



ROI project proposal report

Red Opal Innovations

Shuowei Li

Part 1: Determine user needs

1. Identify the client business area that this project relates to.

Red Opal Innovations (short for ROI) is an information technology company , ROI do consultancy services for IT and network business and design and development of websites, multimedia products and customised applications.

2. Determine the user needs and requirements for both the web and mobile applications, including the interface, system environment and functionality.

ROI HR needs an application to manage contact information of staff members.

Interface: it need to be tidy , easy and clear, and there are menus to direct users to the procedure they need;

multimedia products: include phone , iPad and computer;

customised applications: the applicant contain employees information and including retrieving, inserting, updating and deleting records.

3. Determine the impact that the system environment, for example mobile, tablet or desktop, will have on the user interface design for each.

The main difference that influences the UI design of mobile, tablet and desktop apps is screen size, and desktop is only landscape orientation, and mobile and tablet are suit for both portrait or landscape orientation. . All functional buttons, graphics, and text must change to fit the screen. UI design is also affected by how users interact with devices.

The ability of an app to perform in a user-friendly way no matter what the circumstances are important to overall satisfaction with the app. With phones, we might think less of this issue, because with less screen space there are fewer decisions to make, but on a tablet, this issue deserves some consideration. There is a big difference between landscape and portrait views, Many developers have found that tablets on portrait can follow the design and UI of phones, while tablet landscape can arrange content like a desktop and still give a great user experience. This way, the tablet is neither phone or desktop, but a convenient and user-centric breakpoint between the two.

Accordingly, Take into account the location of buttons and primary and secondary content when designing mobile, tablet and desktop products.

Application layouts should also be different. Due to the smaller screen sizes, smartphones require designers to put only the most important content on the screen. Secondary information can be divided into blocks and across several screens. But desktop apps are slightly different. Thanks to the larger screen size, more elements can be placed on one screen without harming the application's interface. And tablet will be in the middle of the two kind of device.

In the other hand, A tablet application interface shouldn't copy and scale the mobile application interface.

But in order to provide the best experience to users, I will develop separate interfaces for tablet applications, mobile applications and desktop applications while maintaining a consistent UI design. We provide some useful tips below.

4. Determine the target platform for the mobile devices to be used with the application.

Based on current market research, IOS, Android and Linux are the most popular platform for mobile devices, and because of the application is used by all the staff of ROI, not only the HR office, we decide to choose IOS and Android as our target platform, and if we still have budget left, we will consider to add Linux platform as well.

5. Based on the target platform and hardware, choose and evaluate a development environment to use (Note: this may be different for each platform):

- a. Research suitable options for a development environment.
- b. Evaluate the options, using suitable criteria that will allow you to decide between them, including their ease of use.
- c. Select the development environment to use, justifying your selection.

React native VS Xamarin , [ASP.NET](#) web form.

I will choose React native, as React native could be used both for IOS, Android and web, and this will save time and money.

6. Review the organisational goals in the scenario and discuss how you will ensure that your interface design will be consistent with the organisational styles. Discuss the implications of ensuring that they are followed.

I will ensure that all the device that staff use in ROI could access this application easily and simply, the UI will be clear and clean to make sure it is easy to learn how to deal with it. Staff will be clear how to update and branded by being lead with menus.

And of course I will use the Accurate logo en match the whole organisation.

7. Determine and outline the formats that are required for each application interface for:

- d. querying the data (inputs)
- e. displaying reports (outputs).

Mobile: querying the data: REST API get, post , put and delete
displaying reports: JSON payload.

Tablet: querying the data: REST API get, post , put and delete
displaying reports: JSON payload.

Desktop: querying the data: EST API get, post , put and delete
displaying reports: JSON payload.

8. Include a bibliography of the documents and resources you accessed and used.

Xamarin: <https://dotnet.microsoft.com/apps/xamarin/mobile-apps>

How to Make an Android and iOS App in C# on a Mac: <https://www.toptal.com/c-sharp/how-to-make-an-android-and-ios-app-in-c-on-a-mac>

11 Popular Cross-Platform Tools for App Development: <https://hackernoon.com/9-popular-cross-platform-tools-for-app-development-in-2019-53765004761b>

How to Choose the Right Platform for Mobile App Development: <https://steelkiwi.com/blog/how-to-choose-the-right-platform-for-mobile-app-development/>

Target Platform: <https://www.sciencedirect.com/topics/computer-science/target-platform>

User Interface Design Adaptation: <https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/user-interface-design-adaptation>

Designing for Different Screens and Devices: 7 Steps to Creating A Great UX: <https://blog.adobe.com/en/publish/2017/06/01/designing-for-different-screens-and-devices-7-steps-to-creating-a-great-ux#gs.gc33d4>

9. Include space for name and signature of the client.

Name	Position/role	Date	Signature

Part 3

1.

There will be “Home page”, “adding page” and “editing pages”, for the home page information, contents are “Name”, “phone”, “Department”, “Street”, “City”, “State”, “Zip” and “Country”.

One line per employee’s information, and the occupied space ratio respectively are 10%, 10%, 15%, 15%, 5%, 10%, 5%, 5%. The last option of every line are two buttons, which are edit and delete. Which HR could press these two button to update employee’s information or delete the whole employee’s information.

