

Bachelor's thesis
Information Technology
2015

Bethelehem Adugna Burka

HTML-5

-Responsive Web Development



TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

Bethelehem Adugna Burka

HTML5 - RESPONSIVE WEB DEVELOPMENT

During the last 20 years, the smart phone technology has advanced to appear ubiquitously in the field. As a result, the user experience of individuals has shown a large shift from desktop to small devices including mobile phones and tablets. In Addition, Fragmentation and diversity in mobile screen sizes comes out as the main challenge.

The modern experience of websites for a mobile device mainly focuses on applying responsive web design. Modern web design is built to run on devices of various screen size and orientation. HTML, as a base stone of modern websites, has benefited the development process in better performance and data storage system.

In this thesis work, a prototype website was developed using HTML5 and CSS3. This real-time project, commissioned by a textile company, Dozen+1, was conducted to reflect the relationship between user experience and media devices. This thesis work focuses on HTML5 and responsive web design. The project includes a process to develop a website that contends modern web structures.

Furthermore, this thesis work favours HTML5 responsive web tools that are increasing the potential of unifying the different working platform of mobile operating systems which, in return, assists the cost and time consumption. The required tests were finally performed to approve the functionality in different media sizes. Hence, satisfactory results were demonstrated by adjusting a single web document in several media devices which increases the accessibility of a webpage regardless of device fragmentation.

KEYWORDS:

Responsive website, media queries, CSS3, HTML5, local storage, geolocation

CONTENTS

LIST OF ABBREVIATIONS (OR) SYMBOLS	5
1 INTRODUCTION	6
2 HTML5	8
2.1 New Features of HTML5	10
2.2 Graphics in HTML5	12
2.3 Storage in HTML5	14
2.4 Speed	15
2.3 Location	15
3 RESPONSIVE WEB DESIGN	16
3.1 Media Queries	17
3.1.1 Min-Width	17
3.1.2 Max-Width	17
3.2 Display Resolution and Orientaiton	18
4 USER EXPERIENCE	20
4.1 Utility and Usablity	20
4.2 Graphic Design	20
5 REAL-TIME PERFORMANCES	22
5.1 Objective	22
5.2 Method	22
5.3 Results	24
6 CONCLUSION	26
REFERENCES	27
APPENDICES	
APPEDNIX I. HTML5 TAGS	29
APPEDNIX II. Dozen+1 MEDIA QUEIRES	33

PICTURES

Picture 1. Responsive web design	16
Picture 2. Some examples of monochromatic colour	21
Picture 3. Draft of the project	22
Picture 4. Zooming preferences scales	24
Picture 5. Results applied with media queries	25
Picture 6. Implementation of HTML5 new semantic tags	25
Picture 7. The final result of the project	25

FIGURES

Figure 1. Time-line of HTML5	9
Figure 2. Sectional division using<div> and new structures of HTML5	10
Figure 3. Scalable Vector graphics in HTML5	12
Figure 4. Video and audio introduction	13
Figure 5. Mobile screen trends	18

TABLES

Table 1. HTML5 semantic elements <nav>	11
Table 2. HTML5 implementations	13
Table 3. Logical operators in media queries	17
Table 4. Example on min-width media queries	17
Table 5. Example on max-width media queries	17
Table 6: Examples of competing gaps in different sizes	18

LIST OF ABBREVIATIONS (OR) SYMBOLS

W3C	World Wide Web Consortium
API	Application Programming Interface
SGML	Standard Generalized Mark-up Language
DOM	Document Object Model
DTD	Document Type Definition
CERN	The European Organization for Nuclear Research
WHATWG	Web Hypertext Application Technology Working Group
HTML	Hyper Text Mark-up Language
URL	Uniform Resource Locator
HTTP	Hypertext Transfer Protocol
CSS3	Cascading Style Sheets Version 3
DOCTYPE	Document Type Declaration
DIV	Division
PX	Pixel
SVG	Scalable Vector Graphics
GPS	Global Positioning System
JPEG	Joint Photographic Expert Group
PNG	Portable Network Graphics
HEX	Hexadeci

1 INTRODUCTION

The evolution of internet introduced storing and sharing documents through the web for the first time. Correspondingly, the evolution of website development followed this footstep to launch CERN which is the pioneer website. In early mid-90s, when the internet was accessed only on desktops and laptops, web development used HTML and CSS with a small JavaScript to build static pages. HTML contains of the documents in hypertext; CSS provides the presentation style sheets and JavaScript is used for handling pop-up message boxes or image carousels for a web page.

The fundamental challenge with traditional websites is restoring the web contents and functionalities. While the end side of a website is completed, the back side or server side requires a structured database and content management system. The effort that goes into updating the contents and uploading new HTML documents is a major challenge among traditional web development.

The other vital flaw of traditional websites is the prolific growth of different screen sizes within smart devices. Although the traditional web techniques consider the cross-browsers and cross-platforms capabilities, they have a notable drawback in presenting the contents of a webpage in a diversity of screen sizes. Currently, the screen size of smart devices and desktops in the market has expanded approximately between 3.5 – 32 inches [1].

However in modern web technology, CSS introduces a solution to style a page using the 'media' attribute [2]. One of the most powerful schemes in stylesheets that is included in CSS3 is Media queries, which provide syntax to build device-independent websites. This thesis work focuses on the major solutions of time and design management on web content and layout. A single HTML5 document was used to produce a website that fits different screen sizes and orientations for a company Dozen+1.

Dozen+1

'Dozen+1' is a new and contemporary interior textile design group that presented designs for the first time at Heimtextil Trade Fair show in 2015.

The Dozen+1 name derives from the old time British saying 'Baker's Dozen', a traditional phrase that comes from making an extra piece for good luck. The group consists of 13 members from different parts of the world: Finland, France, Britain, Spain, China and Japan, which allows them to bring their own talents and individual style to the company.

Based at Turku University of Applied Sciences in Finland, the group has produced a team that is able to offer a variety of designs consisting of printed, woven, and mixed media. As students, the group see this as a unique opportunity that has allowed them to connect with established textile companies and utilise each student's skills to come together and create a professional team, the Dozen+1 Team[3].

