

# Labwork assignment 2:

## App navigation and Git

### Table of Contents

1	Introduction .....	2
2	App navigation .....	2
3	Git .....	5
4	Learning reflection.....	7
5	Repositories .....	7
6	Literature .....	8

## 1 Introduction

This report follows my progress through the Labwork 2 assignment for the course “Introduction to Mobile App Design and Development R0334-3014”, consisting of introducing application navigation functionality and Git use. The process is following the Labwork 2: UI components and navigation manual, provided in the assignment.

## 2 App navigation

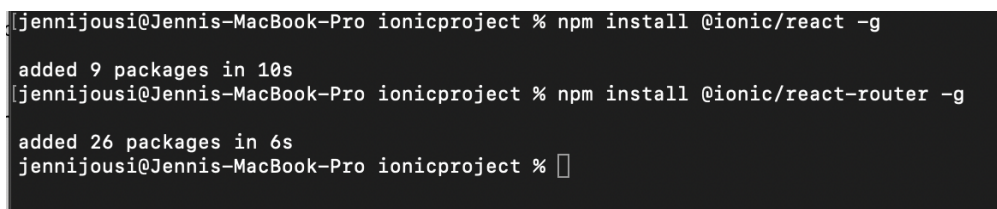
The first step on the manual was to check the versions of NPM, Node and Ionic.

A terminal window titled 'jennijousi - zsh - 124x36' showing the following commands and output:

```
jennijousi@Jennis-MacBook-Pro ~ % npm -v
10.9.3
jennijousi@Jennis-MacBook-Pro ~ % node -v
v22.19.0
jennijousi@Jennis-MacBook-Pro ~ % ionic -v
7.2.1
jennijousi@Jennis-MacBook-Pro ~ %
```

Upon comparing my local versions to the manual, I noticed that I had newer versions in all products. The next step in the manual advised to install Ionic CLI dependencies but the command for this was not provided. I couldn't remember if this was provided in the tutorials I watched earlier so I tried to find out from the provided instructions (Ionic Framework 2025) what this meant: it seemed apparent that I should proceed to the third step and add the Ionic Framework to my already existing React project.

At this point I realised I probably wanted all the installations to go under the project folder I created in the previous assignment, so I ran the installation command in my project folder ionicproject.

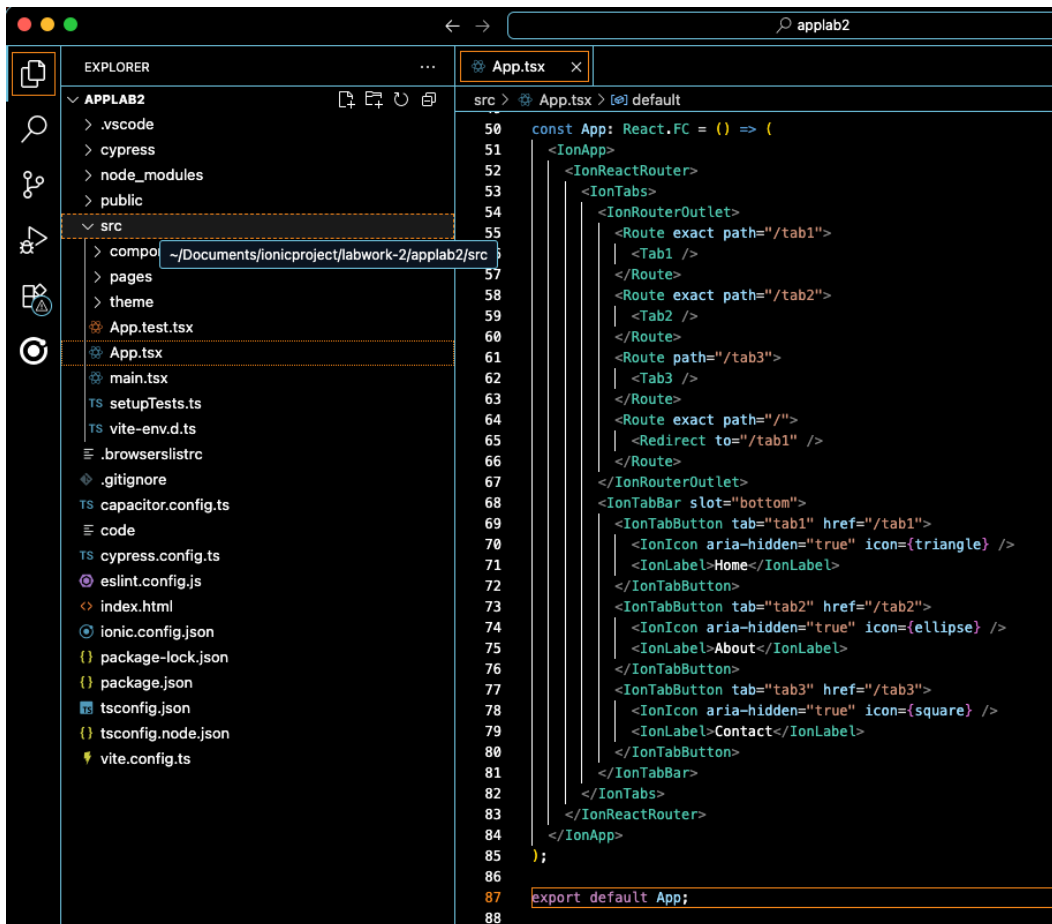
A terminal window showing the following commands and output:

```
jennijousi@Jennis-MacBook-Pro ionicproject % npm install @ionic/react -g
added 9 packages in 10s
jennijousi@Jennis-MacBook-Pro ionicproject % npm install @ionic/react-router -g
added 26 packages in 6s
jennijousi@Jennis-MacBook-Pro ionicproject %
```