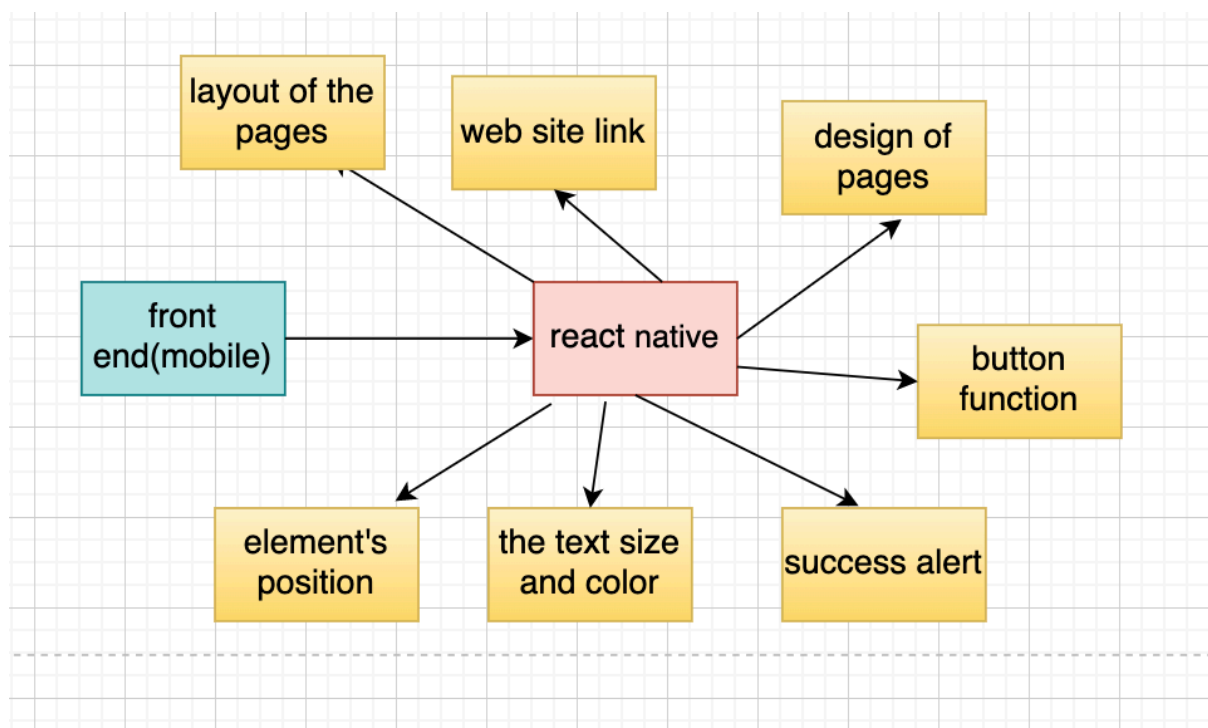
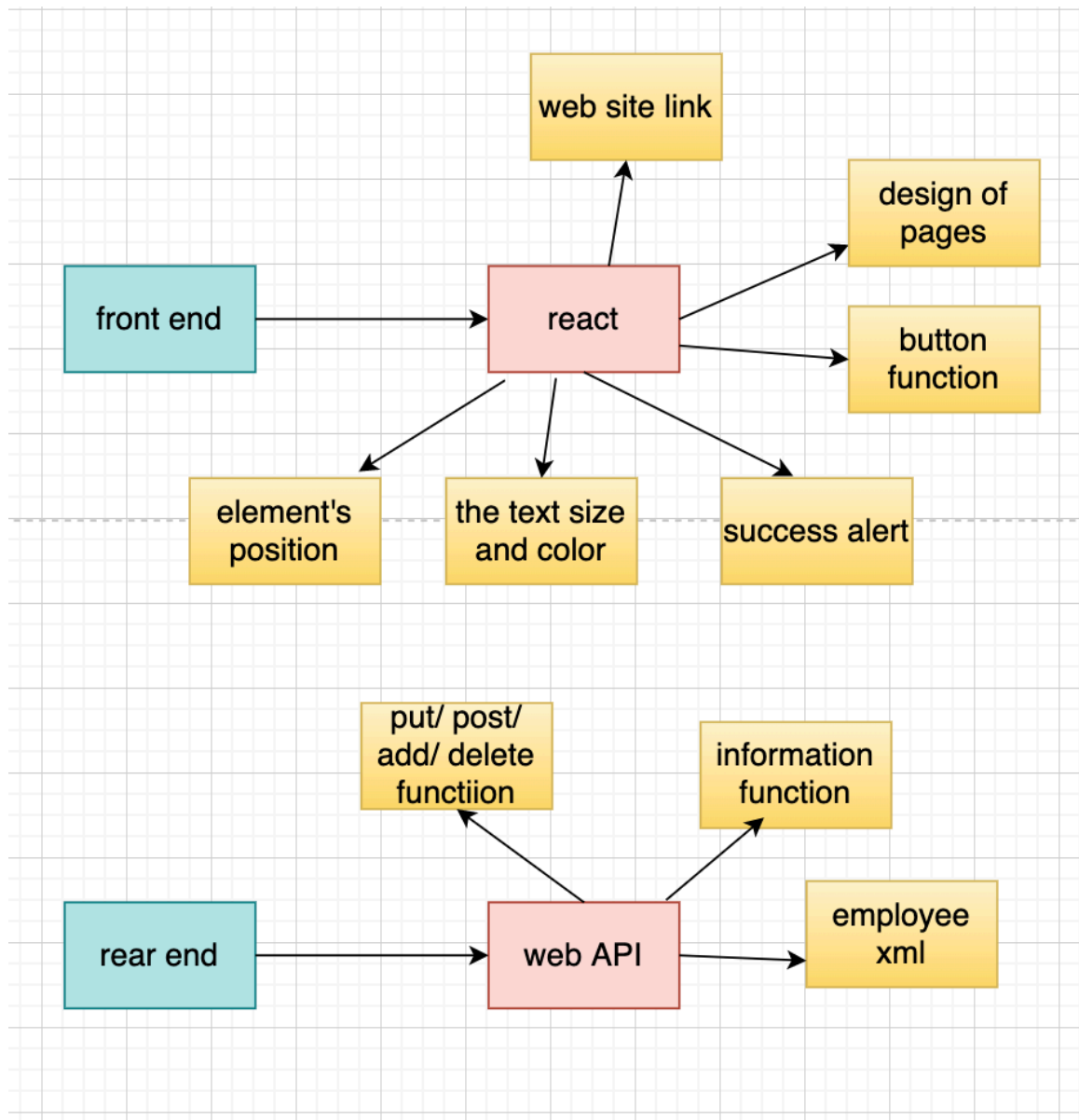
In the top of the home page, there is a green button named “Add Employee”, HR could press this button to add new employee’s information.

For adding page, there are “Add Employee” topic in the top of the page, then two lines per information of this employee, there are “save” and “cancel” button in the end of the page.

For adding page, there are “Edit Employee” topic in the top of the page, then two lines per information of this employee, there are “save” and “cancel” button in the end of the page.

2.

Overpage



3.

Retrieving: When you press the app, you will see the home page, which showed all the employee's information.