2 HTML5

The concept of web documentation was developed in 1989, when Tim Berners Lee proposed the first hypertext system of web documentation. The background concept commenced in CERN, a practical physics laboratory, in response to a need for sharing documents among laboratory researchers. The first web was developed using SGML [4] that is highly dependable on the DOM [5]. SGML consisted of worldwide set standards of elements that can be implemented independent of browser or software diversity. Adjoining the elementary purpose in scientific collaboration, the concept of web was largely accepted when proposed to computing researchers of the time. Therefore, the first prototype was disseminated and implemented into practice in 1990. The proposal consisted of three fundamental specifications that are still prominent in today's web technology.

- HTML: serves as a medium language for editing and inserting contents to the web.
- URL: is a unique address assigned for each particular resource on the web.
- HTTP: applies as the ground application protocol to transfer data between browsers and main servers.

HTML5 is a mark-up language that has extended the functionalities of the web, improving the previous structure of web technology, which reflected a one-way flow of information to the users. Apart from introducing new semantic features [Appendix I], HTML5 contributes new techniques on modern and interactive web environment. HTML5 is a product of cooperative work in W3C and WHATWG which embraces more than one entity including CSS3, JavaScript APIs and HTML5 mark-ups.

In addition, HTML5 tries to balance the compatibility issue of the previous browsers using the new features. The latest version of web browser from Apple, Safari, Google Chrome, Mozilla Firefox, Opera full and Internet Explorer 9.0 support HTML5 functionalities.

Looking back through the history of mark-up language, it is essential to dig in the core motivation for its demand. In the early 90s there was an intense email discussion between early web developers, to identify the main importance and rules of mark-up language. This conversation [6] between Marc Andreessen, Tim Berners-Lee and other computer scientists is a ground to establish the basic elements we use in our web documents today. The discussion answers how elements like `` evolved to standardize a content type. Followed with these basic HTML tags, several improvement and version of HTML evolved.

After the initial proposal in 2008, particular highlights are located in the timeline of HTML5 [7]. An interoperation between web browsers, device production companies and self-oriented developers have contributed to finalize the complete specification of HTML5 which was released on 28th of October 2014 by W3C and WHATWG [8]. As shown in Figure 1, the overall development of HTML has involved media and related application manicuring groups including Apple, Firefox and many more.

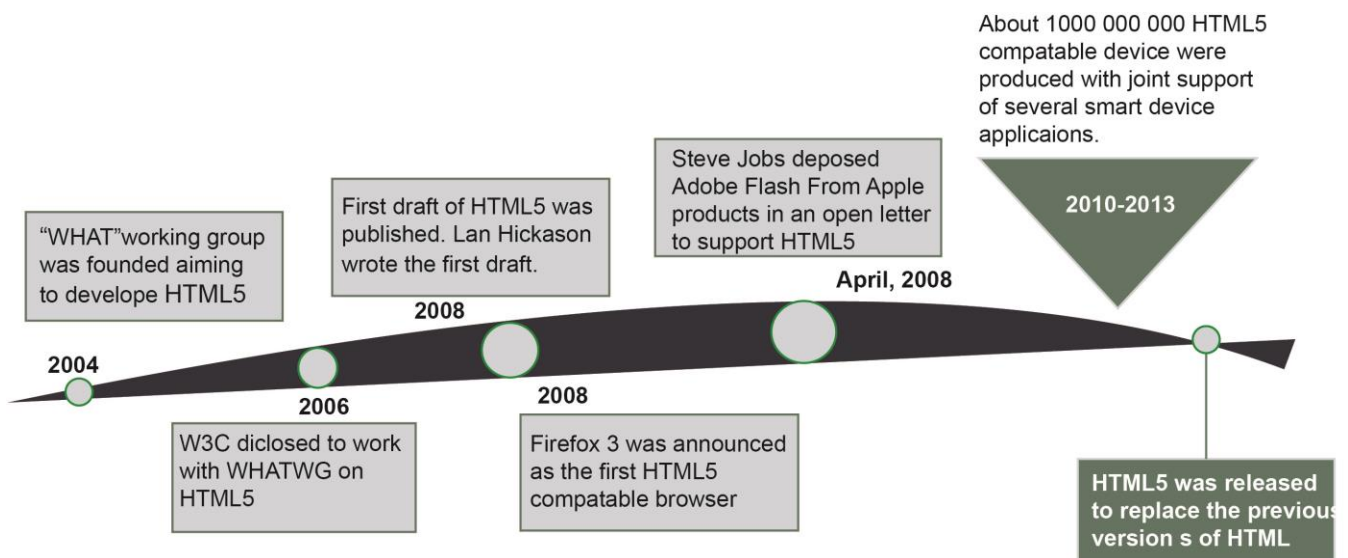


Figure 1. Time-line of HTML5

2.1 New Features of HTML5

The first new feature of HTML5 is plugged into the root element. The root element of HTML5 document is reduced from a bunch of libraries to a simple attribute of `<!DOCTYPE>`.

Similarly, semantic elements[9] like the `<header>` tag, as it is self-evident from the name, defines the structure of its content in relationship to the rest of the structure which at this point is at the top position of the page. Then new semantic tags `<header>`, `<nav>`, `<article>`, `<aside>` and `<footer>` outline and structure the HTML document in a neat division. Previously, `<div>` had the primary presentation syntax for HTML4, assigned as the generic structural element. The element `<div>` signifies for a block division of flow content in an HTML document. The functionality of `<div>` depends on the randomly assigned identifier or class name. As `<div>` is applied to define the different sections of the outlines for the browsers, using random identifiers has a huge risk of disorienting browsers and disorganizing the structure of the page. The reason for this is while assigning an ID for a `<div>` attribute, an author can associate to several synonyms of an identifier which could be misinterpreted by the browser.

