Message> {
    Json(Message {
        contents: name,
        id: age
    })
}
```

Invoke the route we declared above, and feast your eyes on the output! ☺

LEARN MORE

While many open-source products come with terrible documentation, the team behind Rocket is extremely diligent when it comes to updating the backing information. Visit <https://rocket.rs/guide> to feast your eyes on one of the most complete sets of documentation in the industry - a casual look is more than recommended.



CREATE THE IMPOSSIBLE

www.photoshopcreative.co.uk

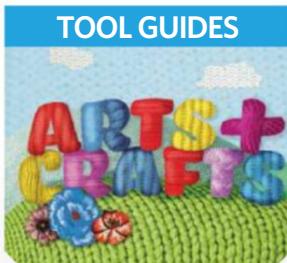
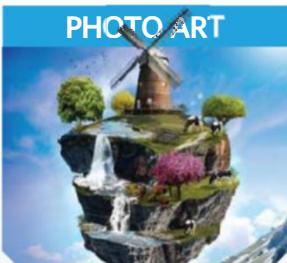


Photoshop® creative

Available from all good newsagents and supermarkets

ON SALE NOW

• Striking imagery • Step-by-step guides • Essential tutorials



BUY YOUR ISSUE TODAY

Print edition available at www.myfavouritemagazines.co.uk
Digital edition available for iOS and Android



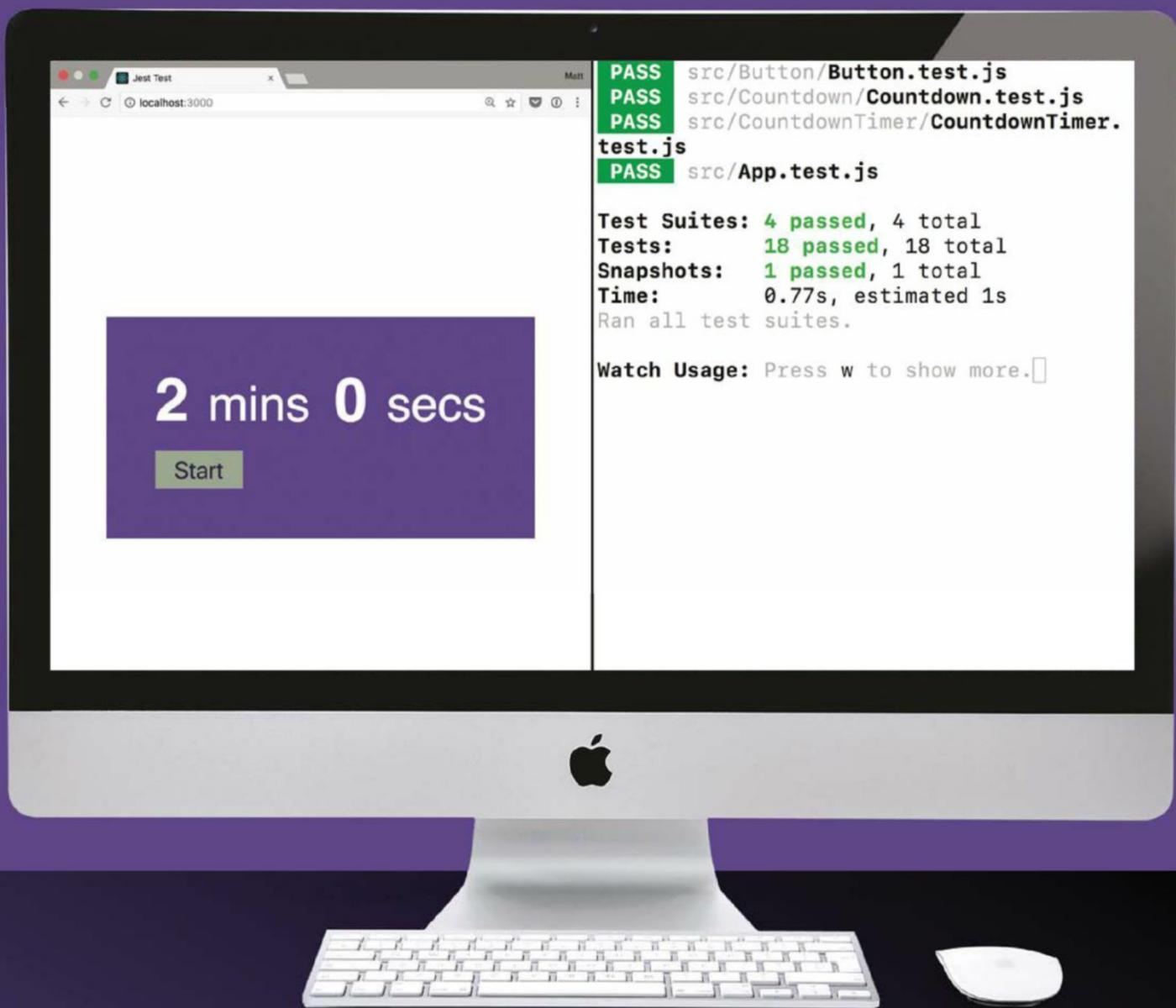
facebook.com/PhotoshopCreative



twitter.com/PshopCreative

Test React with Jest and Enzyme

Learn how to write powerful unit tests that can notify you of issues with components as you work





The rise of JavaScript frameworks such as React have made the creation of dynamic and exciting user interfaces easier for the developer. As a result,

applications built with them soon become awash with complex components with their own unique behaviours.

By creating lots of small, reusable components we can reduce the time that it takes to get a project completed. But what happens when these components need updating? How can we be sure that everything will keep working as we expect it to?

We can always manually test the application we've created as changes are made, but there is always the risk of missing a small feature or an edge case and not realising until it's too late. By automating this process, tests can be created for all types of scenarios and can run in a fraction of the time.

Jest is a testing framework provided by Facebook that is geared towards React. It provides a wide set of tools that help developers write tests alongside their components with minimal setup.

Enzyme is a utility library from the team at Airbnb. It provides an easier way to inspect React components at certain points in time in a similar way to jQuery selectors.

In this tutorial we will be combining Jest and Enzyme to create a few unit tests for an existing countdown component. By making use of the unique features they both provide, we can be sure this component will work as expected as this project progresses.

1. Install dependencies

This project is built using 'create-react-app', which is a tool that sets up a base project that we can use to create React applications. This already includes the Jest framework, but we also need to install Enzyme to help write our tests.

Open up the console, find the project files and install the dependencies using Yarn. If Yarn hasn't been installed, visit yarnpkg.com and set it up first. Then start up the server and find the component we will be testing.

```
No tests found related to files changed since last commit.
Press 'a' to run all tests, or run Jest with `--watchAll`.
```

Watch Usage

- › Press **a** to run all tests.
- › Press **p** to filter by a filename regex pattern.
- › Press **t** to filter by a test name regex pattern.
- › Press **q** to quit watch mode.
- › Press **Enter** to trigger a test run.

```
> yarn install
> yarn start
```

2. Watch for test changes

It is best to have tests running every time we make a change. This helps quickly identify any modifications that might break other functionality.

Jest includes a 'watch mode' that will use Git to check which files have changed since the last commit, and will run the appropriate tests.

In a separate console window, run Jest in 'watch mode'. Keep an eye on this as we start writing tests.

```
> yarn test
```

3. Configure the Enzyme adapter

Since Enzyme can work with other React-like frameworks such as Preact, we need an adapter that connects it to our current version of React. Since we are using v16, that requires the 'enzyme-adapter-react-16' package.

By default, any application created with 'create-react-app' will look for this adapter setup inside 'src/setupTests.js'.

Open up that file and then configure Enzyme to work with this project.

```
import { configure } from 'enzyme';
import Adapter from
  'enzyme-adapter-react-16';
configure({ adapter: new Adapter() });
```

4. Writing the first test

Jest will find any files that end in either 'test' or 'spec' and have either a '.js' or '.jsx' extension. It will then look inside these files for any tests it can run.

A test begins by calling 'it()' with a name labelling the test, alongside a function that describes it. The function will contain at least one 'assertion' – the result of a test that we can match against our expectations.

Keep tests readable

Official Jest documentation uses 'test()' instead of 'it()'. While both function exactly the same, using 'it()' helps to create a more readable and descriptive title for a test.

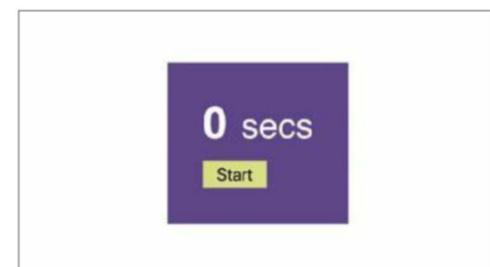
```
// Jest Snapshot v1, https://goo.gl/fbAQLP
exports[`displays correctly 1`] = `
<button
| className="button"
>
|   Button
</button>
`;
```

PASS src/Button/Button.test.js
✓ displays correctly (2ms)

Snapshot Summary
1 snapshot written in 1 test suite.

Test Suites: 1 passed, 1 total
Tests: 1 passed, 1 total
Snapshots: 1 added, 1 total
Time: 0.396s, estimated 1s
Ran all test suites related to changed files.

Watch Usage: Press w to show more.



Top

The component we are testing will open up when the development server starts. If it doesn't, it can be found at localhost:3000

Left

A 'snapshot' is a text file rather than a graphic. They can be found in the '_snapshots__' directory at the same level as the test file

Developer tutorials

Test React with Jest and Enzyme

Open up 'src/Button/Button.test.js' and add a blank test for the Button component. Add a dummy assertion to check Jest is finding the file.