Inserting: if you want to add new employee, press "Add Employee" button which is located on top of home page, add new employee's information in the adding page, press "save" button after you fill the form, then, you could see "success! ", Means add new employee information successfully, and you could press "cancel" button to cancel adding.

Updating: When you press "Edit" button, it will link to edit page, you could edit the information in this page, after edit press "save", when you see "success!" Means update information successfully, and of course you could press "cancel" button to cancel editing.

Deleting: If you would like to delete an employee's information, press "delete" directly, and when users see "success!" means delete information successfully.

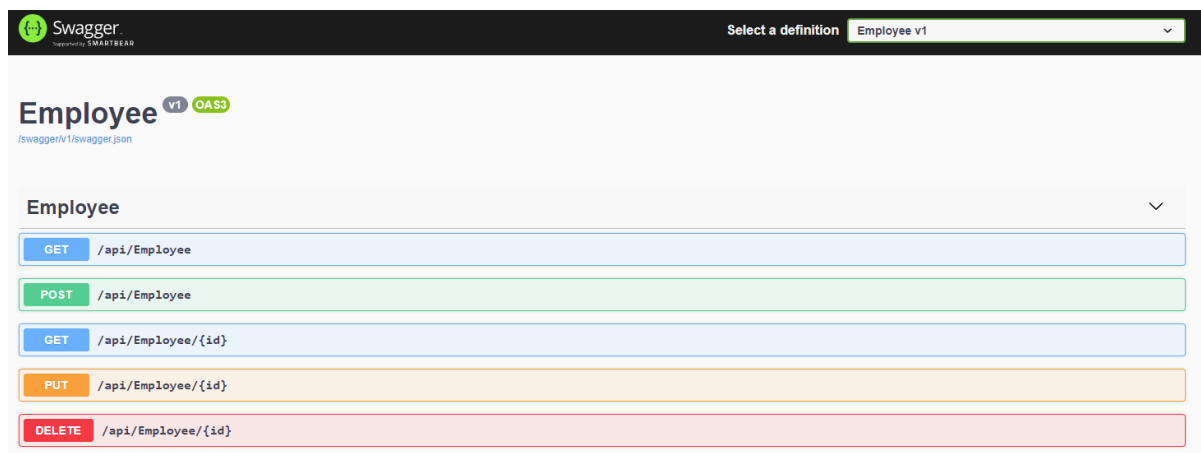
4.

Add Employee

Help

Employees

Name	Phone	Department	Street	City	State	Zip	Country	Actions
Employee 1's information								<div>EditDelete</div>
Employee 2's information								<div>EditDelete</div>
Employee 3's information								<div>EditDelete</div>
Employee 4's information								<div>EditDelete</div>



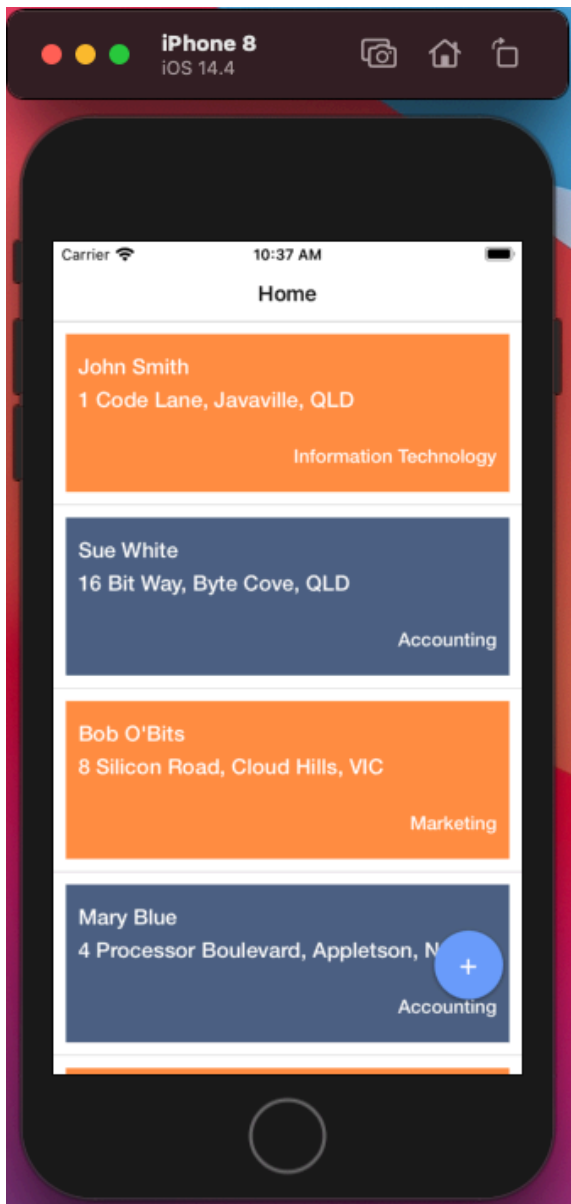
5.

This is a project for storing employee's information, HR could use this project to get all employees information to check, insert new employee's information, update employee's information, and delete employee's information. It is easy and clear to use, just simply follow the menu, and operate the app.

Name	Position/role	Date	Signature

Part 6:

1.



2.

Home

Edit Employee

Name

Phone

Department

Address: Street

Address: City

Address: State

Address: Zip

Address: Country

6.

1) Test Retrieving:

Enter into home page:

Expected result: there are all the 5 employee's information

Real result:

Add Employee
Employees

Name	Phone	Department	Street	City	State	Zip	Country	Actions
John Smith	0299882211	Information Technology	1 Code Lane	Javaville	0299882211	0100	Australia	Edit Delete
Sue White	0388992255	Accounting	16 Bit Way	Byte Cove	QLD	1101	Australia	Edit Delete
Bob O'Bits	0577882255	Marketing	8 Silicon Road	Cloud Hills	VIC	1001	Australia	Edit Delete
Mary Blue	0644559988	Accounting	4 Processor Boulevard	Appletson	0644559988	1010	Australia	Edit Delete
Mick Green	0299881122	Marketing	700 Bandwidth Street	Bufferland	NSW	0110	Australia	Edit Delete

Add Employee

Name

Phone

Department

Address: Street

Address: City

Address: State

Address: Zip

Address: Country

Test result: pass

2) Test Inserting:

Add new employee's information: Name: Lily. Phone: 01234567. Department: Cloud Bee. Street: 3 geez street. City: Sydney. State: NSW. Zip: 2123. Country: Australia.