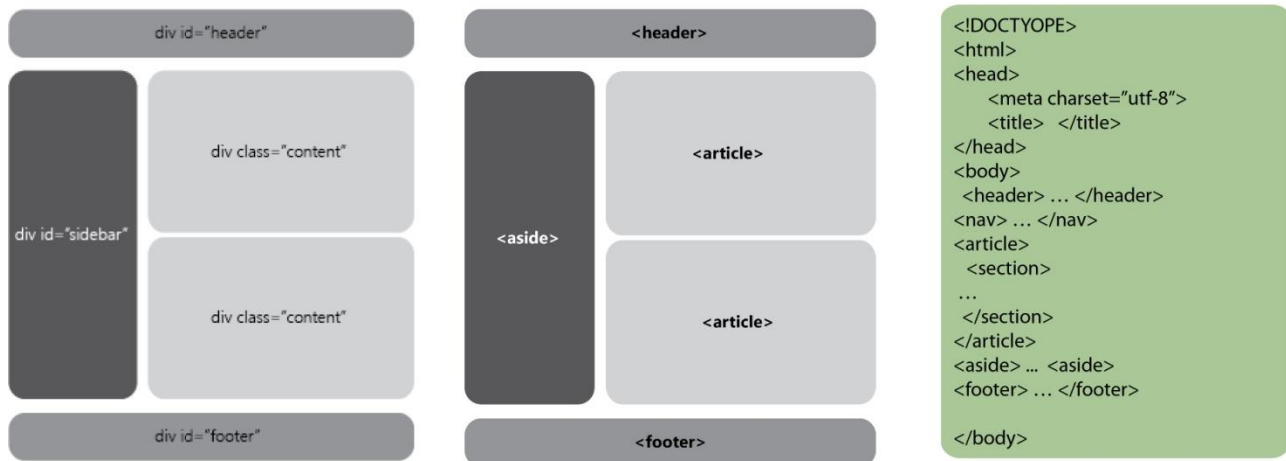


Figure 2. Sectional division using `<div>` and new structures of HTML5

A defined structural framework specifies the content partitions and aids graphical layout at the same time. In addition, with this new semantic standardized definition, it will be easy to detect basic landmark detection to different screen size devices for browsers. Table 1 is a source code

taken from the project to demonstrate the clear structures of the web pages, using the semantic element `<nav>` with the respective styling scripts.

Table 1. HTML5 semantic elements `<nav>`

Source mark-up

```
<!DOCTYPE html>
<head>
...
...
    <link rel="stylesheet" type="text/css" media="all"
href=css/styles.css>
</head>

<body>
...
<nav>

<ul>
<li><a href="designers.html">+DESIGNERS </a></li>
<li><a href="aboutus.html">+ABOUT US </a></li>
<li><a href="portfolio.html">+PORTFOLIO</a></li>
<li><a href="press.html">+PRESS </a></li>
<li><a href="contactus.html">+CONTACT US</a></li>
</ul>

</nav>

</body>
```

Style Sheet

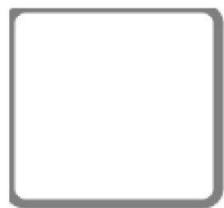
```
nav li a {
  font-family: 'Letter Gothic STD' sans-serif;
  font-size: 3em;
  font-weight:bold;
  color:#000;
  text-decoration:none;
  margin:0 20px;
}
nav li a:hover{
text-decoration:none;
border-bottom:2px solid #88AAA7;}2.2 Graphics in HTML5
```

The visual medium is a dominant content feature of the web. However, in the last ten years, the visual development was limited on CSS and JavaScript. Graphics tools introduced by HTML5 utilised both CSS3 and JavaScript which are solutions for huge range of web graphical conditions. Semantic features as on interactive designing serve beyond the limitation of browsers and device fragmentation. Interactive designs exclude the need for external plug-ins and better interaction with users' platforms.

Canvas and SVG

In HTML5 specification, the `<canvas>` element is one of the new major specifications. HTML5 introduces the `<canvas>` element to provide the whole experience of rendering graphs, game art graphics and other visual objects. The canvas element has neither its own content nor border. It serves as a tool for sketching specific paths, transformations and animated pictures. As Figure 3 explains, a graphical object is created in as part of HTML document.

On the other hand, Scalable Vector Graphics (SVG) are a set of vectors gathered to create a line object that is editable in the original scale. Vector graphics are good ways to balance the quality of images in a slow bandwidth environment. The quality of an image in different sized devices can be supported with the help of vector graphics. The core advantage of vector objects is their stable resolutions while resized.



```
<svg width ="200" height="200">  
  
  <rect  
    x="0" y="0"  
    width="100" height="100"  
    fill="white" stroke="grey"  
    stroke-width="5px"  
    rx="8" ry="8"  
    id="newRect" class="chart" />  
  
</svg>
```

Figure 3. Scalable Vector graphics in HTML5

Multimedia

Multimedia is another limited specification of the web development. In the last 20 years, embedding a video in a website needed strong bandwidth and additional heavy software. Embracing plug-in into the anatomy of a browser was one option to compromise the bandwidth flaws. Media plug-in players such as Adobe Flash and Real Player have been the multimedia deliverer of web documents since 2005[10]. However, after the introduction of HTML5, it became easier to display video and other animated media in websites with a simple tag. The HTML5 tag for video is as easy as <video>.



Figure 4. Video and audio introduction

In addition to the above main expansions, HTML5 came out with a different standard and attributes. API development [11] allows HTML5 to extend in Application technologies and put up a hope to future web frameworks.

Table 2. HTML5 implementations

	HTML5	
<i>JavaScript</i>	HTML	CSS
	HTML5	
<i>API</i>	mark-up	CSS3
<i>Canvas</i>	DOCTYPE	Transform
<i>Multimedia</i>	<canvas>	Colour
<i>Web storage</i>	<audio>	Animation
<i>File API</i>	<video>	Text effects
<i>Drag & Drop</i>	<section>	Web fonts
<i>App cache</i>	<header>	Backgrounds
<i>Geolocation</i>	<footer>	Gradients
	<article>	Borders

2.3 Storage in HTML5

Data storage includes the concept of storing single pages, page tracking keys, and credential keys on user preferences. Up until early 2000s, data storage in web App is executed on client linked server storages space [12]. In the process of such storage, browser cookies come to the image. The browser maximum space on storing cookies extends to 4 kilobytes per domain.

The broad challenge of web performance and web security is a result of the session data in cookies. According to research carried out by Yahoo! [13], 1000-byte cookie adds 16ms of response time on a single request made over a broadband DSL connection. For instance, let us assume that a web page consists of 1 external CSS file and 2 external JS files and 4 images. With a total of eight requests, a 1000 byte cookies will add 128ms to the response time of the request. This result is a significant performance lag on web services.

On the other hand, the security of a session data stored in cookies consumes a lot through encryption of the cookie value or authentication process of the server. To overcome the above challenges of using Cookie sessions, HTML5 introduces storing data on the client side device.

HTML5 local storage is a space within the client's web browser for storing named key/value pairs abstaining from resorting to plug-ins. This local storage saves the data and allows an application to work offline and sync back the data during an available network. The introduction of HTML5 local storage has solved the exiting challenge of storing user's data on a server, as it is possible now to allow Apps to save data on the client's device.

