**Lesson 5 Hands-On48 points**

**Directions**

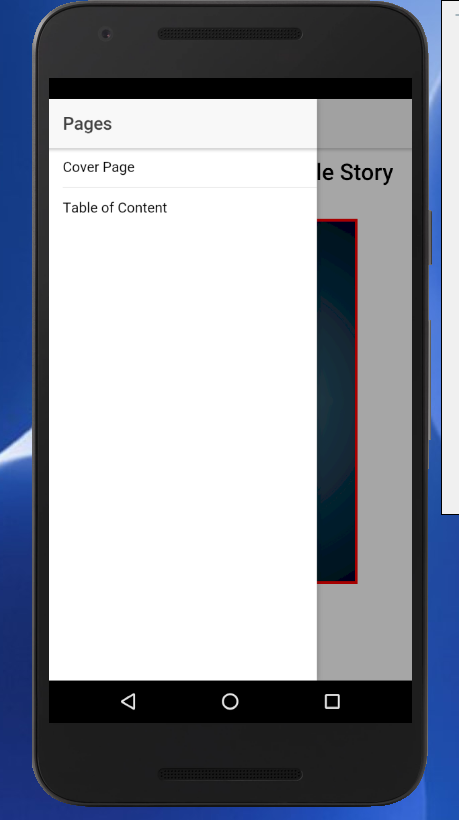
Now that you have learned about Ionic, it is now time to try working with it yourself. Please zip your project directory and upload when you are finished. This Hands-On will be graded, so be sure you complete all requirements.

In this hands-on exercise, you will build a page-turner book app that can be used to display a novel, or a sequence of children's stories, or a comic strip collection, or any book with a simple chapter structure. Follow the directions below and if you have any questions reach out to your instructor/mentor.

1. Build an Apache Cordova app using Ionic that implements a page-turner book app that can be used to display a sequence of content pages in a book. Each content page contains an optional chapter title text, followed by a combination of text and image content, as well as a content page number that starts at 1. Each content page should also provide navigation to the next and previous content pages in the book.

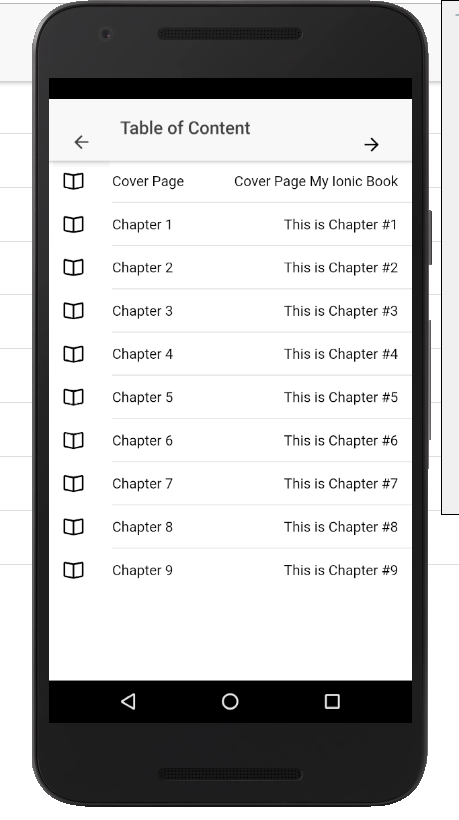


The app turns the pages on the book MyIonic Book – The Whole Story. Above is the cover page. By clicking on the menu at the top left of the screen you will see the following screen.



This menu allows navigation to the Table of Content or the Cover Page. You can also navigate to the Table of Content by clicking on content area of the Cover Page.

1. The first page to be displayed is the book cover page, and the second page is the TOC page (table of contents). After that, there is an arbitrary sequence of content pages that may or may not contain a chapter title text. The TOC page should provide navigation links to each of the content pages that contain a chapter title text, using links that display the chapter title text of those content pages.