Expected result: add lily sussessfully

Real result:

Employees

Name	Phone	Department	Street	City	State	Zip	Country	Actions
John Smith	0299882211	Information Technology	1 Code Lane	Javaville	0299882211	0100	Australia	Edit Delete
Sue White	0388992255	Accounting	16 Bit Way	Byte Cove	QLD	1101	Australia	Edit Delete
Bob O'Bits	0577882255	Marketing	8 Silicon Road	Cloud Hills	VIC	1001	Australia	Edit Delete
Mary Blue	0644559988	Accounting	4 Processor Boulevard	Appletson	0644559988	1010	Australia	Edit Delete
Mick Green	0299881122	Marketing	700 Bandwidth Street	Bufferland	NSW	0110	Australia	Edit Delete
Lily	01234567	Cloud Bee	3 Geez street	sydney	NSW	2123	Australia	Edit Delete

The screenshot shows a web browser window with the address bar displaying 'localhost:3000/employees/6'. The page title is 'Edit Employee'. The form contains the following fields:

- Name: Barb
- Phone: 01234567
- Department: Cloud Bee
- Address: Street: 3 Geez street
- Address: City: sydney
- Address: State: 01234567
- Address: Zip: 2123
- Address: Country: Australia

At the bottom of the form are 'Save' and 'Cancel' buttons. A success message dialog box is open, displaying 'localhost:3000 says Success!' with an 'OK' button.

Test result: pass

3) Test Updating

Update Lily's information: name lily => Barb

Expected result: Lily's name changed into Barb

Real result:

Employees

Name	Phone	Department
John Smith	0299882211	Information Technology
Sue White	0388992255	Accounting
Bob O'Bits	0577882255	Marketing
Mary Blue	0644559988	Accounting
Mick Green	0299881122	Marketing
Barb	01234567	Cloud Bee

Test result: pass

4) Test Deleting

Delete Barb’s information.

Expected result: Barb’s information be deleted.

Real result:

← → ↻ localhost:3000/employees

Home

localhost:3000 says
Delete success.

OK

Add Employee

Employees

Name	Phone	Department	Street	City	State	Zip	Country	Actions
John Smith	0299882211	Information Technology	1 Code Lane	Javaville	0299882211	0100	Australia	<a>Edit <a>Delete
Sue White	0388992255	Accounting	16 Bit Way	Byte Cove	QLD	1101	Australia	<a>Edit <a>Delete
Bob O'Bits	0577882255	Marketing	8 Silicon Road	Cloud Hills	VIC	1001	Australia	<a>Edit <a>Delete
Mary Blue	0644559988	Accounting	4 Processor Boulevard	Appletson	0644559988	1010	Australia	<a>Edit <a>Delete
Mick Green	0299881122	Marketing	700 Bandwidth Street	Bufferland	NSW	0110	Australia	<a>Edit <a>Delete
lily	012343254	ahkfhdk	bfahkfa	dankjfhkdaj	afjd	0123	fdjafh	<a>Edit <a>Delete

Add Employee

Employees

Name	Phone	Department	Street	City	State	Zip	Country	Actions
John Smith	0299882211	Information Technology	1 Code Lane	Javaville	0299882211	0100	Australia	<a>Edit <a>Delete
Sue White	0388992255	Accounting	16 Bit Way	Byte Cove	QLD	1101	Australia	<a>Edit <a>Delete
Bob O'Bits	0577882255	Marketing	8 Silicon Road	Cloud Hills	VIC	1001	Australia	<a>Edit <a>Delete
Mary Blue	0644559988	Accounting	4 Processor Boulevard	Appletson	0644559988	1010	Australia	<a>Edit <a>Delete
Mick Green	0299881122	Marketing	700 Bandwidth Street	Bufferland	NSW	0110	Australia	<a>Edit <a>Delete



Test result: pass

Part 5

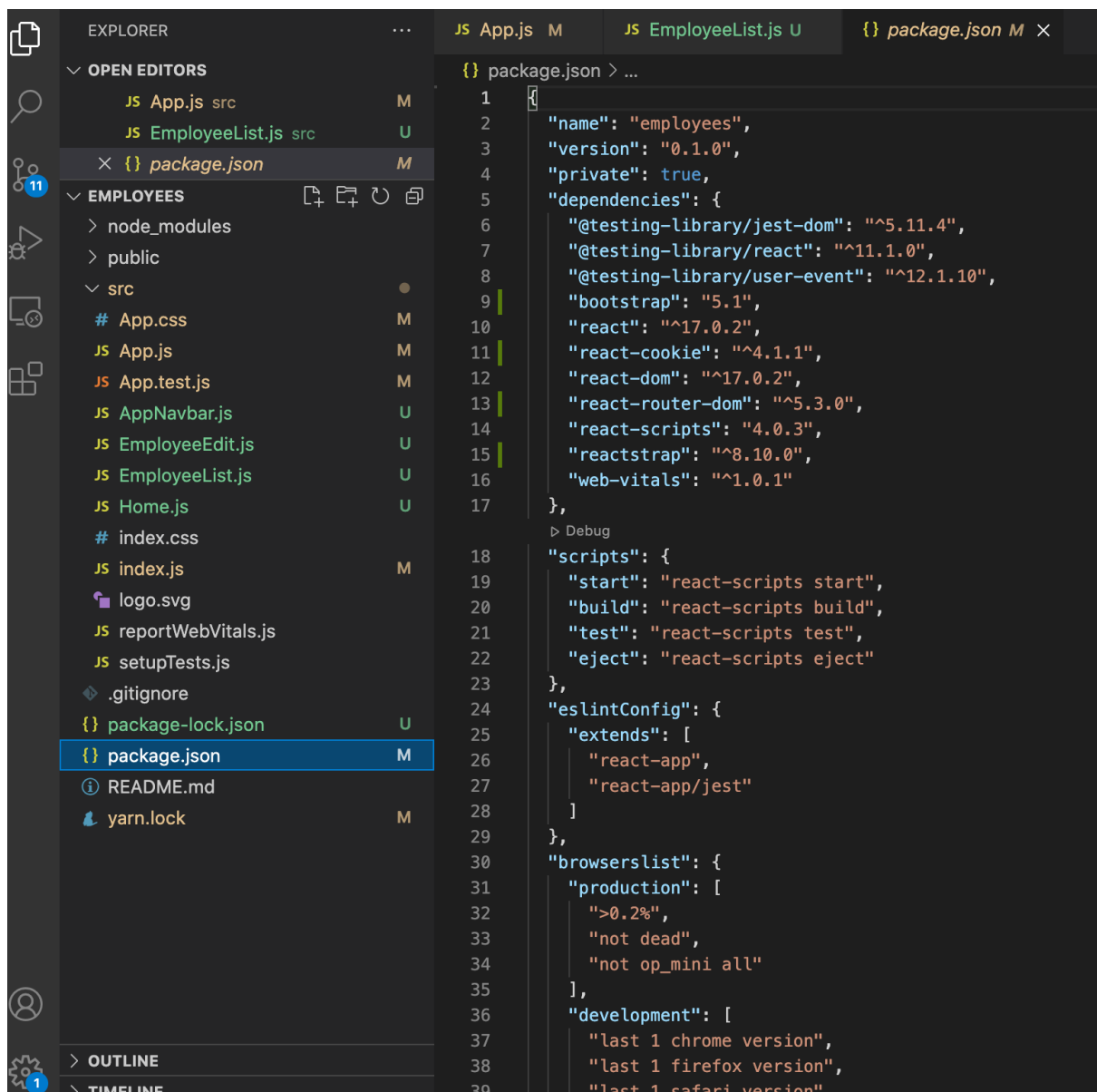
2.

```
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
INFO:werkzeug: * Running on http://127.0.0.1:8080/ (Press CTRL+C to quit)
INFO:werkzeug: * Restarting with stat
WARNING:werkzeug: * Debugger is active!
INFO:werkzeug: * Debugger PIN: 293-325-225
```

