Bootstrap Quick Start

1.2:v4.0.0-beta.2

Created by: Jacob Lett

BootstrapCreative

Dedicated to my wife Colleen,

Thank you for your persistent love and encouragement.

I would like to thank the following for their advice and support with this project: Greg Vance, Dan Joseph, Chris of D.A., Max, and my Parents. Marcus and Joshua thank you for your patience as your Dad was working on this project. Also my mentors for their wisdom and inspiration: Zig Ziglar, Dan Miller, Seth Godin, and Matthew 25:14-30.

Finally, thank you reader for purchasing this book.
My hope is to help you be successful in your
web development projects and career.

TABLE OF CONTENTS

Have a question or found an error? Please email me at https://bootstrapcreative.com/contact

Introduction

INTRODUCTION

Do you remember learning how to write a research paper? Perhaps you learned how to follow the MLA or APA Style Guide. Think how different the reports would look if each student made up their own style and format?

Using the MLA Style guaranteed consistency for anyone who followed that same format even if they were thousands of miles away or written five years apart.

I believe Bootstrap works the same way for web design. It helps developers work more efficiently and write CSS in a clean and consistent manner regardless of where you live or who you work for. It also ensures your website adheres to a mobile first approach and works well across browsers and devices.

I am excited to see what you will build with Bootstrap.

Hi.

My name is Jacob Lett and it's my mission to help you save time learning how to design and build responsive websites.

I earned a bachelors degree in graphic design around the time CSS and web standards were just starting to take hold. Out of frustration not knowing how to fix broken websites generated by Dreamweaver, I learned how to hand code HTML/CSS. Then in 2009 I got my first job as a web designer writing a ton of CSS and realizing I had a long journey of learning ahead.

linkedin

PREREQUISITES

Experience with HTML and CSS is helpful but not necessary. Even absolute beginners who have never written one line of HTML/CSS will be able to follow along the steps.

This training will not require knowing Sass, the command line, or Photoshop.

Recommended Software and Tools

A code text editor so you can benefit from code syntax coloring and other features that make the job of writing code easier. You can find a list of text editor options in the Appendix.

> *Every line of code should appear to be written by a single person, no matter the number of contributors.*
>
> — *Mark Otto, creator of Bootstrap*

Google Chrome because of their nice set of DevTools to help you debug problems and inspect CSS styles and HTML elements. If you do not have this installed you can do so here. Click here to learn more about things you can do with DevTools.

live.js Chrome extension to toggle the ability to automatically reload your browser when you make code changes.

Web Developer Chrome extension to help you test responsive breakpoints and perform other developer focused tasks.

The Mobile Web and Bootstrap

Web designers and developers have had to quickly develop creative solutions to work within the new constraints presented by touchscreens and mobile devices. Bootstrap has been helpful in solving a lot of the challenges faced when building responsive websites.

Building websites today is a lot more challenging and time consuming than it used to be. Some of my first websites were first designed in Photoshop, exported to HTML tables (yes tables) and then linked together with Dreamweaver. If your website did not exceed the width of common monitor resolutions (1024px by 768px) everything would work out fine.

Web standards were quickly introduced because using table markup for grid layout is just bad practice. So HTML tables were replaced with floated divs and tag markup that had meaning – referred to as semantics. This also shifted things away from the majority of the visual design being baked into images and now relying on CSS3 to create borders, shadows, rounded corners, etc.

The first widely used CSS grid system was the 960 grid system (Fig. 1) created by Nathan Smith. This 12,16, 24 column grid system was designed to work well for a fixed desktop resolution of 1024px x 768px. This grid system was widely used and helped designers and developers work from the same grid pixel dimensions.

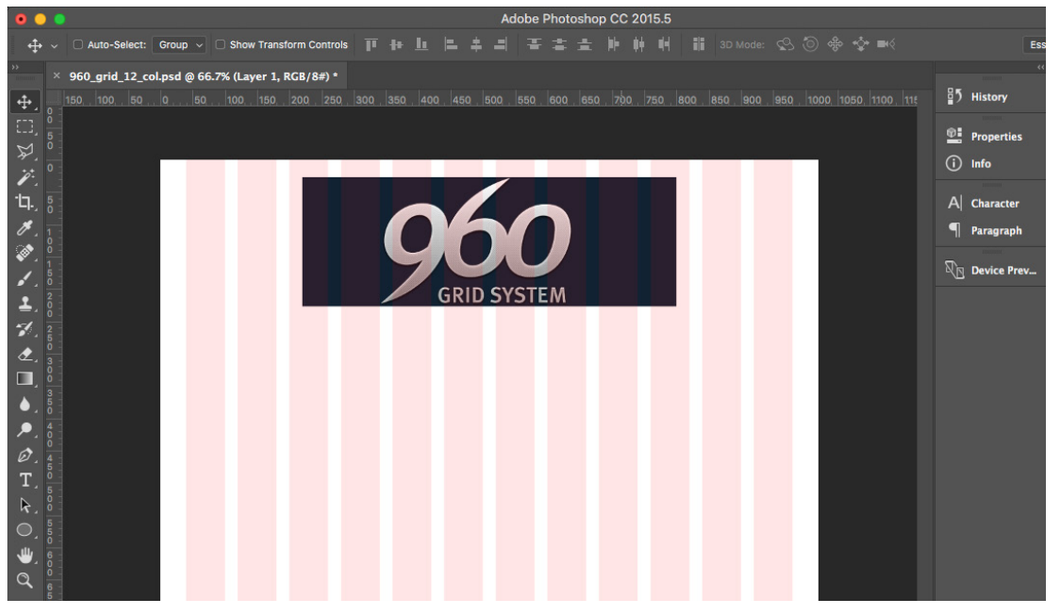


Fig. 1: The 960 grid system helped bring consistency between grid design in Photoshop and the web.

Then in 2007, Steve Jobs introduced the world to the iPhone with Multi-Touch gestures. Now people could access websites anywhere using just their fingers.

Web designers and developers had to quickly develop creative solutions to work within the new constraints presented by smartphones and tablets.

These Constraints Include:

- Smaller screens
- Increased pixel densities with retina displays
- Ability to switch between portrait and landscape orientation

- Multi-touch gestures
- Slower data connections
- Distracted user attention (one eyeball and one thumb).

At the start, the concept of responsive design did not exist. And so mobile devices had to scale down websites to fit the smaller screens. For the user, in order to read the text they would have to double tap the screen or pinch and zoom.

Website owners quickly realized it was not a good experience to display their homepage at a zoomed in level. The meta tag below was introduced to remove this default scaling and give the site creator more control.

```
<meta name="viewport"
content="width=device-width, initial-
scale=1">
```

Mobile Applications

One approach is to build a dedicated experience as a mobile app. This gives the developer the most control and could utilize the device user interface components and to help with navigation. Major drawbacks include: it requires an app developer, considerable amount of marketing to direct exisitng traffic to the mobile app, and overcoming low rates of user adoption. Also, any links to outside pages required them to open in a web browser window.

Adaptive Design

Another approach is to build multiple versions of a website and use server side detection to then present custom code for that device or viewport size.

You could decide to have your mobile site on a separate domain for example m.domain.com. The server will then automatically serve all mobile traffic to that domain.

The server could also perform dynamic serving of page content so that you have just one domain name.

The downsides to this approach is it requires complex server side detection code and is harder to maintain multiple site versions.

Responsive Design

Responsive design was introduced to help designers build one site on one domain that responds to a users viewport. The two necessary elements for a responsive design are a meta viewport tag to disable scaling and media queries to alter the design as the page gets smaller. Responsive design is a lot less expensive and

easier to maintain than the other mobile strategies. This has added to its rapid growth and adoption.

A big challenge with responsive design is finding a balance between the content needs for both mobile and desktop. A desktop site has a lot of visual real estate that is often filled with carousels, videos, large parallax background images, and large blocks of text.

If you load a feature-rich website on a mobile device you often increase the page load for mobile visitors. This is due to the large images and videos which are scaled down to mobile.

Fig. 2 - Desktop First Responsive Site

	Desktop	Mobile
Data Speed	Fast	Slow
Width	Wide	Narrow
Height	Unlimited	Unlimited
Retina Display Probability	Medium	High
Page File Size	Large	Large

> *End-users don't care about your responsive web or your separate sites,* ***they just want to be able to get stuff done.***
>
> — *Brad Frost, author of Atomic Design*

Mobile First and Progressive Enhancement

In the desktop first approach, you sacrifice the mobile experience because you have a lot of images and text content. In an article from Zurb on mobile first design it said, "Roughly 80% of the screen size is taken away when you start with mobile first design, you have to think about how to utilize your space in a much more conservative manner."

A mobile first approach considers the goals of a mobile user and presents the content to help them achieve those goals. It removes all of the fluff and filler content and presents a concise collection of content that loads fast and is easy to use.

Fig 3. Mobile First Responsive Site

	Mobile	Desktop
Data Speed	Slow	Fast
Width	Narrow	Wide
Height	Unlimited	Unlimited
Retina Display Probability	High	Medium
Page File Size	Small	Medium +

The chart above shows the workflow flipped so the site is built mobile first and then enhancements are added as the viewport gets wider. Notice how the mobile site is loading a small file size on a slow data speed? That is as Google would say, being mobile friendly.

But some might say. "Ok now the mobile site looks good but now the desktop looks too basic and lacks flair."

A great way to solve this is to progressively enhance the page as your data speed and screen width increases. Everything you add to the page will be enhancing the design and if it doesn't load for some reason your page is still usable.

Screens are small, connections are slow, and people often only give you their partial attention or short bursts of their time. ***Designing for mobile first forces you to embrace these constraints***

— Luke Wroblewski, Mobile First

The best way to do this is with JavaScript media queries to determine viewport width and then load in content to the page. I created a small plugin called IfBreakpoint.js to help detect Bootstrap 4 breakpoints with JavaScript. I also recommend reading this article on ways to progressively load images with media queries.

One creative solution that has transformed the web and made responsive design easier for web designers has been the Bootstrap CSS framework. We will take a closer look at Bootstrap in the next section.