The major options for web storages include:

- Web Storage
- Web SQL Database
- Indexed DB

2.4 Speed

One major factor of web performance in online experience is the display time. Display time measures the time spent from a user's click until the page is fully loaded to the screen. In several situations, after a page has loaded to the users' screen, sites continue to perform background work in response to the user's initial action. In such cases, browsers continue to perform downloading a user's document which has a significant impact on the responsiveness of the page. Therefore, the speed of a website is determined by this elapsed time in combination with display time and CPU processing time.

Considering the risks of timing, the core solution relies on understanding a specific network in which a website is to be launched. In most web development cases, it is an initial step to study the network and calculate the resource and user access that is available in a particular medium. As far as a standard speed is concerned, there are few ways that will improve the performance of display time quality.

- Efficiently structure mark-up
- Minimize byte to download
- Optimize media usage
- Avoid quirks mode[14]

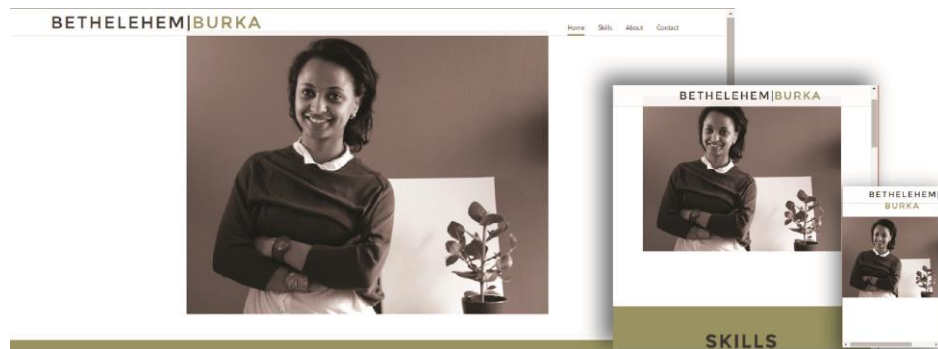
In addition, image and SVG usage ought to be chosen carefully for web development. JPEG formats are preferred to present pictures and photography visual objects. On the other hand, PNG formats are the best options in graphical arts as they are efficient to balance compatibility and CPU decoding.

2.5 Location

A location of a user is calculated with several methods including IP address, wireless network connection and signal exchanging towers or an implanted GPS hardware. Geolocation API [15] generates an updated communication between a users' location and a local server. It creates a link on exchanging a location (latitude and longitude) and gives back related information with location area.

3 RESPONSIVE WEB DESIGN

A basic introduction of responsive design [16] could be a mind-set design of a structural layout considering the further enhancement possibilities of a page. Adoption to various screen sizes is a major factor considered as an identity of a responsive design. Screen size identification of responsive web development involves either server-side or client-side detection of a device type. Designing a responsive page primarily starts from understanding the variety of users and devices of the implementation.



Picture 1. Responsive web design

A responsive web development involves three core elements. A web platform is considered to fulfill the requirements of a responsive design if it contains the following core elements.

- Flexible and Fluid grids
- Dynamic contents
- CSS3 Media queries

Above all the steps of the designing process, the foundation of the design is required to meet flexible and fluid layouts that include grids. Grids hugely assist in the process of breaking and redisplaying proper portions of a website. Secondly, a responsive design is planned to show a dynamic reflection in its content. For instance, an image as content needs to be flexible with the specific design requirement.

3.1 Media queries

Media queries are a section in a Cascade Style Sheet that specifies the media match type of a device or orientation. Media queries were introduced in CSS3 in 2011. As a result, they check the condition of a query on a return value of true or false. The condition specifies the type of orientation or screen size. It is possible to add logical operators like ‘and’, ‘not’, and ‘only’ to combine, negate, and restrict respectively. Below are some examples of using these logical operations.

Table 3: Logical operators in media queries

```
@media (min-width:700px) and (orientation:landscape) {...}
```

```
@media (not all) and (monochrome) {...}
```

```
<link rel="stylesheet" media="only screen and (color)" href="style.css">
```

3.1.1 min-width

After the introduction of media queries, the “device width” that is specified is described with minimum and maximum width measurement limits.

Table 4: Example on min-width media queries

```
@media only screen and (min-width:320px) {...}
```

If [device width] is greater than or equal to ‘i’, then the CSS style sheet will be applied to the screen (monitor or device). For instance, a screen size of 310px will fall in the regular style sheet instead since it has a value less than 320px.

3.1.2 max-width

Table 5: Example on max-width media queries

```
@media only screen and (max-width:320px) {...}
```

The above section is translated in the condition of screen (monitor and device) with a screen size of 320 or less is required to inherit the style sheet. As a result, all other wider-than-this screen sizes are dropped. To fill the gap of the screen that can be left out, it is possible to integrate two media scripts.

Table 6: Examples of completing gaps in different sizes

```
@media only screen and (min-width:480px) and(max-width:640px) {...}
```

```
@media only screen and (max-width:479px) {...}
```

These queries define the design translation rules in HTML5 and CSS3 based on the devices' screen width.

3.2 Display Resolution and orientation

The variation of mobile devices is directly affecting the display resolution of digital devices. The display resolution of a digital device is the number of distinct pixels in each dimension that can be displayed. Trends in screen sizes [17] show a noticeable gap in a rapid speed. In general, the screen sizes of most devices that are currently in the market are adopting portrait, landscape or even a complete square. Many new devices have 100 different screen sizes in between ranges.

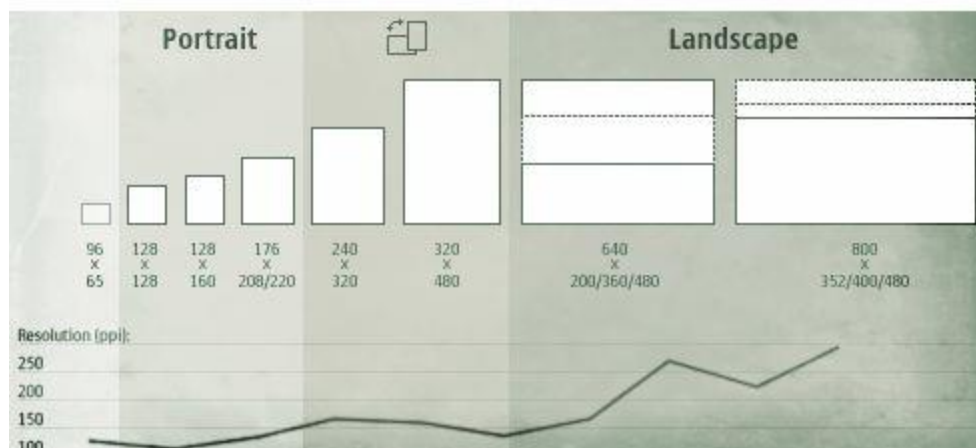


Figure 5. Mobile screen trends

As the usage figure might vary in any instance, assuming the screen size possibilities with the future devices is one solution. Grouping a certain range of similar screen sizes can be another solution. Although it is a possible measure, involving over a hundred of screen size and resolution in a test can be overwhelming. On the other hand, the orientation of a device is another definition of display property, which classifies the layout of the web page in horizontal and vertical orientation of smart devices.