I did not receive any warnings. I installed the listed plugins to VS code, the only one I didn't include was Atom one dark theme since I already had a dark theme in my VS code and didn't want to have possible conflicts.

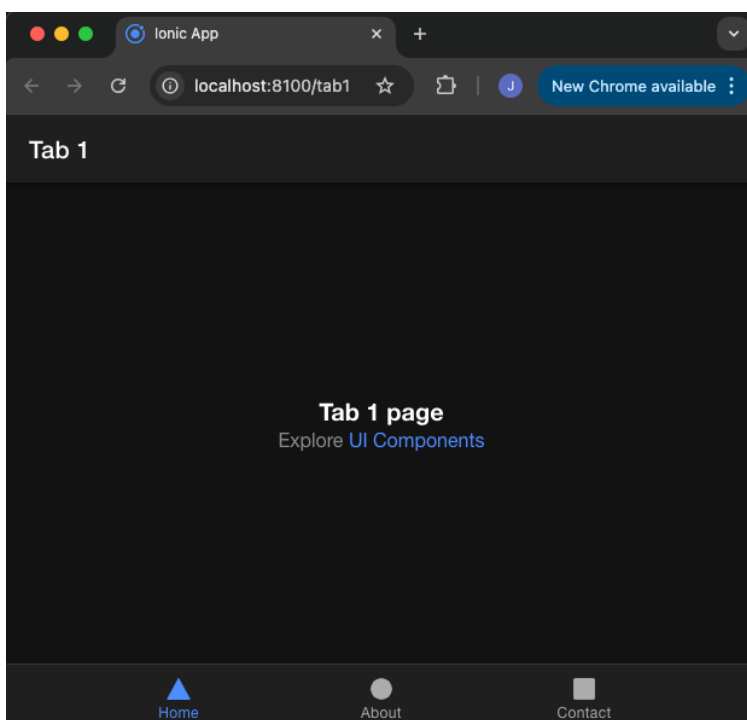
I reviewed the listed Starter commands with the ionic start - - list command and focused on the listed React commands. However, the manual did not advise to use the given tabs command, so I proceeded with the given instructions to launch the project with ionic start command. When reviewing the items on the CLI, I noticed an info text indicating that my previous git project was found - this made me realize I needed to create a new remote repository in order not to lose my Lab1 assignment work.

I entered the applab2-directory and proceeded with the ionic serve command. I skipped step 5 in the manual as I did not previously download any Android Studio related software and decided to proceed with the required steps. I opened the project folder in Visual Studio code and proceeded to view the main.tsx file. I made the required tabname changes in the app.tsx file and launched the app with the ionic service command.



```
50 const App: React.FC = () => {
51   <IonApp>
52     <IonReactRouter>
53       <IonTabs>
54         <IonRouterOutlet>
55           <Route exact path="/tab1">
56             <Tab1 />
57           </Route>
58           <Route exact path="/tab2">
59             <Tab2 />
60           </Route>
61           <Route path="/tab3">
62             <Tab3 />
63           </Route>
64           <Route exact path="/">
65             <Redirect to="/tab1" />
66           </Route>
67         </IonRouterOutlet>
68         <IonTabBar slot="bottom">
69           <IonTabButton tab="tab1" href="/tab1">
70             <IonIcon aria-hidden="true" icon={triangle} />
71             <IonLabel>Home</IonLabel>
72           </IonTabButton>
73           <IonTabButton tab="tab2" href="/tab2">
74             <IonIcon aria-hidden="true" icon={ellipse} />
75             <IonLabel>About</IonLabel>
76           </IonTabButton>
77           <IonTabButton tab="tab3" href="/tab3">
78             <IonIcon aria-hidden="true" icon={square} />
79             <IonLabel>Contact</IonLabel>
80           </IonTabButton>
81         </IonTabBar>
82       </IonTabs>
83     </IonReactRouter>
84   </IonApp>
85 };
86
87 export default App;
88
```

The newly changed tab names were visible in the application UI, as indicated in the below screenshot.