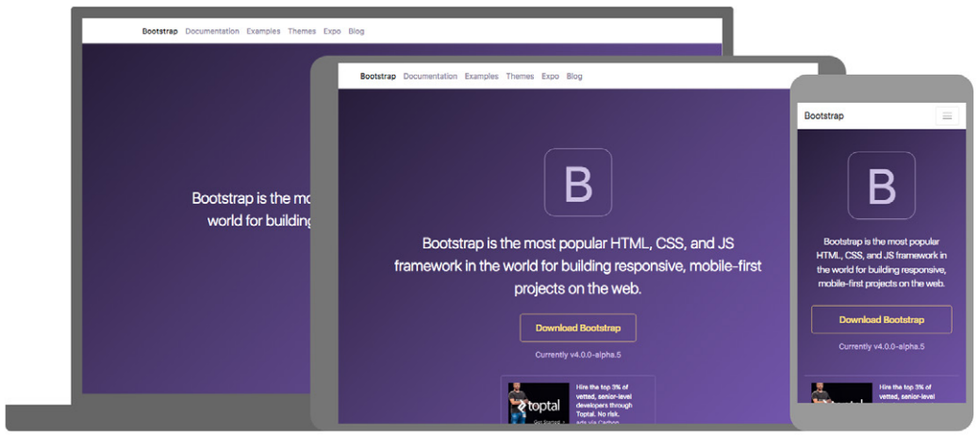
Bootstrap Cuts Development Time in Half

I remember building my first few responsive websites. I wasted so much time writing the same type of styles over and over for each new project. I also found it difficult to find plugins that worked well together and had cohesive design style

I then heard about Bootstrap and I liked how it included javascript components and had really comprehensive documentation. The documentation was extremely detailed and easy to follow. At first, it was hard to know what classes did what but after using it on a few projects I was amazed at how quickly I could create a working prototype of a design. The time I saved meant I could complete more projects in less time and make more money in the process.

The more I used Bootstrap, the more I felt like it could be a global standard because it removes a lot of routine tasks when building responsive sites.

Bootstrap was created by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. It is now an open source project hosted on GitHub and has seen rapid growth and global use in web applications and websites.



Bootstrap CSS Framework History

- Before 2011 — An internal Twitter tool
- August 2011 — Released as open source
- January 2012 — Bootstrap 2
- August 2013 — Bootstrap 3
- August 2015 — Bootstrap 4 Alpha
- August 2017 — Bootstrap 4 Beta
- ?? — Bootstrap 4

A Toolkit Built in Style Guide Form

When Bootstrap was first created at Twitter it was built as a toolkit of reusable components with additional documentation and code snippets on how to use them. This helped a team of multiple developers work on a project and have a cohesive methodology on

how to build layouts. The documentation and ease of implementation, made it easy to share and reference with others, regardless of their skill level.

So, **the initial intent of Bootstrap was to be a living style guide documentation for a team of developers** to code in the same way following a set of pre-defined rules and components.

Today, Bootstrap Can Be Used in Two Main Ways:

1. Linking to a precompiled version via CDN or locally
2. Linking to a customized build using the Sass source files

On the next page (Fig. 5) I explore the pros and cons of each method and also break them into smaller sub-methods to help you decide which is best for your project.

A System of Components

At this point, you might be wondering what is a component and why does Bootstrap use them?

Well, one definition I found was, "A component is a minimal software item that can be tested in isolation." The keyword in that phrase is **isolation**.

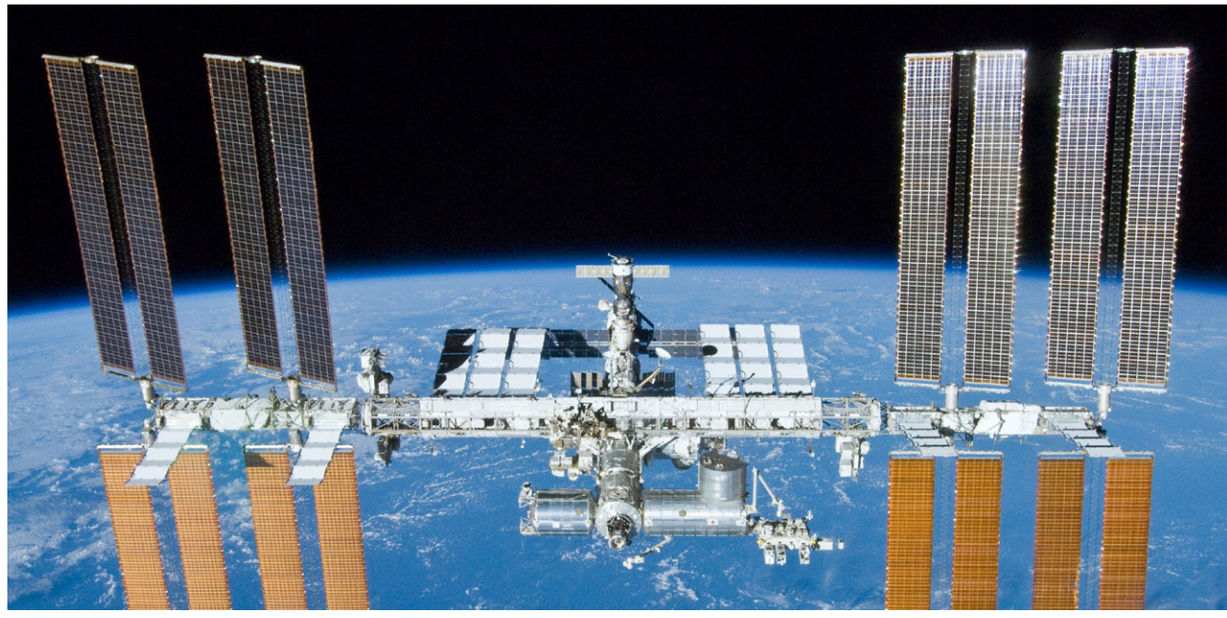


Fig. 4: The International Space Station is made up of isolated components that perform a specific purpose. Combined they create a system of linked components.

Since CSS cascades down to child elements, how do you isolate things and write styles to target specific components and leave everything as is? The solution Bootstrap presented is the use of prefix class naming and sub-classes for variations.

Mark Otto wrote on his blog, "Each class name begins with a prefix. Class name prefixing makes our code more durable and easier to maintain, but it also better enables us to **scope styles to only the relevant elements.**"

Fig. 5 - The Different Ways to Use Bootstrap

Method		Pros	Cons	Use Cases
B1a	**Link to CDN Minified Files** Difficulty: beginner	• Fast • Easy to setup • No preprocessor needed	• Lacks unique visual style • Some code bloat of unused components	• Cases where custom branding is not a priority • Backend layouts • Prototypes
B1b	**Link to CDN Minified Files + Custom Stylesheet** Difficulty: beginner	• Fast • Easy to setup • No preprocessor needed	• The time to inspect and overwrite Bootstrap styles • Some code bloat of unused components	• The method used in this book • Production sites that require unique branding
B1c	**Link to CDN Minified Files + Custom Stylesheet from Sass Files** Difficulty: intermediate	• Some setup time • You gain the benefits of Sass with mixins, variables, and multiple files.	• The time to inspect and overwrite Bootstrap styles • Some code bloat of unused components • Requires knowledge of Sass and how to compile it	• Production sites that require unique branding
B2a	**Link to Custom Sass Build** Difficulty: advanced	• More setup time • Gain the benefits of Sass • Removes code bloat	• Knowledge of Sass and how to compile it	• Production sites that require unique branding
B2b	**Link to Custom Sass Build + Docs Build** Difficulty: advanced	• More setup time • Gain the benefits of Sass • Removes code bloat • Create updated documentation	• Knowlege of Sass and how to compile it • Knowledge of Jekyll and how to compile it • Documentation updates over time	• Production sites that require unique branding • Multiple developers work on a single project

Is a CSS Framework Like Bootstrap Even Necessary?

If you are an experienced web designer or developer you are probably wondering what the benefits are using Bootstrap in your project. Prior to using Bootstrap, I used a boilerplate I wrote myself that consisted of a reset, basic grid, typography, utilities, and media queries. Below are the benefits I have experienced from now using Bootstrap for my projects.

Helps You Save Time

I admit I was the worst at documenting my own work. I would use my boilerplate on a project and then want to make an update to it a month later. But by then, I totally forgot my naming convention. So I would have to spend time reading my code to try to understand what I did. If I couldn't figure it out, I would add new code and leave the old code alone to prevent breaking something. Yup, sound the code bloat alarm.

Bootstrap has amazing documentation on each component. So if I want to update a project I worked on a few months ago that uses Bootstrap I know where to go to find documentation if I get stuck. Also, the more I use Bootstrap the more it is burned into my brain and the less time is spent searching the docs.

> *Documentation is a love letter that you write to your future self.*
>
> — *Damian Conway, Perl Programmer*

Helps You Avoid Cross-browser Bugs

Prior to using Bootstrap I would get the dreaded emails from clients saying their website they just paid me for doesn't look good on X device. And of course, it is a device I do not currently own or have access to. After hours or searching on Google you finally find a fix on Stack Overflow. You find comfort knowing it is a common problem with Android devices and not something you caused. Clients assume you know how to fix everything.

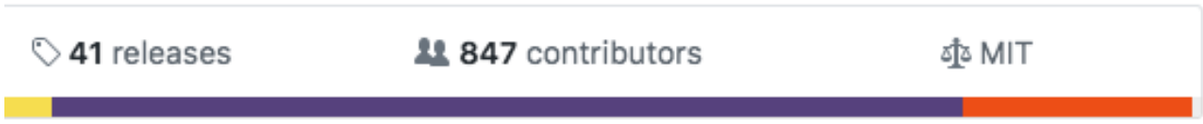


Fig. 6: A large number of contributors helps improve the framework over time.

Being an open source project, anyone can submit browser bugs and code fixes for it. This is an extremely valuable asset to a developer because you gain confidence knowing your code has been improved by a community to address common browser bugs. No matter how good you are, there is no way you can be aware of every browser inconsistency and the fix necessary.

Helps You Follow Best Practices

I studied graphic design in college and self-taught myself HTML & CSS from books, YouTube, and blog posts. This mixture of knowlege worked to some degree but I know there are a lot of knowledge gaps. I hit my lack of understanding head-on when I first learned Bootstrap with all of the new terminology written for software engineers.

Bootstrap is not just a framework but a methodology of best practices for front-end design.

Gather a room full of the smartest web designers and developers and let them discuss at length what they think is the best way to write CSS and to organize a project. The result being a distilled version of best practices agreed upon by a large collection of your peers.

Helps You Avoid jQuery Plugin Soup

I know some JavaScript but writing a full-fledged plugin is out of my reach. So I often collected various jQuery plugins into a project to achieve the look and functionality I was looking for.