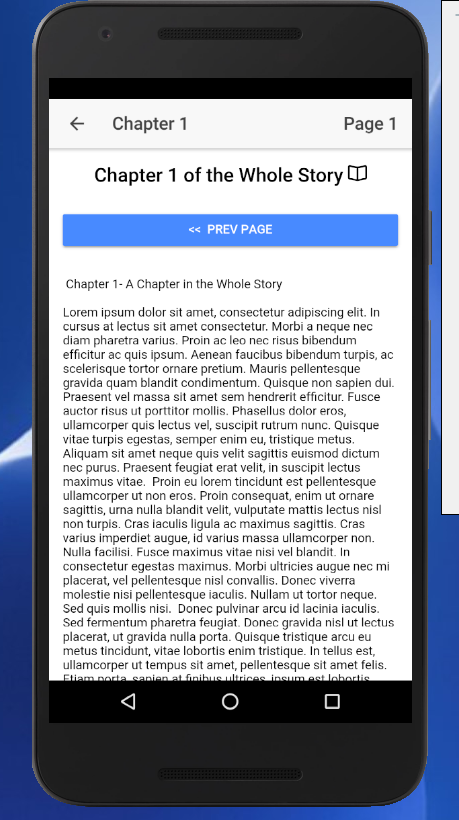
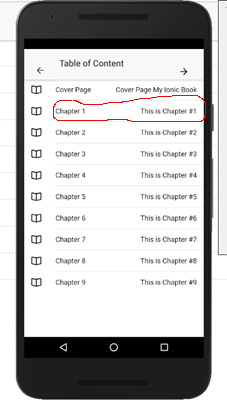
The following Table of Content will be displayed when you click on the menu entry or the Cover Page. By clicking on the Arrow at the top right of the screen you will go to the first page of the book.

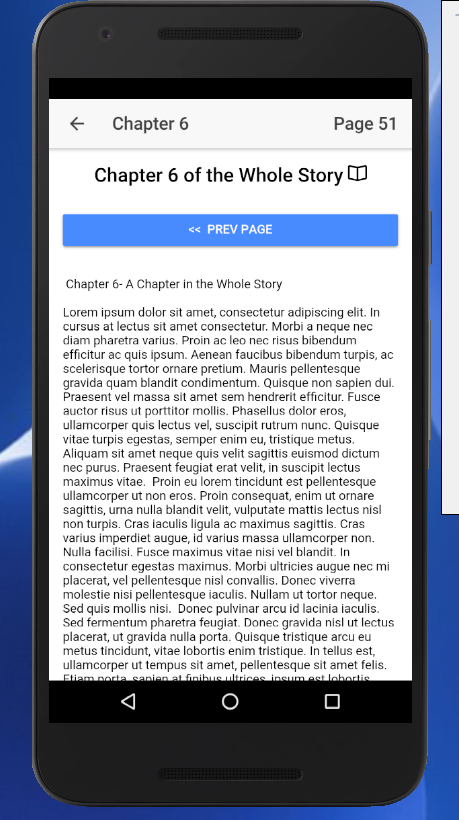
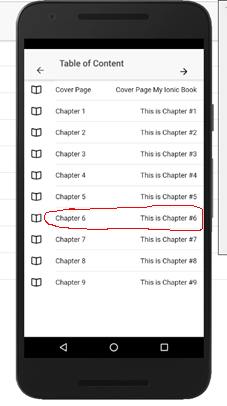
If you click on any of the Chapters you will be taken to the 1st page of that chapter. You can get back to the Cover Page by clicking on the Cover Page TOC entry or clicking on the Arrow at the top left of the page. If you navigated to the TOC by menu you will see a menu icon instead and you can use that to navigate to the Cover Page.

1. The initial screen displayed when you start the app for the first time will be the cover page, which contains the book title text and optionally any additional content, such as an image. Tapping on the cover page should navigate to the TOC page. From there, you can either navigate to the next content page in the usual way, or you can navigate to a particular content page by tapping on the chapter title text listed in the table of contents.

Through the TOC page you can go to the 1st page of any chapter in the book. As covered above you can navigate to the TOC from the Cover Page by tapping on the Cover Page or by the menu at the top left of the screen.

When you click on a chapter it will take you to the first page of that chapter as shown below.





