INDUSTRIAL TRAINING REPORT

REACT - NATIVE

AFAAN ANSARI

19th April, 2019

Table of Contents

CHAPTER 1: INTRODUCTION..... 2

1.1) STUDENT DETAILS..... 2

1.2) COURSE / TRAINING YOU OPTED FOR 3

1.3) ABOUT THE COMPANY..... 4

CHAPTER 2: OBJECTIVE OF THE TRAINING / COURSE..... 9

CHAPTER 3: OVERVIEW OF WHAT YOU LEARN..... 10

CHAPTER 4: INTERNSHIP / TRAINING PLAN..... 12

CHAPTER 5: WEEKLY SUMMARY..... 13

CHAPTER 6: SUMMARY OF PROJECT..... 18

CHAPTER 7: CONCLUSION..... 26

CHAPTER 8: REFERENCES..... 27

CHAPTER 1 : INTRODUCTION

1.1) STUDENT DETAILS

NAME	ANSARI MOHAMMED AFAAN
ENROLLMENT NO.	SS16IF033
EMAIL ID	affanmohhd@gmail.com
PHONE NO.	8097862247
COLLEGE NAME	GOVERNMENT POLYTECHNIC MUMBAI
DEPARTMENT	INFORMATION TECHNOLOGY
YEAR	3rd YEAR
SEM	6
DURATION	15 Day's
COMPANY	PEN MOUSE DESIGN TECH PVT. LTD.
SUBJECT	INDUSTRIAL TRAINING
COURSE CODE	IT16319

1.2) COURSE / TRAINING YOU OPTED FOR

- **REACT - NATIVE**

React Native lets you build mobile apps using JavaScript. It uses the same design as React, letting you compose a rich mobile UI from declarative components.

With React Native, you don't build a "mobile web app", an "HTML5 app", or a "hybrid app". You build a real mobile app that's indistinguishable from an app built using Objective-C, Java, Kotlin, or Swift. React Native uses the same fundamental UI building blocks as regular iOS and Android apps. You just put those building blocks together using JavaScript and React.

React Native lets you build your app faster. Instead of recompiling, you can reload your app instantly. With hot reloading, you can even run new code while retaining your application state.

React Native combines smoothly with components written in Objective-C, Java, Kotlin, or Swift. It's simple to drop down to native code if you need to optimize a few aspects of your application. It's also easy to build part of your app in React Native, and part of your app using native code directly - that's how the Facebook app works.

The focus of React Native is on developer efficiency across all the platforms you care about - learn once, write anywhere. Facebook uses React Native in multiple production apps and will continue investing in React Native.