But I often ran into the following problems:

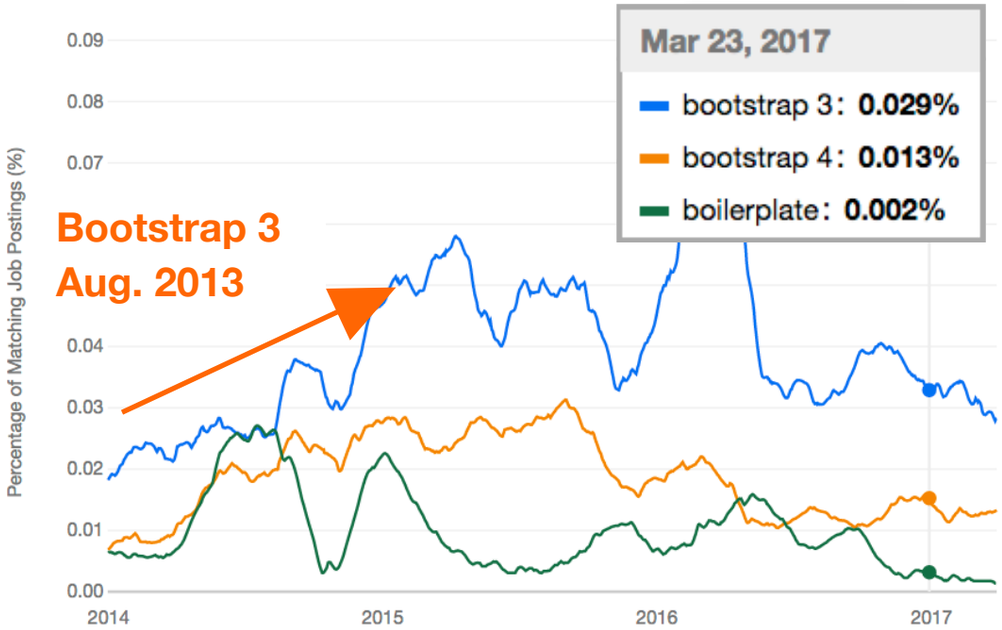
1. Plugins would not work across browsers
2. Plugin CSS styles would conflict with other CSS styles
3. Plugins would be dependent on different versions of jQuery

Bootstrap contains a collection of jQuery components that you know are stable on modern browsers compatible and works with your jQuery version. Also, the styling matches all of the other components in the your project.

Helps You Get a Job

Bootstrap has 73% of the design framework market share as of May 2017. This popularity correlates to the demand for people to know the framework to either update existing systems and or create new ones.

So this will make you more marketable to prospective employers. Indeed.com, a popular job search engine, shows Bootstrap has a lot of job postings compared to other CSS frameworks.



After Bootstrap 3 was released in August 2013 there was an increase in job postings thereafter. I believe we will see a similar spike when v4 is released. Source:indeed.com

Framework Name	# Sites that Use
Bootstrap CSS *(update to v4?)*	12,559,226
HTML5 Boilerplate	4,219,959
960 Grid System	437,120
Semantic UI	10,803
Unsemantic	74,386

Source: BuiltWith as of May, 2017

Summary

When your time is billable, every minute you shave off will greatly improve your bottom line. Plus it makes development more fun because you are not declaring redundant CSS properties.

I hope I have shown you how Bootstrap can save you time and make you a better developer in the process. In addition, you will be following industry best practices vetted by an open source community.

Now that you know the history and benefits of using Bootstrap let's dive into what's new in version 4.

What’s New in Bootstrap 4

FLEXBOX GRID SYSTEM

The most important element of any CSS framework is the grid system. The Bootstrap grid has been used on many websites worldwide which make it extremely stable. This cross-browser support is why you probably are considering using Bootstrap for your website (it was for me).

In this section, I will provide an overview of the grid and provide examples to help you quickly apply it to your projects.

Important

Before you begin a project, you should know what set of web browsers you are going to support. This will actually determine what version of Bootstrap you use because Bootstrap 4 is not supported on IE9 and below.

What Versions of Internet Explorer Do You Need to Support?

So how do you know what browsers to support? If you are redesigning an existing site, I suggest looking at your Google Analytics to see what browser the majority of your site visitors use. Look for trends to determine if it makes sense to remove support for an older browser.

If you have no analytics to work with I suggest looking at StatCounter to see the top browsers in your country. But from my experience, it is best to have a clear understanding of your ideal site visitor. Because there are a lot of factors, tools like StatCounter do not factor in. One of those being corporate environments that are slow to upgrade to newer browsers.

One way to get a clear picture of your target site visitor is through surveys or live interviews. Talk to likely site visitors and ask them what browsers they use and if there are any IT restrictions preventing them to upgrade browsers.

Once you have your data and some assumptions follow the decision chart below (Fig. 7) to determine what Bootstrap version you should use.

Bootstrap Version Decision Chart

I need to support IE10+	Use Bootstrap 4
I need to support IE9+	Use Bootstrap 3
I need to support IE8+	

What is Flexbox?

In Bootstrap 3 and for the majority of websites, the only way to build multi-column layouts was to set column widths and use floats. Then on mobile, you would just remove the float and width property so that it would change to be one column.

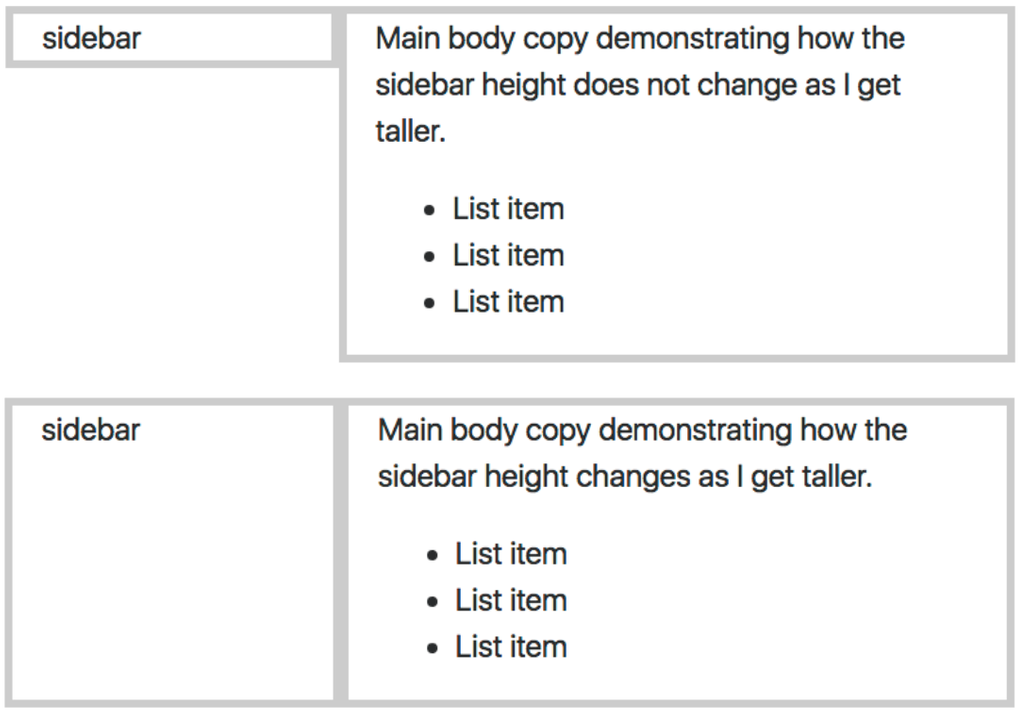


Fig. 8: Demonstrates how a flexbox grid adjusts the height of sibling columns while a float grid does not. Top: Float columns; Bottom: Flexbox columns

Now with flexbox, or flexible box, you will be able to build complex grid layouts with more control and flexibility to adapt the layout as the viewport changes.

If you are familiar with an UL and LI relationship, flexbox is very similar in how it has sub items or flexbox items inside a parent wrapping container. But since flexbox is a display property it can be applied to any parent and child HTML elements and does not have its own HTML element like `<flexbox>`.

Fig. 9: One of the most exciting features of flexbox is how it handles vertical alignment.

Keep in mind, Bootstrap is a CSS framework that builds upon the core language of CSS. So flexbox is the core CSS technology that Bootstrap uses for grid layout and is not a component created by Bootstrap. So it is helpful to know the fundamentals of flexbox in case you need to override something.

Here are some additional sources to learn more about flexbox: Solved by Flexbox and my Bootstrap 4 flexbox utility classes cheat sheet.

12 Column Grid

So now that you understand flexbox and why it's superior to floats for layout, lets look at how Bootstrap uses this for their grid system.

The Bootstrap grid system is based on a 12 column grid because the number 12 is divisible by 12, 6, 4, 3, 2. So your column sizes inside each row will need to equal 12. This math makes the grid more flexible for a wide range of layouts.

Common Grid Layout Examples:

- 2 column grid
  .col-sm-6 + .col-sm-6 = 12
- 2 column golden ratio grid
  .col-sm-8 + .col-sm-4 = 12
- 3 column grid
  .col-sm-4 + .col-sm-4 + .col-sm-4 = 12

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Fig. 10: By default, Flexbox item widths are equally distributed to fill the width of the container. If you set the column class to `.col` *it makes each column horizontal at all breakpoints. So it doesn't respect the 12 column grid like you think it would as shown in this example.*

1	2	3	4	5	6	7	8	9	10	11	12
13	14	15									

Fig. 11: In this example, I changed the column class to `.col-sm-1` *instead of* `.col` *which restricts the column width even after it has been wrapped. Columns widths are equally distributed to fill the width of the container. Since Bootstrap uses a 12 column grid, it breaks after 12 and the remaining columns are equally distributed.*

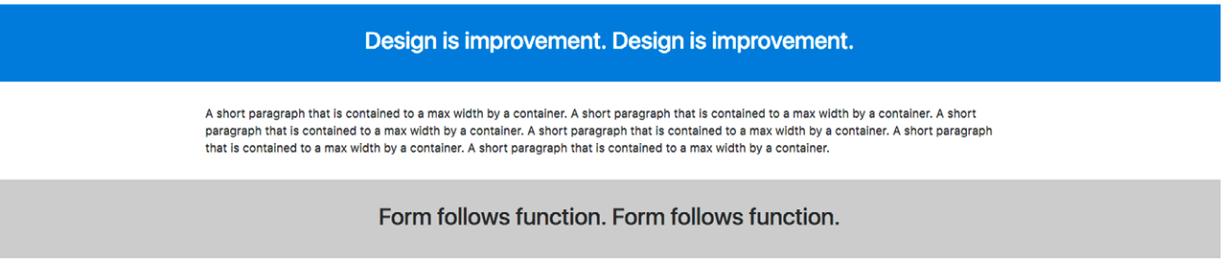
The Bootstrap Grid System Has Three Main Parts: CRC

When working with the Bootstrap 12 column grid you have to keep in mind the order of elements and that there are always three parts: a Container, a Row, and any number of Columns.

Tip: You can remember CRC by thinking of a **C**abin, a **R**oom inside the cabin, and finally **C**hairs inside the room.

If you want all of your page content to be constrained to a max-width, you would just need one `.container` on your entire page. Or if you would like your columns to fill the entire window and have no max-width use `.container-fluid`. Then use a series of row blocks with column divs to build your grid. If your design has no

horizontal color banding you could set `.container` to the body tag. However, there is a design trend to have horizontal background colors with the content set to a max-width.



The way I achieve this effect is using a section tag with a background color set to it.

```
section.bg-primary > .container > .row > .col-sm-6
```

If you are familiar with a HTML table structure the grid system is very similar.