4 USER EXPERIENCE

The experience of a visitor on a web page deploys an impact on the evaluation of a web development process. User experience is an emotional impact that is ingrained in the web design with the intention of achieving a maximum user satisfaction.

4.1 Utility and usability

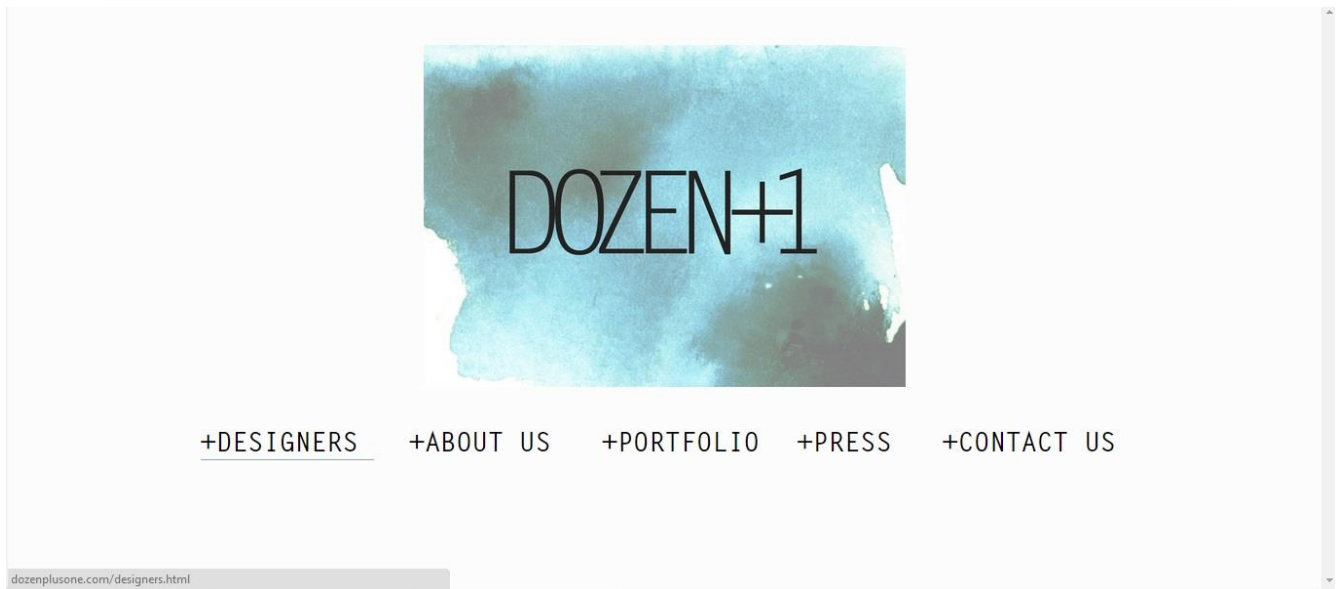
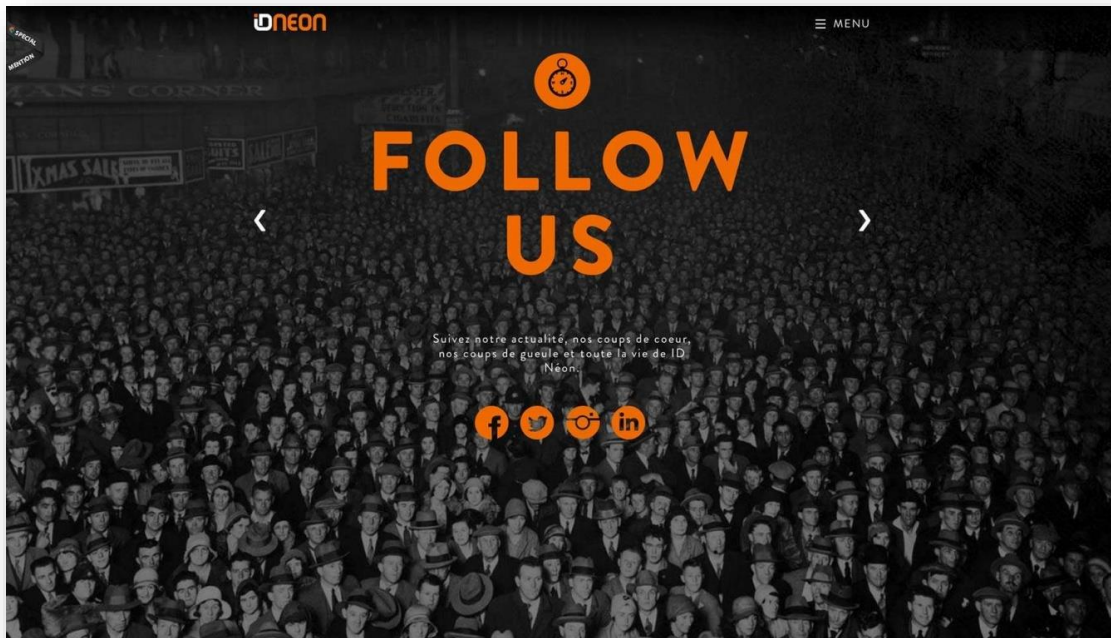
Utility is the fundamental element for measuring the functional quality of a website. The major question every company wishes to answer is 'how interesting or useful a website is to its user?'. Pre-knowledge on the interest of the targeted audience is a starting point to meet the goal in the utility and usability.

