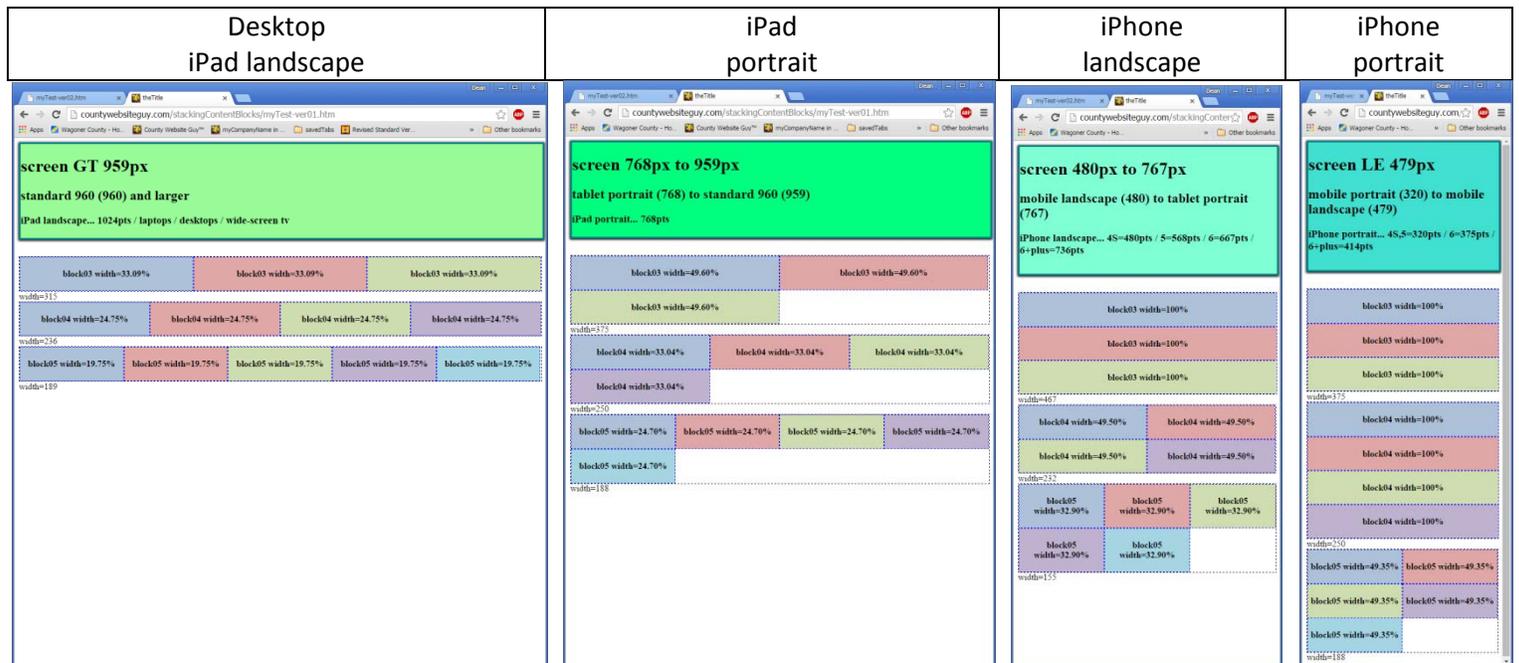


Website Maintenance Training

Overview

The purpose of this training is to familiarize you with the coding changes that were made to format some of your web pages with blocks of content that produce a mobile-friendly page. Currently, the only way to achieve a formatted page is to use table layouts. However, especially in iPhone portrait mode, there is not enough room for all of the content and as much as two-thirds of the content could be truncated. Additionally, the conventional wisdom is that tables are not to be used for layout.

The solution to this problem is described in the book *Responsive Web Design* by Ethan Marcotte, where he talks about how the content should re-arrange itself to fit the device viewport. Long story short, as a result of reading Ethan's book and my own accidental discovery of how a one-cell table behaved in the ok.gov CMS, I developed a set of classes that would adapt themselves, and thereby the content in them, to the device viewport. As the viewport is resized from 1024 pixel width desktop and iPad landscape to iPad portrait to iPhone landscape and finally iPhone portrait, the blocks will optimally re-arrange themselves to fit the content to the relevant viewport.