For example:

```
table > tr > td
        is like
.container > .row > .col-sm-6
```

Container

`.container` or `.container-fluid`

This is the parent container that determines if the columns should be full-width or not.

Row

`.row`

A horizontal wrapping container for the series of columns it contains.

Columns

`.col` or `.col-*`

A column is a vertical division similar to a table cell. This is where your content goes and has built-in margin to the left and right to prevent text and images from touching each other.

Columns also have grid tiers which tell the columns how they should look at different breakpoints. In the example below I used .col-sm-6 which essentially says, “When the browser window is 576px or higher make this column span 6 of the 12 columns. For anything below 576px make it full width.”

So when you declare a grid tier you are saying make it this size for the specified tier and up.

Fig. 10: **Bootstrap Grid Tiers and Breakpoints**

xs	Extra small <576px	portrait mobile
sm	Small ≥576px	landscape mobile
md	Medium ≥768px	portrait tablets *navbar collapse*
lg	Large ≥992px	landscape tablets
xl	Extra large ≥1200px	laptops, desktops, TVs

Conclusion

The Bootstrap CSS Grid System is extremely customizable and helps you be more efficient in building page layouts. And now with flexbox instead of floats, you will have more control on how your columns are positioned on the page.

Key points

- Uses a 12 column grid and the number of columns has to equal 12
- **CRC** – `.container > .row > .col-*-*`
- Most projects will just need one .container unless you want to do colored row banding
- Setting a grid tier like `.col-sm-6` says for sm and up
- Columns have horizontal padding to create the gutters between individual columns, however, you can remove the margin from rows and padding from columns with .no-gutters on the .row.
- Grid columns without a set width will automatically layout with equal widths.

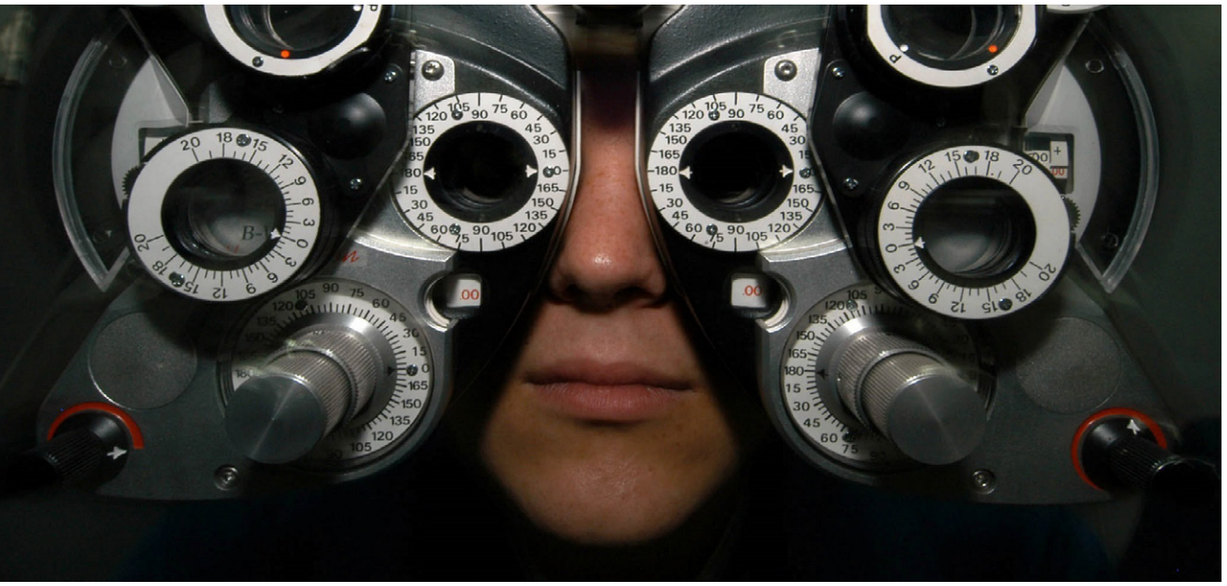
View documentation

The Design Process

The design process helps designers bring clarity to their solution gradually decreasing the changes of re-work and project delays. It emphasizes a design strategy based on research and branding and not on trends.

Before we start using Bootstrap I want you to be familiar with the design process I follow for most projects. This four phase process has milestones that require approval by the decision maker before moving onto the next phase. These approvals help to eliminate costly re-work at later stages, delayed projects, and stress.

For example, changing a navigation item name in the design phase could break the layout and require time to re-write styles to accommodate.



The last time I was at the eye doctor I was fitted for glasses. As I was looking through one of these, my doctor flipped lenses and asked me questions like "Can you see better with these lenses? How about now? Better or worse?" With each lens flip, my vision would get clearer and clearer. The design process acts in a very similar way in bringing more clarity to the project you are building.

Starting out with a blank sheet of paper can be daunting. But by asking a series of questions before you start your design, you can begin to bring clarity to the final solution.

> *"Research is the gathering of facts. In the absence of facts, you have assumptions. And assumptions are the enemy of design."*
>
> *— Mike Monteiro*

Below are the four phases of the design process: strategy, prototype, design, and delivery. As you progress through the phases you gain more clarity on the final design solution. Each phase requires client sign-off before moving on.

These checkpoints or milestones help decrease the chances of late changes causing project delays and increased costs.

1. Strategy

Interviews and research to define a clear brand strategy and design strategy to achieve business objectives.

Approvals needed

- Brand strategy
- Design strategy

2. Prototype

You will define the site architecture, page naming, and site content to be presented.

Approvals needed

- Wireframe & architecture
- Site copy
- Clickable prototype

3. Design

Work through branding and visual style. The client should approve code and not flat Photoshop files.

Approvals needed

- Clickable design of homepage
- Development site

4. Delivery

Publish final site and collect analytics for three months. Report back and review performance.

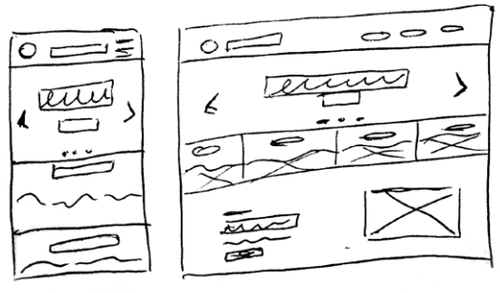
Approvals needed

- Final signoff
- Analytics report

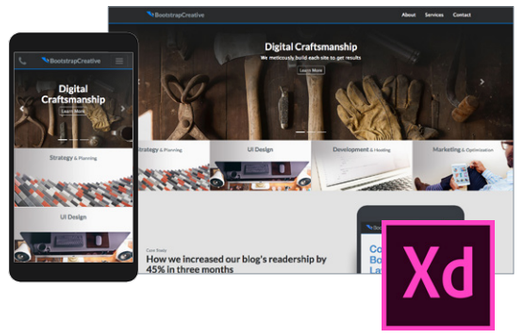
Phases Used in This Book

In this book, we will be spending our time focusing on phases two and three. These phases are closely related and could also be broken up into UX and UI. Prototypes focus on identifying content, naming, hierarchy and site architecture. The design phase converts wireframes into final layouts with graphics and branding.

This book will not be able to demonstrate how to generate pencil thumbnail sketches or perform design exploration in a graphics program. Both of these require a deep dive into design theory, branding, and design software. If you would like to learn these skills, please email me and tell me this is something you would be interested in.



2.
Prototype



3.
Design

Build a Homepage

INTRODUCTION

Are you ready to start building something? To help make the projects you build more practical, I am going to use my business name and pretend I am a medium sized web design agency. I am hiring you to help me design a website to help me gain leads online.

On the next page, I provide a sample brand strategy to help you learn about my company and the people I serve. Since I already have a logo and other marketing material I would like to continue using similar colors and fonts for consistency.

If you ever end up freelancing, you will often work with businesses who do not have an existing brand strategy and you will need to help them develop one. This initial project demonstrates your concern for brand consistency and how it helps differentiate them from their competition.

In addition, I provide a completed design strategy outlining the goals and expectations for the homepage design. This strategy is what we used to guide us through developing a wireframe and then a clickable prototype.

Design is the bridge that brings the business strategy to life. It can build intangible brand value for a business, so the business is worth more. We all know that the intangible brand value of Coke far outweighs the value of its tangible assets.

- Geoff Suvalko

Brand Strategy

Web design services company who helps businesses tell their story online. They obsess over the creative process to ensure efficiency and accuracy in meeting business objectives.

Audience

- Local professional service businesses who depend on a steady stream of leads.
- Businesses who have existing desktop websites and want to redesign to make them responsive.

How They are Unique

Digital craftsmen who believe design needs to tell a story to achieve measurable results.

#0275d8 #292b2c

#636c72 #e2e2e2

Lato Bold

Helvetica Neue Regular

Helvetica Neue Medium

Helvetica Neue Bold

Design Strategy

BootstrapCreative would like to attract business opportunities by demonstrating their expertise and past success with other companies.

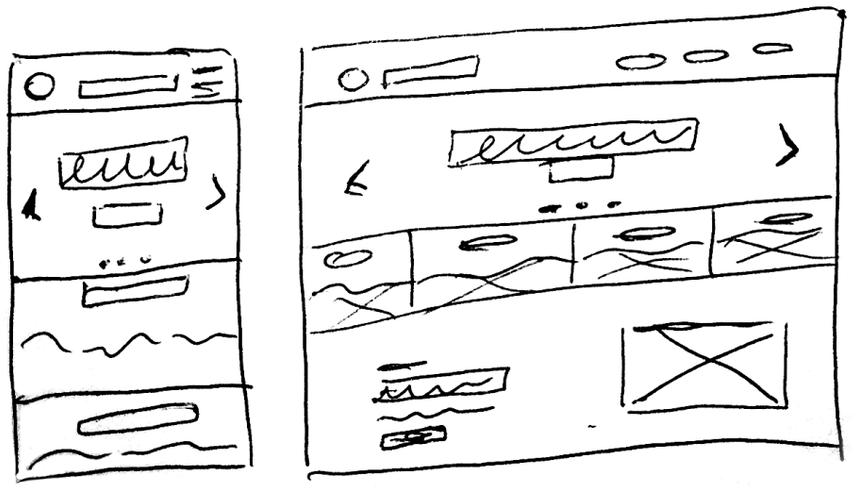
Design considerations

- Customers are tech savvy and primarily marketing professionals or small business owners
- Marketers who are very data driven and expect a ROI or measurable return on investment
- BootstrapCreative spends a lot of time researching their clients needs and business objectives and champion craftmanship.

Solution

Homepage will feature a carousel to highlight unique selling points. Article headings will clearly demonstrate growth and measurable return on investment.
Service offerings will be prominent so it is clear what BootstrapCreative provides. The high quality design will demonstrate competency and the ability to achieve quality work.