Part 6

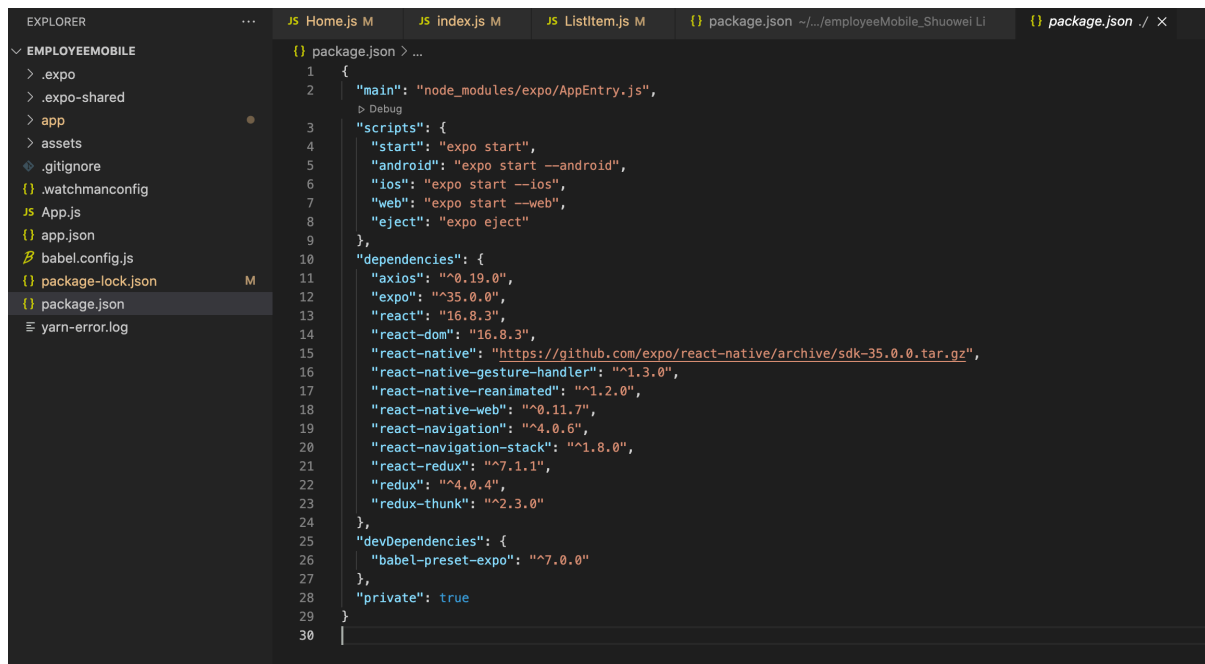
1. The environment is react and react native

2. The react web application



```
1 {
2   "name": "employees",
3   "version": "0.1.0",
4   "private": true,
5   "dependencies": {
6     "@testing-library/jest-dom": "^5.11.4",
7     "@testing-library/react": "^11.1.0",
8     "@testing-library/user-event": "^12.1.10",
9     "bootstrap": "5.1",
10    "react": "^17.0.2",
11    "react-cookie": "^4.1.1",
12    "react-dom": "^17.0.2",
13    "react-router-dom": "^5.3.0",
14    "react-scripts": "4.0.3",
15    "reactstrap": "^8.10.0",
16    "web-vitals": "^1.0.1"
17  },
18  "scripts": {
19    "start": "react-scripts start",
20    "build": "react-scripts build",
21    "test": "react-scripts test",
22    "eject": "react-scripts eject"
23  },
24  "eslintConfig": {
25    "extends": [
26      "react-app",
27      "react-app/jest"
28    ]
29  },
30  "browserslist": {
31    "production": [
32      ">0.2%",
33      "not dead",
34      "not op_mini all"
35    ],
36    "development": [
37      "last 1 chrome version",
38      "last 1 firefox version",
39      "last 1 safari version"
40    ]
41  }
42 }
```

React native mobil application



The screenshot shows a Visual Studio Code editor with the Explorer sidebar on the left and the package.json file open in the main editor. The Explorer sidebar shows the project structure for 'EMPLOYEEEMOBILE', including files like .expo, .expo-shared, app, assets, .gitignore, .watchmanconfig, App.js, app.json, babel.config.js, package-lock.json, package.json, and yarn-error.log. The package.json file is selected and its content is displayed in the main editor. The package.json file is a standard JSON object with the following structure:

