



SCC306 Internet Applications
Engineering
Coursework Element 2: Responsive &
Accessible Web Design

Matthew Bradbury and Phillip Benachour

Please note that this workbook should be anonymous. Do not include your name in the workbook.

Introduction

In this workbook, we will be looking at both responsive and accessible web design. In Part 1, we will focus on how to build an accessible website, whilst Part 2 looks at how to approach responsive web design.

For both parts of the exercise we will use the same template website and build upon it throughout. In other words, once you have completed Part 1, the same website will be used for Part 2 - no need to start again.

The template website is provided on Moodle. The template consists of these files: index.html, style.css, icon.png, aerial.png, banner.png.

Once you are finished with both parts of this workbook, please submit all of your code and this workbook (as a single zipped folder) using the submission point provided in Moodle.

Part 1: Accessible Web Design

The first part of this workbook is to examine accessibility for the supplied website. This involves identifying the issues that are causing the web page to be poorly accessible to some audiences, and then fixing a subset of these by modifying the web page.

To begin with, we recommend you review the BBC's Standards and Guidelines for Mobile Accessibility:

<http://www.bbc.co.uk/accessibility/forproducts/>

You may also find the following resources useful:

<https://tenon.io/>

<http://wave.webaim.org/>

<https://www.w3.org/WAI/standards-guidelines/>

Throughout this section you should refer to appropriate accessibility standards or guidelines by citing them. You may choose which citation style you use, but should be consistent throughout the workbook.

Accessibility Issues

For the first task, and using the resources given above, review the existing website and assess to what extent it meets the criteria, particularly those described in the BBC's guidelines.

Create a list of issues, being clear at each stage what the issue is, and referring to the guidelines to be clear which specific accessibility feature is being violated. You should explain how the issue violates the guidelines/standards/recommendations you reference. To accompany this text, provide evidence of each using a screenshot, you may include a screenshot of the DOM if necessary.

List of Issues

- The first accessibility issue I could immediately identify in the website would be the contrast between the blue and black in the tab fields (or menu) of the website. It directly violates the 1.4 Distinguishable standards for 1.4.3 Contrast (Minimum) in the WCAG 2.1 guidelines, which states that the visual presentation of text and images of text has a contrast ratio of at least 4.5:1, except for a few special scenarios, as in our case, the contrast between the colours is 1.47:1, as per stated in <https://webaim.org/resources/contrastchecker/> . In the checker, this combination of colours fails every test, as it would significantly create struggles for users with low vision or colour deficiencies to access content without using additional assistive technologies. Users who are blind, have low vision, or especially colour deficiencies are not going to be able to differentiate the content in this menu bar:

Home People Courses About us Contact us

- The second thing I notice is that there are no image alternatives to the images on the website. If for an instance, images fail to load or users have a vision impairment, they won't be able to process what the image is, so we need an alt attribute to associate alternative text content with an image in any HTML document. This violates two main guidelines, one would be in the 1.2 Time-based Media standards for 1.2.3 Media alternatives and 2.5 Input Modalities standards for 2.5.3 Label in Name, as both state that if a person has a vision impairment, and is not able to see the images, there should be an alternative tag line to help the screen reader recognise the image and put a label on it, presenting what it is.
- Another problem that the website encounters is that it is not indicating a language, which is definitely a must, as for people who are listening or using assistive technologies like screen readers would need to know what's the language of the website so the speech synthesizer is adjusted. A word in English can mean something totally different in another language. This is directly violating the BBC Accessibility for Products section for Mobile Accessibility Guidelines – Editorial (Indicating Language). Looking at the code, we see that it is fully missing there.
- The 4th problem the website has is that the main page has not been identified as unique and hasn't been clearly identified. It is directly violating the BBC mobile accessibility guidelines – Structure (Unique page/screen titles). As the page title is usually the first thing people see on their monitors, it acts as a confirmation of where they have arrived at, generally helping people orientate where are they and if they are accessing the right resources. Accessibility wise it helps vision impaired users who cannot view or distinguish the whole page at once. It is used for identification, so it is pretty sufficient to be included in the design of a website. This happens by including a <title> tag in the head of the html document. In our website it is clearly missing, so we view the title like that:

 index.html# 

- The 5th problem concerns spacing and more specifically disturbs the BBC mobile accessibility guidelines for Design (Spacing). It states that there should be an inactive space around all actionable elements in a website. This is for anyone that finds it challenging to interact with small and tiny controls, especially ones that are tightly grouped together. If a person with motor or vision impairment happens to be in such a situation, it would be nearly impossible for them to continue and do it appropriately. In our most actionable element, we do not have any spacing between the different buttons:

, which makes it less accessible for people with the disabilities said above. The minimum space possibly set would be 1 pixel, however, the more there is the better in many instances, as the space shall be larger.

- The 6th issue directly infers the BBC mobile accessibility guidelines for Structure(Headings), which states that there should be a logical and hierarchical heading structure, which is consistent around the whole website. It enables people to understand the content of the page quicker, as well as, allows screen reader users to navigate easily within the screen. In our website, this consistent layout is disrupted in places, more specifically can be seen in here:

Facts and Figures

Students at Lancaster University are among the most satisfied in the UK according to the 2015 National Student Survey, in which the University achieved 91% on the overall satisfaction score, placing it firmly amongst the top 10 UK universities. Summer 2011 saw the opening of our brand new £20m Sports Centre, ranked sixth in the THE Student Experience Survey 2015. 97% of Lancaster University graduates have gone into work or further study within six months of graduating. As one of the few collegiate universities in the UK we welcome each student to one of our colleges on campus. In addition to our eight undergraduate colleges, we have one dedicated to our postgraduate students. [Read More](#)

News

Lancaster Academics to unlock patient information to create a healthier North West

The headings here are different in style, which makes it way harder to people to follow the outline and navigate within the page.

- The 7th issue I've found is concerning and directly impacting the BBC Mobile Accessibility Guidelines for Design (Visible focus), which states that all actionably and focusable elements must have a visible state change. In our case, for instance in the tab bar menu, whenever we hover over an element, nothing changes, so there really isn't a way to easily recognise if the element is clickable or actionable straightaway. This is an issue for sighted keyboard and switch device users that track progress as they navigate throughout a webpage, as well as, people controlling with a TV interface or generally touch users, as they would receive a confirmation that an element is interactive. The issue could be easily seen here:

A hovering change of state should be introduced so people can navigate easily.