WIREFRAME

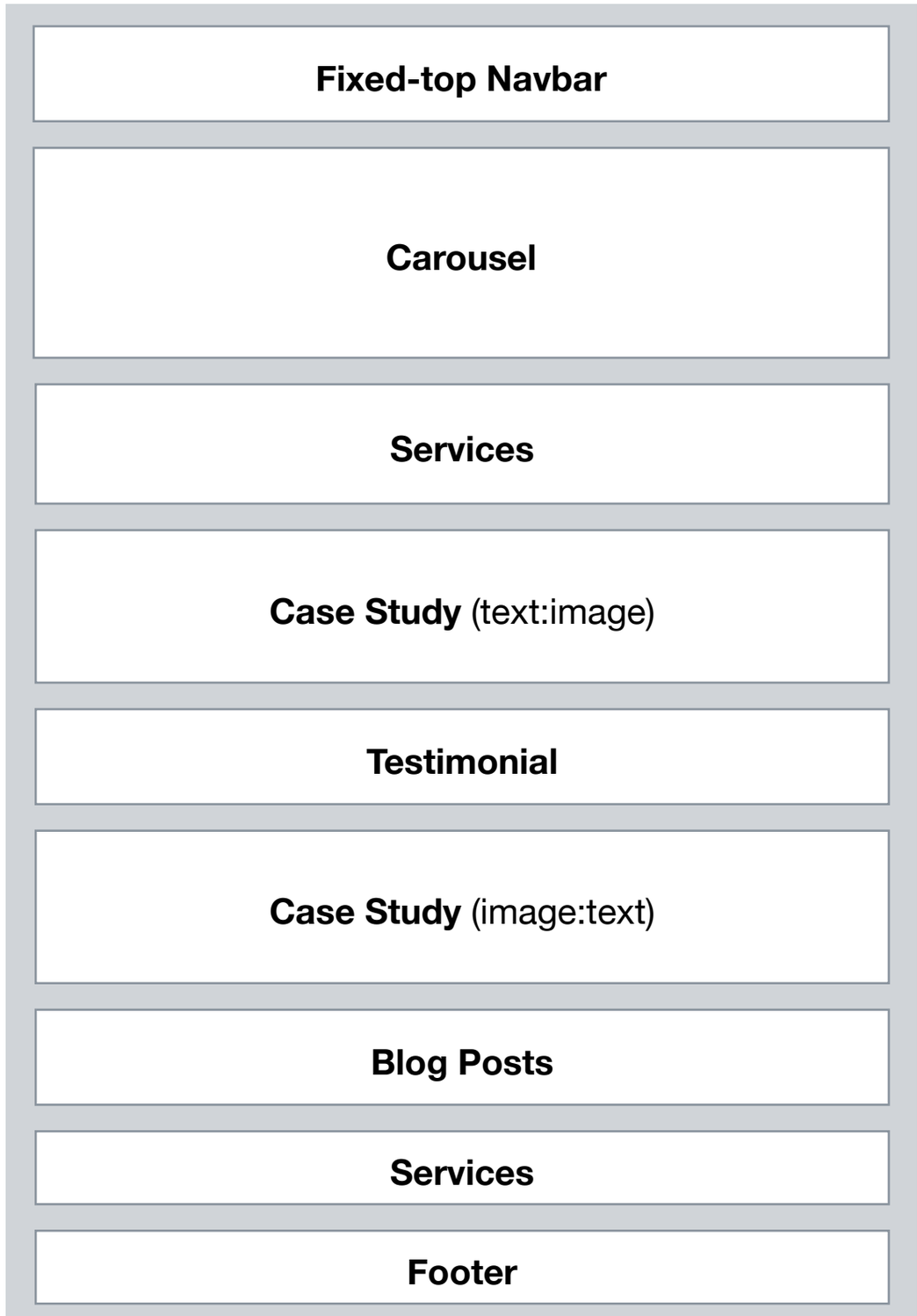


Wireframing is the most important step of any design. It forces you to think about how things will be organized and function.

What content is really needed? What behaviors would someone want to perform on this page? What will they look for the most?

Once we have a vision of what we want to build the next step is to explore the existing Bootstrap components to see what we can re-use in our project.

Wireframing tools: Balsamiq, Moqups, Powerpoint, Inkscape, and Google Slides. Site architecture tools: Coggle and MindNode.



BOOTSTRAP COMPONENTS USED

Now that we have a wireframe, the next step is to select existing Bootstrap components that we can use in our project.

Browse Bootstrap's documentation or if you purchased the Pro package, the Components Visual Reference is a handy tool to quickly review components.

Make notes of components that need further customization and those that need to be written from scratch. This step helps to eliminate code duplication and saves time in re-using existing components in your project.

A. Navbar

B. Carousel

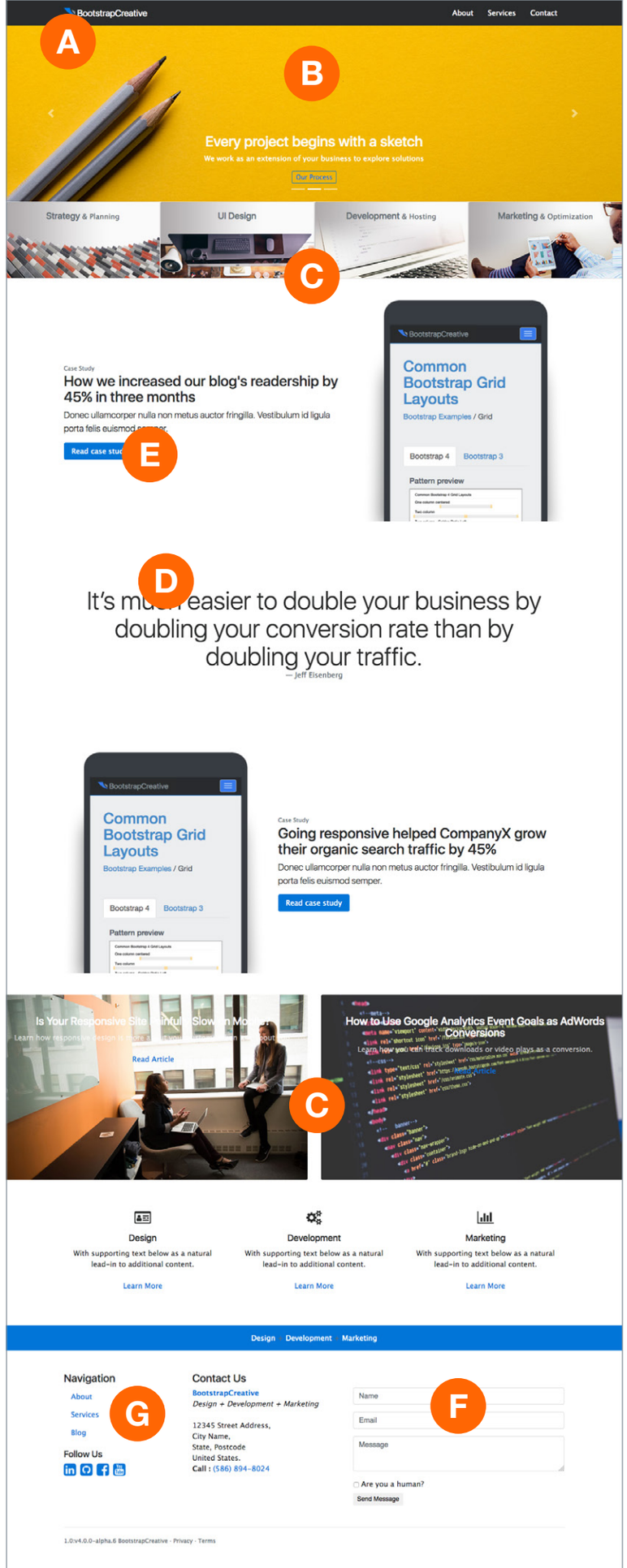
C. Cards

D. Blockquote

E. Buttons

F. Forms

G. Nav

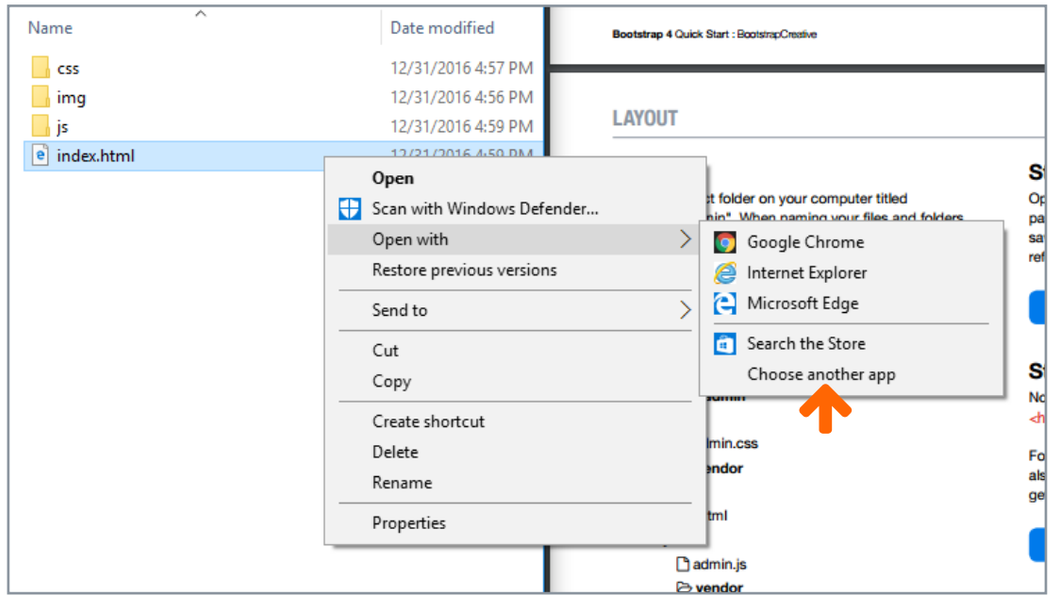


Step 1

Download the starter project .zip which contains the folder structure, images, and blank files. Save and unzip the folder where you would like to keep your website projects. I personally use XAMPP to run a local server and generally save all of my web projects in my **htdocs** folder that way I can test my projects at localhost/project-name. If you have never setup a local server before, you can follow my tutorial to learn the steps.

- **homepage-prototype**
  - **css**
    - homepage.css
  - **img**
  - index.html
  - **js**
    - homepage.js
    - defer.js
    - **vendor**
      - jquery-3.1.1.min.js

Get .zip bootstrapcreative.com/b4hp01



Windows - If this is your first time using Sublime Text you will also need to right click the files with extensions .html, .css, .js and hover over **OPEN WITH** *and click* **CHOOSE ANOTHER APP** *to then choose Sublime Text. This will make sure these files open in your text editor.*

Step 2

Open `index.html` in your text editor. Now copy and paste the page boilerplate code into `index.html` and save. This is a basic HTML page template without any reference to Bootstrap or outside scripts.