```
it('displays correctly', () => {
  expect(true).toBe(true);
});
```

5. Snapshot the component

The 'react-test-renderer' package imported at the top of the test file will render the component that gets passed to it. We can then use this to create repeatable tests without involving the browser.

Jest can create a snapshot of this component and refer to it in subsequent tests. If anything has changed it will let us know.

Remove the 'expect' line from the previous step and create a snapshot test. Convert the Button component to JSON and tell Jest to use it to compare against when the test runs next time.

```
<Button>Button</Button>.toJSON();
expect(wrapper).toMatchSnapshot();
```

6. Use Enzyme to render Button

While the test renderer works great for simple tests like snapshots, they can be tricky to inspect and manipulate after they have been created. Enzyme enables us to use jQuery-like selectors to target specific elements within a component to make testing easier.

Create a new test, but this time use Enzyme to render only the component under test. This is known as a 'shallow render'. We will add our assertion for this test in the next step.

```
it('accepts custom classes', () => {
  const wrapper = shallow(<Button
    className="myClass">Button</Button>);
```

Avoid external code

Make sure tests focus only on code written for your own project. By mocking any calls to external libraries, you can be sure any bugs present in their code will not be picked up by your tests.

7. Check the class on Button

The 'wrapper' variable now contains a reference to the rendered component. We can call methods upon it to find out more information about how it rendered, such as 'prop0' to see the value of a specified prop.

The Button component renders a styled version of a regular <button> element. Add an assertion that checks the 'className' prop contains what we supplied it with.

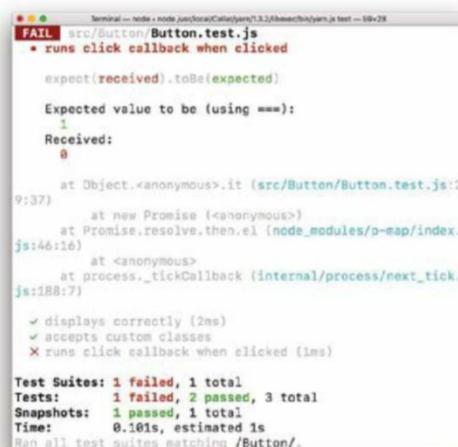
```
expect(wrapper.prop('className'))
  .toContain('myClass');
```

8. Test the Button click action

We need to check when the button is clicked that the right event callback will be called. Since we do not want to test a specific callback, we can ask Jest to create a fake function called a 'mock' that takes its place. Jest will capture any calls to it that we can later check in the assertion.

Create a new test that checks the 'onClick' callback. Use Enzyme to simulate clicking the button and check the callback was called once.

```
it('runs click callback when clicked', () =>
{
  const onClick = jest.fn();
  const wrapper = shallow(<Button
    onClick={onClick}>Button</Button>);
```



```
wrapper.simulate('click');
expect	onClick.mock.calls.length
  .toBe(1);
});
```

9. Fix the failing test

It seems the test we added in the last step is failing. Jest will tell us what it asserted (0), what we were matching against (1), and the line number of the assertion that needs addressing.

In this case, the Button component does not pass through all the props it gets to the rendered <button> element. Open up 'src/Button/index.js' and pass the props through.

```
return (<button className={`button
${props.className ?
  props.className : ''}`}
 {...props}>
  { props.children }
</button>)
```

10. Fix the snapshot test

While making that other test pass, we broke the snapshot test from earlier. Jest highlights exactly what changed between the last snapshot and now to help identify what happened.

This highlights the main benefit of testing components as they are created. Without them, a change that seemed initially harmless ended up breaking the button.

The props we passed through in the previous step are overwriting the 'className' prop. Move the props spread before 'className' to fix the issue.

```
return (<button className={`button
...
${props.className ?
  props.className : ''}`}
 {...props}>
  { props.children }
</button>)
```

11. Testing required props

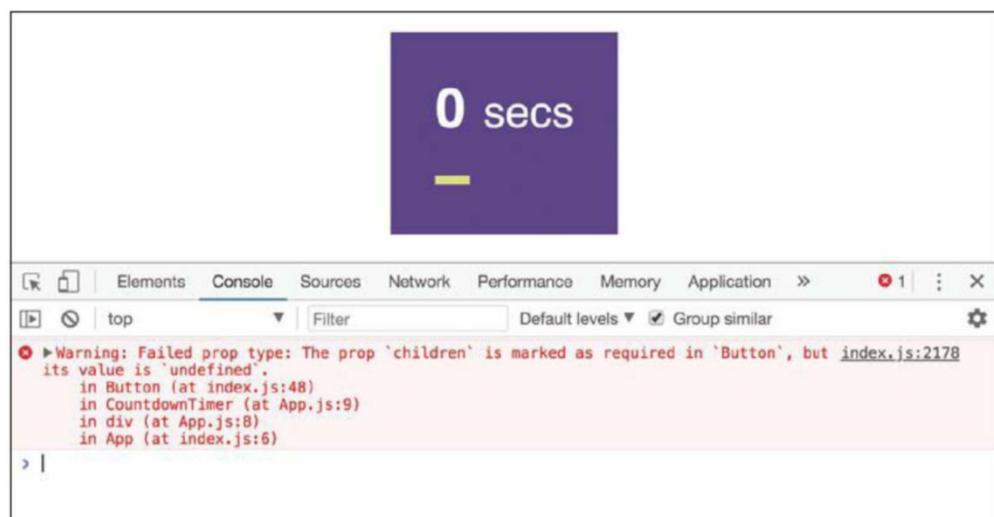
When using React, we can set prop types to denote which props are required. The only requirement we

Top

When coming across a failed snapshot test, if the change to a component was deliberate the snapshot can be updated by pressing the "u" key.

Right

When a required prop is missing, it only logs the error to the console rather than throwing an error. Matching against an error will fail the test.



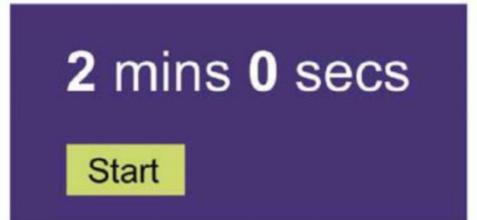
Shallow render

`shallow(<CountdownTimer/>)`



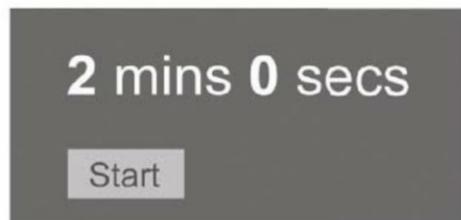
Full render

`mount(<CountdownTimer/>)`



Static render

`render(<CountdownTimer/>)`



Rendering methods used in Enzyme

There are three different types of rendering methods that Enzyme can perform - 'shallow', 'full' and 'static'. The case under test will determine which method to use.

'Shallow' rendering is the most common and will limit the rendering to just the component under test rather than any of its children. This means any errors caused when rendering those children will not fail the test.

'Full' rendering will render everything from the parent downwards. This can be particularly useful for testing those wrapped in higher order components. These tests can take longer to run, but any component doing any direct manipulation of the DOM will need to do a full render.

'Static' rendering will only render the component once as static HTML. It can be faster but will not provide any interaction.

have for our button is that it has some kind of content inside to label it.

When a required prop is missing, React will log an error to the console. By mocking the 'console.error' method like we did earlier with the click event, we can check that it was detected correctly.

As the 'console' object is global, it needs to be overridden through the 'global' object. Create a new test that mocks the console and checks an error is logged.

```
it('requires button text', () => {
  global.console = {};
  error: jest.fn();
  const wrapper = shallow(<Button />);
  expect(console.error).toBeCalled();
});
```

12. Restore the console

Each test should be able to run in isolation. Any side effects should be cleaned up after each test runs.

The test from the previous step overwrites the behaviour of the console object. Any test that runs after it would use this stripped-down version and may fail because of it.

At the start of the test, keep a reference to the original console object. Once the test is finished, replace it.

```
const oldConsole = global.console;
[...]
global.console = oldConsole;
```

13. Start testing for Countdown

The actual counting down happens within the Countdown component. This takes a 'time' prop as a Date

object and will count down towards it. When it's done, it calls the 'onComplete' function prop if one was passed.

Open up 'src/Countdown/Countdown.test.js' and add a basic test to make sure it renders without error.

```
it('displays correctly', () => {
  shallow(<Countdown />);
});
```

14. Group timer tests together

Components can be quite complex, so it is best if similar tests are grouped together. This can be achieved by using a 'describe' block. Any tests in a block will be shown in the output from Jest.

These blocks can be particularly useful if they share a similar setup, which we will look at in the next step. Below the initial test, create a block to hold our timer tests.

```
describe('timer functionality', () => {
  //Tests will go here
});
```

15. Set up and tear down

Jest defines special 'beforeEach' and 'afterEach' blocks that will run before or after every test within that block. Similarly, 'beforeAll' and 'afterAll' will run before or after all tests run. This can save having to repeat the same logic for each one.

Inside the new block, set up a Date object for two minutes time. There's no need to tear down with 'afterEach' since we are rewriting the same variable.