1. In this assignment, you do not have to implement any data persistence. That means that when you start the app again in the future, it does not have to remember what page you were on last. It is OK always to go back to the cover page every time. Storing and retrieving the last known current page as a bookmark would be a nice feature to include, but would add too much development time to this project. However, if you do find that you have time to add that feature, feel free to try it out!
2. Of course, you can use google search and stackoverflow.com, as well as the many available tutorials and videos on the web. You can also use any GitHub repos that you can find that might help you develop your solution. If you find a GitHub example that is very close to these requirements, do try to customize in some interesting ways, because you will want a distinctive code example on your own GitHub account to show potential employers what you can do. However, you are permitted to copy and use any public domain code in your solution that you find in any resources that you may find helpful.
3. Provide the link to your project solution on GitHub to your instructor for verification.

<https://github.com/gojohill/myIonicProject>

The following might help to understand the structure of the book content.

Cover page

* Book title text
* Optionally any additional cover page information, such as an image.
* Next page navigation to the TOC page



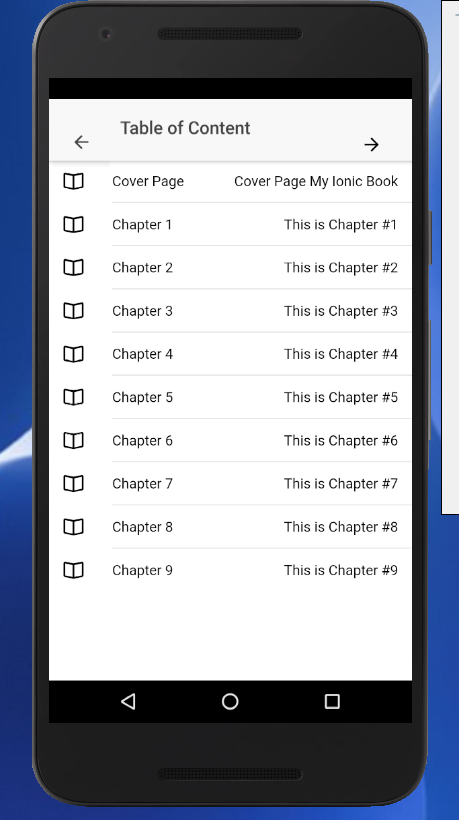
Navigation to TOC by touching screen or through menu

Cover Image and Author

Book Title and Sub Title

TOC page (table of contents)

* Direct navigation links to each of the content pages that contain chapter title text
* Next page navigation to the first content page
* Previous page navigation to the cover page



Clicking on any Chapter will take you to the first page of that chapter.

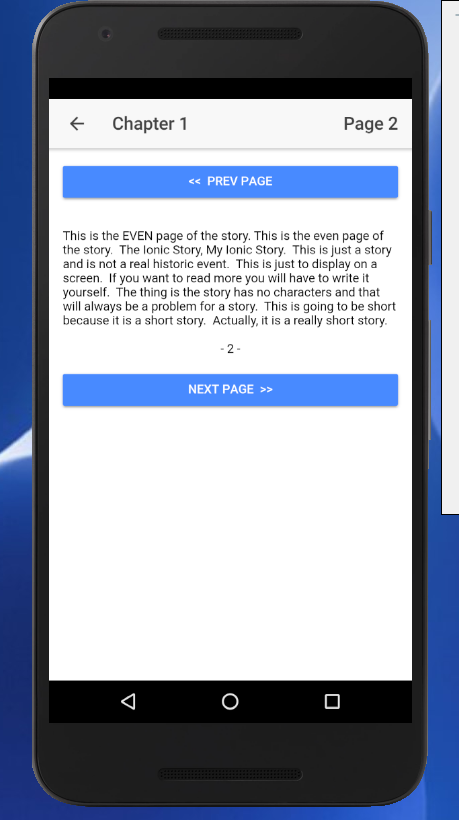
The arrows at the top of the screen will take you back to the Cover Page or on to the first page of the book.