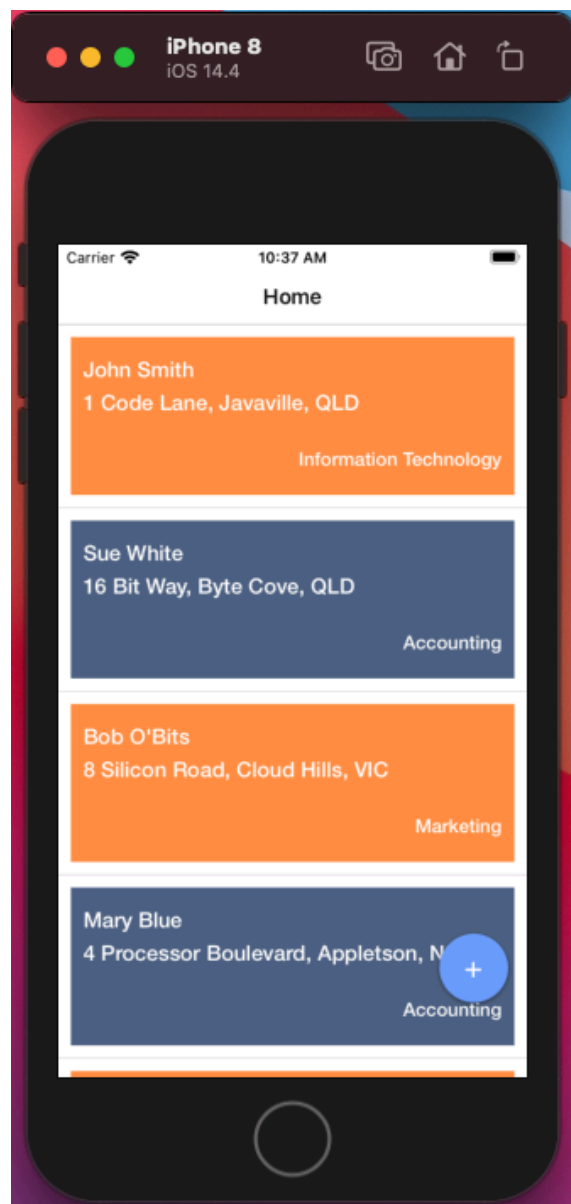
```
{
  "main": "node_modules/expo/AppEntry.js",
  "scripts": {
    "start": "expo start",
    "android": "expo start --android",
    "ios": "expo start --ios",
    "web": "expo start --web",
    "eject": "expo eject"
  },
  "dependencies": {
    "axios": "^0.19.0",
    "expo": "^35.0.0",
    "react": "16.8.3",
    "react-dom": "16.8.3",
    "react-native": "https://github.com/expo/react-native/archive/sdk-35.0.0.tar.gz",
    "react-native-gesture-handler": "^1.3.0",
    "react-native-reanimated": "^1.2.0",
    "react-native-web": "^0.11.7",
    "react-navigation": "^4.0.6",
    "react-navigation-stack": "^1.8.0",
    "react-redux": "^7.1.1",
    "redux": "^4.0.4",
    "redux-thunk": "^2.3.0"
  },
  "devDependencies": {
    "babel-preset-expo": "^7.0.0"
  },
  "private": true
}
```

6

Get: when we press app, then we could see all the 5 employees' information

Expected result: view all 5 employees' information

Real result:



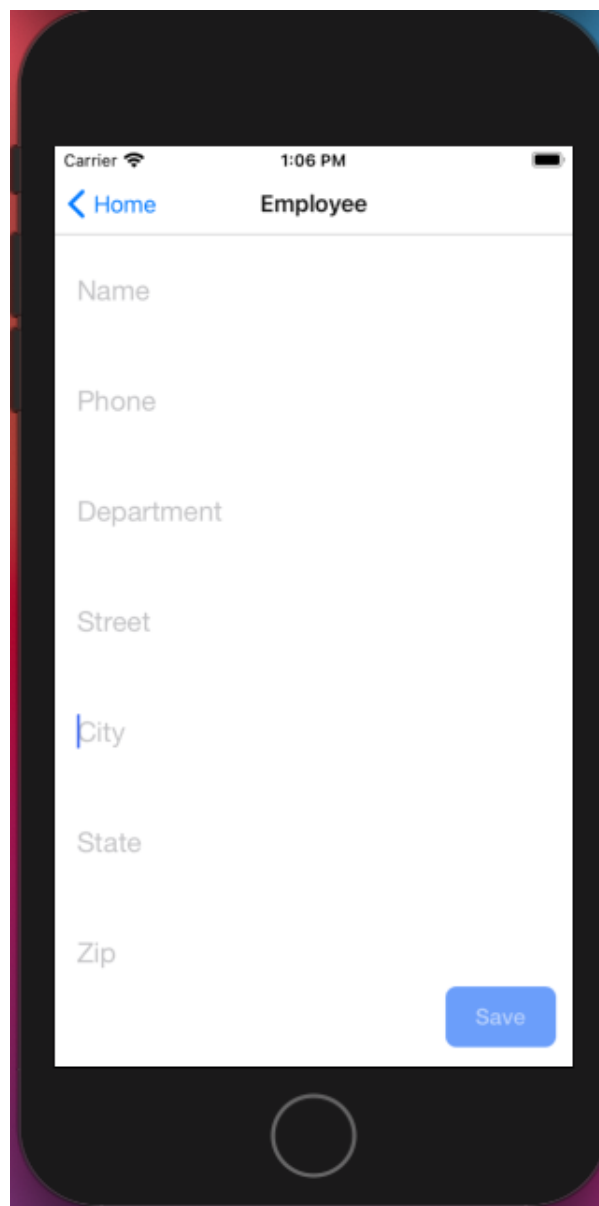
Test result: pass

Insert: press “+” in the bottom right of the main page, add the new employee’s information and press save

Expected result: add John information successfully

Real result:

(Add new
information
here)



(Fill the
Information)