```
let twoMinutes;
beforeEach(() => {
  twoMinutes = new Date();
  twoMinutes.setMinutes(
```

```
twoMinutes.getMinutes() + 2);
});
```

16. Check the initial values

We are about to write a test to check the countdown actually counts down. But before we get any further, we should check the timer displays the correct initial time.

Use Enzyme to shallow render the Countdown component set for two minutes, then check those values are showing up.

```
it('counts down each second', () => {
  const wrapper = shallow(
    <Countdown time={twoMinutes} />);
  expect(wrapper.find('.countdown__mins .countdown__value').text())
    .toEqual("2");
  expect(wrapper.find('.countdown__secs .countdown__value').text())
    .toEqual("0");
});
```

17. Fake JavaScript timers

The Countdown works using 'setInterval', which would require time to pass before it would update the display. Unfortunately waiting a second for each test to run would soon build up.

Jest provides functionality that can fake the passage of time so we can test things without delay.

Use the fake timers to advance time by one second and check the countdown progressed correctly. Be sure to tell Jest to use the fake timers before creating the component with Enzyme.

```
jest.useFakeTimers();
```

Developer tutorials

Test React with Jest and Enzyme

2 mins 0 secs

Start

1 min 1 sec

Start

Top

The internal 'timeRemaining' state is recalculated each second. From this, the render function calculates the minutes and seconds remaining

Right

It's always a good idea to test as much as possible, including any potential edge cases. For example, what happens when a unit is in single digits?

```
[...]
jest.runTimersToTime(1000);
expect(wrapper.find('.countdown__mins
  .countdown__value').text())
  .toEqual("1");
expect(wrapper.find('.countdown__secs
  .countdown__value').text())
  .toEqual("59");
```

```
Terminal — node ./node_modules/.bin/jest src/Countdown/Countdown.test.js
● timer functionality › counts the timer down each second
  expect(received).toEqual(expected)
    Expected value to equal:
      "1"
    Received:
      "2"

    at Object.it (src/Countdown/Countdown.test.js:46:71)
      at new Promise (<anonymous>)
        at Promise.resolve.then.el (node_modules/p-map/index.js:46:16)
          at <anonymous>
            at process._tickCallback (internal/process/next_tick.js:188:7)

PASS  src/Button/Button.test.js
  Test Suites: 1 failed, 1 passed, 2 total
  Tests:       1 failed, 5 passed, 6 total
  Snapshots:  1 passed, 1 total
  Time:        0.587s, estimated 1s
  Ran all test suites related to changed files.

Watch Usage: Press w to show more.
```

18. Mocking out Date.now

We have a problem – the timer still says that there are two minutes left. What happened?

Every time the interval callback fires, the component is using 'Date.now()' to calculate the time remaining. The fake timers Jest creates will not touch the Date object, which means 'now' is the exact same time as it was when the component was first created.

To fix this we need to mock 'Date.now'. Jest lets us specify return values for mocked functions, which enables us increase the time artificially. Let's do that by adding the following to the top of the test.

```
let fakeNow = Date.now();
```

```
const mockDateNow = jest.fn()
  .mockReturnValueOnce(fakeNow)
  .mockReturnValueOnce(fakeNow + 1000);
global.Date.now = mockDateNow;
```

19. Force Enzyme to update

Even after mocking 'Date.now', it looks like the test we created still does not pass.

While Enzyme will keep the render up to date in response to any direct actions – such as with a button press, for example – any external call to update the state will not cause a re-render. Since our timer is external we need to prompt Enzyme to re-render for our changes to be reflected.

Just before the second block of assertions, instruct Enzyme to update.

```
wrapper.update();
```

20. Know when the timer runs

'CountdownTimer' is the component that controls Countdown. It defines the target time when the button is pressed. A callback from Countdown lets this component know when the countdown is running or not, which sets the 'running' value in state.

In addition to props, Enzyme exposes the component state as well. By firing the right method on the component instance manually, we can check it sets the right state.

Open src/CountdownTimer/CountdownTimer.test.js and add a test for the 'running' state value.

```
it('knows when a timer runs', () => {
  const wrapper = shallow(
    <CountdownTimer />);
  expect(wrapper.state('running'))
    .toBe(false);
  wrapper.instance().setTimer();
  expect(wrapper.state('running'))
    .toBe(true);
});
```

21. Disable the button when the timer runs

We know that the 'setTimer()' method will update the state, but we do not know whether that will disable the button. We need a separate test for that.

Enzyme enables us to find components within the rendered component by name. From here we can query its props like normal.

We can manually update the state as that aspect is not under test. This works much like setting state inside the React component itself. Create a final test that checks the button's 'disabled' prop after altering the state.

```
it('disables the start button', () => {
  const wrapper = shallow(
    <CountdownTimer />);
  expect(wrapper.find('Button')
    .prop('disabled')).toBe(false);
  wrapper.setState({ running: true });
  expect(wrapper.find('Button')
    .prop('disabled')).toBe(true);
});
```

1 min 3

Start

Special offer for readers in *North America*



6 issues FREE

When you subscribe*

FREE
resource
downloads
every issue

21 PAGE EXPERT FREE 70 MINUTES OF JS VIDEOS

web designer™

@WebDesignerMag /WebDesignerUK

HTML5 CSS3 jQuery JavaScript

SEO TODAY

20 HOT NEW TOOLS

Must-know libraries and frameworks

REACTJS

Ten pages on testing and form controls

+ Faster builds with better boilerplates • Get started with Rust • Create custom reusable Sketch components • Build a WebGL racing game • Animated menus

The path to search success in 2018

- How to futureproof your site
- Get to grips with technical SEO
- Optimise for mobile and voice
- 10 essential tools and resources

“The only magazine you need to design and develop stunning websites”



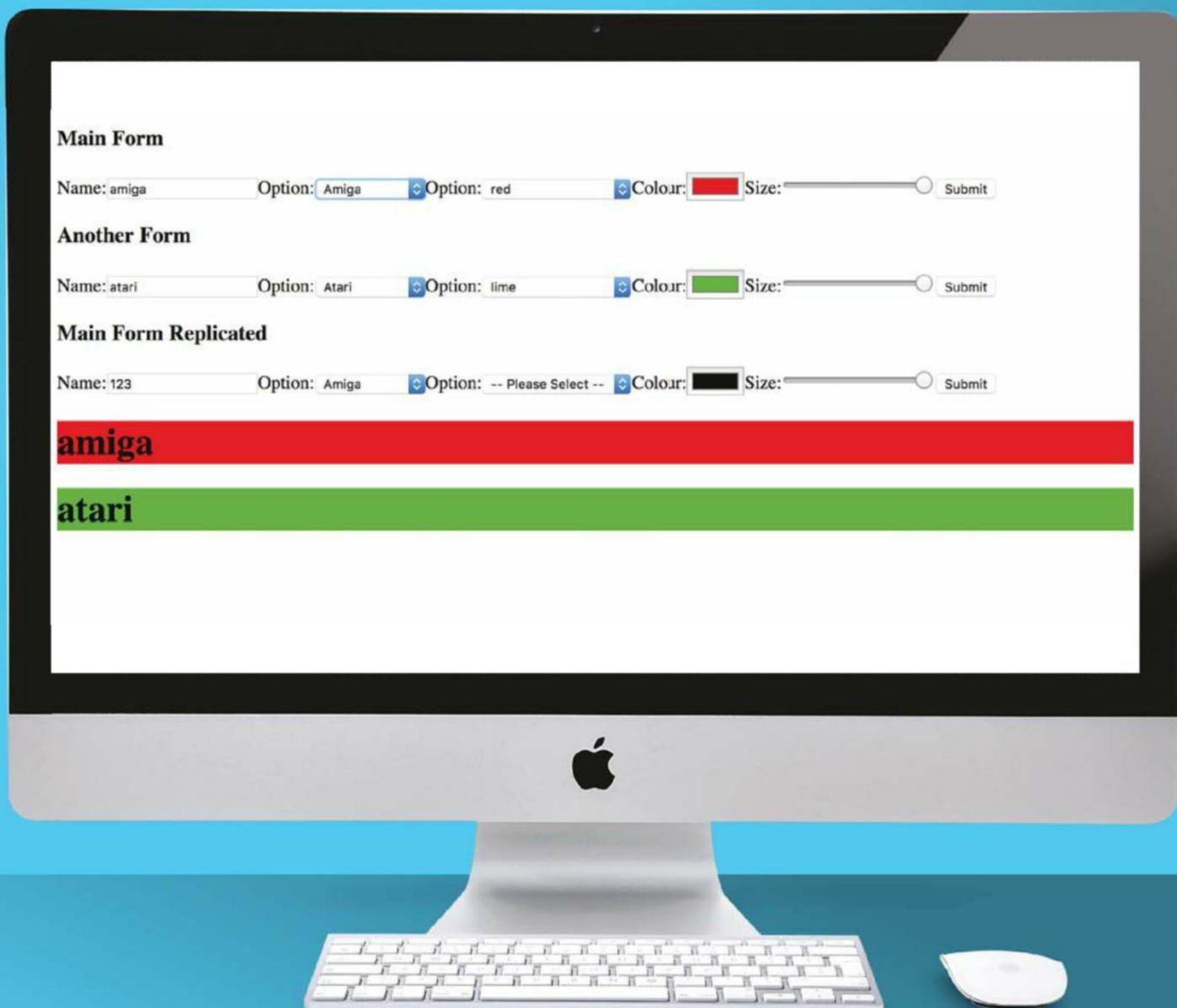
Order hotline **+44 (0) 344 848 2852**
Online at **myfavouritemagazines.co.uk/WEDPQ17**

***Terms and conditions** This is a US subscription offer. 6 free issues refers to the USA newsstand price of \$14.99 for 13 issues being \$194.87, compared with \$103.20 for a subscription. You will receive 13 issues in a year. You can write to us or call us to cancel your subscription within 14 days of purchase. Payment is non-refundable after the 14 day cancellation period unless exceptional circumstances apply. Your statutory rights are not affected. Prices correct at point of print and subject to change. Full details of the Direct Debit guarantee are available upon request. UK calls will cost the same as other standard fixed line numbers (starting 01 or 02) are included as part of any inclusive or free minutes allowances (if offered by your phone tariff). For full terms and conditions please visit: bit.ly/magtandc Offer ends May 31 2018.

**Expires
31 May
2018**

Code dynamic form controls using React

Use the React framework to quickly and easily set up an interactive form that controls data and visuals





ReactJS is a framework created by Facebook for building user interfaces, especially those that are heavily dependent on data integration. The

framework allows for easier development of advanced front-end functionality by taking care of the management of components created. Your task for developing with ReactJS is to define the initial settings required by ReactJS to manage the user interface. A knowledge of JavaScript fundamentals, along with an understanding of object-oriented programming and the MVC pattern will help to make learning ReactJS significantly easier.

This tutorial focuses on using ReactJS to create a reusable form component called 'TextForm' that can be used to control text presentation. The form will act as a container for user input elements, which will link to event handlers within the component that are designed to respond to event interactions.

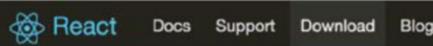
Rendering of 'TextForm' is performed by creating a new 'instance' of the TextForm class. Each instance is defined with its own properties based on the specification of the TextForm class. Each form will have the ability to decide how to change both content and presentation attributes of their assigned content areas.

1. Document initiation

Initiate the page document with links to load ReactJS and the required CSS for styling. Links to these files are placed inside the head section. This example uses the online quick-start version of ReactJS, but you should look at downloading and installing ReactJS - see the following:

<https://react-cn.github.io/react/downloads.html>

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8" />
<title>Forms</title>
<link rel="stylesheet" type="text/css"
href="styles.css" />
<script src="https://unpkg.com/react@16/umd/
```