Content page

* Optional chapter title text (only for the first page in a chapter)
* Text and image content
* Content page number
* Direct navigation link to the TOC page
* Next page navigation to the next content page, if there is one
* Previous page navigation to the previous page, if there is one



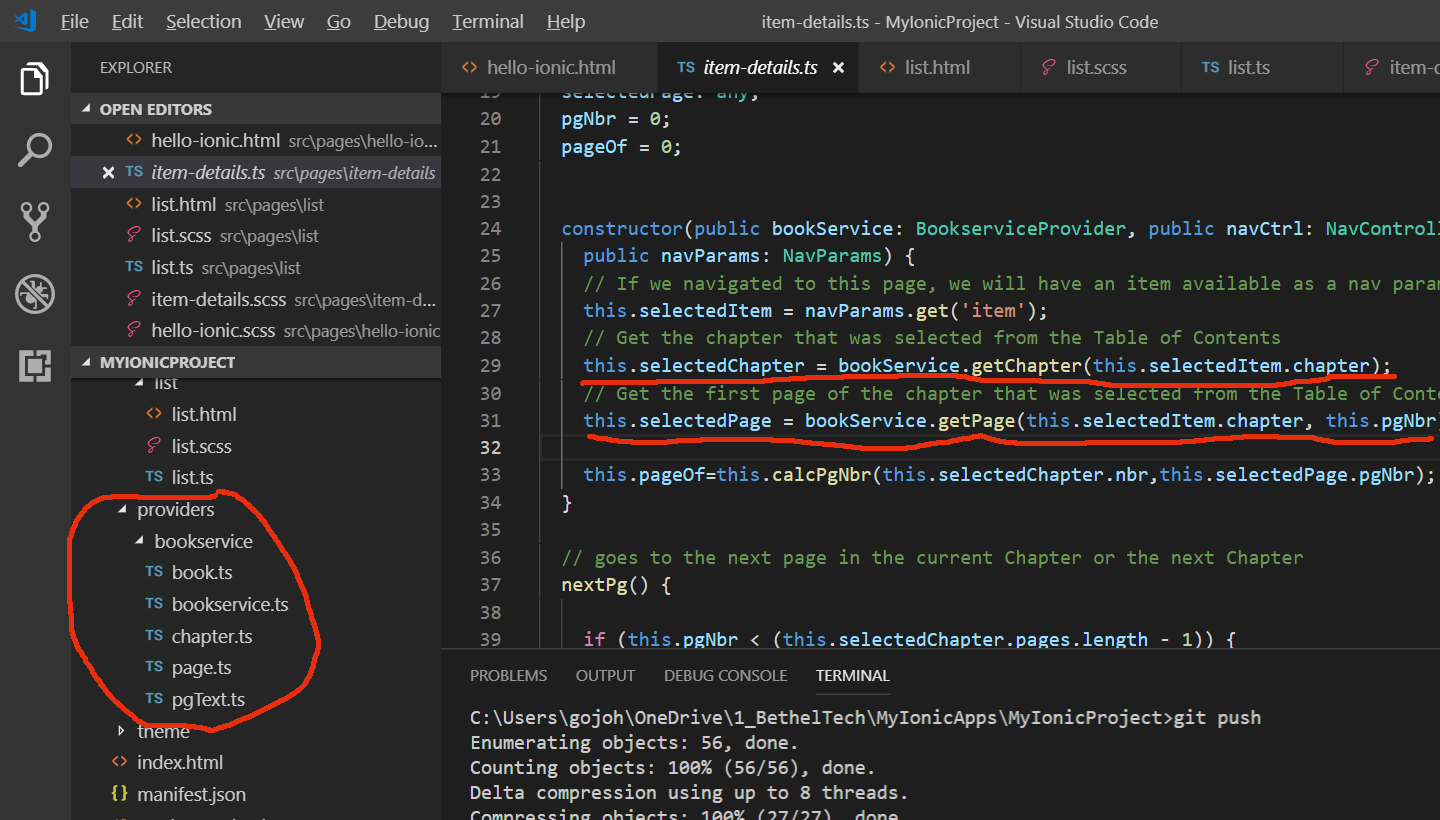
The page will also scroll. By clicking on the next page button you will get the following:



The next page will take you to the next page and eventually the next chapter and so forth. Until you reach the end of the book.

Extra credit:

To implement the book, I used Ionic Provider.



The provider is used to make the book data available throughout the app. The implementation is accomplished through book.ts, chapter.ts and page.ts. pgText.ts is used to populate the book.

The first content page in the book should have chapter title text. All subsequent content pages may or may not have chapter title text.

Don't worry or spend too much time on creating the text and images of the actual content pages. It is most important that you get the page navigation working properly. Just create enough content pages to show that your navigation logic is working correctly.