- The 8th problem I've encountered is concerning the BBC Mobile Accessibility Guidelines – Design (Content resizing), which states that users must be able to control themselves the font sizing and user interface scaling, meaning that when it doesn't happen or exist in a website, users with vision impairment, or generally users who like to zoom in to view clearer are in a disadvantage and unable to adapt towards the size of the content to see. This happens when someone tries to zoom in on the website, it surely can be done in a better way:

Located on a beautiful campus in the North West of England, the University places great emphasis on a strong student experience and employability and gives students access to academics who are experts in their field. Lancaster is one of only a handful of universities with a

Accessibility Fixes

From the list you have identified above, choose 4 unique issues, each of which require a different *type* of fix. In the following section, work through each of these issues, first by identifying the offending code/content, using a snippet to highlight exactly where the issue has occurred.

Then, using the knowledge you have gained from the Accessibility lecture and associated readings, provide the fix to address each accessibility problem. To do this, document the fix using code/content snippets, and provide *evidence* of how it resolves the accessibility problem in the rendered web page. Use a small textual description to aid your explanation of *how* you achieved the fix. Refer to appropriate guidelines/standards/recommendations as to how your change has now resolved the accessibility problem.

Accessibility Fix 1

Explanation, code extracts, screenshots/ evidence

Here, we are going to fix the first issue we identify in our list, which is the contrast between the blue and black in the tab field at the top of our page. The code that causes this problem is the following in our .css file, where it customises the `ul{}` and `li a{}` elements:

```
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333;
}

li {
  float: left;
}

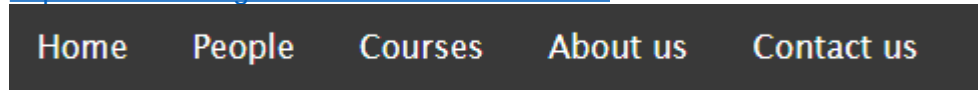
li a {
  display: block;
  color: blue;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  outline: none;
}
```

The lines that identify the colours would be the `background-color: #333` in the `ul` element and the `color: blue` in the `li a` element. In order to fix this, we need to change the colour and achieve a ratio higher than 4.5:1 as per the guidelines state. The fix is the following:

```
li a {
  display: block;
  color: #FFFFFF;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  outline: none;
}
```

This changes the colour of the text in the menu element, making it white, which achieves a contrast ratio of 12.63:1, meeting the guidelines of a 4.5:1 minimum:

<https://webaim.org/resources/contrastchecker/>



Home People Courses About us Contact us

We've resolved the issue and now meet the guideline for minimum contrast of text, being more accessible to people with vision disabilities. Our solution even passes the 1.4.6 Contrast (Enhanced) guidelines, as our contrast ratio is higher than the one specified there being at least 7:1, which means we now have an optimal accessibility with the text.

Accessibility Fix 2

Explanation, code extracts, screenshots/ evidence

Here we are going to fix the second issues in our list, which is the missing alternative tag line for images, which prevents screen readers, subtitles and etc to recognise images, so by putting one on the image section in our code, we are going to help people with vision impairments associate alternative text content with an image. The issue is the missing alt attribute:

```
▼ <div id="header">
...    == $0
    </div>
▼ <div class>
    
  </div>
```

What I am going to do to fix it, is add this alt attribute and put a meaningful associative text for the respective images:

```
▼ <div id="header">
    
  </div>
▼ <div class>
    
  </div>
▶ <div id="menu-bar">...</div>
▼ <div id="campus">
    <h2>University Campus</h2>
..   
    == $0
```

Accessibility Fix 3

Explanation, code extracts, screenshots/ evidence

In this section, we're going to look at the missing indicating language component. The missing part is located in the the very first lines where the `<html>` tag is opened:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta name="viewport" content="width=device-width, initial-scale=1.0">
5   <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
```

Now, in order to fix this issue, we have to add a language specification to the html tag. This is going to help people who are listening or using assistive technologies like screen readers who would need to know what's the language of the website so the speech synthesizer they use is adjusted and doesn't interpret words in a different way than the specified language. The fix is done like that:

```
2 <html lang="en-GB">
```

Now, we have specified that the language of the website is British English. We've checked it works and per the BBC mobile accessibility guidelines for Editorial(indicating language) have been met now.

Accessibility Fix 4

Explanation, code extracts, screenshots/ evidence

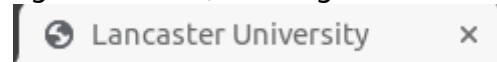
In this section, we're going to be looking at the page title issue and specifically how the lack of it undermines people orientating themselves within websites and apps. Within the BBC mobile accessibility guidelines - Structure(unique page/screen titles), we need to introduce a title for our unique page, so that vision impaired user who cannot perceive the whole page at once can still identify what resources are they looking at and orientate around the web. The troubled section in our code is the following, within the head of the html file:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta name="viewport" content="width=device-width, initial-scale=1.0">
5     <link rel="stylesheet" type="text/css" href="style.css">
6 </head>
```

We need to add a title tag in there, specifying the name of the page. We're going to do that in the following way:

```
<head>
    <title>Lancaster University</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
```

The title tag is fixing our issues, making it look like that:



Now people can immediately know what unique page are they on and orientate themselves.

Part 2: Responsive Web Design

In this part of the workbook we will be focusing on building a responsive website, using the same code/content you have worked on and improved in the previous part of this exercise.

To begin with, you will familiarise yourselves with the techniques, tools and approaches necessary to build a responsive website. You will then move on to applying these concepts to make the existing website responsive.

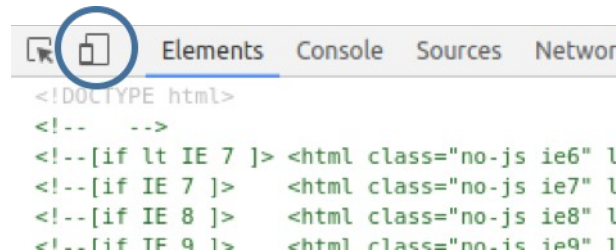
Getting Started

These initial exercises will help you to get started with this element of the coursework.

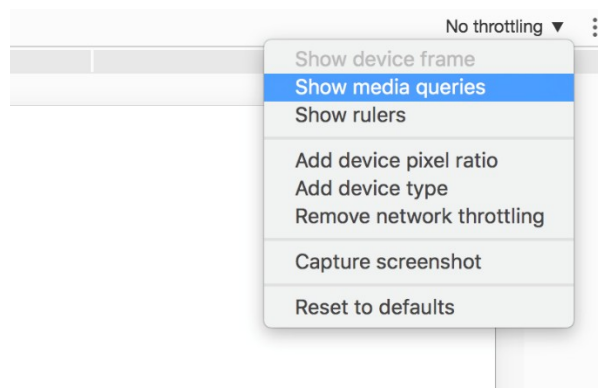
Exercise 1: Chrome developer tools device mode

In this exercise, you will use the Chrome developer tools with device mode enabled. Device mode allows you to spoof the User Agent (UA) and emulate screen sizes of different mobile devices. This tool is particularly useful for checking if your website will be viewable on a range of devices and can also be used to identify the breakpoints of existing responsive websites.

