

## COMM049: Mobile Application Development

### Refining the UI

Autumn 2016

#### Introduction

The View for your app is now a bit slicker, but it is way off being suitable for a mobile device. We will now start to progressively refine the app so that we will end up with a UI that is suitable for a much wider range of devices.

You should use your own judgement/creativity in addressing this. However, I will start you off with a few pointers.

You can build on your own project, or use the archive file on SurreyLearn. If using your own project, don't forget to remove the line in the *application.js* file that "requires" turbolinks.

#### Some background on styling

I will talk a little about SCSS in the session today and you will find it makes your styling documents more succinct. You don't have to use it but it is worth trying it out if you are not familiar with it.

Do include a version of normalize.css in your project.

One thing to be aware of. The *application.css* manifest file includes the line:

```
*= require_tree .
```

This can be a bit of a pain as you may find that the styling files are compiled in a different sequence to what you were expecting. I strongly recommend you alter the manifest to compile the files in the order *you* want. For example:

```
*= require normalize.css
*= require scaffolds.scss
*= require articles.scss
*= require comments.scss
*= require_self
*/
```

(The *require\_self* statement ensures that any css in your *application.css* file will come at the end of the css asset that is served back to the browser).

#### Altering the Index page

Just to set the direction, the first thing to do is to change the table layout of the *index.html.erb* file in */views/articles* (Why?).

I suggest you change it to something like this code:

## Making the application more dynamic

```
<p id="notice"><%= notice %></p>
```

```
<section>
```

```
<h1>Listing Articles</h1>
```

```
<% @articles.each do |article| %>
```

```
<article class="blog">
```

```
<h1><%= article.title %></h1>
```

```
<h2><%= article.author %> | <%= article.date %></h2>
```

```
<section>
```

```
<%= link_to 'Show', article %> |
```

```
<%= link_to 'Edit', edit_article_path(article) %> |
```

```
<%= link_to 'Destroy', article, method: :delete, data: { confirm: 'Are you sure?' } %>
```

```
%>
```

```
</section>
```

```
</article>
```

```
<% end %>
```

```
</section>
```

```
<%= link_to 'New Article', new_article_path %>
```

Note that I am just showing the titles of the articles, and not the content.

Now, what you can do is to use a similar JavaScript approach to revealing the content of a blog article by clicking “Show” and have it revealed on the same page.

But do think carefully about the overall structure. You might want to change it so that the landing page of your application is the latest blog article. Then use a link to allow the user to go to the index of older articles.