Lori, Amanda and I worked with Courtney Langston (product manager, Oklahoma Interactive) to implement the classes as a “user enhancement”, first into the OK.gov etsqa test system and then into the production system. While this allows us to make the Wagoner County website mobile-friendly by eliminating content truncation and achieving a formatted page without using tables for layout, Oklahoma Interactive does not support this type of implementation. They will provide help (at their customary hourly rate) if technical assistance is ever needed.

The mobile-friendly classes can be used by selecting View / Source code from the editor menu bar and writing the relevant <div> tags. Eg- to create a line with four (4) blocks, repeat the following div and class coding four (4) times...

```
<div class="fblock04">
...content...
</div>
```

There are classes for 1 to 10 blocks per line. Also, specialized helper classes for the Contact Us functionality, clearing floats, buttons, panels, hiding/showing blocks and blocks of irregular dimensions (eg- 25/75, 33/66, 25/50/25, etc). Standard classes in addition to the hide/show classes are used to make the menus at the bottom of the page and the “style” attribute is used for additional html element styling, as needed.

Up to now, I have been using this technique for parts of the website for which there has not been assigned for making changes a departmental person. In order to make the remainder of the website mobile-friendly, the consensus was that departmental website mobile-friendly training would be provided to everyone who had an OK.gov CMS id. This would then allow me to proceed with making mobile-friendly changes to the rest of the Wagoner County website. This would be that training.

Wagoner County departmental website mobile-friendly training

- Will keep you updated with a list of pages as I get the changes made.
- Show in editor source code mode the changes made to the departmental home page demo and other relevant pages.
- Then switch to editor design mode and demonstrate “View / Show blocks” menu item. This will make visible the various blocks that make up the page. Eg- H1, H2, P, DIV, etc. What you are interested in is where the DIV blocks are.
- How often do you make changes to the website?
 - frequently?, infrequently?, monthly?, occasionally?, yearly?, never?
- What kind of changes do you make to the website?
 - Change blocks of content?
 - Name/address / phone number changes?
 - Rework a few parts of a page?
 - Rework many parts of a page?
 - Create new pages?
- While there is always the possibility that you could accidentally break something when you make changes , observing the following guidelines will help to keep that from happening...
 - For blocks of content and name / address / phone numbers, when deleting text, be sure to STOP after you delete the desired text. If backspacing, do NOT keep backspacing after the desired text is gone or bad things might happen. If you are using the delete button, then do NOT keep deleting after desired text is gone, otherwise bad things might happen. eg- “I am text to be deleted and replaced”. Position cursor at end of sentence and hit backspace button MAKING SURE TO STOP after it is deleted the letter “I” at the beginning of the sentence. Then you can type your new text.
 - For editing a few parts of a page, as long as what you are moving around is WITHIN A DIV, probably should be ok.
 - For editing many parts of a page, will probably cross DIV boundaries, so be sure to check with Dean, Amanda or Lori.
 - For creating new pages you will want to check with Dean, Amanda or Lori.
- ...so please be careful what changes you make in design mode on pages which contain mobile-friendly code.
- And as always, if you are not comfortable with making the changes you need to make, please do not hesitate to contact Dean, Amanda or Lori.
- FYI... here are a couple of examples of “broken” pages...



County Street Map



OSU Extension Service

Notices and Upcoming Events

- Test line 1...
- **Test line 2...** - further text...
- **Test line 3...**
- test line 4...
- test line 5...

Notices and Upcoming Events

New To Wagoner County?

Find Helpful Information about the County

New To Wagoner County?



Find Helpful Information about the County

Economic Development



Commissioner Meetings



Search Land Record



ds

Bid List



Land Records Filing Fees

- Therefore, if you think that you might have bumped something, please contact Dean, Amanda or Lori for help fixing it.
- Also, there is the consideration that web page content must be Accessible as per Section 508 Refresh and WCAG 2.0. This subject was covered in detail during the past year. If you would like me to resend any of the information that was distributed, please do not hesitate to ask!
- Following two (2) pages contain four (4) examples of mobileFriendly pages built using design mode and then a quick technical refresher of the four types of web sites. Eg- static, liquid (or fluid), adaptive and responsive.