Downloads

Download the starter kit to get everything you need to [get started with React](#). The starter kit includes React and some simple example apps.

[Download Starter Kit 0.14.3](#)

Development vs. Production Builds

We provide two versions of React: an uncompressed version for development and a minified version for production. The development version includes extra warnings about common mistakes, whereas the production version includes extra performance optimizations and strips all error messages.

If you're just starting out, make sure to use the development version.

```
react.development.js"></script>
<script src="https://unpkg.com/react-dom@16/
umd/react-dom.development.js"></script>
<script src="https://unpkg.com/babel-
standalone@6.15.0/babel.min.js"></script>
*** STEP 3 HERE
</head>
<body>
*** STEP 2 HERE
</body>
</html>
```

2. Content template

Page layout is defined from the use of normal HTML elements placed within the page body. These elements need a way for ReactJS to reference, so each of the containers has a unique ID attribute applied.

```
<div id="form1"></div>
<div id="form2"></div>
<div id="form3"></div>
<div id="output1"></div>
<div id="output2"></div>
```

3. JavaScript initiation

ReactJS code must be placed inside a script container that has a type attribute of 'text/babel' - or it will fail. This first step of the JavaScript defines a helper function called '\$' that will be used as shorthand to find and return elements on the web page.

```
<script type="text/babel">
var $ = function(cssRule){
  return document.querySelector(cssRule);
}
*** STEP 4
</script>
```

4. React content rendering

This step uses the ReactJS 'render' method and outputs

MVC patterns

ReactJS was created to make management of webpage content easier to control. Understanding the concept of models/views/controllers will help you to understand concepts of ReactJS.

Welcome

Left

A combination of HTML within JavaScript results in this initial content rendering created in step 3 using the render method of ReactDOM within ReactJS

Main Form

Name: 123 Option: Amiga Option: -- Please Select -- Colour: Size: Submit

Another Form

Name: 123 Option: Amiga Option: -- Please Select -- Colour: Size: Submit

Main Form Replicated

Name: 123 Option: Amiga Option: -- Please Select -- Colour: Size: Submit

Welcome

Bottom

Despite the form controls only being defined once, ReactJS allows multiple 'instances' of the form to be placed wherever they are required

Developer tutorials

Code dynamic form controls using React

content to a target. Parameter one is the HTML content to output. Parameter two uses the function defined in step 3 to reference the 'output1' element created in step 2.

```
ReactDOM.render(  
  <h1>Welcome</h1>,  
  $('#output1')  
)
```

5. React component

The form will be controlled by a React component we give the name 'TextForm' that uses 'extend' to gain functionality from React.Component. This step initiates TextForm with a render method that will be used to return HTML content when the TextForm class is called.

```
class TextForm extends React.Component {  
  *** STEP 15 HERE  
  render() {  
    return (  
      *** STEP 6 HERE  
    );  
  }  
  *** STEP 18 HERE
```

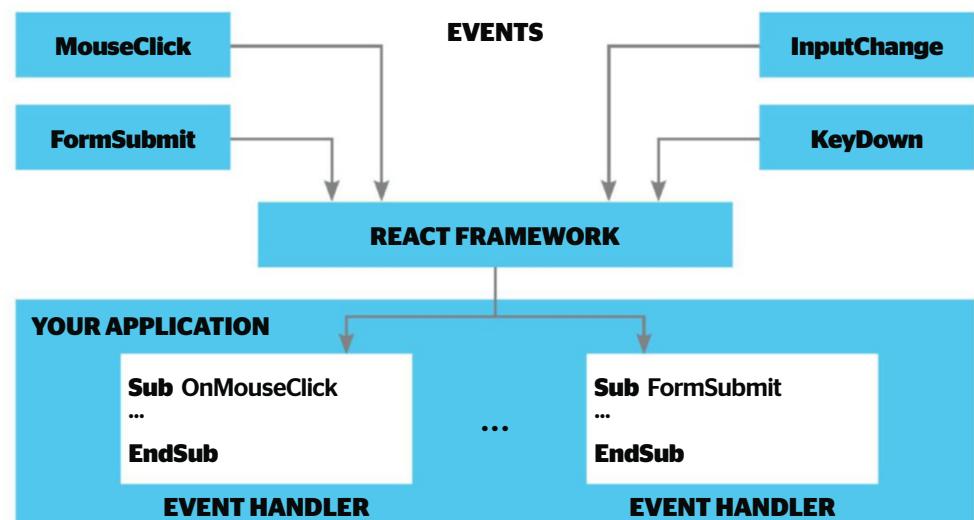
6. Form template definition

The render method will produce the HTML for an interactive form. First, the form container and the input element use 'onSubmit' and 'onChange' to apply event handlers defined within the TextName class in later steps. The 'this.props' allows access to properties provided as part of the render call.

```
<form onSubmit={this.handleSubmit}>  
  <h3>{this.props.title}</h3>  
  <label>  
    Name:
```

Object-oriented JS

ReactJS is heavily based on object-oriented principles using JS. Learning about object-oriented programming concepts will make it significantly easier to learn about the ReactJS framework.



```
<input type="text" onChange={this.  
  handlerTextChange}  
  value={this.state.text} />  
</label>  
  *** STEP 7 HERE  
</form>
```

```
</select>  
</label>  
*** STEP 8 HERE
```

7. Drop-down list

Another form element defined with normal HTML. Like the previous step, an event handler is applied to 'onChange', to be defined in a later step. The element's value is linked to the class's 'state.text' data for ReactJS to automatically make sure the element shows the current value of 'state.text'.

```
<label>  
  Option:  
  <select onChange={this.handlerTextChange}  
  value={this.state.text}>  
    <option value="amiga">Amiga</option>  
    <option value="atari">Atari</option>  
    <option value="amstrad">Amstrad</option>  
    <option value="commodore">Commodore</option>
```

8. Another list

Another drop-down list. This time using the 'handlerBackgroundChange' method to respond to changes. The data value presented is linked to the 'state.colour' of the TextForm object - described in a later step.

```
<label>  
  Option:  
  <select onChange={this.  
  handlerBackgroundChange}  
  value={this.state.colour}>  
    <option>-- Please Select --</option>  
    <option value="#ffff00">Yellow</option>  
    <option value="#00ff00">lime</option>  
    <option value="#ff0000">red</option>  
    <option value="#0000ff">blue</option>  
</select>  
</label>  
*** STEP 9 HERE
```

From this page
Check the console for data results.

OK

Option: lime Colour: Size: Submit

Option: -- Please Select -- Colour: Size: Submit

Top
Notification of form submission method linked to the form as 'onSubmit' control. Data processing before submission occurs here

Right
Form submission handler has been set to output the current version of the object's state data to the console. Each instance of FormName has its own values.