1.3) ABOUT THE COMPANY

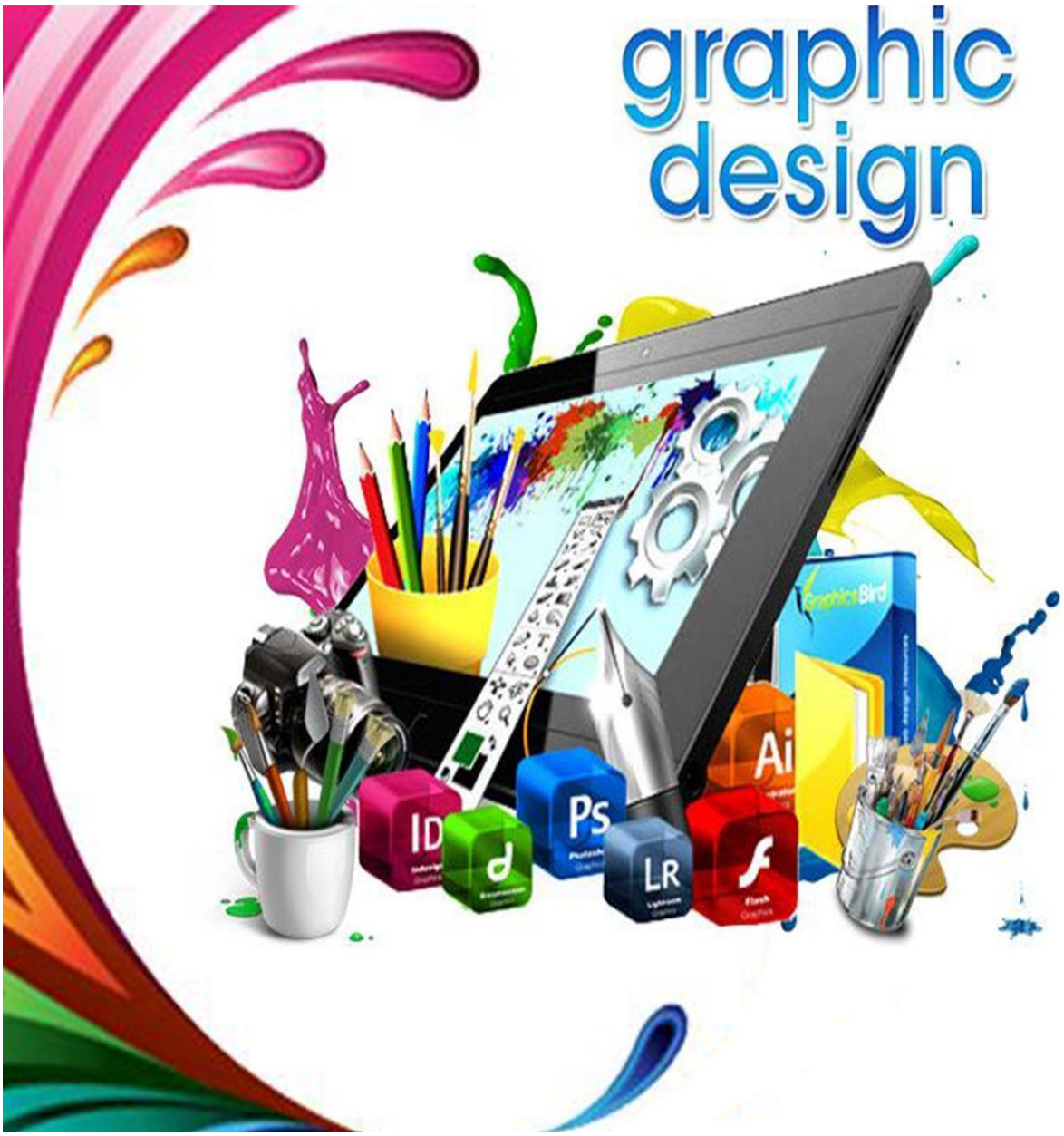
Pen Mouse Design Tech established by Mr. Taj Khan (Director), Mrs. Sadiya Khan (Director) & Mr. Azim Khan (Director) at Jan, 2018. We provide top class services and superior customer support for CM Launcher Theme & Keyboards Theme.

At Pen Mouse Design Tech professionalism and expertise rules and we believe our designs improve your mobile look to new heights of success in the shortest possible time.

Our unique Keyboard & Theme designing have always satisfied our customers.