Usability [18] on the other hand is how difficult or easy a website is to use by the targeted users. It is about providing an equal amount of information regardless of the device fragmentation or designed layout. Decision on hiding contents and functionality is affected by the knowledge on the targeted audience. Other technical rules include the three click rule of designing apply in this section of the designing phase. The three-click rule suggests that a user should be able to find the information they are seeking with no more than three clicks.

4.2 Graphic design

Graphic design in responsive website is initially employed with three elements that determine the identity of the site. Graphic design is required to embrace content, performance and usability in the process of visual development. Respectively, the very first step of any graphic design is to absorb a draft template that reflects the prioritized elements of a site. Background outlines or clear sketches placed from the beginning are main structures that can craft the final page.

Respectively, the visual components of a website consist of the appropriate colour and style which can engage with the user's emotional experience. Well-balanced contrast of colours and consistent use of fonts and presentation deploys a graceful brand that will magnify positive impression. Monochromatic colour set is an optical concept material that shows the sensitive power of one colour. Some examples of this technique are shown in Picture 2 [19].



Picture 2. Some examples of monochromatic colour

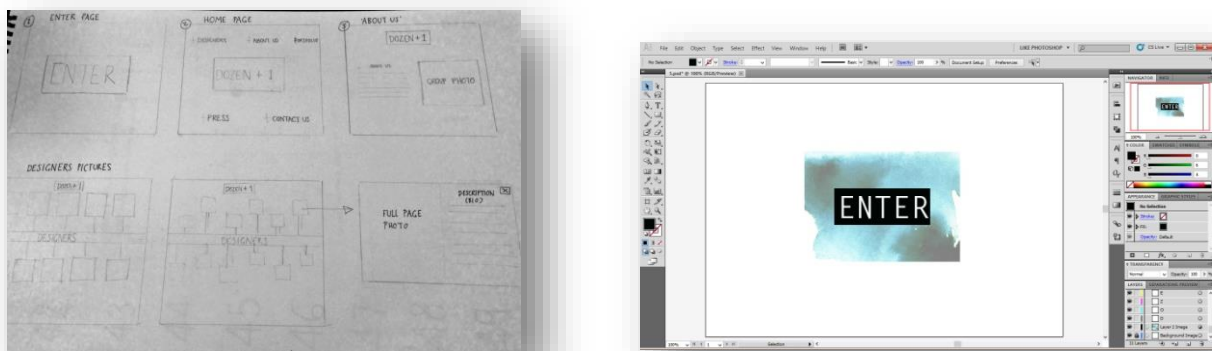
5 REAL-TIME PERFORMANCES

5.1 Objective

One of the main objectives in the path to develop this project was to be able understand the designing techniques of modern web development. Responsive design solves the previous issues of maintenance and consistency in traditional web designing. This project attempted to balance the structure, design, and application of these modern web approaches.

5.2 Method

The first step of any web development is sketching the draft layout. Placing on a paper what the end product looks, gives an opportunity to maximize efficient designs. The leading identity of this project in terms of designing was 'white and simple'. As shown in Picture 3, clean elements of the pages were set on paper. The website was intended to follow a monochromatic colour set. As a result, the next step was to choose one colour to define the consistency of the pages. Light blue with the hex colour of #88AAA7 was the root colour used in the website. At this stage, it was easy to identify the contents that could be included in the diverse device layouts.



Picture 3. Draft of the project

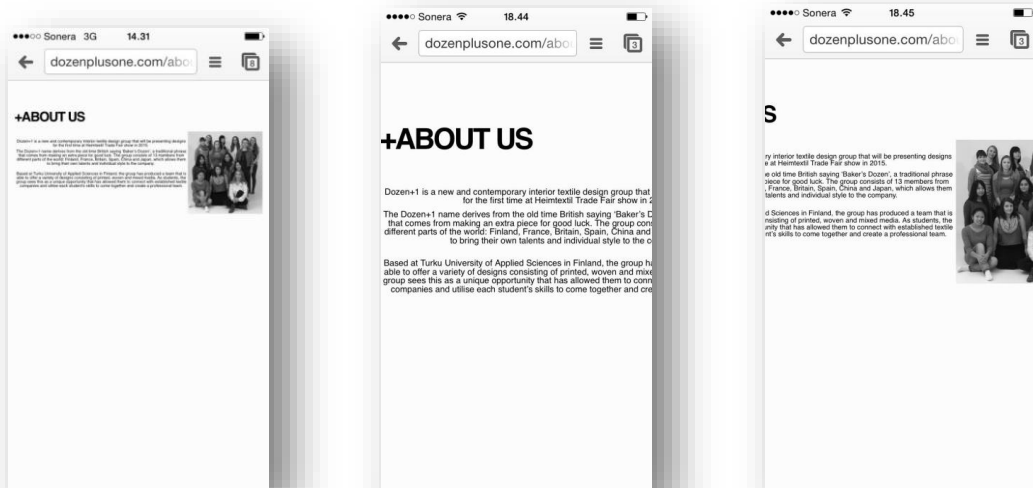
The first draft of the website was developed using traditional layouts techniques. In the process of transforming the page into a modern website, it was needful to find out the user ranges, orientations and pioneer devices. A static non-responsive web page requires zooming

to access the pages in smaller devices. The guide in the flexible pictures and contents were to minimize zooming preferences as the page fits to the device view point.

Next, media queries were applied in the style sheet to apply responsiveness. The media that were covered with the CSS styles were screen sizes 780px, 480px and 320px. Scalable units including 'em' and percentage were used for font sizes, padding and margin measurements.

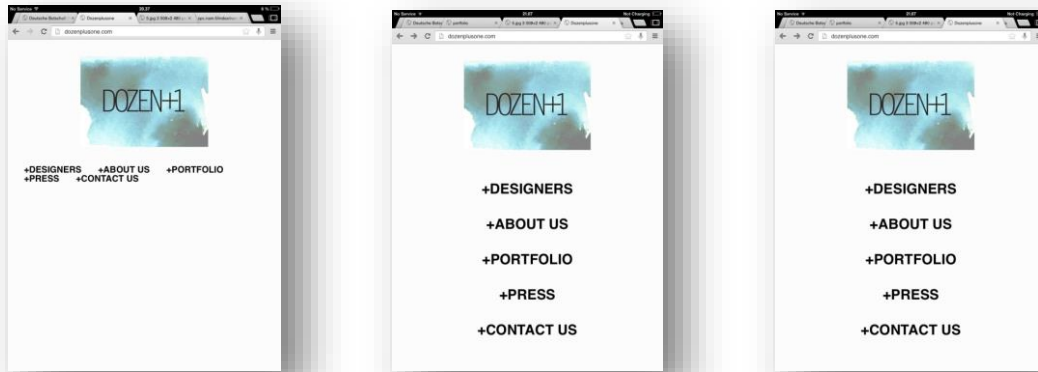
5.3 Results

The structure of this thesis work is evaluated with the significant results of the test that was performed on the targeted devices. According to the first result, the view of the screen had to be adjustable to zoom in and out with scalable proportion as shown on picture (Picture 4). The viewport meta tag was used to control the scale and width of the website on devices. The initial scale provides the scale property when the page first loads as the maximum scale, which supports the users to be able to zoom.