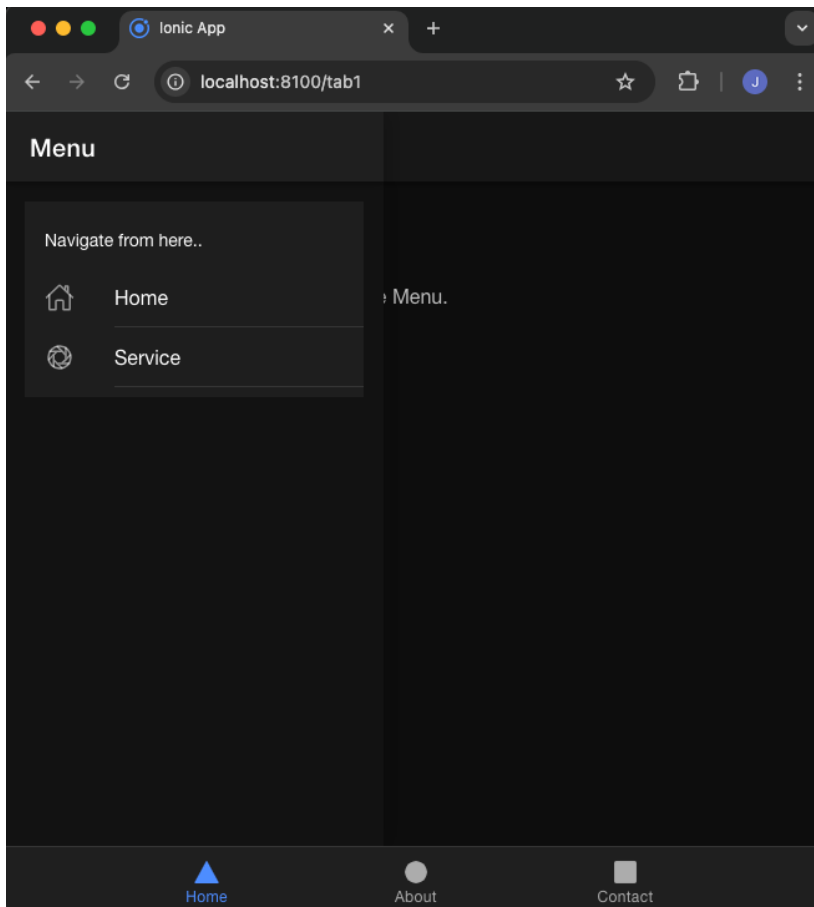


Then I proceeded to create the Service.tsx file. I copied the library import from Tab1.tsx file and changed the annotations according to the manual.

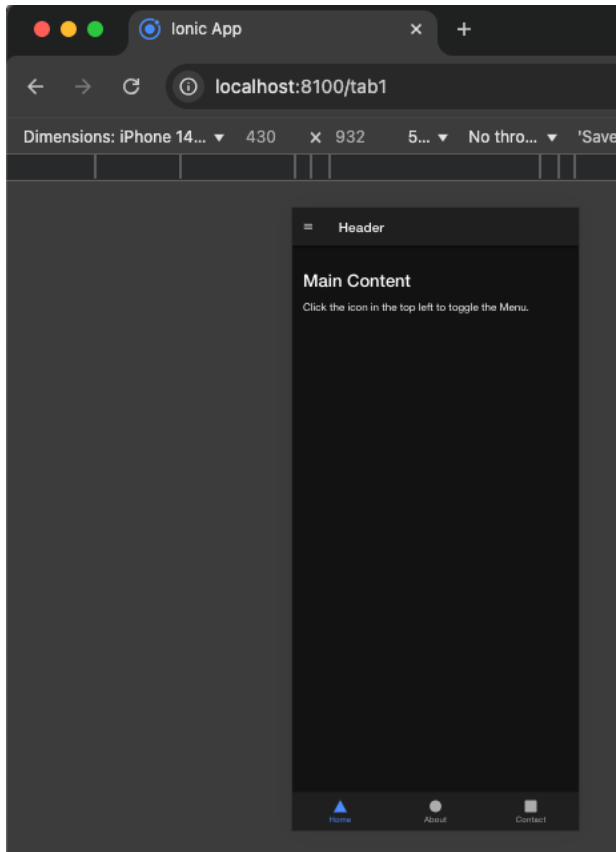
I struggled a lot with Action 4, as the file indicated in the manual differed greatly from the few items in my file. At this point, I asked ChatGPT some guidance on the strategy going forward; would I need to recreate all items indicated in the teacher's file, having over 40 lines more than my file, or would it be sufficient to only add items that were referring to the added Service and Menu items. While ChatGPT indicated that it would be enough to just add the needed references (and ignore the dozens of other references missing from my file) I decided to make a Git commit in order to be able to make a rollback to the previous working version if necessary - the changes required to my files were so drastic and not indicated in the manual.

