Launch Screen - The first screen users will see when opening the app. It is made up of the following components

* A View Controller - This will act as the container for each screen
  1. At the very bottom of the panel on the right, there is a search bar. Use that, and then drag and drop the View Controller onto the main screen
* An Image viewer - this is for the background photo of the skeletal spine
  1. From the bottom portion of the right-side panel, drag and drop an image viewer into the view controller. Stretch it to cover the whole VC
  2. Ensure that the image you wish to use is within the project repository
  3. Select the image viewer, and then in the right-hand panel, go to the attribute inspector tab
  4. Under the Image view section, choose the image that should go there
  5. Set the view to “scale to fill”
  6. Make sure it’s opaque, not hidden, and clipped to bounds
  7. In the bottom panel, click the add new constraints button, make sure there is 0 spacing to nearest neighbour on each side

Title Screen - This page has the same look and feel as the launch screen, except it has a title label and a log in button

* A Label - this is for the project title
  1. From the bottom portion of the right-side panel, drag and drop an image viewer into the view controller
  2. Select the label, and then in the right-hand panel, go to the attribute inspector tab
  3. Select the plain text type and then type the title right under that. Make sure it is black, centred, enabled and highlighted
  4. Set a white shadow with an offset of 3
  5. Set the background to be a light text colour. Ensure that it is not opaque in this case.
* A button - this is for the project title
  1. From the bottom portion of the right-side panel, drag and drop a button into the view controller
  2. Select the button, and then in the right-hand panel, go to the attribute inspector tab and ensure the settings look like this:



* 1. Add another VC. Hold down the control key. Click the button and drag the mouse over to the next VC. Select ‘show modally’. Now when the button is clicked, it will lead to that VC (page), which is the Study tab, allowing the user to select the “medical cases” mode or the “speed round” mode.
* Stack view – Horizontal and vertical stacks help with keeping component aligned relative to other objects, so the layout looks the same across all platforms
  1. Drag and drop the stacks from the object library in the bottom right
  2. Add constraints, the same way they were added for the image in the launch screen
  3. The objects that should be within the stack can now be dragged over the stack in the document outline until the blue line gets shorter and the object is now indented under the stack in the document outline
  4. The object’s constraints may need to be adjusted

Study Tab – this page allows users to choose whether they would like to go through cases, or be given stand-alone multiple-choice questions

* Buttons – Added same way as described in the title screen section. Should be called “medical cases” and “speed round”. Each button will lead to a Spinner screen where the user can be more specific about the types of questions/cases wanted.
* Stack view– Horizontal and vertical stacks help with keeping component aligned relative to other objects, so the layout looks the same across all platforms
  1. Drag and drop the stacks from the object library in the bottom right and set it up, same as it was done for the title screen
  2. Ensure the two buttons are within the stack view

Category Screen (both) – There will be two identical screens like this. One for the med cases, and one for the speed round. It allows users to choose a surgical category and anatomical region from two spinners. The cases/ questions will then be provided accordingly

* Button – Added same way as described in the title screen section. Should be called “Go”. Used to move to the next page once selections have been made.
  1. If it is the spinner for the Speed round, the next page will be the questions page.
  2. If it is the spinner for the medical cases, the next page will be the category info page
* Label – There are two labels on this page provide instructions on what to do with each spinner. Set them up like the label from the title screen, but make the background blue
* Stack view – add in a stack view to contain the labels, spinners, and button
* Picker view
  1. Drag and drop the Picker View from the object library in the bottom right. Position it and place constraints appropriately.
  2. Create a class of type UIViewController and set it to be the custom class of this page.
  3. Open the assistant editor to help you make a reference of the picker view in the code of the class.

    @IBOutlet weak var SurgicalCategoryPicker: UIPickerView!

* 1. Create an array to hold the values you need to appear in the picker view

    var SurgicalCategoryOptions = ["All","Trauma", "Tumour", "Deformity", "Infection", "Degenerative", "Pediatric", "Adult", "Random"]

* 1. Define function to tell the app the number of rows this picker view will hold

func pickerView(\_ pickerView: UIPickerView, numberOfRowsInComponent component: Int) -> Int

in this case, the return value will be the size of the array created in step 4.

* 1. Define function to tell the app the number of columns this picker view will hold

func numberOfComponents(in pickerView: UIPickerView) -> Int

* 1. Define function to tell the app the data to be returned when the picker view is used by the user. In our case, the index of the value in the picker view that they chose will be used to search the array created in step 4.

Questions page – This page will have a question at the very top, and then 4 choices for the user to select the correct answer

* Button – Added same way as described in the Study tab section. Should be called “Next MCQ”. Used to move to the next question once selections have been made
* Labels - There are 4-5 labels on this page. Set them up like the labels from the spinner screen
  1. The first one provides the question and has a blue highlight.
  2. The rest of them are the answer options what to do with each spinner. Set them up like the label from the title screen, but make the background blue
* Stack view – add in 2 stack views to contain the labels
  1. The blue highlighted question label should have 0 spacing between the left and right edges of the screen
  2. The grey highlighted answer options should have some spacing, so they don’t quite reach the side edges of the screen.
* Connection to the database should be documented at a later time.

Category Info Page –

* Stack view – add in 2 stack views to contain the labels
  1. The blue highlighted “A random case has been chosen for you” label should have 0 spacing between the left and right edges of the screen
  2. All the other objects should have some spacing, so they don’t quite reach the side edges of the screen.
* Button – Added same way as described in the study tab section, but also with a blue background. Should be called “BEGIN”.
* The info for this case has been hard coded so far.

Case Info Page –

* Labels - There are 4-5 labels on this page. Set them up like the labels from the spinner screen
  1. Three of them are subheadings, highlighted in blue
  2. Two of them are just info, no background.
* Stack view – add in 2 stack views to contain the labels
  1. The 3 blue highlighted subheading label should have 0 spacing between the left and right edges of the screen
  2. The 2 blue info labels should have some spacing, so they don’t quite reach the side edges of the screen.
* Button –
  1. One button should be called “Continue” Which allows the user to go to the next page, whether it be the MCQ questions of the next prompt. The questions on the page now will only be questions relevant to the case.

A Navigation Controller is placed ahead of the study section of the app in order to allow the user to go back to their previous page at any time. The way it works: for anything you want to lead to the study page, you use a segue from there to the navigation controller. The navigation controller will always segue to the Study Tab.