Main Form

Name: amiga Option: Amiga Option: red Colour: Size: Submit

Another Form

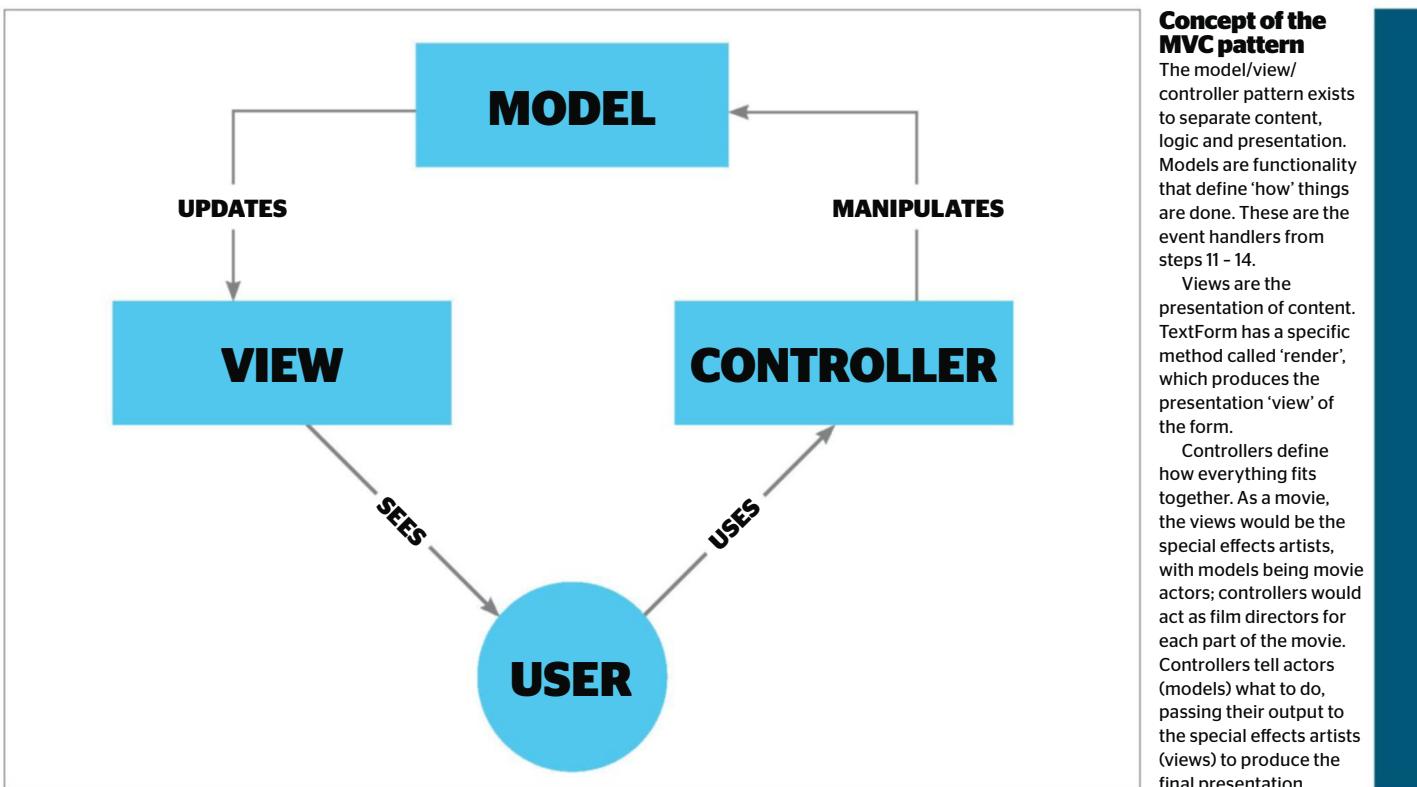
Name: atari Option: Atari Option: lime Colour: Size: Submit

Main Form Replicated

Name: 123 Option: Amiga Option: -- Please Select -- Colour: Size: Submit

amiga

Console was cleared
undefined
▼ {text: "amiga", colour: "#ff0000"}
 colour: "#ff0000"
 text: "amiga"
 ► __proto__: Object

**Concept of the MVC pattern**

The model/view/controller pattern exists to separate content, logic and presentation. Models are functionality that define 'how' things are done. These are the event handlers from steps 11 - 14.

Views are the presentation of content. TextForm has a specific method called 'render', which produces the presentation 'view' of the form.

Controllers define how everything fits together. As a movie, the views would be the special effects artists, with models being movie actors; controllers would act as film directors for each part of the movie. Controllers tell actors (models) what to do, passing their output to the special effects artists (views) to produce the final presentation.

9. Colour selector

A colour input linked to the same 'state.colour' data source provides choice to update the colour data using the value of either the drop-down list or this colour picker input. Using the same state data value as the drop down, both input elements will influence each other. This is called data binding.

```

<label>
  Colour:
  <input type="color" onChange={this.
  handlerBackgroundChange} value={this.state.
  colour}/>
</label>
*** STEP 10 HERE
  
```

10. Size range and submit

The last two input elements of the form are a range slider and the form submit button. The range slider is set to have the class 'handlerSizeChange' method applied for responding to any changes. No special ReactJS features for the submit button – it will trigger a submit event on the parent form.

```

<label>
  Size:
  <input type="range"
  min="0.5" max="5" step="0.1"
  onChange={this.handlerSizeChange}>
</label>
<input type="submit" value="Submit" />
  
```

11. Handler methods: background

A handler method is a function that can be called in

response to an event of an input element. This step defines a handler for input elements of the form defined in steps 6 - 9, which results in background colour. See next step about 'props' and 'state'. Note how the \$ helper created in step 3 is used.

```

handlerBackgroundChange(event) {
  this.setState({colour: event.target.
  value});
  $(this.props.target).style.background =
  event.target.value;
}

*** STEP 12 HERE
  
```

12. Handler methods: size

Another handler is required to change the size of the text. Each handler has the ability to access data stored as part of the TextForm object via 'this'. Look at steps 11 - 13 using the 'props.target' data value to reference the DOM target; see step 15 on how 'props' are passed to the FormName object.

```

handlerSizeChange(event) {
  this.setState({size: event.target.value});
  $(this.props.target).style.fontSize =
  event.target.value+"em";
}

*** step 13 here
  
```

13. Handler methods: text update

Updates to the text presentation are managed by this step's handler method. Take note of how steps 11-13 use the object's 'setState' method from ReactJS to update the FormName object's data values. This step sets the value of

'text' to match the input value provided through the 'event' parameter.

```

handlerTextChange(event) {
  this.setState({text: event.target.value});
  ReactDOM.render(
    <h1>{event.target.value}</h1>,
    $(this.props.target)
  )
}
*** STEP 14 HERE
  
```

14. Handler methods: form submit

The final handler defined in this tutorial will manage the form submission. Unlike traditional HTML forms, data is stored inside the FormName's 'state' data container object instead of DOM input elements. This step outputs the 'state' data container to the console when the form is submitted.

```

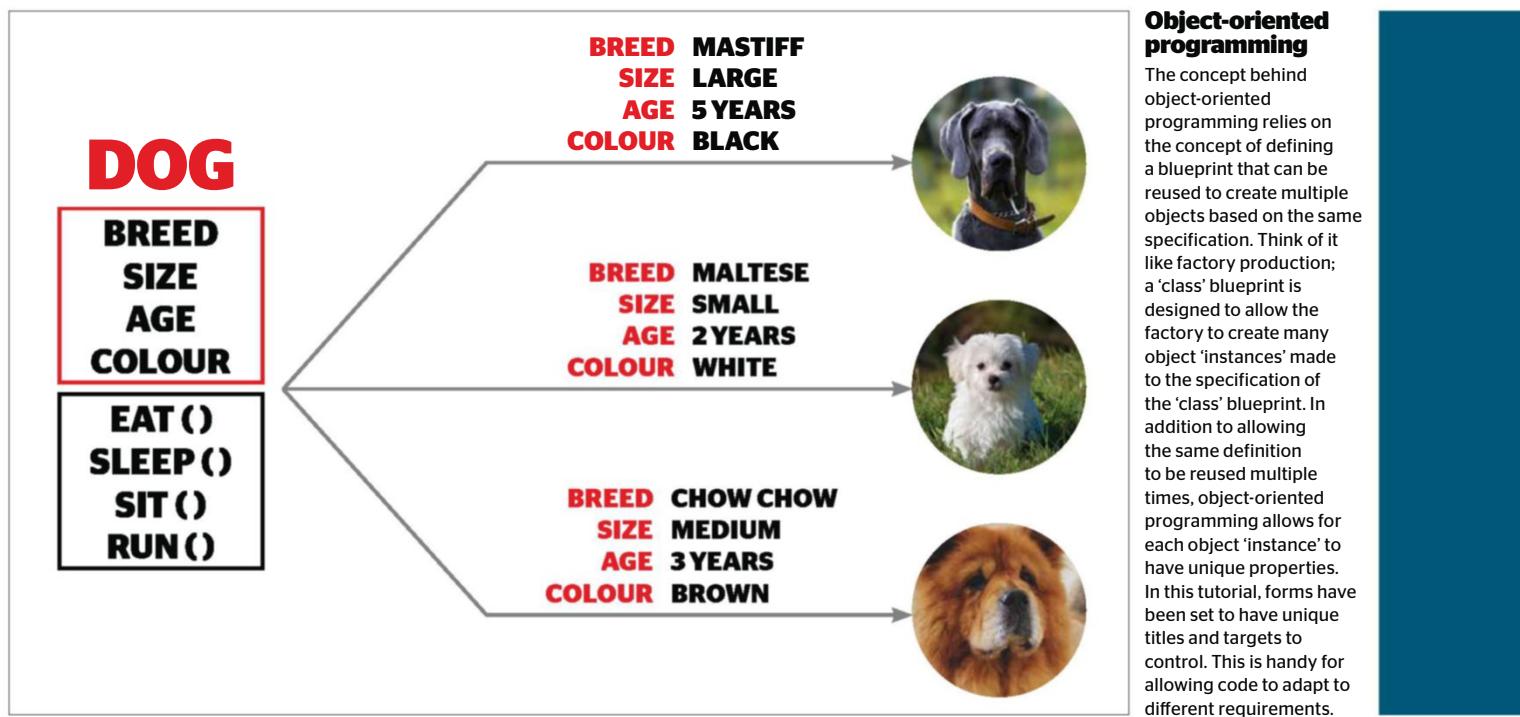
handlerSubmit(event) {
  alert('Check the console for data
  results.');
  console.log(this.state);
  event.preventDefault();
}
  