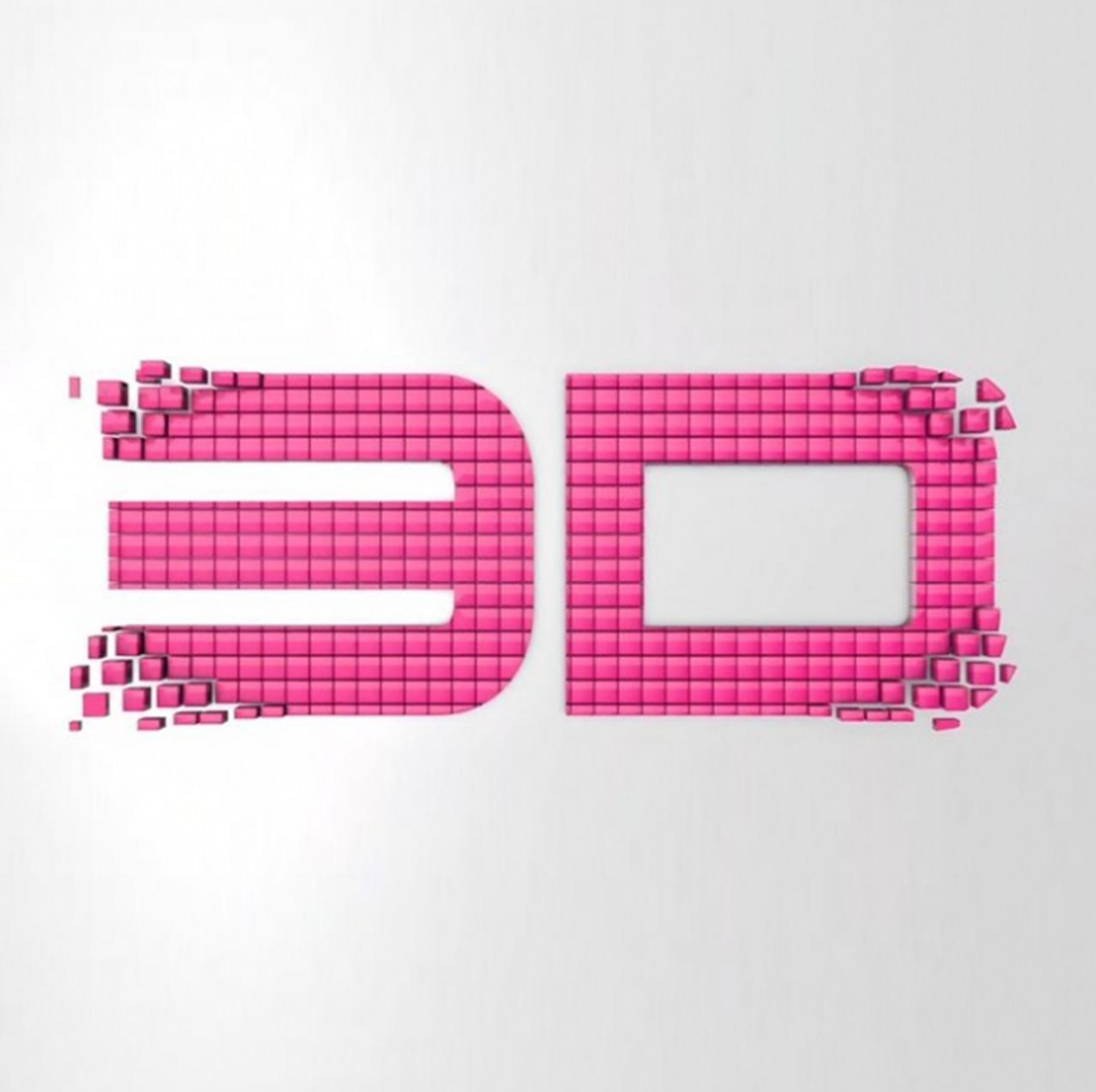
➔ **SERVICES**

Graphic Design



Mobile Applications Development

3D Modeling



➔ CONTACT

❑ **Our office Address -**
UG-175/176, 1st floor, The dreams mall, bhandup west Mumbai 78

If you'd like to get in touch, you can send me an email directly at info@penmousedesigntech.com or contact me on +91-9172624399

CHAPTER 2 : OBJECTIVE OF TRAINING / COURSE

The objective of the course is to learn the react-native , basic of other programming languages which we can use with react-native , it include javascript.also the implementation of react-native application in industrial level (i.e real world application).

As the react-native is derived from react , that is a package ,so it can be understand that react - native is also package , we also to learn many more package compatible with react-native environment (i.e all the package is node packages as node package manager [**npm**] ,also react & react - native is a node package)

Other packages include :-

1. React - navigation
2. Native - base
3. Firebase
4. React - firebase
5. React - vector - icons
6. React - redux

And many more which we can find it in github & npm library (https://github.com/jondot/awesome-react-native#navigation)

CHAPTER 3 : OVERVIEW OF WHAT YOU LEARN

In this training / internship we have learned the react - native ,

And the it's implementation on real world ,how it can be applicable

As , we know that the development of an mobile application can be done by native development . Which run only in Android , and cannot be implemented in IOS devices . if we need to implement it in IOS then we need to some other programming language by the development xcode environment , which is available only in mac devices .

So , to overcome this there is a concept hybrid development , by this we don't need to write a separate code for IOS , both will work in a single environment and with same code.

There are many hybrid development framework (XAMARIN , ICONIC , INTEL XDK , PHONEGAP , FRAMEWORK 7 , SENCHA TOUCH , KENDO UI , REACT - NATIVE)

They all works same but react - native is dissimilar to all other . because the react -native works with javascript to the core (java) , which works with device OS , which is unlike to other . so it is more benefit to use react - native more than other.

We have learned the concept of react - native , and the javascript concept which can be used with the development (function , class , states etc.)

And the it's component like button , flatlist , keyboardavoidingview , view , safeareaview , text , status , alert , slider , animation , textinput etc.

For the multi- screen (i.e for the navigation) we have use the react-navigation , which for the navigators .

For better animated components there is a package native- base

For store the data in the google-firebase which is nosql by the firebase and react-firebase package

For icons there is a package react-vector-icons

For the management of the states there is a package react - redux ,which can be to manage states , and store the temporary data in cache memory , for getting and storing data in api format from the network in redux .

And also the are advanced topic which we have learn are - networking , generating signed apk etc.

Also, the most important is stylesheet api , css for styling and alignment

We can also see to the api .

CHAPTER 4 : INTERNSHIP / TRAINING PLAN

This training is conducted by the pen mouse design pvt. Ltd of the mobile development : react - redux

In this we have learn the every part of react -redux ,

And this we have learned weekly basic of certain concept of it .

And in this training is of the duration of one month which is conducted in 2 weeks .

And in each week we have learned a new concept of react native and corresponding to that project we have done a task, that is in form of a small project .