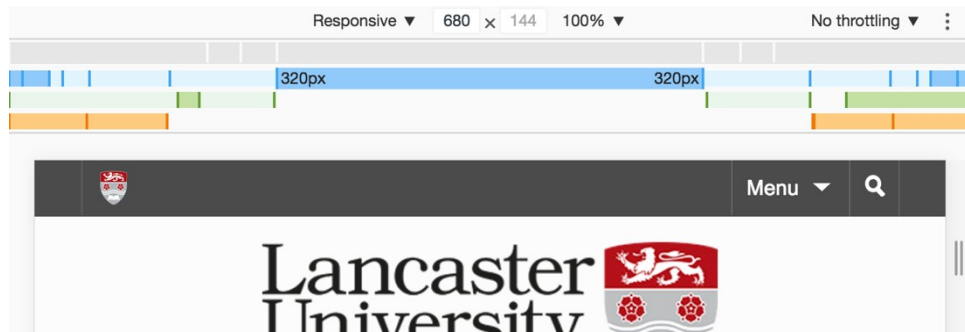
To use device mode in Google Chrome, go to developer tools and select device mode:



Click on the settings button on the top right-hand corner of the page and enable “Show media queries”:



At the top of your browser, there will be the breakpoints for the webpage. Go to <http://www.lancaster.ac.uk/> and see what breakpoints they have designed, e.g.:



More information about device mode is available at:

<https://developers.google.com/web/tools/chrome-devtools/device-mode/>

Next, examine how these websites change when you select devices of different resolutions. Identify the breakpoints of each site (resize the page until a change is triggered):

- <http://www.youtube.com>
- <https://modules.lancaster.ac.uk>

Exercise 2: W3schools responsive web design

We **strongly** recommend that you go through the *entire* responsive section available at W3Schools to familiarise yourself with the fundamentals.

http://www.w3schools.com/css/css_rwd_intro.asp

Responsive Web Design Assignment

For this exercise, you will make changes to the web page you have already been working on to enable a responsive design. This should react to different screen sizes and widths. To do this, you need to provide screenshots of your website at each breakpoint as instructed. Use the Chrome developer tools in device mode to capture the **entire page** (take multiple screenshots or scale the page if required). You must also provide an explanation of the methods used to make the website look the way it is at **each breakpoint**. To support this, include extracts of your code accordingly.

For any additional features you may have added, show and explain code and screenshots. Finally, explain how the feature improves the responsiveness of the website.

Once you have completed both parts of this workbook, upload your website and workbook to Moodle (include index.html, style.css, icon.png, aerial.png, banner.png, this lab book document, and any other files you may have used).

In this new website, we expect to see the following principles:

- Media Queries
- Fluid Grids
- Fluid Images

You may also consider using features that facilitate intrinsic web design, such as with intrinsic grids and the intrinsic sidebar. The coursework should target the feature set of Chrome 107.

The responsive website you create should cater for a range of devices:

- Mobiles (425px)
- Tablets (768px)
- Laptops (1024px)
- Large Laptops/Desktops (1440px)

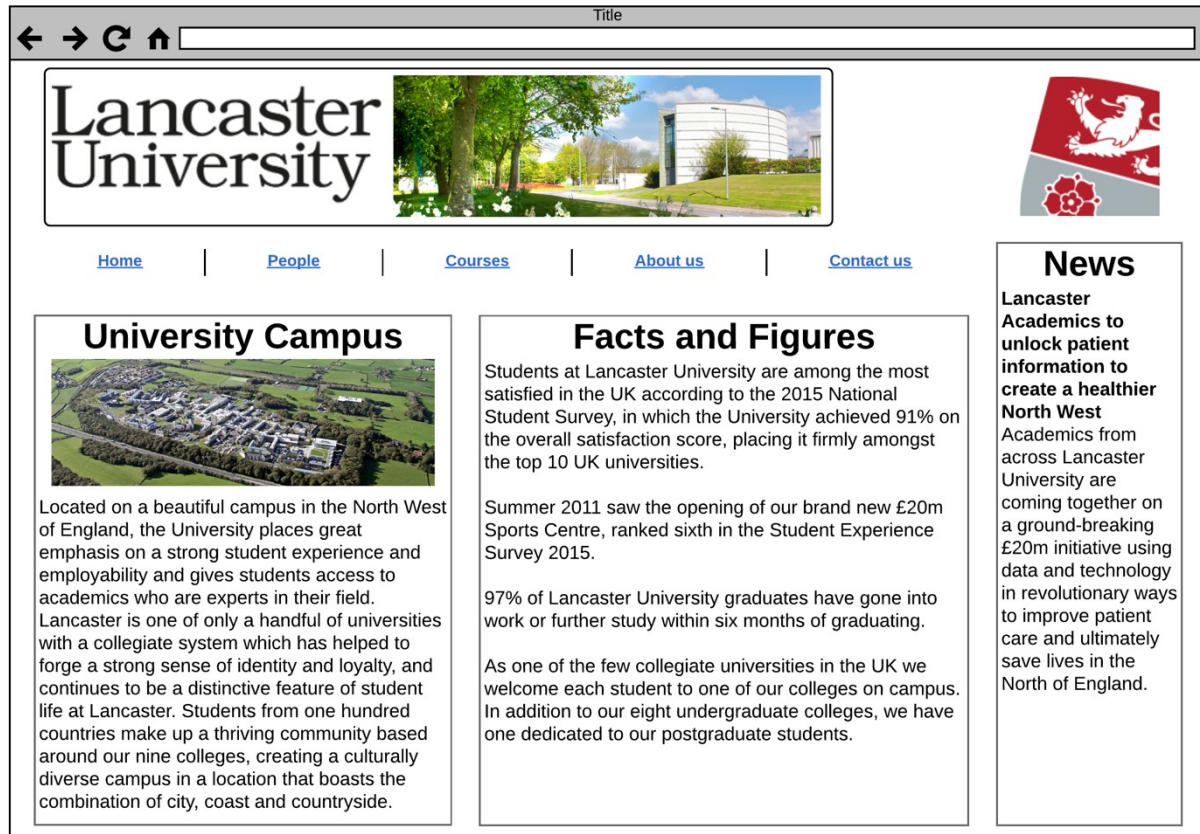
Additional marks will be available to students who provide and *justify* up to two other responsive features appropriately. Examples include but are not limited to:

- Using a touch friendly 'burger' or vertical menu on a break in design.
- Creating a collapsible box, only showing the most important content until expanded.