At this stage I disconnected the Labwork 1 remote repository from git, created a new remote, and made a local commit of the changes so far. This way, I could easily trash any changes that would cause my app to break, as I couldn't be sure which items were necessary in order for the app to be able to run. Then I proceeded to add the imports and elements that were missing from my file, I found the syntax highlight to be particularly useful here, as I had missed some imports and syntax error was able to point that out before I even tried to run the app. I also commented out some import "ExploreContainer" that received a syntax error since it was not being used. I then added the route path to App.tsx and edited the titles and texts of tab2.tsx and tab3.tsx files.

When running the app with ionic serve command, the app proceeded to run in my browser, with a Menu bar indicated on the Home page and About and Contact pages showing the titles as edited.



Since I didn't have any Android or iOS device virtualized for this assignment, I checked with Chrome developer tools how the application would look in a mobile device; I used the given Android and iPhone devices listed in the options.



I made a git commit on the changes and pushed them to the new remote repository I made earlier.

The repository that contains the work regarding app navigation is:

<https://github.com/jemajii/applab2>

### 3 Git

I realised that I had probably misunderstood the requirements for having a git repository returned already for Labwork 1. When creating that repository I thought that we would be using the same repository and just add assignment directories under that same repository, so now I need to revise my strategy regarding Git.

I did check my git version, even though I have already used git for both labwork assignments. My version was 2.37.1 (Apple Git-137.1). Before starting a new Ionic app, I realised I needed to switch away from my applab2 directory in order not to overwrite my work. I read the whole assignment through and decided I needed a new repository for GitDemo. I created the following repository:

<https://github.com/jemajii/GitDemo>

In order to create some files to push to the remote repository I followed the instructions with the command `ionic start GitDemo tabs - - type react`, then proceeded to my new directory and ran `ionic serve`.

Your Ionic app is ready! Follow these next steps:

- Go to your new project: `cd ./GitDemo`
- Run `ionic serve` within the app directory to see your app in the browser
- Run `ionic capacitor add` to add a native iOS or Android project using Capacitor
- Generate your app icon and splash screens using `cordova-res --skip-config --copy`
- Explore the Ionic docs for components, tutorials, and more: <https://ion.link/docs>
- Building an enterprise app? Ionic has Enterprise Support and Features: <https://ion.link/enterprise-edition>

```
jennijousi@Jennis-MacBook-Pro ionicproject % cd GitDemo
```

```
jennijousi@Jennis-MacBook-Pro GitDemo % ionic serve
```

```
> vite --host=localhost --port=8100
```

```
[vite] → Local: http://localhost:8100/
```

```
[vite] → press h + enter to show help
```

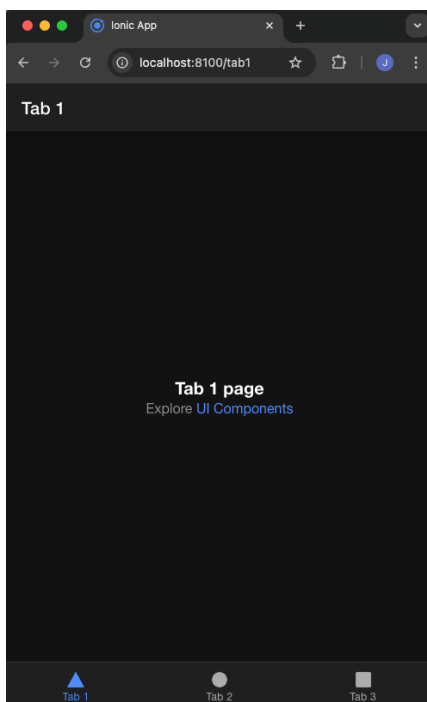
```
[INFO] Development server running!
```

```
Local: http://localhost:8100
```

```
Use Ctrl+C to quit this process
```