Copy Code bootstrapcreative.com/b4hp02

Step 3

Copy the minified Bootstrap CSS code and paste above the closing `</head>` tag.

Copy Code bootstrapcreative.com/b4hp03

Step 4

Next, copy the minified Bootstrap JavaScript code and paste it above the closing `</body>` tag.

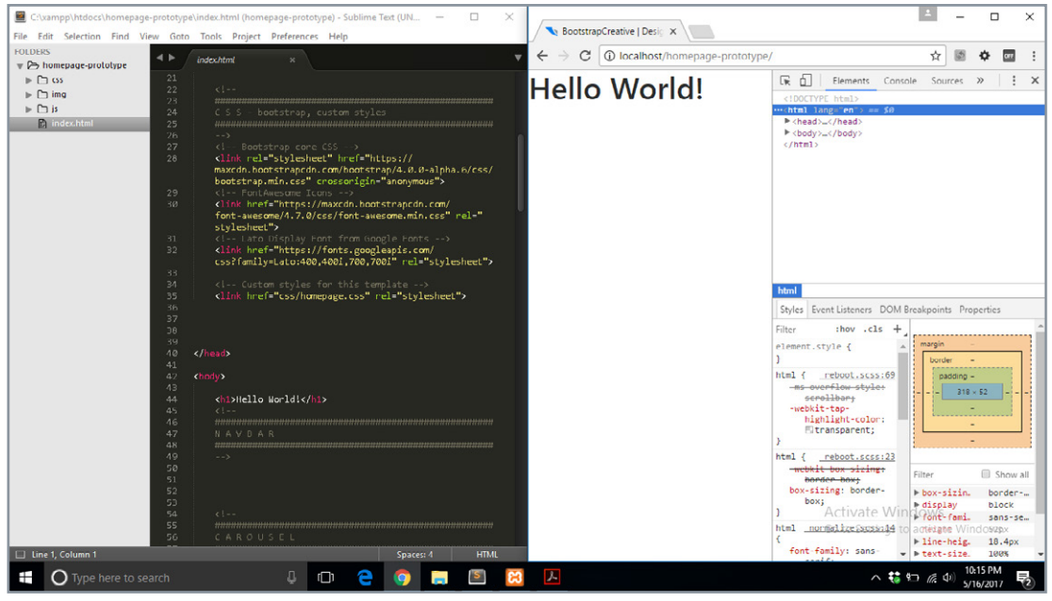
This code snippet contains Bootstrap scripts and its dependencies, which are jQuery and tether.js (which is used for tooltips).

What's a dependency? In other words, Bootstrap requires other code to work properly and so depends on additional libraries created by another developer. Dependencies have to be referenced/loaded before Bootstrap can run.

Copy Code bootstrapcreative.com/b4hp04

This snippet also includes a script tag with some JavaScript code inside. This code loads a JS file `defer.js` after the entire page is finished loading. This helps increase page load time by removing the non-critical code from the initial page load. Things like Disqus comment scripts, social widgets, and enhancements like animations.

Then open your `index.html` file inside your browser by dragging and dropping it onto a blank browser tab window. You are now able to navigate your project files, edit them, and see the results in your browser window.



Your screen should now look like this. I recommend using this workspace to force yourself to view your design mobile first.

Step 5

For our homepage layout, we want some content to be fluid width and some content to be constrained to the breakpoints. To achieve this we will be using the `.container` class on a per section basis instead of having all of our body content inside of it.

Example: `nav > .container`

Next, copy and paste the navbar code to add it below the opening `<body>` tag.

Copy Code bootstrapcreative.com/b4hp05

Step 6

Next, copy the carousel code and paste to replace the carousel comment section and replace this line of code `<h1>Hello World!</h1>`.

Copy Code bootstrapcreative.com/b4hp06

Step 7

Next, copy the image cards code and paste it to replace the image cards comment section.

Copy Code bootstrapcreative.com/b4hp07

Step 8

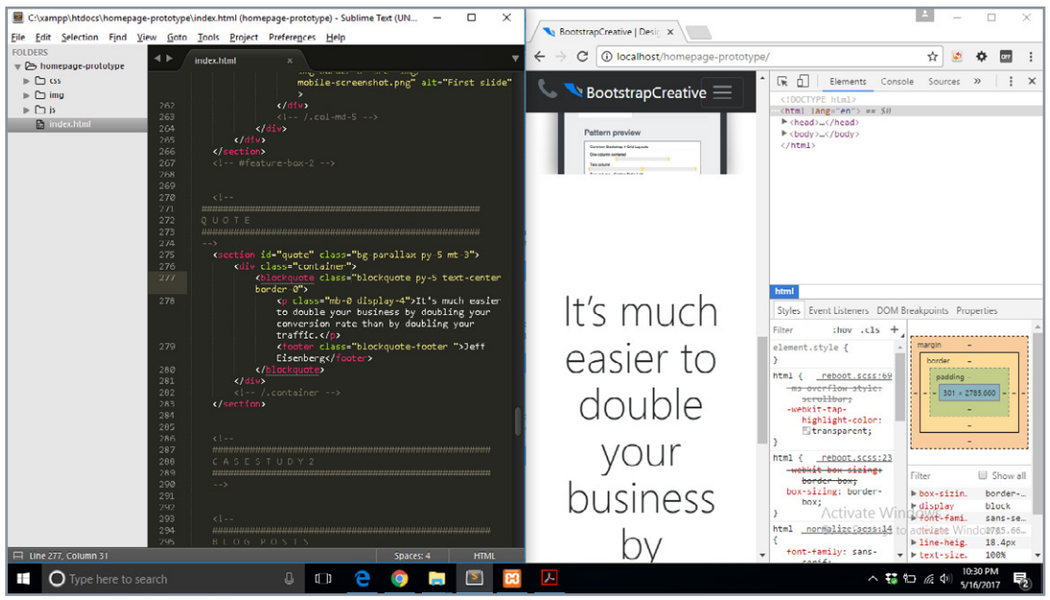
Next, copy the case study code and paste to replace the case study 1 comment section.

Copy Code bootstrapcreative.com/b4hp08

Step 9

Next, copy the quote code and paste to replace the quote comment section.

Copy Code bootstrapcreative.com/b4hp09



Your screen should now look like this.

Step 10

Next, copy the case study 2 code and paste to replace the case study 2 comment section.

Copy Code bootstrapcreative.com/b4hp10

Step 11

Next, copy the blog posts code and paste to replace the blog posts comment section.

Copy Code bootstrapcreative.com/b4hp11

Step 12

Next, copy the services code and paste to replace the services comment section.

Copy Code bootstrapcreative.com/b4hp12

Step 13

Next, copy the brand message code and paste to replace the brand message comment section.

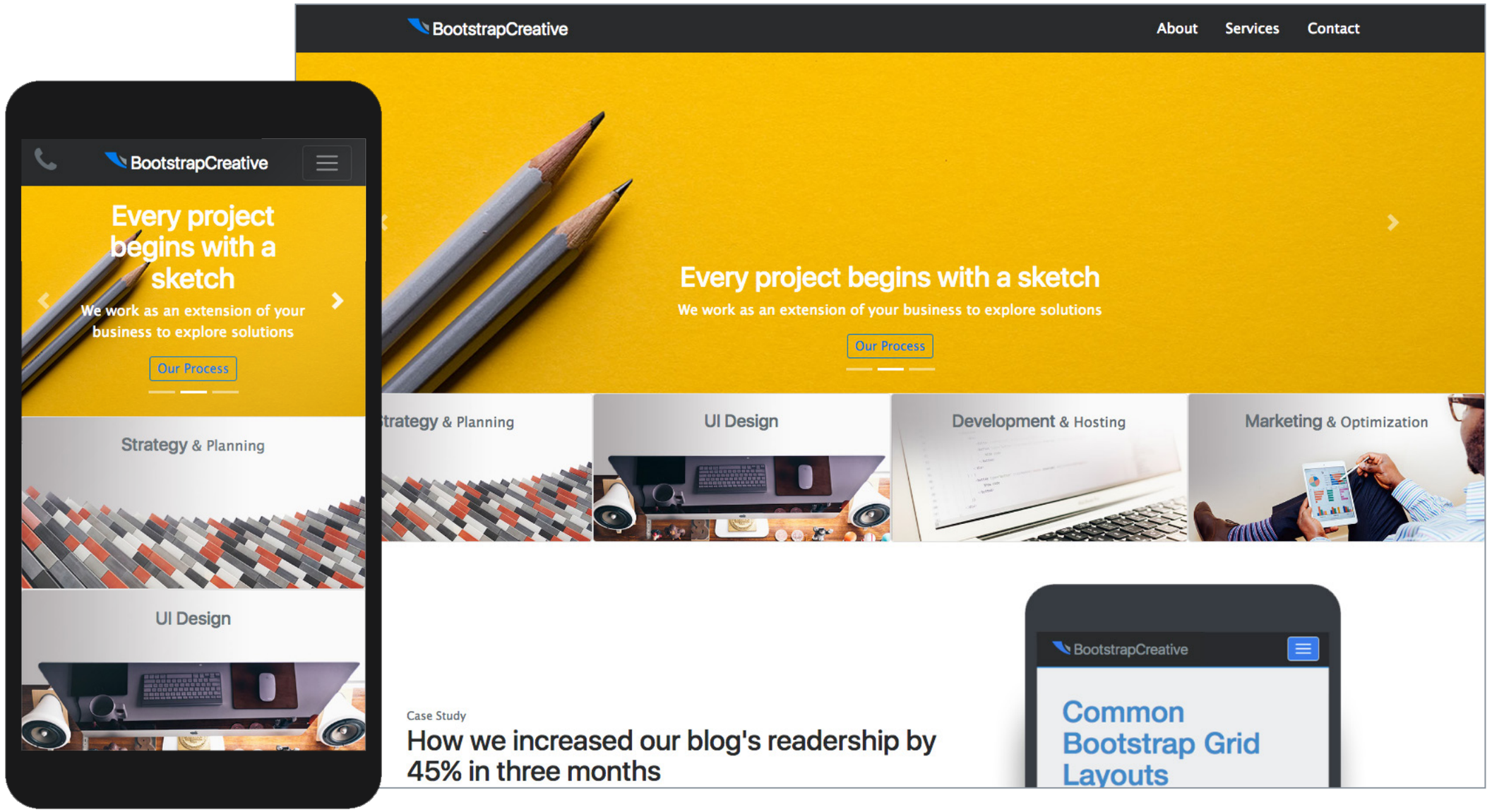
Copy Code bootstrapcreative.com/b4hp13

Step 14 *Final Step*

Next, copy the footer code and paste to replace the footer comment section.