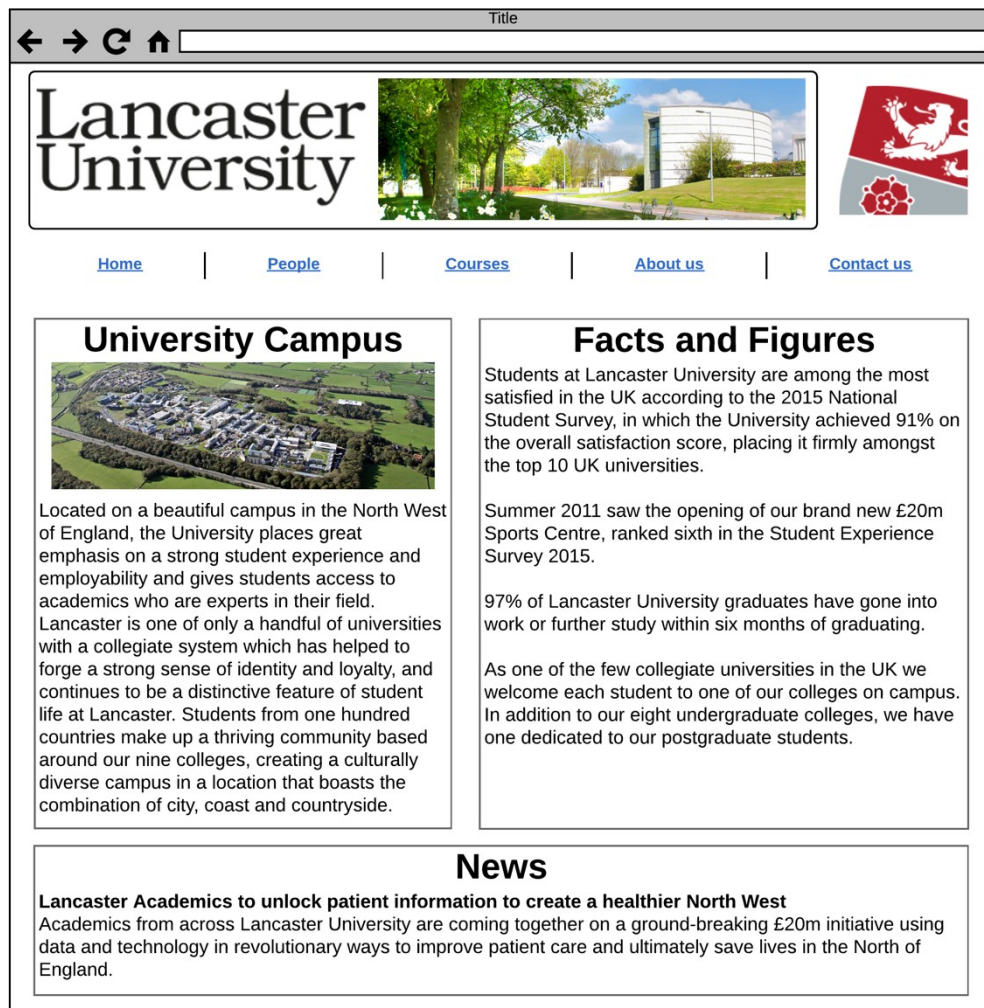
The following are mock-ups of what we expect the website to look like at each breakpoint. Whilst you are free to experiment with the design, when making your website try to replicate the **structure** shown in the mock-ups as much as possible. To evidence your responsive website, download the workbook from Moodle and fill in each section.

Note: You may use any framework, tools, or techniques that you wish to complete this task. However, this assignment can be accomplished with the use of CSS alone, as shown in Exercise 2.

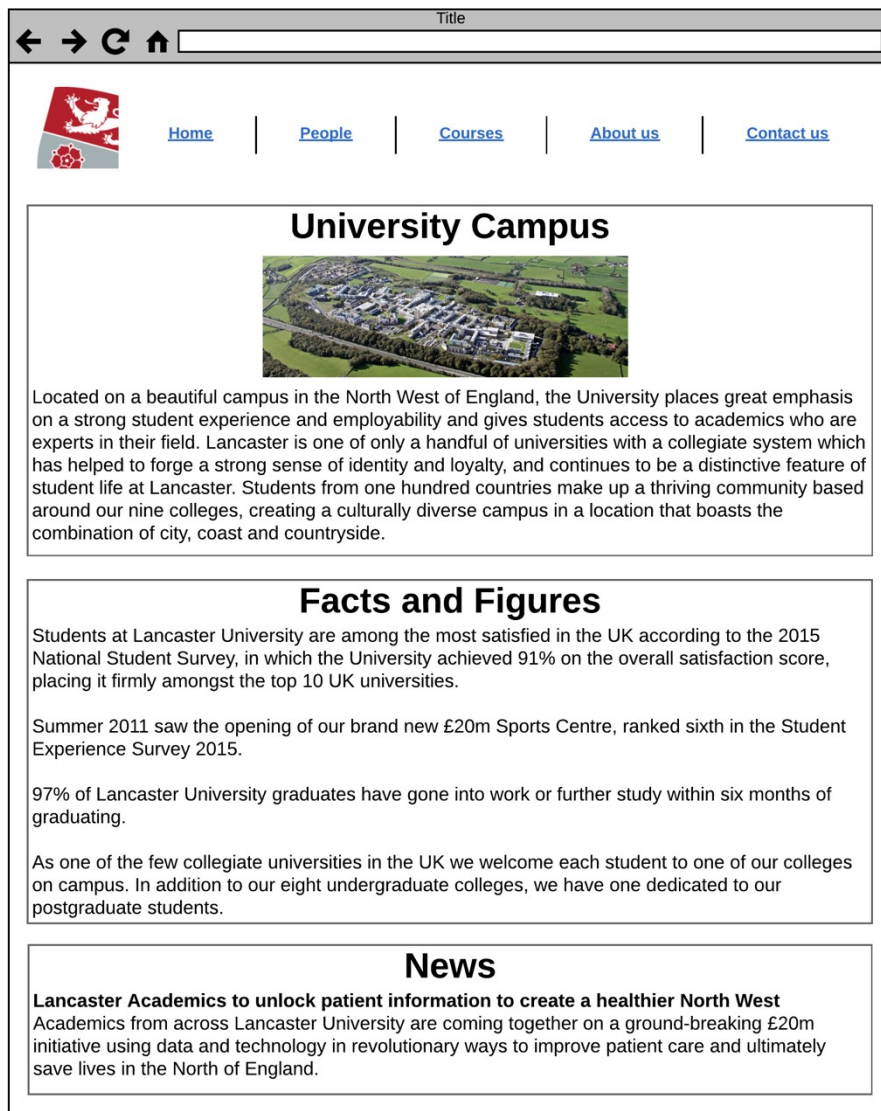
Large Laptop/Desktop (1440px)