```

15. Constructor definition

The constructor is executed as soon as the object is created. ReactJS components accept one parameter for their constructor; an object describing all of the properties provided in the call to the function – see steps 18 - 20. These 'props' are made accessible to handlers through 'super'.

Developer tutorials

Code dynamic form controls using React



Object-oriented programming

The concept behind object-oriented programming relies on the concept of defining a blueprint that can be reused to create multiple objects based on the same specification. Think of it like factory production; a 'class' blueprint is designed to allow the factory to create many object 'instances' made to the specification of the 'class' blueprint. In addition to allowing the same definition to be reused multiple times, object-oriented programming allows for each object 'instance' to have unique properties. In this tutorial, forms have been set to have unique titles and targets to control. This is handy for allowing code to adapt to different requirements.

```
constructor(props) {
  super(props);
  *** STEP 16 HERE
}
```

16. Object state data

This step also defines the 'state' object that the FormName methods will use to store data. Initially this is set to store a default value for 'text', which can be overwritten as well as other data items being added by method functions.

```
this.state = {text: '123'};
```

17. Binding the handlers

Handlers created in steps 11 - 14 need to be attached to their associated form elements. ReactJS provides all functions with 'bind' method, allowing automatic calls by any elements in 'this' instance of FormName referencing the method functions in the HTML created in steps 6 - 12.

```
this.handlerBackgroundChange =
  this.handlerBackgroundChange.bind(this);
this.handlerSizeChange =
  this.handlerSizeChange.bind(this);
this.handlerTextChange =
  this.handlerTextChange.bind(this);
this.handlerSubmit =
  this.handlerSubmit.bind(this);
```

18. Rendering: first form

ReactJS is used to render a version of its TextForm to the page area with ID 'form1'. This version of the form has properties for 'title' set to 'Main Form' and 'target' being set to '#output1'. Take note how the \$ helper from step 3 is used to find the page location to produce the output.

```
ReactDOM.render(
```

```
<TextForm title="Main Form"
  target="#output1" />,
  $('#form1')
);
```

19. Rendering: another form

Another form using the same technique is declared for output at the page location using ID 'form2'. This version of the form provides different values for the 'props'. Specifically, the 'target' attribute results in the form controlling another text item.

```
ReactDOM.render(
  <TextForm title="Another Form"
    target="#output2"/>,
    $('#form2')
);
```

20. Rendering: replica form

A third and final version of the TextForm object is defined. Although using a different value for 'title', the 'target' props attribute is set to the same value used for the

original TextForm object. This results in this version of TextForm controlling the same target text as the original version.

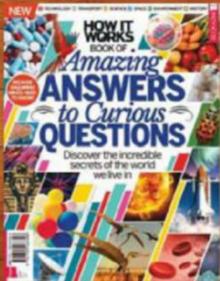
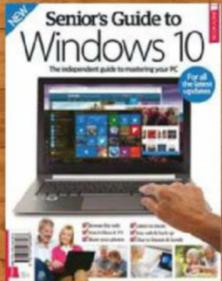
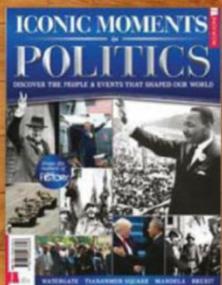
```
ReactDOM.render(
  <TextForm title="Main Form Replicated"
    target="#output1"/>,
    $('#form3')
);
```

21. Initiate styling

Create a new file called 'styles.css'. This is where you should place any of your custom page styling. This step starts the CSS style definitions by setting both the padding and margin of the body and HTML container to zero. This avoids visible spacing being presented around the page.

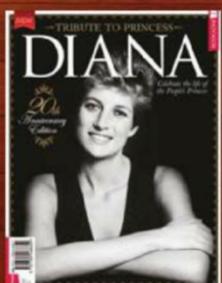
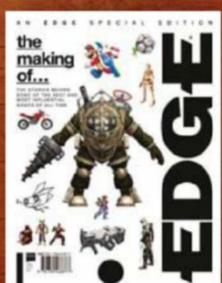
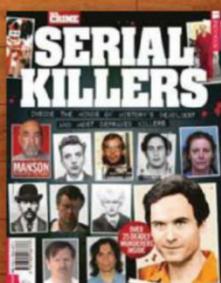
```
html, body{
  padding: 0;
  margin: 0;
}
```

```
constructor(props) {
  super(props);
  this.state = {text: '123'};
//PROPS IS WHERE PRESENTATION DETAILS SHOULD BE STORED
  this.handlerBackgroundChange = this.handlerBackgroundChange.bind(this);
  this.handlerSizeChange = this.handlerSizeChange.bind(this);
  this.handlerTextChange = this.handlerTextChange.bind(this);
  this.handlerSubmit = this.handlerSubmit.bind(this);
}
```



Discover another of our great bookazines

From science and history to technology and crafts, there are dozens of Future bookazines to suit all tastes



Get great savings when you buy direct from us



1000s of great titles, many not available anywhere else



World-wide delivery and super-safe ordering



www.myfavouritemagazines.co.uk
Magazines, back issues & bookazines.



Get your listing in our directory

To advertise here contact *Chris*

chris.mitchell@futurenet.com

+44 (0)1225 687832

HOSTING LISTINGS



Featured host: Netcetera

netcetera.co.uk
03330 439780

About us

Formed in 1996, Netcetera is one of Europe's leading web hosting service providers, with customers in over 75 countries worldwide.

As the premier provider of data centre colocation, cloud hosting, dedicated servers and managed web hosting services in the UK, Netcetera offers an array of services designed to more effectively manage IT

infrastructures. A state-of-the-art data centre environment enables Netcetera to offer your business enterprise-level colocation and hosted solutions.

Providing an unmatched value for your budget is the driving force behind our customer and managed infrastructure services. From single server to fully customised data centre suites, we focus on the IT solutions you need.

What we offer

- Managed hosting** - A full range of solutions for a cost-effective, reliable, secure host.
- Cloud hosting** - Linux, Windows, Hybrid and Private Cloud Solutions with support and scalability features.

- Data centre colocation** - Single server through to full racks with FREE setup and a generous bandwidth.
- Dedicated servers** - From QuadCore up to Smart Servers with quick setup and fully customisable.

5 tips from the pros

1. Reliability, trust & support

Reliability is a major factor when it comes to choosing a hosting partner. Netcetera guarantees 100 per cent uptime, multiple internet routes with the ability to handle DDOS attacks, ensuring your site doesn't go down when you need it.

knowledgeable staff available 24/7 to provide you with assistance when you need it most. Our people make sure you are happy and your problems are resolved as quickly as possible.