Copy Code bootstrapcreative.com/b4hp14

SUMMARY



Great job. You now have a working homepage using Bootstrap 4. We used Bootstrap's default styles to get a working prototype to quickly demonstrate functionality. The prototype gets the basic idea down but it needs some refinements.

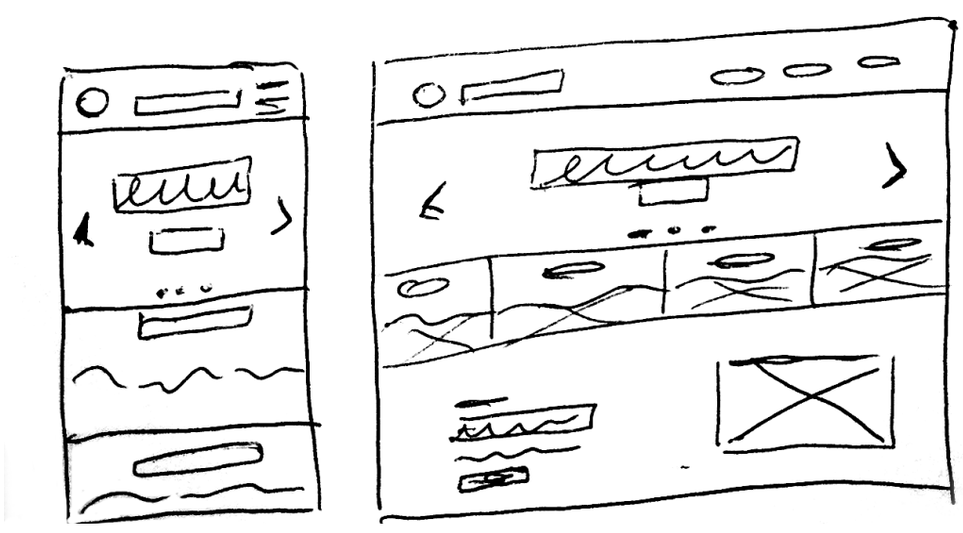
In the next section, we will write some custom CSS styles and add some scripts to make everything look the way we want. See you there.

Where to Start

If we refer back to the design process on page 30, we are now in the design phase. In this phase, we will explore design options and then begin to build a polished design that we can use for testing and approval.

Design Exploration - Fast Forward

Now that you have an approved wireframe, architecture, and page content you can begin exploring different visual solutions. Every designer has their own method but I recommend sketching with paper and pencil first.



It is easier to flush out multiple ideas on paper than it is to draw shapes in a design program. Also, if it doesn't work as a sketch it will never work as a final design.

Once you have a sketch you like you can then move to a design program like Photoshop, Adobe XD, or Sketch. These apps will help you work out aesthetics, colors, typography, spacing, and graphics before writing any code. Some designers skip this step and go right to code and design in the browser. I personally think this limits your creativity especially if you use Bootstrap. **The UI of Bootstrap should not influence your design**. Following this progression will greatly reduce the chances of your site looking too much like a Bootstrap site.

Since design is such a broad topic and would require a lot of explanation I am going to skip past this process and move right to development.

Clickable Design

Before responsive design, it was much easier to convert a desktop design and convert it to code because everything was fixed width and not much would change.

Today, we have to design for different breakpoints, devices, and bandwidths. Each with their own set of nuances and challenges. Because of this, I recommend never showing a client a flat Photoshop design. Ok, let's look at how we are going to improve the design.

DESIGN *WHAT WE WILL CHANGE*

A. Navbar Branding

Add blue border to the bottom and change link color

B. Carousel Captions

Move the captions to the top

C. Image Cards

Adjust image widths

D. Vertical Centered Content

Make the text vertically centered with the image

E. Parallax Background Image

Set a background image that is fixed on scroll

F. Add Right Arrow to Nav Links

Add arrow with CSS and not HTML

G. Text Overlay on Image Cards

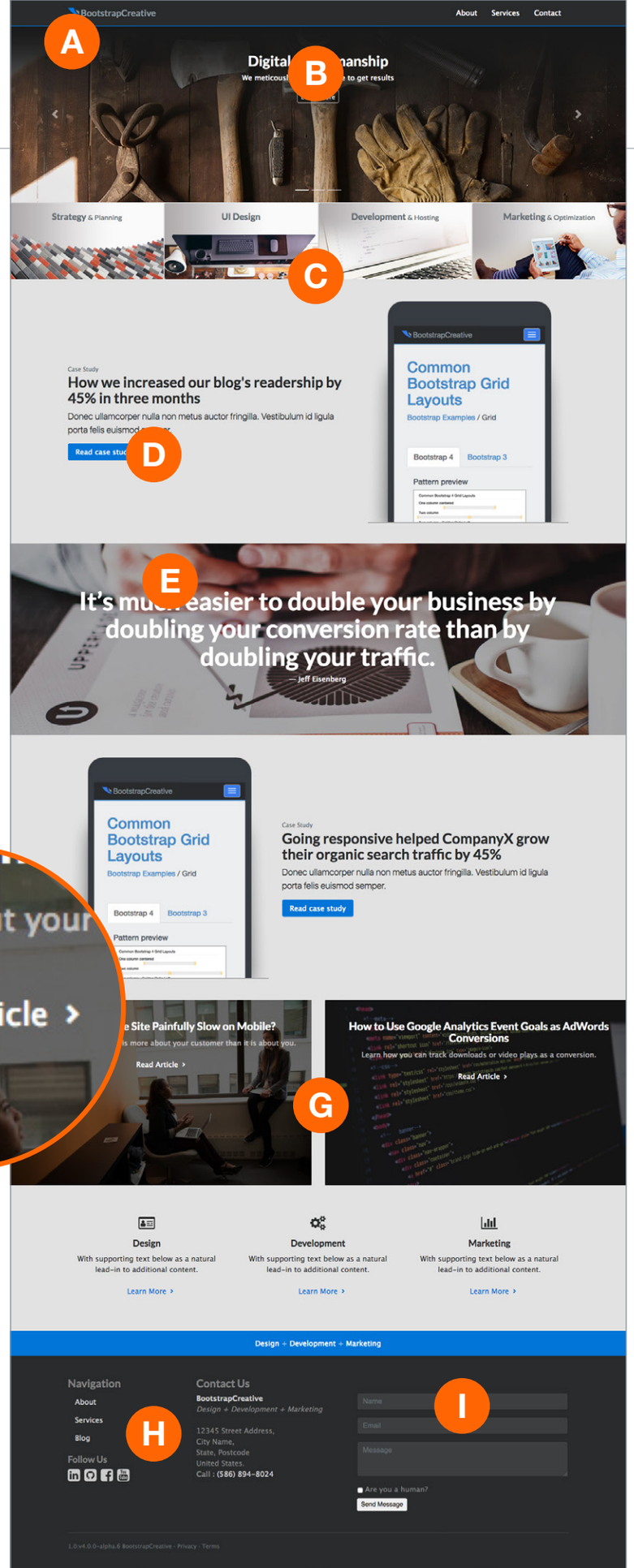
Add a black overlay on top of the image to improve text readability

H. Nested Columns

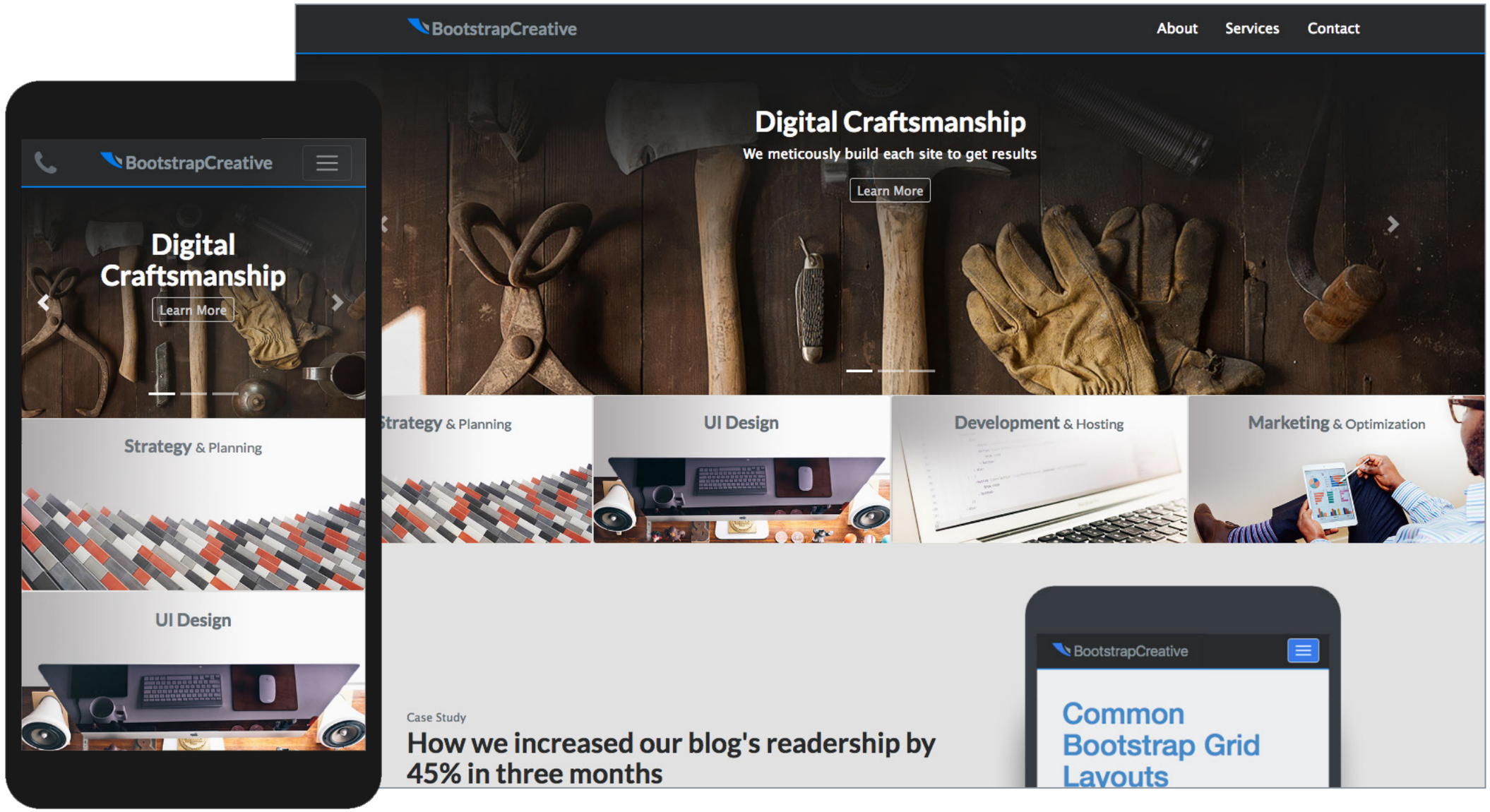
Add columns with breakpoints that make sense

I. Muted Form Inputs

The form inputs have too much contrast and are hard on the eyes



SUMMARY



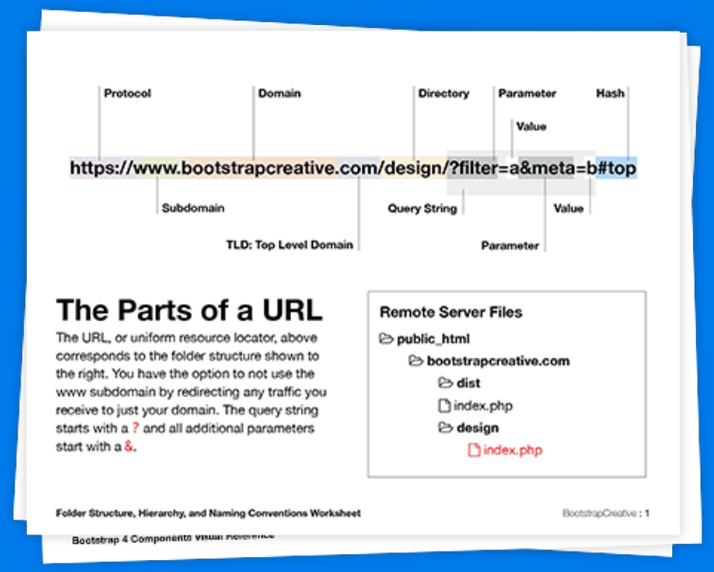
Great job. You now have a polished homepage using Bootstrap 4. We pushed our prototype of default Bootstrap components to match our desired design.