Carrier 1:16 PM

< Home Employee

John

123

IT

23 aaa

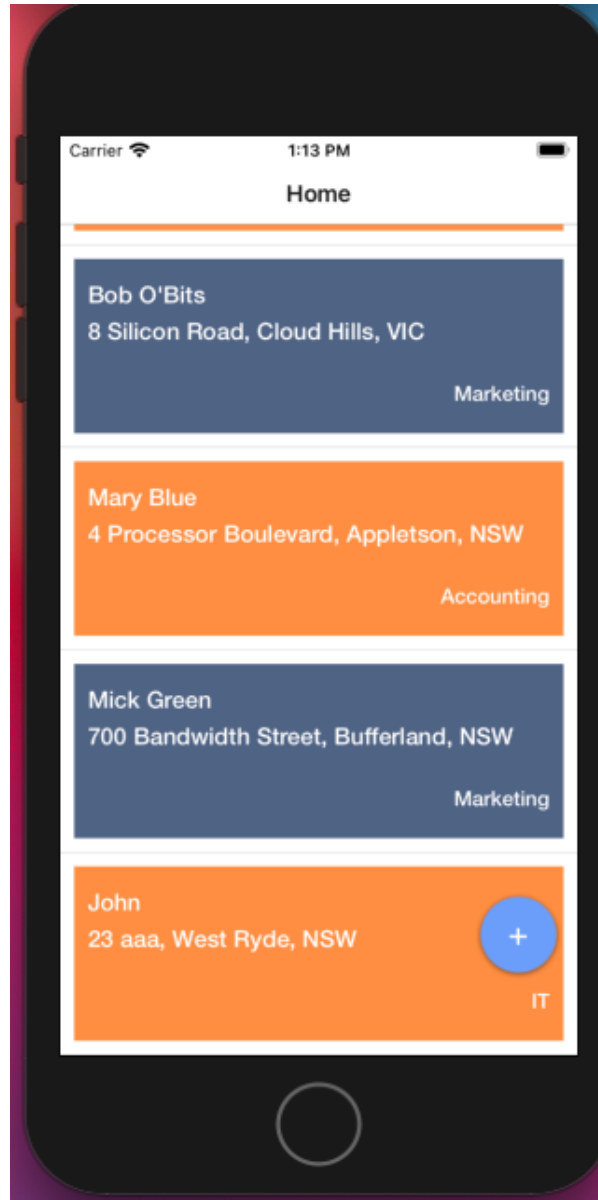
West Ryde

NSW

132

Save

(We can see
John in the
bottom)



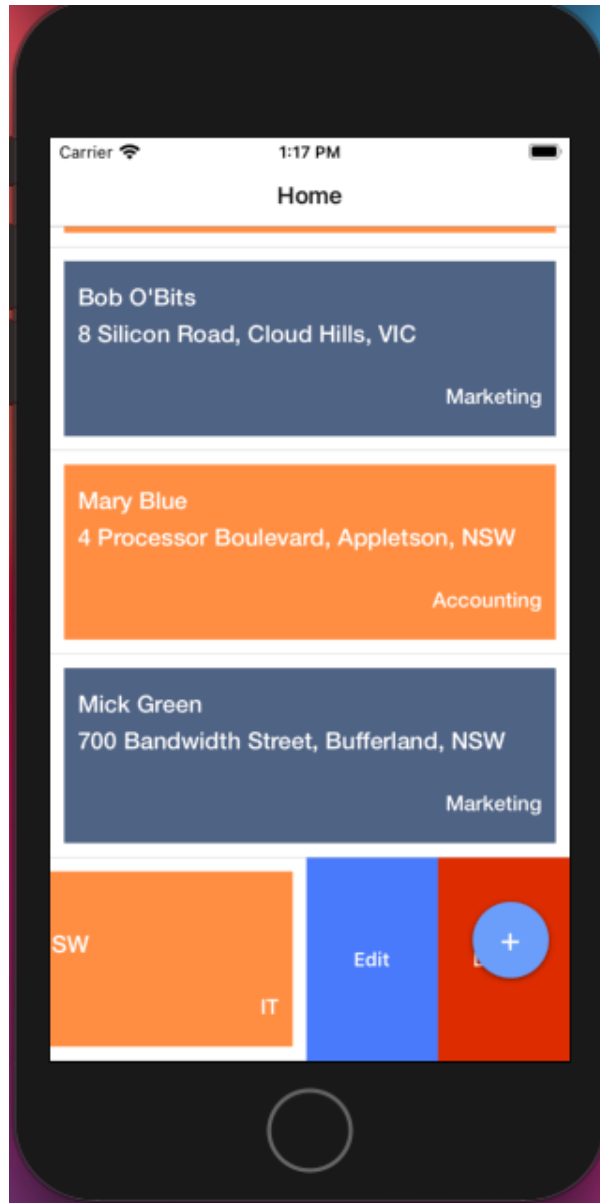
Test result: pass

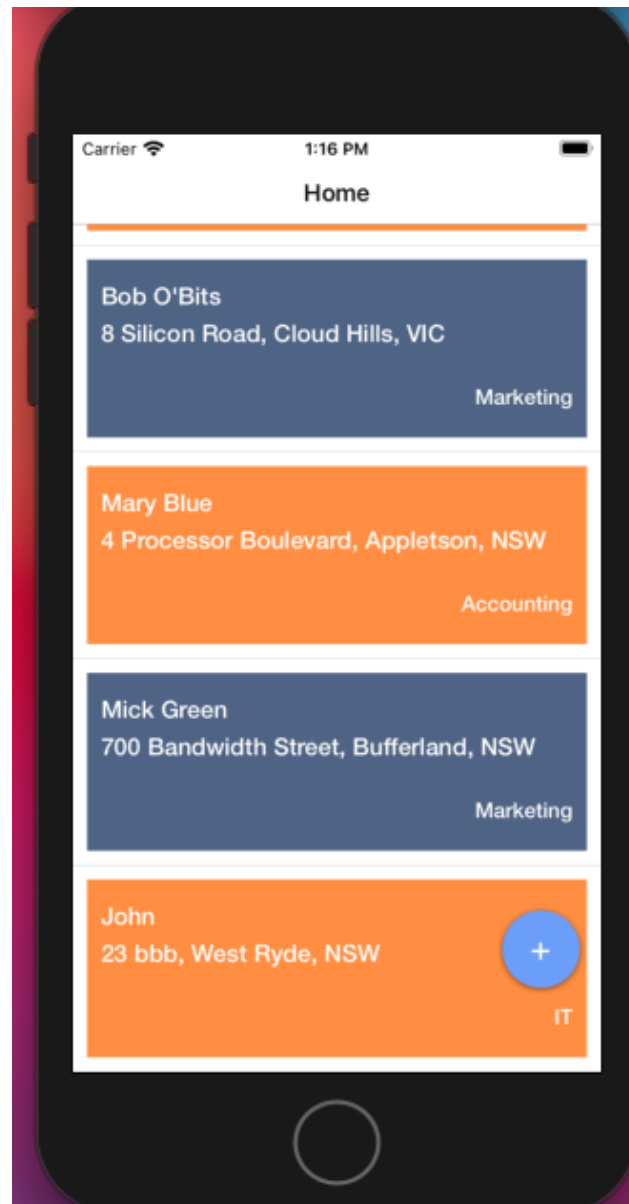
Update: right click John, press edit then we could edit John's information