How to Make Mobile-Friendly Web Pages using the OK.gov CMS Editor Design Mode

OK.gov mobileFriendly examples

Here are the different ways that you can make the body of an OK.gov web page mobile-friendly. Note that if you use tables to format your content, as will be seen in refresher notes below regards static websites, your content might be truncated.

Just display the link and resize your browser from right to left, thereby shrinking the size of the viewport. This will let you test how your pages will look in desktop and iPad-landscape, iPad-portrait, iPhone-landscape and iPhone-portrait.

- [liquid - the page contains all text](#)
as a web page is resized, the text will flow. therefore, by definition, all-text pages are inherently mobile-friendly!
- [liquid - the page contains one \(1\) or more images and block\(s\) of text](#)
be sure to test at smaller viewport sizes to see how they look. note that the image(s) dynamically resize as the viewport gets smaller.
- [enable right-side bar](#)
this is done by entering content in relevant fields in the "Right Content" tab of the editor. The CMS will then handle the stacking of the right-side bar underneath the body of the page at the appropriate viewport size. It is the clients responsibility to manage the display of content in the rest of the body of the page.
- [adaptive - using the Stacking Content Block classes](#)
these classes provide the ability to control the formatting of content in a table-like manner, but the blocks of content will dynamically stack as the viewport is resized. these classes were developed in order to make the Wagoner County website mobile-friendly as it was desired to have the ability to control the format of content and the vendor did not provide any tools that would do this - so we wrote our own.

Quick Refresher on Different Types of Web Sites

Now a quick refresher on the differences between the various types of web sites. eg- static, liquid (or fluid), adaptive and responsive.

The quickest way to demonstrate these differences is to let you [display each of the \(4\) types in the liquidaptive website](#) and then resize your web browser from right to left, thereby shrinking the size of the viewport, observing the changes and noting how each type behaves differently. Following are some of the behaviours that you will see. [Note that the initial screen displayed is "Responsive" - use the drop-down list to pick the other three (3) types.]

Once you are comfortable changing the website type and resizing the browser from right to left to shrink the viewport, here is something else for you to try... [Matt Kersley's Responsive Web Design Testing Tool](#) Enter "www.liquidaptive.com" in the URL field and hit 'enter'. You can even change the type of website in the drop-down field. Nice, yes? - lets you see five (5) different viewport widths at once! (240 - 320 - 480 - 768 - 1024)

Here are some things to look for with each type...

- static - traditional way of building websites when formatting of the content is needed. Formatting is usually achieved using tables from CMS or with coding using the "display: block" property. Note how the content is truncated when the browser is resized from right to left.
- liquid (or fluid) - text will flow and images will resize to fit the viewport. Note that the content areas get smaller and smaller all squeezed together in the viewport. Also that it is not possible to format your content with fluid. not a

problem if you don't need formatting and the result after flowing to fit smaller viewports is ok with you.

- adaptive - note that as you move the right side of the browser to the left, the screen will "jump", then scroll smoothly for a while, then "jump" again, scroll smoothly, etc. Observe how after each "jump" the screen looks differently, even that the blocks of content start stacking on each other. Note that the **block-stacking only happens when the screen "jumps"**. Technical note - this is achieved by the use of media-query (@media) break-points that are coded in the CSS stylesheet.

Regards the OK.gov CMS, the header, menu bar and footer are adaptive as of the GEN4 upgrade (prior to this upgrade they were Static) and present nicely in all four (4) types of common devices - iPhone portrait, iPhone landscape, iPad portrait, iPad landscape and desktop. Note that the body of the web page is the responsibility of the client.

- responsive - main feature of responsive is that **text continuously flows, images continuously scale and content blocks dynamically stack** as the right browser margin is moved to the left. Technical note - the dynamic block-stacking is achieved by the use of media-query (@media) break-points that are coded in the CSS stylesheet.