In the next section, we will follow the same workflow to build our CMS dashboard for this site.

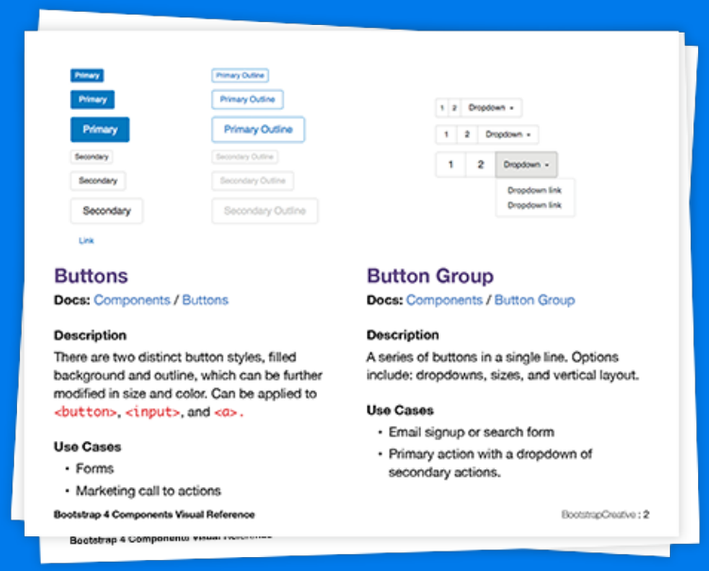
Videos & Quick Reference Guides

In addition to the book, there are videos and visual reference guides available to help you save time and be more efficient.

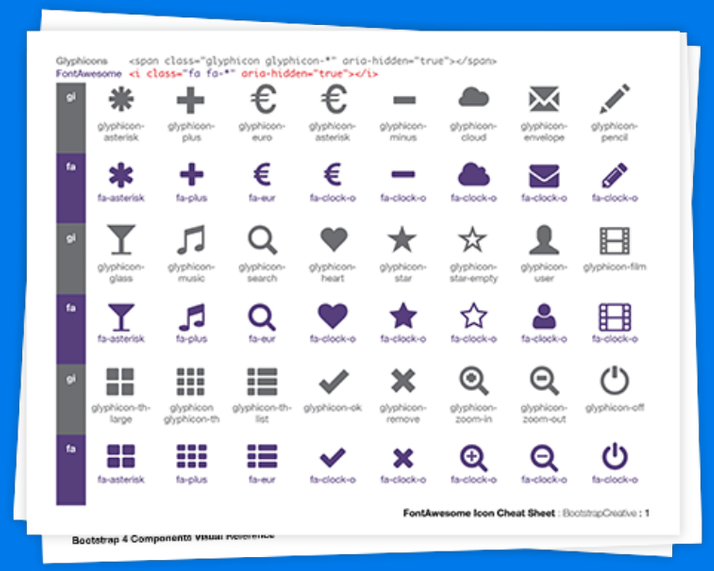
FontAwesome Icon Cheat Sheet



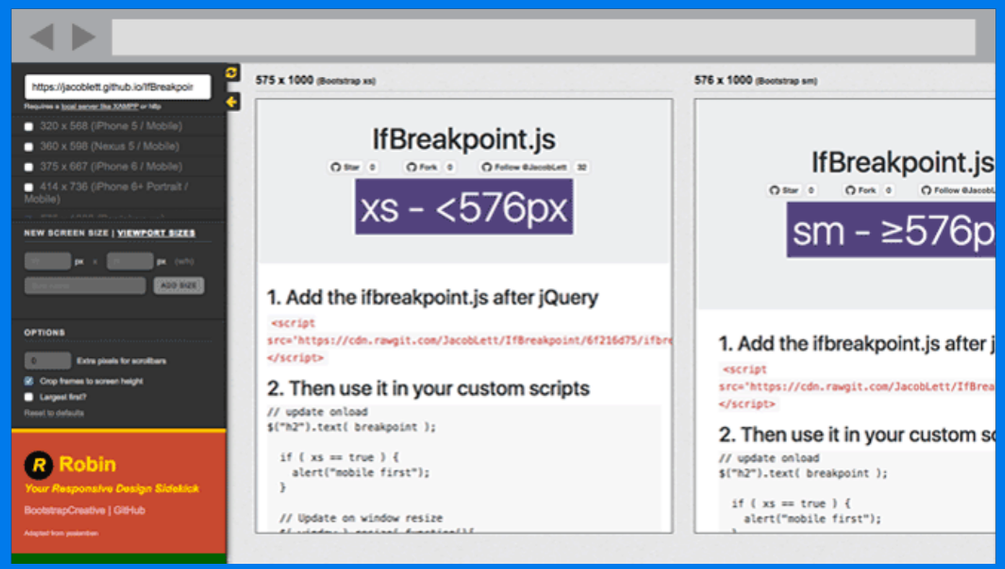
Web Design Primer



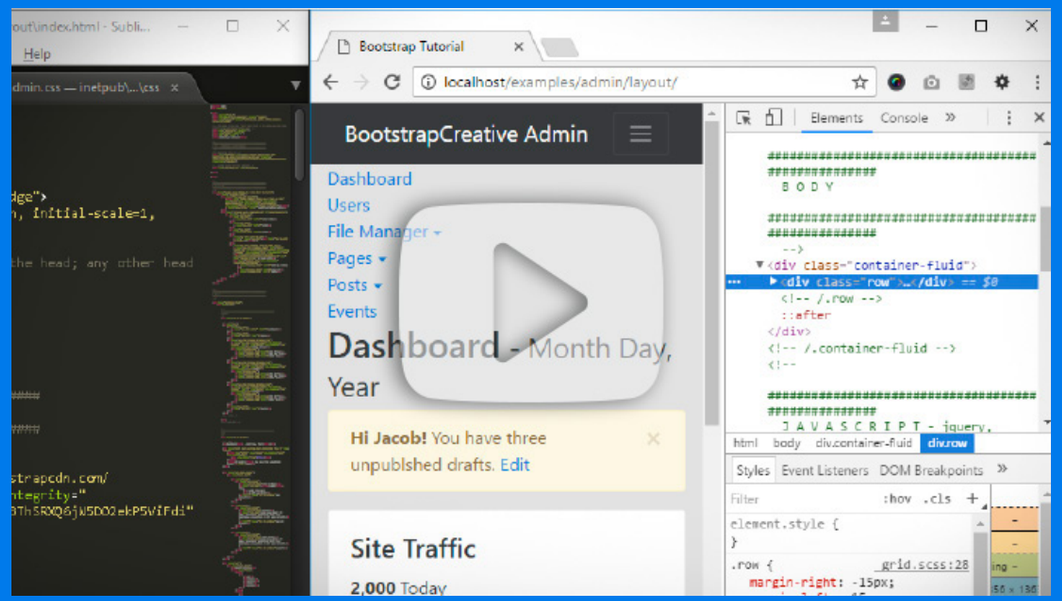
Bootstrap 4 Components Visual Reference



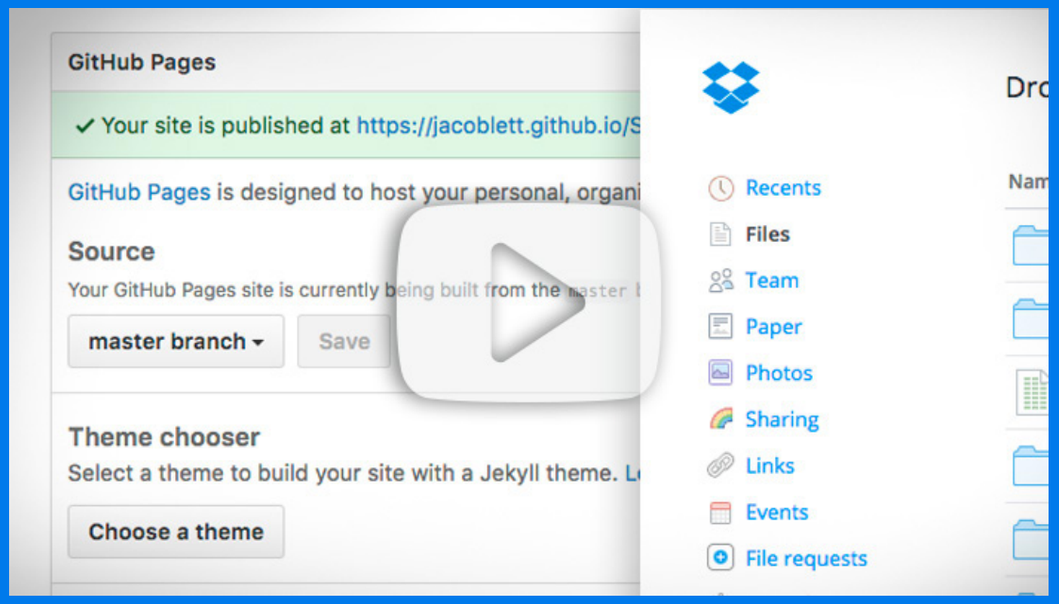
FontAwesome Icon Cheat Sheet



Robin - Responsive Breakpoint Preview Tool



Google DevTools - Inspect and Test CSS



How to Share Prototypes Fast

And More!

Buy now and start learning Bootstrap 4 and responsive web design basics.

Buy Now