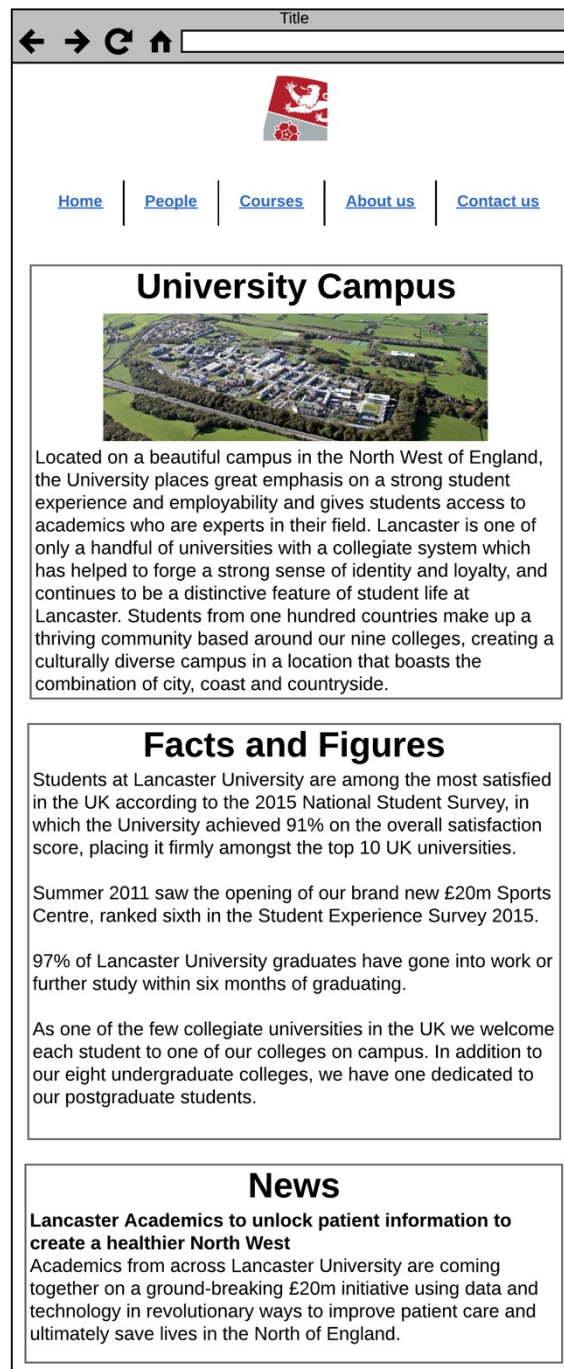
Laptop (1024px)



Tablet (768px)

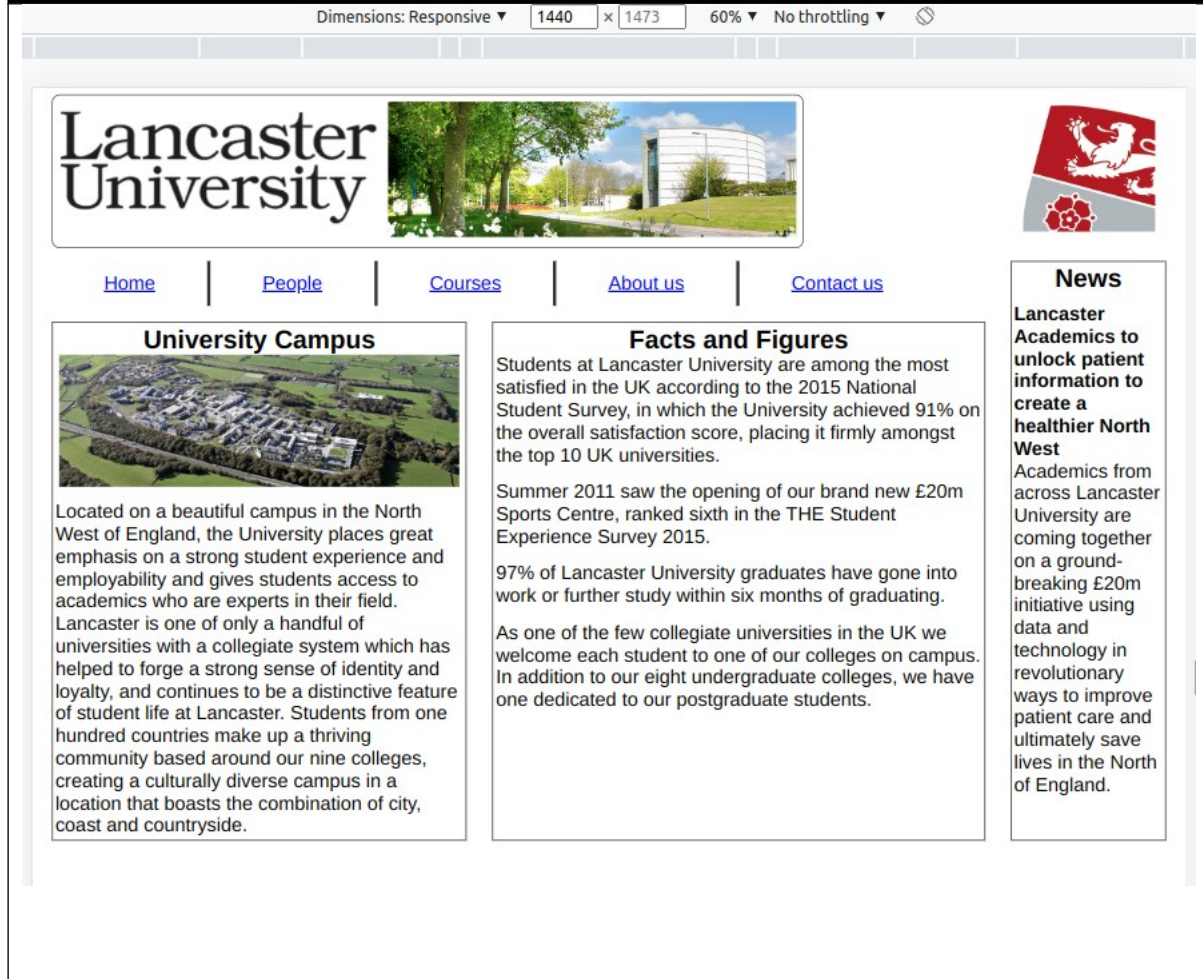


Mobile (425px)



Laptop Large/Desktop (1440px)

Screenshot at 1440px (website and breakpoint using device mode)



