# Agenda

1. Who am I and what am I going to talk about
   1. Intro about me
   2. Small intro about the talk
2. Where to start
   1. Importance of inspiration
   2. UI Parade and Dribble
   3. AWWARDS
   4. Behance
3. Pitfalls and solutions
   1. How to HTML5 Data Attributes Effectively
   2. Using Flexbox to solve daily issues
   3. SASS to rescue
   4. Gulp
4. Tools
   1. Color tools
   2. Font tools
   3. Icons
   4. CSS Animations

# Introduce myself

Good morning guys, hope everyone is having a good time at M3. My name is kishan wanigasinghe and I go by Doctor K, KWub.

This is my first M3 talk. So, apologies in advance

Also, I want to thank all the sponsors of this event, thank you for making this happen.

A little bit about myself,

I’m from Sri Lanka. I’ve graduated from Franklin University 3 years ago. After graduation I started working for Fusion Alliance as a .net dev, and, I’ve been working for them ever since. Now, couple of years later I’m still a .net dev, but mostly focused on front end stuff.

Started as a regular developer, few years later

I managed to be in-charge of couple of UI implementations.

So, What I learned from my experiences and where to start as a beginner for web UI, what to look out for and some tools.

# Part 1: Where to start

Let’s talk about where to start.

I think the first step for getting into building a better UI is looking at other designs and processes. Seeing how others do it and brainstorming ideas.

A while back, I used to think that I had the best designs, obviously. But after seeing other designs I knew it was definitely not the case.

So I started to look for examples and started collecting creative ideas from others.

As a personal experience, I can tell you that more creative things I got exposed to, the better my designs got. As an example, just take a look at these two things I designed,

There’s a big difference haa?

\*\*show my websites

So, here are some of my favorite websites for good design examples.

* UI Parade & Dribble
  + UI Parade and Dribble are two awesome websites. These two are a great place to get examples on good web component designs, color pallets and see a constant stream of new ideas. Most of the time I would take a look at couple of examples and then start to develop my own things or just take one I love and modify it.
* AWWWARDS
  + Next up, Awards is a great place to see people competing to win the best of the month and best of the year website awards. This website features good websites that are actually implemented and live. Every month a panel of judges will rate websites on various aspect and pick a winner.
* Behance
  + Lastly, Behance is a great website with a bunch of creative professionals. There is a lot of projects well documented from the sketches to the actual project. I love to go check projects here to see their design process and to see all the things they covered.

Now that we know a little bit about where to find examples and inspiration, next up I would like to talk about some common things that we should all look out for.

# Part 2: What to look out for

In this section I’m going to talk about using HTML attributes, CSS flexbox, SASS and a little bit about using gulp.

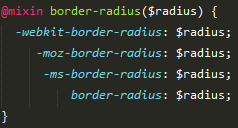
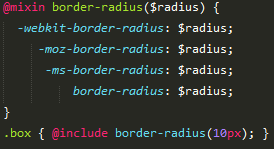
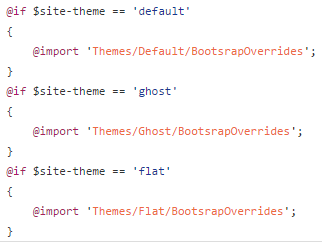
## HTML attributes

* + HTML attributes are a great functionality we have. Using them will save us some time and maybe we’ll manage to end up with clean css and javascript. Here I’m going to demo you couple of instances where we can effectively use html attributes.

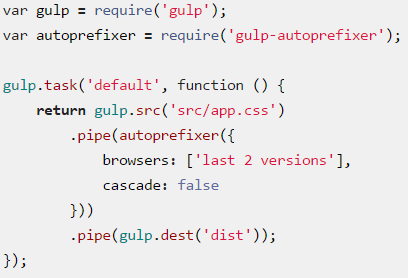
## CSS3 Flexbox

* + Flexbox is the new layout model for CSS.
  + Traditionally we used tables and later on we used the box model for our layouts
  + Each of these models had issues
  + Using tables caused unpredictable rendering issues unnecessary nesting and markup
  + Then box model was confusing and it didn’t do well few things like vertical centering, item positioning and stuff like that.
  + With flexbox we managed to overcome most of those issues. We get full control over the items and how they render.
  + Only drawback right now is the browser support, IE 11 is the only supported version. So if you don’t have to worry about that you might be in luck.
  + Personally, I think couple of the koolest things with flexbox are equal height columns item ordering. Before flexbox it was a very difficult to implement that.
  + Let’s get in to a demo to see few things we can do with flexbox.