Picture 4. Zooming preferences scales

The meta data in the HTML document, “<meta name “viewport” content= width =device-width, initial-scale=1, maximum scale =1>”, is used in the project to allow the preferences of zooming. On the other hand, the orientation of the landing page of the website was designed to fit both mobile and desktop layouts. Rules in media queries specify how to display the same HTML document in different orientation.



Picture 5. Results applied with media queries

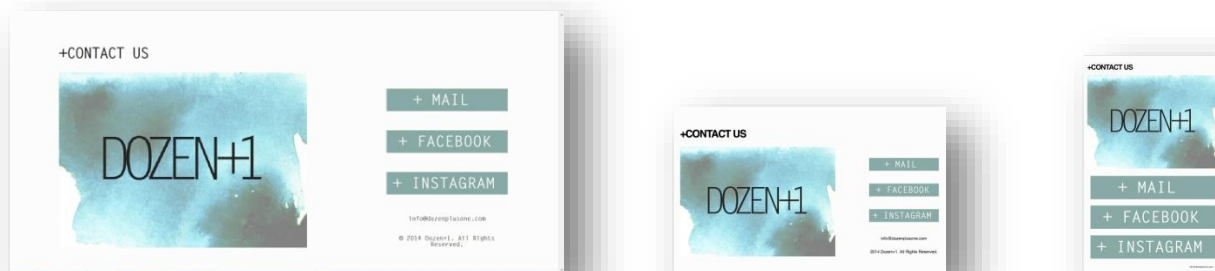
New semantic features like `<ruby>` [Appendix I], a tag introduced in HTML5 as to represent Ruby annotation, was applied to be part of HTML documents instead of using external images.

```
<h2>+<ruby>にしまら あずさ</ruby></h2> <p>(Azusa) </p>
```



Picture 6. Implementation of HTML5 new semantic tags

As a final result, Picture 7 shows the responsive web page of Dozen+1.



Pictue 7. The final result of the project

6 CONCLUSIONS

Web development is a field that is moving beyond adopting smart devices and orientation. The new frameworks and web based application which are based on HTML5 and similar technologies drive the discovery of new techniques and approaches. New frameworks like Bootstrap and AngularJs have proved the potential HTML5 and CSS3 have in further transformation of websites to framework-based templates.

The key advantages of responsive web design are:

- Accessibility from desktop to mobile
- improved Search Engine Optimization in web address and web page
- Lower overall cost than developing separate websites and native application

On the contrary, the challenges that hinder responsive web applications are the time it requires to structure the contents and test. In addition, responsive web development require more time and resources on testing and inspecting elements of several different versions of the layouts. As a result, testing using real devices was one major challenge in this project. The second limitation of this project was on using media queries in providing support for IE8 and older browsers.

In conclusion, during this project, it was easy to note down the strong power of simple and neat design with the blend of the user expectation and experience. Understanding the audience early in the phase and choosing customer based approaches results in applicable web pages. In general, the use of responsive web design proves in handy in today's market, which is flooded with a wide variety of devices with various sizes and resolutions.

References

- [1] Hjerde, M. (2008, April 28). More on mobile screen size trends [Online]. Available: <http://sender11.typepad.com> The Book of CSS3: A Developer's
- [2] Gasston, P. "Media Queries" in the Future of Guide to Web Design, 1st ed. San Francisco, CA: No Starch Press, 2011, ch 2, pp. 7-20.
- [3] Dozen+1. (2014). about us [online]. Available: <http://dozenplusone.com>
- [4] Goossens, M. and Saarela, J. (1995). Practical Introduction to SGML (2nd ed) [online]. Available: <http://mactex-wiki2.tug.org/TUGboat/Articles/tb16-2/tb47goo1.pdf>
- [5] Pilgrim, M. "Detecting HTML5 Features" in HTML5: Up and Running, Loukides, M. Ed., 1st ed. Sebastopol, CA: O'Reilly Media, 2010, pp. 15.
- [6] Pilgrim, M. "A long digression into how standards are made" in HTML5: Up and Running, Loukides, M. Ed., 1st ed. Sebastopol, CA: O'Reilly Media, 2010, pp. 2-7.
- [7] wix. (2012, July 11). The Authentic Infographic History of HTML5 [Online]. Available: <http://www.wix.com>
- [8] Lardinois, F. (2014, Oct 28). W3C Declares HTML5 Standard Complete [Online]. Available: <http://techcrunch.com/2014/10/28/w3c-declares-html5-standard-done/>
- [9] MacDonald, M. "Introducing the Semantic Elements" in HTML5 the missing manual, 2nd ed. Sebastopol, CA: O'Reilly Media, 2014, ch. 2, sec.1, pp. 38-39.
- [10] Powers, S. "HTML5 Audio and Video Elements: By-Default" in HTML5 Media, St. Laurent, S. Ed., Sebastopol, CA: O'Reilly Media, 2011, ch. 1, pp.1-7.
- [11] Gustafson, J. M. "The components of an HTML5 application" in HTML5 web Application Development By Example, Birmingham, UK. Packt Publishing, 2013, ch. 1, pp. 9.
- [12] Pilgrim, M. Dive Into HTML5, The Past, Present & Future of Local Storage For Web Applications. [Online]. Available: <http://diveintohtml5.info/storage.html>