Code extracts and explanation

The way I've done the main part of this media query breakpoint is by doing the following code:

```
/* Media query for desktop: */
@media only screen and (min-width: 1440px) {
  .container{
    display: grid;
    height: 100vh;
    grid-template-columns: 0.8fr 0.6fr 0.3fr 0.3fr;
    grid-template-rows: 0.1fr 0.1fr 0.1fr;
    grid-template-areas:
      "header header none logo"
      "menu-bar menu-bar menu-bar news"
      "campus facts facts news"
    ;
    grid-column-gap: 2rem;
    grid-row-gap: 1rem;
    padding-left: 1em;
    padding-right: 1em;
  }
}
```

the @media only screen line defines that the following layout is only going to be for the desktop media query with minimum width of the screen being 1440 pixels. Afterwards I've created a container and specified that the container is going to be a grid, while also specifying the number of columns and rows in the template and their sizes. In this section I've gone for 4 columns in grid-template-columns, each taking a different proportion of the screen and 3 rows, all of them being the same size. In the line grid-template-areas I've defined the position of all of the elements and the places of the grid they are going to take, creating the main layout.

```
#header{
  grid-area: header;
  border: 2px solid #333;
  border-radius: 10px;
  padding: 0.5em;
}

#logo{
  grid-area: logo;
  align-self: center;
  justify-self: center;
}

#menu-bar{
  grid-area: menu-bar;
}
```

Afterwards I've defined each of the elements in the .css file by their html tag id

name and called them so that I can modify them following the template. For each of the main elements, there's a border around, just like we see in the #header, and almost every element has a form of padding around so that it can be adjusted and effective towards responsiveness and accessibility. The main model of formatting elements can also be seen in this snippet of code:

```
#campus{
  grid-area: campus;
  border: 1px solid #333;
  padding: 0.2em;
}

#campus > h1{
  text-align: center;
  margin-block-start: 0em;
  margin-block-end: 0em;
}

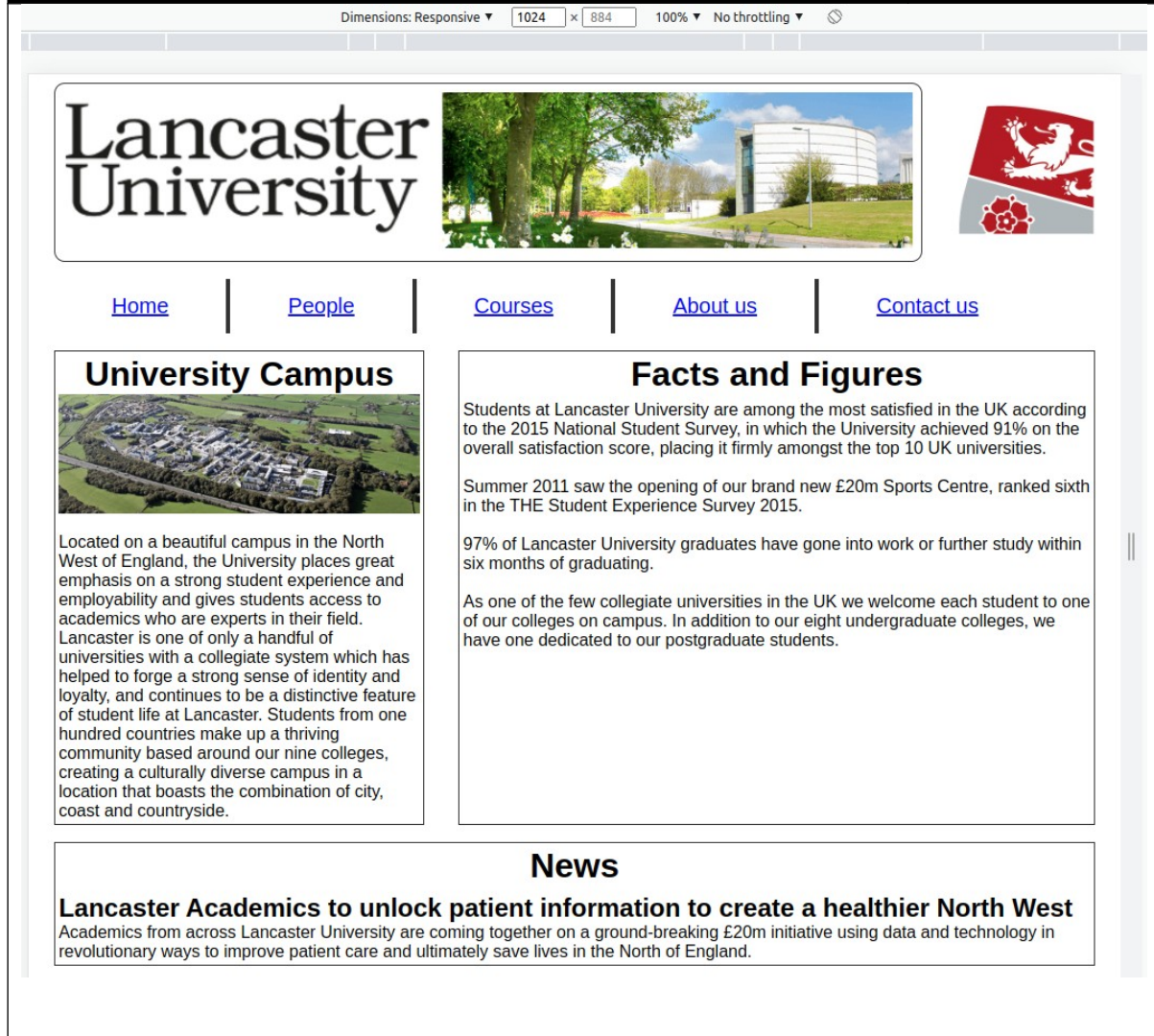
#campus > img{
  display: block;
  margin-left: auto;
  margin-right: auto;
  width: 50%;
}

#campus > p{
  font-size: 24px;
  margin-block-start: 0em;
  margin-block-end: 0em;
}
```

Firstly, I'm looking at the main id of the element, creating a border around it, giving it a bit of space to breathe with the padding, and then it is followed by other methods, each specifying a different component of the whole element, h1 is going to be the header, img is going to be the image in it, while p would be the normal text. I've modified each in design, so that it follows the template as well, which is achieved by aligning both texts and images, and playing with the format of the normal text.