2. Secure and dependable

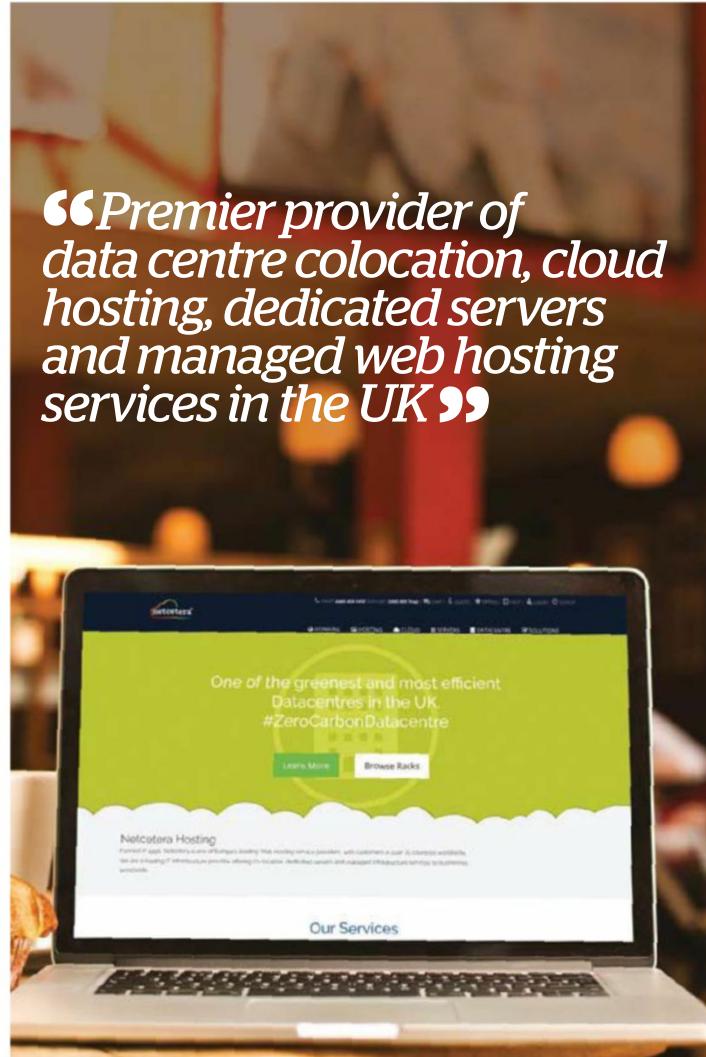
Netcetera prides itself on offering its clients a secure environment. It is accredited with ISO 27001 for security along with the options of configurable secure rackspace available in various configurations.

4. Value for money

We do not claim to be the cheapest service available, but we do claim to offer excellent value for money. We also provide a price match on a like-for-like basis, as well as a price guarantee for your length of service.

5. Eco-friendly

Netcetera's environmental commitment is backed by use of eco-cooling and hydroelectric power. This makes Netcetera one of the greenest data centres in Europe.



Testimonials

Roy T

"I have always had great service from Netcetera. Their technical support is second to none. My issues have always been resolved very quickly."

Suzy B

"We have several servers from Netcetera and their network connectivity is top-notch, with great uptime and speed is never an issue. Tech support is knowledgeable and quick in replying. We would highly recommend Netcetera."

Steve B

"We put several racks into Netcetera, basically a complete corporate backend. They could not have been more professional, helpful, responsive or friendly. All the team were an absolute pleasure to deal with, and nothing was too much trouble, so they matched our requirements 100 per cent."

Supreme hosting



cwcs.co.uk
08001777000

CWCS Managed Hosting is the UK's leading hosting specialist. They offer a fully comprehensive range of hosting products, services and support. Their highly trained staff are not only hosting experts, they're also committed to delivering a great customer experience and are passionate about what they do.

- Colocation hosting
- VPS
- 100 per cent network uptime

UK-based hosting



cyberhostpro.com
0845 5279 345
Cyber Host Pro are committed to providing the best cloud server hosting in the UK; they are obsessed with automation. If you're looking for a hosting provider who will provide you with the quality you need to help your business grow, then look no further than Cyber Host Pro.

- Cloud VPS servers
- Reseller hosting
- Dedicated servers

Cluster web hosting



fasthosts.co.uk
0808 1686 777
UK-based and operating 24/7 from dedicated UK data centres. Fasthosts keep over one million domains running smoothly and safely each day. Services can be self-managed through the Fasthosts Control Panel.

- Dedicated servers
- Cloud servers
- Hosted email



Budget hosting



hetzner.com
+49 (0)9831 505-0

Hetzner Online is a professional web hosting provider and experienced data centre operator. Since 1997, the company has provided private and business clients

with high-performance hosting products as well as the infrastructure for the efficient operation of sites. A combination of stable technology, attractive pricing, flexible support and services has enabled Hetzner Online to strengthen its market position nationally and internationally.

- Dedicated/shared hosting
- Colocation racks
- SSL certificates



All-inclusive hosting



1and1.co.uk
0333 336 5509

1&1 Internet is a leading hosting provider that enables businesses, developers and IT pros to succeed online. Established in 1988, 1&1 now

operates across ten countries. With a comprehensive range of high-performance and affordable products, 1&1 offers everything from simple domain registration to award-winning website building tools, eCommerce packages and powerful cloud servers.

- Easy domain registration
- Professional eShops
- High-performance servers

SSD web hosting



bargainhost.co.uk
0843 289 2681

Since 2001, Bargain Host have campaigned to offer the lowest possible priced hosting in the UK. They have achieved this goal successfully and built up a large client database, which includes many repeat customers. They have also won several awards for providing an outstanding hosting service.

- Shared hosting
- Cloud servers
- Domain names

Value Linux hosting



patchman-hosting.co.uk
01642 424 237

Linux hosting is a great solution for home users, business users and web designers looking for cost-effective and powerful hosting. Whether you are building a single-page portfolio, or you are running a database-driven eCommerce website, there is a Linux hosting solution for you.

- Student hosting deals
- Site designer
- Domain names

Flexible cloud servers



elastichosts.co.uk
020 7183 8250

ElasticHosts offer simple, flexible and cost-effective cloud services with high performance, availability and scalability for businesses worldwide. Their team of engineers provide excellent support 24/7 over the phone, by email and with a ticketing system.

- Cloud servers with any OS
- Linux OS containers
- 24/7 expert support



Get your listing in our directory

To advertise here contact *Chris*

chris.mitchell@futurenet.com

+44(0)1225 687832

COURSE LISTINGS



Featured: **Northcoders**

northcoders.com
Twitter: @northcoders
Facebook: Northcoders

About us

Northcoders is the coding bootcamp for the north, based in the heart of Manchester and built upon northern values of grit, determination and community spirit. No matter what your background, you can fast-track your career and become a web or software developer in 12 weeks at their

full-time bootcamp, or fit their course around your life with their 24-week part-time bootcamp. Their internal career support team will help find you work as a developer, setting up interviews with your choices of Northcoders Hiring Partners across the north of England.

What we offer

- **Full-time:**
Fast-track your career in just 12 weeks.
- **Part-time:**
Fit our curriculum around your life in 24 weeks.

5 tips from the pros

1. Get started with coding

The best way to know if coding is for you is to just try it! We recommend the free, online JavaScript track of Codecademy to get you started with the basics.

for you, set aside a few evenings each week to really start making progress! If coding is for you, this should be fun.

4. Be prepared

We'll be with you every step of the way when you apply. Make sure you go through all the materials we recommend and ask for help if you're stuck.

5. Get social

With Northcoders, you're not just on a course, you're part of a community that will stay with you long after you graduate. Make the most of it!

2. Do your research

Make sure you read plenty of student reviews to make sure you're applying somewhere reputable. Read their blog and have a look at their social channels.

3. Throw yourself in

Once you've decided it's right



Becoming part of this vibrant, caring community was something I hadn't expected before the course, but now I couldn't be without it. To be a Northcoder is to be enlightened, inspired and supported.

Joanne Imlay

Primary school teacher to software developer at Careicon



Northcoders delivered their part of the bargain in spades. They provided tremendous assistance in turning me into the full product - a well-rounded, capable, future tech employee - and they have the contacts to deliver the opportunities for such people.

Joe Mulvey

Maths teacher to software developer at Auto Trader



udemy

UDEMY

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

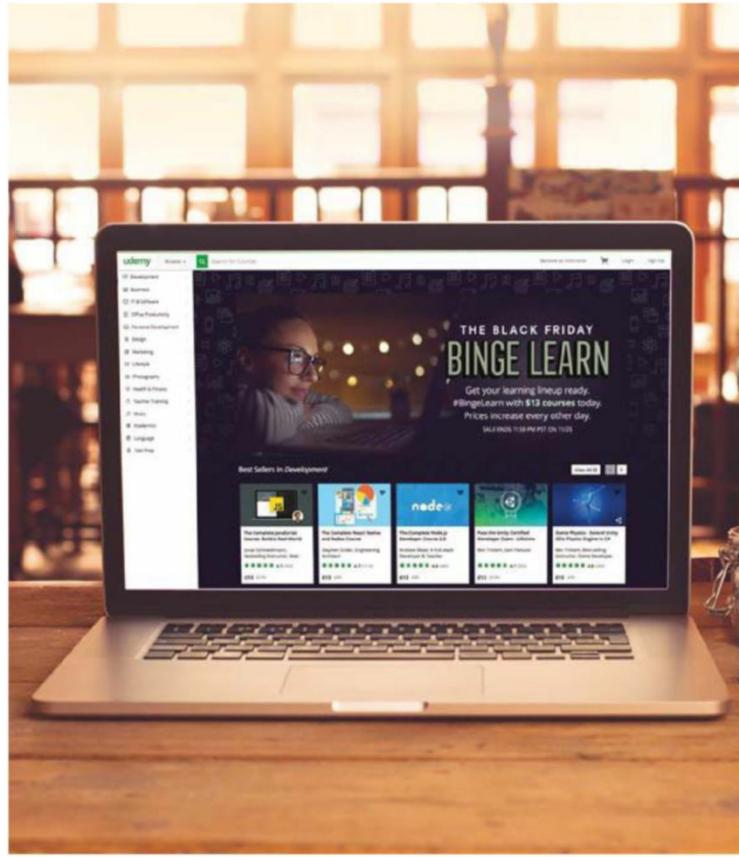
Twitter: [@udemy](https://twitter.com/udemy)

Facebook: [udemy](https://facebook.com/udemy)

The inspiration for Udemy began in a small village in Turkey, where founder Eren Bali grew up frustrated by the limitations of being taught in a one-room school house. Realising the potential of learning on the internet he set out to make quality education more accessible. Udemy is now a global marketplace for learning and teaching online. Students can master new skills by choosing from an extensive library of over 40,000 courses including HTML, CSS, UX, JavaScript and web development.