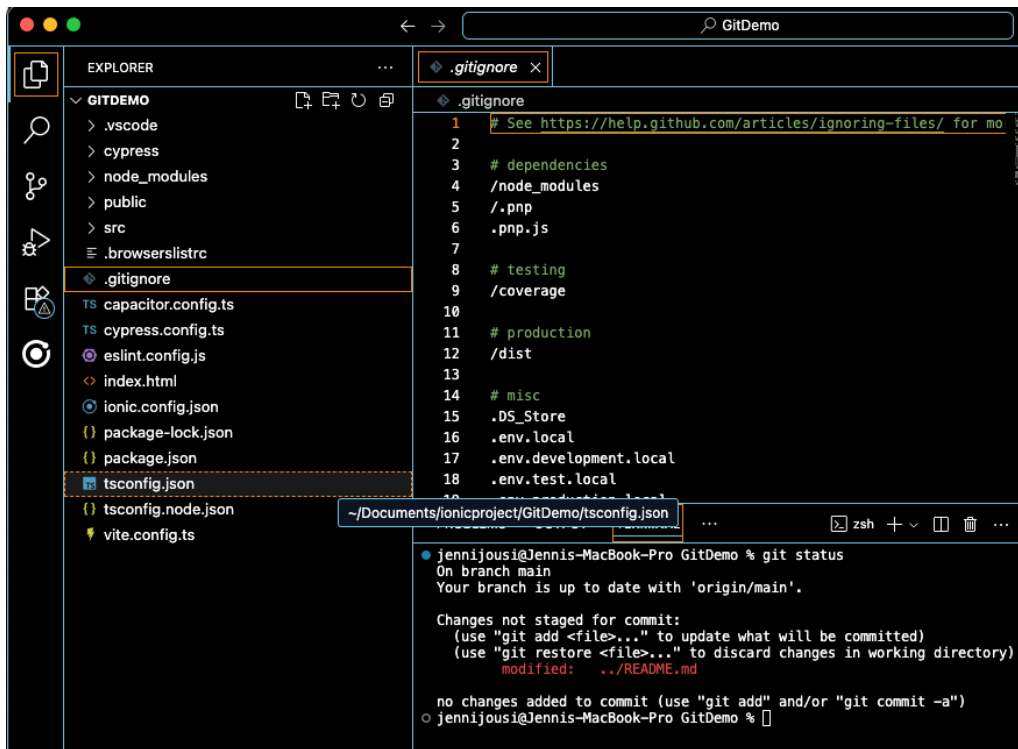
```
[INFO] Browser window opened to http://localhost:8100!
```

This launched the app in my browser.



Since I had already created my remote repository, I proceeded to commit my local changes and push them into my remote. I had also created username and tokens already with Labwork 1, so I could skip all authentication parts of the instructions.

When connecting git with VS code, I opened the GitDemo directory in VS code and Git was already connected without any configuration necessary.



I did not use the git clone command now, since I didn't need any repository locally on my computer. I added readme.md files to applab2 repository (not to the GitDemo repository since I believe it doesn't require one).

#### 4 Learning reflection

##### Ionic Framework

This week's assignment showed some tricky parts as I needed to add elements on the UI that I didn't know how to do or what dependencies they might have in Ionic Framework. I try to avoid doing assignments in a way where I only copy lines that I don't understand, but now I must admit that I don't fully understand how the application menu was configured. I probably couldn't be able to develop it again without instructions, but I do understand the basic principles of including a new page, routing it and showing it only on the Menu (not in the footer bar).

##### Git

As I am somewhat familiar to Git and already used it in Labwork 1 while solving all issues independently, this assignment wasn't challenging to me. The only issue might be that since we are using multiple repositories throughout this course, it will create many danger zones when not remembering which repository we are pushing into. However, the reason why git is so widely used is most definitely the possibility to roll back any unwanted changes.

However, I did run into some Git issues when examining my repositories in GitHub. Since I created my repository for Labwork 1, I forgot to disconnect Git from the root directory - the parent directory for the directories I created for this assignment. So I did end up creating a completely new repository for the app navigation work. That is why git history is showing different commit history than what I described here. I did learn a lot from this mess and fixing the issue increased my Git competence in many ways.

#### 5 Repositories

Labwork2 - appwithionic: Application navigation development

<https://github.com/jemajii/applab2>

GitDemo - a local directory pushed into a remote repository

<https://github.com/jemajii/GitDemo>

## 6 Literature

Ionic Framework. Ionic packages. Referenced 28.9. 2025. <https://ionicframework.com/docs/intro/cdn>