Laptop (1024px)

Screenshot at 1024px (website and breakpoint using device mode)



Code extracts and explanation

What changes in the code from the above dekstop (1440 px) media query has been presented in the following snippet of code:

```
/* Media query for laptop: */
@media only screen and (min-width: 1024px) {
  .container{
    display: grid;
    height: 100vh;
    grid-template-columns:0.4fr 0.5fr 0.15fr;
    grid-template-rows: 0.2fr 0.2fr 0.2fr 0.2fr;
    grid-template-areas:
      "header header logo"
      "menu-bar menu-bar menu-bar"
      "campus facts facts"
      "news news news"
    ;
    grid-column-gap: 2rem;
    grid-row-gap: 1rem;
    padding-left: 1em;
    padding-right: 1em;
  }
}
```

We've changed the min width, so we specify the laptop format would be from 1024 to 1440 px, as well as, made the grid responsive to the smaller format by making it a 3 column grid with 4 rows. The designing of the format and everything else follows the idea of the dekstop media query, elements have been split up in their subelements, and if necessary, anything is easily declarable and modifyable. Specifically here, the menu bar is a more interesting element, as it stayts alone on a row by itself. The following format and placement has been achieved by the following:

```
#menu-bar{
  grid-area: menu-bar;
}

ul{
  justify-self: center;
  overflow:visible;
  background-color: #FFFFFF;
}

li a{
  display: flex;
  align-items: center;
  color: #0000FF;
  font-size: 20px;
  text-decoration: underline;
  padding-right: 54px;
  padding-left: 54px;
  border-right: 4px solid #333;
}

li:last-child a{
  border-right: none;
}
```

The ul element is the whole element, while the li a would be every cell on the menu bar (the different categories - home, people, courses and etc). They have significant padding so that it could be formatted right, as well as a border on the right, except for the last child a, so that there are those small columns in the middle between the options.

Tablet (768px)

Screenshot at 768px (website and breakpoint using device mode)

[Home](#)[People](#)[Courses](#)[About us](#)[Contact us](#)

University Campus



Located on a beautiful campus in the North West of England, the University places great emphasis on a strong student experience and employability and gives students access to academics who are experts in their field. Lancaster is one of only a handful of universities with a collegiate system which has helped to forge a strong sense of identity and loyalty, and continues to be a distinctive feature of student life at Lancaster. Students from one hundred countries make up a thriving community based around our nine colleges, creating a culturally diverse campus in a location that boasts the combination of city, coast and countryside.

Facts and Figures

Students at Lancaster University are among the most satisfied in the UK according to the 2015 National Student Survey, in which the University achieved 91% on the overall satisfaction score, placing it firmly amongst the top 10 UK universities.

Summer 2011 saw the opening of our brand new £20m Sports Centre, ranked sixth in the THE Student Experience Survey 2015.

97% of Lancaster University graduates have gone into work or further study within six months of graduating.

As one of the few collegiate universities in the UK we welcome each student to one of our colleges on campus. In addition to our eight undergraduate colleges, we have one dedicated to our postgraduate students.

News

Lancaster Academics to unlock patient information to create a healthier North West

News

Lancaster Academics to unlock patient information to create a healthier North West

Academics from across Lancaster University are coming together on a ground-breaking £20m initiative using data and technology in revolutionary ways to improve patient care and ultimately save lives in the North of England.

Code extracts and explanation

The code for the tablet media query differs a bit from the others. While the others only have minimum or maximum width, I've decided to give a bit more options for the tablet, as there is a large amount of varying tablets by screen and size. I've done the following:

```
/* Media query for tablet: */  
/*768 px*/  
@media only screen and (max-width: 1023px) and (min-width:426px) {  
  .container{  
    display: grid;  
    height: 100vh;  
    grid-template-columns: 0.1fr 0.9fr;  
    grid-template-rows: 0.2fr 0.6fr 0.6fr 0.4fr;  
    grid-template-areas:  
      "logo menu-bar"  
      "campus campus"  
      "facts facts"  
      "news news"  
    ;  
    grid-row-gap: 1rem;  
  }  
}
```

This way it goes from minimum width of 426 px to max width of 1023px, so transitions are smoother between devices. The grid is in the form of 2 columns and 4 rows, and the formatting of everything is the same as the other media queries above this one. Only difference is that here, apart from the very first row, where there are 2 elements on it, the row and the menu bar splitting it 10/90, the other rows fully manage 1 component. That's why it looks like following a 1 page model.

Mobile (425px)

Screenshot at 425px (website and breakpoint using device mode)



[Home](#)

[People](#)

[Courses](#)

[About us](#)

[Contact us](#)

University Campus



Located on a beautiful campus in the North West of England, the University places great emphasis on a strong student experience and employability and gives students access to academics who are experts in their field. Lancaster is one of only a handful of universities with a collegiate system which has helped to forge a strong sense of identity and loyalty, and continues to be a distinctive feature of student life at Lancaster. Students from one hundred countries make up a thriving community based around our nine colleges, creating a culturally diverse campus in a location that boasts the combination of city, coast and countryside.

Facts and Figures

Students at Lancaster University are among the most satisfied in the UK according to the 2015 National Student Survey, in which the University achieved 91% on the overall satisfaction score, placing it firmly amongst the top 10 UK universities.

Summer 2011 saw the opening of our brand new £20m Sports Centre, ranked sixth in the THE Student Experience Survey 2015.

97% of Lancaster University graduates have gone into work or further study within six months of graduating.

Facts and Figures

Students at Lancaster University are among the most satisfied in the UK according to the 2015 National Student Survey, in which the University achieved 91% on the overall satisfaction score, placing it firmly amongst the top 10 UK universities.

Summer 2011 saw the opening of our brand new £20m Sports Centre, ranked sixth in the THE Student Experience Survey 2015.

97% of Lancaster University graduates have gone into work or further study within six months of graduating.

As one of the few collegiate universities in the UK we welcome each student to one of our colleges on campus. In addition to our eight undergraduate colleges, we have one dedicated to our postgraduate students.

News

Lancaster Academics to unlock patient information to create a healthier North West

Academics from across Lancaster University are coming together on a ground-breaking £20m initiative using data and technology in revolutionary ways to improve patient care and ultimately save lives in the North of England.

Code extracts and explanation

The phone media query has been made in the following way:

```
/* Media query for mobile: */
@media only screen and (max-width: 425px) {
  .container{
    display: grid;
    height: 100vh;
    grid-template-columns: 1fr;
    grid-template-rows: 0.2fr 0.3fr 0.6fr 0.6fr 0.4fr;
    grid-template-areas:
      "logo"
      "menu-bar"
      "campus"
      "facts"
      "news"
    ;
    grid-row-gap: 1rem;
  }
}
```

It's limited to 425 px maximum width, and its following strictly a 1 column format, with many rows, as each element has its own row. The formatting is the same as the other queries and can be easily seen in this snippet:

```
#news{
  grid-area: news;
  border: 1px solid #333;
  padding: 0.2em;
  padding-bottom: 0.5em;
}

#news > h1{
  text-align: center;
  margin-block-start: 0em;
  margin-block-end: 0em;
}

#news > h2{
  margin-block-start: 0.5em;
  margin-block-end: 0em;
}

#news > p{
  margin-block-start: 0em;
  margin-block-end: 0em;
}
```

Additional Feature 1

Screenshot at relevant breakpoint

University Campus



Located on a beautiful campus in the North West of England, the University places great emphasis on a strong student experience and employability and gives students access to academics who are experts in their field....