40,000+ courses: There is a course for every designer and dev.

Self-paced learning: Learn how to code at your own pace.



THE IRON YARD

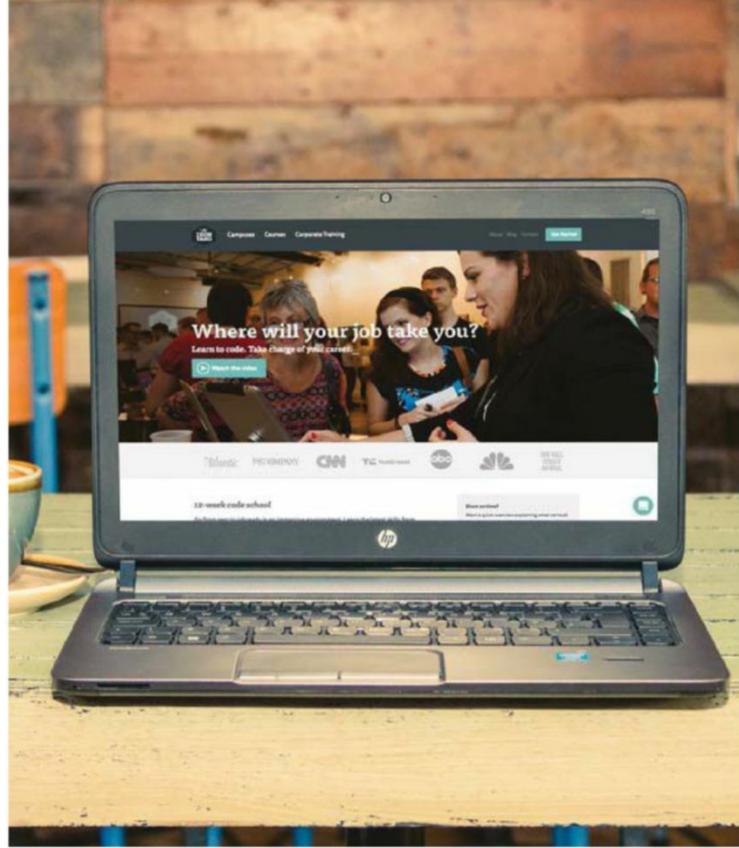
[theironyard.com](https://www.theironyard.com)

Twitter: [@TheIronYard](https://twitter.com/TheIronYard)

The Iron Yard is one of the world's largest and fastest-growing in-person code schools. It offers full-time and part-time programs in backend engineering, frontend engineering, mobile engineering and design. The Iron Yard exists to create real, lasting change for people, their companies and communities through technology education. The in-person, immersive format of The Iron Yard's 12-week courses helps people learn to code and be prepared with the skills needed to start a career as junior-level software developers.

12-week code school: Learn the latest skills from industry pros.

Free crash courses: One-night courses, the perfect way to learn.



WE GOT CODERS



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

hello@wegotcoders.com

We Got Coders is a consultancy that provides experts in agile web development, working with startups, agencies and government. Take one of their 12-week training courses that covers all that is required to become a web developer, with highly marketable full-stack web development skills.

- Classroom-based training
- Real-world work experience
- Employment opportunities

FUTURELEARN



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

feedback@futurelearn.com

Choose from hundreds of free online courses, from Language & Culture to Business & Management; Science & Technology to Health & Psychology.

Learn from the experts. Meet educators from top universities who'll share their experience through videos, articles, quizzes and discussions.

- Learn from experts
- Free courses
- All-device access

GYMNASIUM



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

help@thegymnasium.com

Gymnasium offers free online courses, designed to teach creative professionals in-demand skills.

Courses are all self-paced and taught by experienced practitioners with a passion for sharing practical lessons from the design trenches.

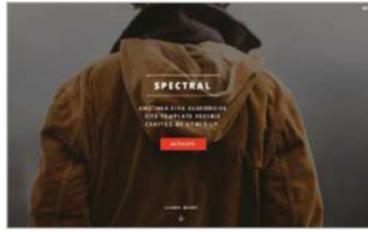
- Gain real-world skills
- Get expert instruction
- Career opportunities

Free with your magazine

Instant access to these creative resources...

Essential assets and resources

Get textures, fonts,
backgrounds and more



Exclusive video tutorials

Learn to code/create with
HTML, CSS, JS & PHP



Tutorial project files

All the assets you'll need
to follow our tutorials



Plus, all of this is yours too...

- All-new tutorial files to help you master this issue's HTML, CSS and JavaScript techniques
- 70 minutes of expert JavaScript video from Pluralsight (www.pluralsight.com)
- 13 hi-res greeting card mockups and 10 wooden picture frame mockups from Sparklestock (www.sparklestock.com)

→ Log in to www.filesilo.co.uk/webdesigner

Register to get instant access
to this pack of must-have
creative resources, how-to
videos and tutorial assets

Free
for digital
readers, too!
Read on your tablet,
download on your
computer

The image shows a magazine cover for 'Web Designer' magazine with the headline 'SEO TODAY'. It also features a laptop displaying the 'FileSilo.co.uk' website, specifically the 'Web Designer' section, showing various digital resources. A white coffee mug is placed next to the laptop.

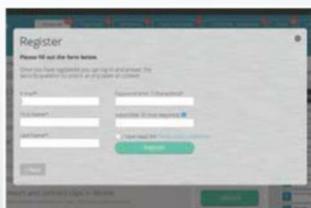


The home of great downloads – exclusive to your favourite magazines from Future!

- Secure and safe online access, from anywhere
- Free access for every reader, print and digital
- Download only the files you want, when you want
- All your gifts, from all your issues, in one place

Get started

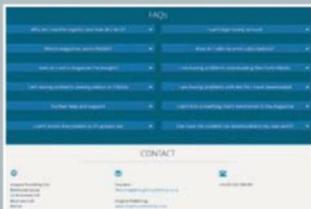
Everything you need to know about accessing your FileSilo account



- 01** Follow the instructions on screen to create an account with our secure FileSilo system. Log in and unlock the issue by answering a simple question about the magazine.



- 02** You can access FileSilo on any computer, tablet or smartphone device using any popular browser. However, we recommend that you use a computer to download content, as you may not be able to download files to other devices.



- 03** If you have any problems with accessing content on FileSilo, take a look at the FAQs online or email our team at the address below.

filesilohelp@futurenet.com

An incredible gift for subscribers



Subscribe today & unlock the free gifts from more than 50 issues

Access our entire library of resources with a money-saving subscription to the magazine – that's more than 900 free resources

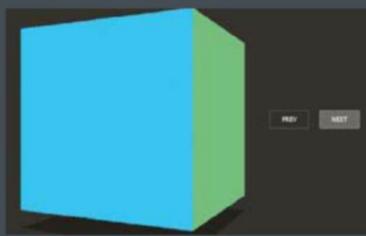
Over 60 hours of video guides

Let the experts teach you to create and code



More than 400 tutorials

Get the code you need to get creative



Over 210 creative assets

Templates, fonts, textures and backgrounds



Head to page 32 to subscribe now



Already a print subscriber?
Here's how to unlock FileSilo today...

Unlock the entire Web Designer FileSilo library with your unique Web ID – the ten-digit alphanumeric code printed above your address details on the mailing label of your subscription copies – also found on any renewal letters.

More than 900 reasons to subscribe

+
More added every issue

NEXT MONTH

UX 2018

CURRENT TRENDS, BEST PRACTICE, ESSENTIAL
TIPS & TECHNIQUES FOR TODAY

HOT NEW CSS PROPERTIES

Get a closer look at the properties that will soon be part of your everyday arsenal

CREATE A FACE TRACKING APP

Add graphical elements that track to the face with the option to download and keep

WHAT IS THE WEB PAINT API?

We reveal how JS and CSS can talk to each other to create dynamic backgrounds

Visit the **WEB DESIGNER** online shop at



myfavouritemagazines
myfavouritemagazines.co.uk

for the latest issue, back issues and specials

ALL IN YOUR NEXT
WEB DESIGNER
Issue 274 on sale
Thursday 26th April 2018





Did you know that European forests, which provide wood for making paper and many other products, have grown by 44,000km² over the past 10 years? That's more than 1,500 football pitches every day! [†]

Love magazines? You'll love them even more knowing they're made from natural, renewable and recyclable wood



[†]UNFAO, Global Forest Resources Assessment 2005-2015.

Two Sides is a global initiative promoting the responsible use of print and paper which, when sourced from certified or sustainably managed forests, is a uniquely powerful and natural communications medium.

There are some great reasons to #LovePaper
Discover them now,
twosides.info



moqups



Mockup, wireframe, diagram, prototype
& collaborate with one flexible solution.

moqups.com

OFFER EXPIRES
MARCH 1ST 2018

Save an extra 30% on all new Moqups plans
Discount code: **NETMAG30**

**SAVE
AN EXTRA 30%**