Expected result: Change John's address from aaa to bbb

Real result:

(Right click
Then we
Could
See “edit”
Button)





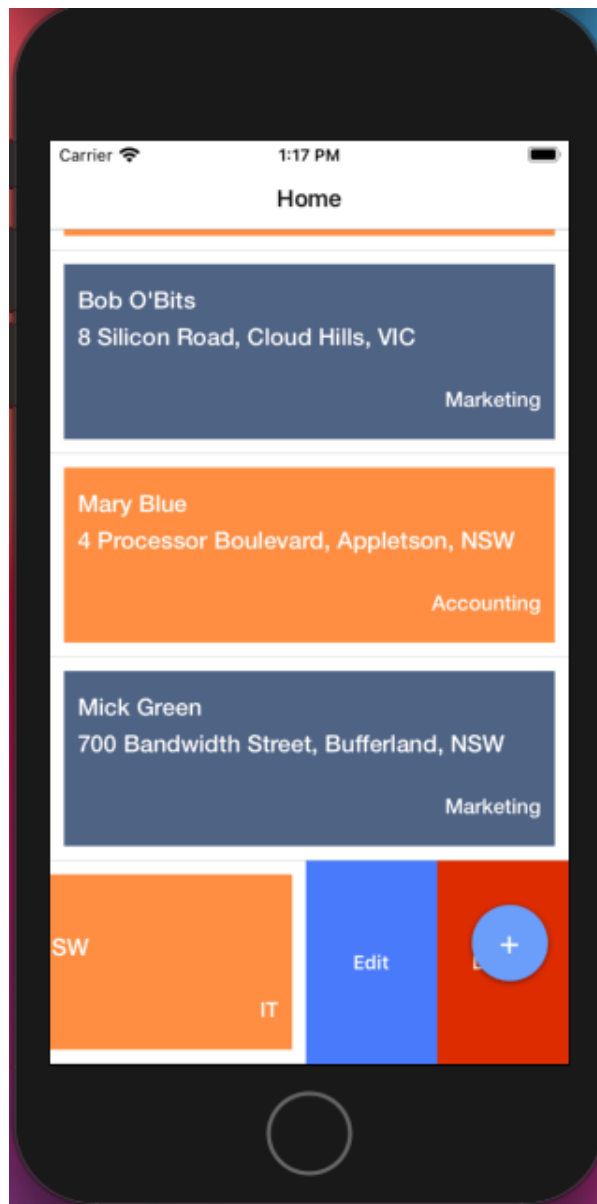
Test result: pass

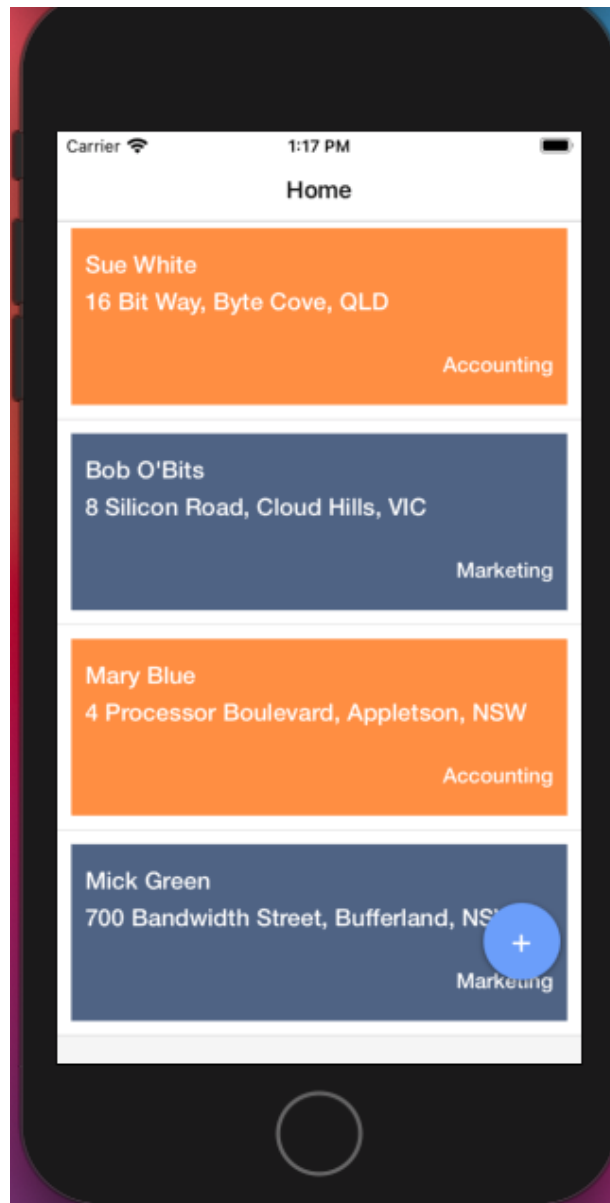
Delete: right click John, press delete then we could edit John's information

Expected result: John's information is deleted

Real result:

(Right click
Then we
Could
See “delete”
Button, which
Is under the “+”
Button:-D)





Test result: pass