- [13] YUI Team. (2010). Performance research, part 3: When the cookie crumbles. Yahoo! User Interface Blog (March 2007) [Online]. Available: <http://yuiblog.com/blog/2007/03/01/performance-research-part-3/>
- [14] Castro, E. "The DOCTYPE and Standards vs Quirks mode" in HTML for the World Wide Web, 5th ed. Berkeley, CA. Peachpit Press, 2002, ch. 1, pp. 39.
- [15] Holdener, A. T. "Geolocation API in Code" in HTML5 Geolocation, 1st ed, Sebastopol, CA. O'Reilly Media, 2011, ch. 3, pp. 31-35.
- [16] Frain, B. "HTML5 for Responsive Designs" in Responsive Web Design with HTML5 and CSS3, Birmingham, UK. Packt Publishing, 2012, ch. 4, pp. 97.
- [17] Hjerde, M. (2008, April 28). More on mobile screen size trends [Online]. Available: <http://sender11.typepad.com>
- [18] Hartson, R. and Pyla, P. "From Usability to User Experience" in The UX Book: Process and Guidelines for Ensuring a Quality User Experience, 1st ed. San Francisco, CA. Morgan Kaufmann, 2012, ch. 1, sec. 3, pp. 7-20.
- [19] (2015). iDneon [Online]. Available: <http://idneon.ch>

Appendix I	HTML5 Tags
<select>	Defines a control for a menu of options. <select> is used with conjunction with the tag <option>
<nav>	Defines a section that contains a navigation elements and links.
<article>	Represents an article content that can stand independently by its
<aside>	Represents content that which is related with the main content but could also standout by its own. A “pull quote” from a longer article is presented with a large typeface on the same page
<header>	Defines the header of a page or selection. It often contains a logo, the title, and navigation tab lists.
<footer>	Defines the bottom part of a page containing the copyright and contains contact and additional information.
<main>	Represents the main content area in the HTML document. <main> tag exits only once in a document excluding tags like <header><footer><nav> and so on.
<figure>	Represents a figure in the document. This is different from the tag providing illustration options, diagrams and

<figcaption>

This tag provides a caption under a figure and used with <figure> element.

<data>

Associates to a particular element that allows storing extra information on standard, semantic HTML elements.

<time>

Declares the date and time within an HTML document.

<mark>

Indicates a text that is highlighted for reference purpose.

<ruby>

Represents content to be marked by Ruby/Rubi* annotation

**Ruby: (also spelt rubi) characters are small, annotative glosses that can be placed above or to the right of character when writing logographic languages such as Chinese or Japanese to show the pronunciation.*

<rt>

Marks the ruby text component of a ruby annotation.

<rp>

Marks a parenthesis around a ruby annotation.

<bdi>

Marks a span of text that should be isolated from its surrounding in order to allow embedding a base for the unknown direction. <bidi> is quite useful in case of user generated content. It would be convenient to identify if its read from right-to-left or left-to-right in the case of

Arabic.

<wbr> Provides a line break opportunity, in a case a long word strings without a space. This improves the readability.

<embed> Represents an external integration, often non-HTML, application or interactive content

<video> Specifies video on an HTML document, further specifying how the video should play. Examples could be Preload, auto play, loop and more.

<audio> Specifies to embed a music file in HTML. Any content between the opening and closing <audio> tags is fall-back content.

<source> Provides an option to embed multiple media resources to give more alternatives files that can be supported to the browser.

<track> Specifies external timed text tracks for media elements. It could include subtitles, caption, description, chapters and metadata.

<canvas> Provides a graphical space for designing graphs, games and visual images.

<math> Define mathematical formula.

<keygen>	Represents a key-pair generator control
<output>	Represents the result of a calculation.
<progress>	Specifies the completion progress of a task
<meter>	Represent a scalar measurement, with a known range
<details>	Represents a widget from which the user can obtain additional Information or controls.
<summary>	Specifies a summary, caption, or legend for a give <details>
<menuitem>	Represents a command that the user can invoke
<menu>	Represents a list of commands

Appendix II

```
/** media queries **/  
  
@media screen and (max-width: 780px) {  
  
nav li a {  
  
    font-size: 2.5em;  
  
    margin-bottom:20px;  
  
    padding-top:1em;  
  
    }  
  
#portfolio li {  
  
    width: 45%;  
  
    }  
  
#d1 ,#d2, #d3,#d4, #d5, #d6, #d7, #d8, #d9, #d10, #d11, #d12, #d13{  
margin:0;}  
  
#designer li {  
  
    float: none;  
  
    display: inline-block;
```

```
width: 45%;  
margin-bottom: 30px;  
}  
  
#bar{  
display:none;}  
  
#gphoto img {  
width: 60%;}  
  
h1 {  
font-size: 3em;  
line-height: 1em;}  
  
#dphoto img{  
width:90%;  
margin-left:30px;  
  
margin-top:30px;  
float:none;}  
  
h2{  
  
margin-top:1em;}
```

```
}  
  
#co{  
  
float:right;  
  
width:27%;  
  
text-align:top;  
  
}
```

```
@media screen and (max-width: 480px) {  
  
#bkg  
  
#bkg1{  
  
width:100%;  
  
margin-bottom:25px;}  
  
#bkg2{  
  
width:100%;  
  
margin-bottom:25px  
  
}  
  
#contact li {  
  
    float: none;  
  
    display: inline-block;  
  
    width: 100%;  
  
}  
  
nav li a {  
  
    font-size: 2.5em;  
  
    margin-bottom:20px;  
  
    display:block;  
  
    text-align:center;
```

```
}
```

```
#portfolio {
```

```
    text-align: center;
```

```
}
```

```
#portfolio li {
```

```
    float: none;
```

```
    display: inline-block;
```

```
    width: 80%;
```

```
    margin-bottom: 30px;
```

```
}
```

```
#d1 ,#d2, #d3,#d4, #d5, #d6, #d7, #d8, #d9, #d10, #d11, #d12, #d13{
```

```
margin:0;}
```

```
#designer li {
```

```
    float: none;
```

```
    display: inline-block;
```

```
    width: 80%;
```

```
    margin-bottom: 30px;
```

```
}
```

```
#gphoto img {
```

```
display: block;

float: none;

width: 90%;

margin-bottom: 30px;}
```

```
  p{

font-size:1.2em;
```

```
}
```

```
h1 {

text-align:center;

font-size: 2.5em;

line-height: 1em;}

#dphoto img{

margin-top:30px;

float:none;

width:90%;
```

```
}  
  
h2{  
  
font-size:2.5em;  
margin-top:1em;}  
  
#co{  
  
float:none;  
width:100%;  
  
}  
  
}
```

```
@media only screen
and (max-width : 320px) {
  #gphoto img {

    display: block;

    float: none;

    width: 90%;

    margin-bottom: 30px;}

  p{

    font-size:1.2em;

  }

}
```