[Read More](#)

University Campus



Located on a beautiful campus in the North West of England, the University places great emphasis on a strong student experience and employability and gives students access to academics who are experts in their field. Lancaster is one of only a handful of universities with a collegiate system which has helped to forge a strong sense of identity and loyalty, and continues to be a distinctive feature of student life at Lancaster. Students from one hundred countries make up a thriving community based around our nine colleges, creating a culturally diverse campus in a location that boasts the combination of city, coast and countryside.

[Read less](#)

This screenshot is for the 1440 dekstop breakpoint, but the feature has been added for all of them.

Code extracts and explanation

The first feature I've implemented is a responsive Read More button, which has the function to hide some specified part of the text, and whenever clicked, it reveals that part, making the pages less cluttery. In order to do that, I've created and named a few elements in the html in the following way:

```
<p>Located on a beautiful campus in the North West of England, the University plac  
<span id = "more">Lancaster is one of only a handful of universities with a colleg  
<!--<a href="#">Read More</a-->  
<button onclick="readMore()" id="button" class="button">Read More</button>
```

The text I want to hide under the read more button is one with the id = "more", while the button itself has been initialised, containing a JavaScript function, that does the animation and effect of hiding and showing the text:

```
function readMore() {  
  var dots = document.getElementById("dots");  
  var moreText = document.getElementById("more");  
  var btnText = document.getElementById("button");  
  
  if (dots.style.display === "none") {  
    dots.style.display = "inline";  
    btnText.innerHTML = "Read more";  
    moreText.style.display = "none";  
  } else {  
    dots.style.display = "none";  
    btnText.innerHTML = "Read less";  
    moreText.style.display = "inline";  
  }  
}
```

This allows us to expand and shrink the visible text. The last piece of the puzzle here would be the cs, which helps us hide the needed part initially, so that the button can do its job:

```
#button{  
  font-size: 15px;  
}  
  
#more{  
  display: none;  
}
```

The display: none method does the job. Additionally, for some of the media queries I've provided a bigger font size for the button so that it follows the format of the template.

Additional Feature 2

Screenshot at relevant breakpoint


The second feature I've introduced is the interactive and responsive sidebar menu:

Dimensions: Responsive ▾


1440 × 1008

50% ▾ No throttling ▾

Lancaster University



Open Sidebar



[Home](#)

|

[People](#)

|


[Courses](#)

|

[About us](#)

|

[Contact us](#)

University Campus

Located on a beautiful campus in the North West of England, the University places great emphasis on a strong student experience and employability and gives students access to academics who are experts in their field. Lancaster is one of only a handful of universities with a collegiate system which has helped to forge a strong sense of identity and loyalty, and continues to be a distinctive feature of student life at Lancaster. Students from one hundred countries make up a thriving community based around our nine colleges, creating a culturally diverse campus in a location that boasts the combination of city, coast and countryside.

Read less

Facts and Figures

Students at Lancaster University are among the most satisfied in the UK according to the 2015 National Student Survey, in which the University achieved 91% on the overall satisfaction score, placing it firmly amongst the top 10 UK universities.

Summer 2011 saw the opening of our brand new £20m Sports Centre, ranked sixth in the THE Student Experience Survey 2015.

97% of Lancaster University graduates have gone into work or further study within six months of graduating.

As one of the few collegiate universities in the UK we welcome each student to one of our colleges on campus. In addition to our eight undergraduate colleges, we have one dedicated to our postgraduate students.

Lancaster Academics to unlock patient information to create a healthier North West

Academics from across Lancaster University are coming together on a ground-breaking £20m initiative using data and technology in revolutionary ways to improve patient care and ultimately save lives in the North of England.

[Services](#)
[Clients](#)

[People](#) |
 [Courses](#) |
 [About us](#) |
 [Contact us](#)

Campus

Lancaster University is one of the most beautiful campuses in the world, the University is on a strong and employability access to experts in their field of only a handful of colleges in the collegiate system forge a strong sense of loyalty, and distinctive feature of Lancaster University. Students from all over the world make up a vibrant community based around our rich history and a culturally diverse location that combines city, coast

Facts and Figures

Students at Lancaster University are among the most satisfied in the UK according to the 2015 National Student Survey, in which the University achieved 91% on the overall satisfaction score, placing it firmly amongst the top 10 UK universities.

Summer 2011 saw the opening of our brand new £20m Sports Centre, ranked sixth in the THE Student Experience Survey 2015.

97% of Lancaster University graduates have gone into work or further study within six months of graduating.

As one of the few collegiate universities in the UK we welcome each student to one of our colleges on campus. In addition to our eight undergraduate colleges, we have one dedicated to our postgraduate students.

News

Lancaster Academics to unlock patient information to create a healthier North West

Academics from across Lancaster University are coming together on a ground-breaking £20m initiative using data and technology in revolutionary ways to improve patient care and ultimately save lives in

Code extracts and explanation

There is a open sidebar button at the top right of the page, which opens another sidebar menu with new options on the left of the page. I've done it in the following way:

```
<div id="sidebar" class="sidebar">
  <a href="javascript:void(0)" class="closebtn" onclick="closeNav()">&times;</a>
  <a href="#">Services</a>
  <a href="#">Clients</a>
</div>

<div id="sidebarbutton">
  <button class="opensbd" onclick="openNav()">&#9776; Open Sidebar</button>
</div>
```

This introduces both the new options in the menu and the open and close buttons for the navigation. The close button can only be seen after the sidebar has been opened, the functions are the following:

```
function closeNav() {
  document.getElementById("sidebar").style.width = "0";
  document.getElementById("sidebarbutton").style.marginLeft = "0";
}

function openNav() {
  document.getElementById("sidebar").style.width = "250px";
  document.getElementById("sidebarbutton").style.marginLeft = "250px";
}
```

This closes and opens the elements. The final thing missing is the css which is defining the whole style for the gadget:

```
/* The sidebar menu */
.sidebar {
  height: 100%;
  width: 0;
  position: fixed;
  z-index: 1;
  top: 0;
  left: 0;
  background-color: #111;
  overflow-x: hidden;
  padding-top: 60px;
  transition: 0.5s; /* 0.5 second transition effect to slide in the sidebar */
}

.sidebar a {
  padding: 8px 8px 8px 32px;
  text-decoration: none;
  font-size: 25px;
  color: #818181;
  display: block;
  transition: 0.3s;
}

.sidebar a:hover {
  color: #f1f1f1;
}

/* Position and style the close button (top right corner) */
.sidebar .closebtn {
  position: absolute;
  top: 0;
  right: 25px;
  font-size: 36px;
  margin-left: 50px;
}

.opensbd {
  font-size: 20px;
  cursor: pointer;
  background-color: #111;
  color: white;
  padding: 10px 15px;
  border: none;
}
```