## SASS

* + Next up we’re gonna talk about SASS. Syntactically Awesome Styling Sheets.
  + So, some of you might ask, why use SASS when I’m completely happy with the way I do my styling?
  + How about variables in my CSS, that’s what got me to use SASS.
  + Traditionally if you have a color or something that get repeated over and over in your css, when you have to change that, it’s a pain.
  + But, with SASS, it’s just a one liner if you use a variable everywhere.
  + There are more features like that so let’s get into another demo
  + Let's talk about some other benefits of SASS as well, the following are some of the highlights,
    - As I mentioned above, the reuse of code. This allows, easy maintenance and quicker updates.
    - Import other CSS files and create one CSS file at compile time. This helps to reduce the number of http calls to download css files.
    - Mixins, mixins are like smaller functions for styling. The following example show a smaller example of how a mixin can be used,
    - 
    - Now whenever you style a component you can include this and send in a value as a parameter to get the border radius attributes for your element.
    - 
    - In this example I’m including the mixin that we made earlier and sending in 10px as the value. This will apply a 10px border radius to my element with all four attributes mentioned.
  + Another great feature is the operators.How cool would it be if you could use if else statements in your css? Well, with SASS you can,
    - 
    - 
    - In this example, we have a if else statement that defines what SASS file needs to be imported depending on the value of site theme variable.
    - As you can see in the example, this will only import the ghost styling. So, none of the other styling will be in our output css. This helps to reduce excess that we dont need, when we dont need to have it.
    - You can apply this to do something else, I thought this was a great use to keep my code separated and clean and output only the needed css for my website.

## Gulp

* + <https://css-tricks.com/gulp-for-beginners/>
  + So what is gulp, gulp is a task runner. I’ve used a little bit of Gulp for couple of project and it has been a very useful tool to save time. You’ll have to install node js on your computer as Gulp requires that.
  + Let’s see some of the things that Gulp can do for you to save time,
    - Do you remember those browser prefixes that you have to add in certain cases? Well, gulp can do all that automatically for you. You just need to add that to the pipeline.
    - In this example we installed the autoprefixer plugin and added that to the pipeline like below,
    - 
    - Just like that you can have plugins to automatically compile your SASS files when you save them, minify your CSS and javascript and do all sorts of things that you want. Essentially you’re automating all these things that matter less.

# Part 3: Tools

So this is the last section of my talk. Im going to talk about few tools I use daily. These tools are small, but if im in a hurry, these are very useful.

## WhatFont

* + WhatFont, this is a tool that tells me what font styles are applied to a certain piece of text in a website. This is a handy tool when I want to figure out what kinda fonts and styles are applied to those fonts quickly. You just click at any piece of text and it there will be a pop up with all the info about that text.

## TintUI, Adobe Color Wheel, 0to255

* + These are some of the color pallette tools i use regularly. TintUI and 0to255 are two of my favorite website. TintUI have commonly used color palettes from different OSs and platforms.
  + 0to255 is very useful when I have to find a different shade of a color. This will list down different shades from lightest to the darkest for any given color. This was helpful every time when i needed to do hover styling for components. One thing to note is that if you use SASS, it has built in functionality to get this done very easily. You just need to define how dark or light you want the color to be and it’ll return you those values.

## Google Fonts

* + If you take a look at most modern website, they have some kinda custom web font built into them. How do i know? I use WhatFont hahaha
  + Google Fonts is a great source of CDN for web fonts. With this you dont have to host your own fonts, you can let google handle all that for you. These days, my favorites are Open Sans and Droid Sans. Seems to look really nice when I have them as my font family for websites.
  + To you google fonts, you just select the font and the variations you need, select the url and add it to your html, then you use that font family. Just like, 3 steps, you have custom fonts.

## FontAwesme, Fontello

* + So do you remember those old days when you need to have icons you had to add pictures to you page? Not any more. With CSS font icons you just need to add the class to any element that you need to have an icon.
  + Font icons gives you the flexibility of changing the colors and resize them without getting your icons pixelated.
  + Font Awesome and Fontello are two of my favorite font icon sites. FontAwesome has more icons and Fontello had multiple types of font icons.
  + All and all both very good source if you want have font icons in your site.

## Animate.css

* + The lastly I want to talk about animations. With CSS 3 you dont need to use JQuery to animate your elements. You can define animations in your css.
  + There are many premade CSS animation solutions. For me, I usually use Animate.css website as my go-to for most of my animation stuff. Very easy to use, you reference the css file and then add the animation class that you need. Also, you can detect when an animation is done with jquery.

# Wrap Up

So, that wraps up my talk about where to begin as a web ui developer, what to look out for and some of the tools to use. I hope you enjoyed my talk and got something out of it.

If you have any questions, now is the time.

thank you for stopping by!

And enjoy the rest of M3