The weekly concept of 2 weeks are :-

1. Usage of redux api
2. Usage of persist-redux api
3. Converting it into apk

CHAPTER 5 : WEEKLY SUMMARY

The task which is performed during the training are summarized weekly , below :-

1. Usage of Redux Api

⇒ **To use React Redux with your React app:**

npm install react-redux

⇒ **Provider**

React Redux provides <Provider />, which makes the Redux store available to the rest of your app:

```
<Provider store={store}>
   <App />
</Provider>
```

⇒ **connect()**

React Redux provides a connect function for you to connect your component to the store.

```
export default connect(
  mapStateToProps,
  mapDispatchToProps
)(Counter)
```

⇒ **mapStateToProps**

It should take a first argument called state, optionally a second argument called ownProps, and return a plain object containing the data that the connected component needs.

This function should be passed as the first argument to connect, and will be called every time when the Redux store state changes. If you do not wish to subscribe to the store, pass null or undefined to connect in place of mapStateToProps.

```
function mapStateToProps(state, ownProps?)
```

⇒ **mapDispatchToProps**

As the second argument passed in to connect, mapDispatchToProps is used for dispatching actions to the store.

dispatch is a function of the Redux store. You call store.dispatch to dispatch an action. This is the only way to trigger a state change.

```
const mapDispatchToProps = (dispatch) => ({
   name1: () => dispatch({type: Actions.NAME}),
});

export default connect(mapStateToProps,
mapDispatchToProps)(CounterComponent);
```

⇒ **Project : -**

Counter App (to save the desired state in Redux memory)

2. Usage of persist Redux Api

⇒ **npm install redux-persist**

⇒ **Basic usage involves adding persistReducer and persistStore to your setup. IMPORTANT Every app needs to decide how many levels of state they want to "merge". The default is 1 level.**

```
import { createStore } from 'redux'
import { persistStore, persistReducer } from 'redux-persist'
import storage from 'redux-persist/lib/storage' // defaults to localStorage for web and
AsyncStorage for react-native

import rootReducer from './reducers'

const persistConfig = {
  key: 'root',
  storage,
}

const persistedReducer = persistReducer(persistConfig, rootReducer)

export default () => {
  let store = createStore(persistedReducer)
  let persistor = persistStore(store)
  return { store, persistor }
}
```

⇒ **Project : -**
Incrementer / Decrementor App (to save the consistent state in memory)

3. Converting into APK

⇒ **Generating a signing key**

You can generate a private signing key using keytool. On Windows keytool must be run from C:\Program Files\Java\jdkx.x.x_x\bin.

$ keytool -genkeypair -v -keystore my-release-key.keystore -alias my-key-alias -keyalg RSA -keysize 2048 -validity 10000

⇒ **Setting up gradle variables**

1. **Place the my-release-key.keystore file under the android/app directory in your project folder.**
2. **Edit the file ~/.gradle/gradle.properties or android/gradle.properties, and add the following (replace ***** with the correct keystore password, alias and key password),**

```
MYAPP_RELEASE_STORE_FILE=my-release-key.keystore
MYAPP_RELEASE_KEY_ALIAS=my-key-alias
MYAPP_RELEASE_STORE_PASSWORD=*****
MYAPP_RELEASE_KEY_PASSWORD=*****
```

⇒ **Adding signing config to your app's gradle config**

Edit the file android/app/build.gradle in your project folder, and add the signing config,

```
...
android {
    ...
    defaultConfig { ... }
    signingConfigs {
        release {
            if (project.hasProperty('MYAPP_RELEASE_STORE_FILE')) {
                storeFile file(MYAPP_RELEASE_STORE_FILE)
                storePassword MYAPP_RELEASE_STORE_PASSWORD
                keyAlias MYAPP_RELEASE_KEY_ALIAS
                keyPassword MYAPP_RELEASE_KEY_PASSWORD
            }
```

```
        }
    }
    buildTypes {
        release {
            ...
            signingConfig signingConfigs.release
        }
    }
}
...
```

⇒ **Generating the release APK**

```
$ cd android

$ ./gradlew assembleRelease
```

⇒ **Project : -**

Any app to covert it in apk to use it in physical device

CHAPTER 6 : SUMMARY OF PROJECT

1. QUESTION AND ANSWER APP

During the internship period we have worked on an project that had covered all the concept that we have learned .

The topic is question and answer app .

The external packages used are :-

1. React-native-vector-icons
2. React-navigation
3. React-navigation-material-bottom-tabs
4. native-base
5. react-redux

In this first we have login to that user account , if account is not present then register account . Also, anonymous login is possible (with the help of filebase)

Then there is a subject names , and with respect to that there is questions and we have to answer the question

The navigator used is nested form , all three navigators are use

Icons are also used

To add question in the firebase , admin panel is there

The answer we give is stored in the admin panel

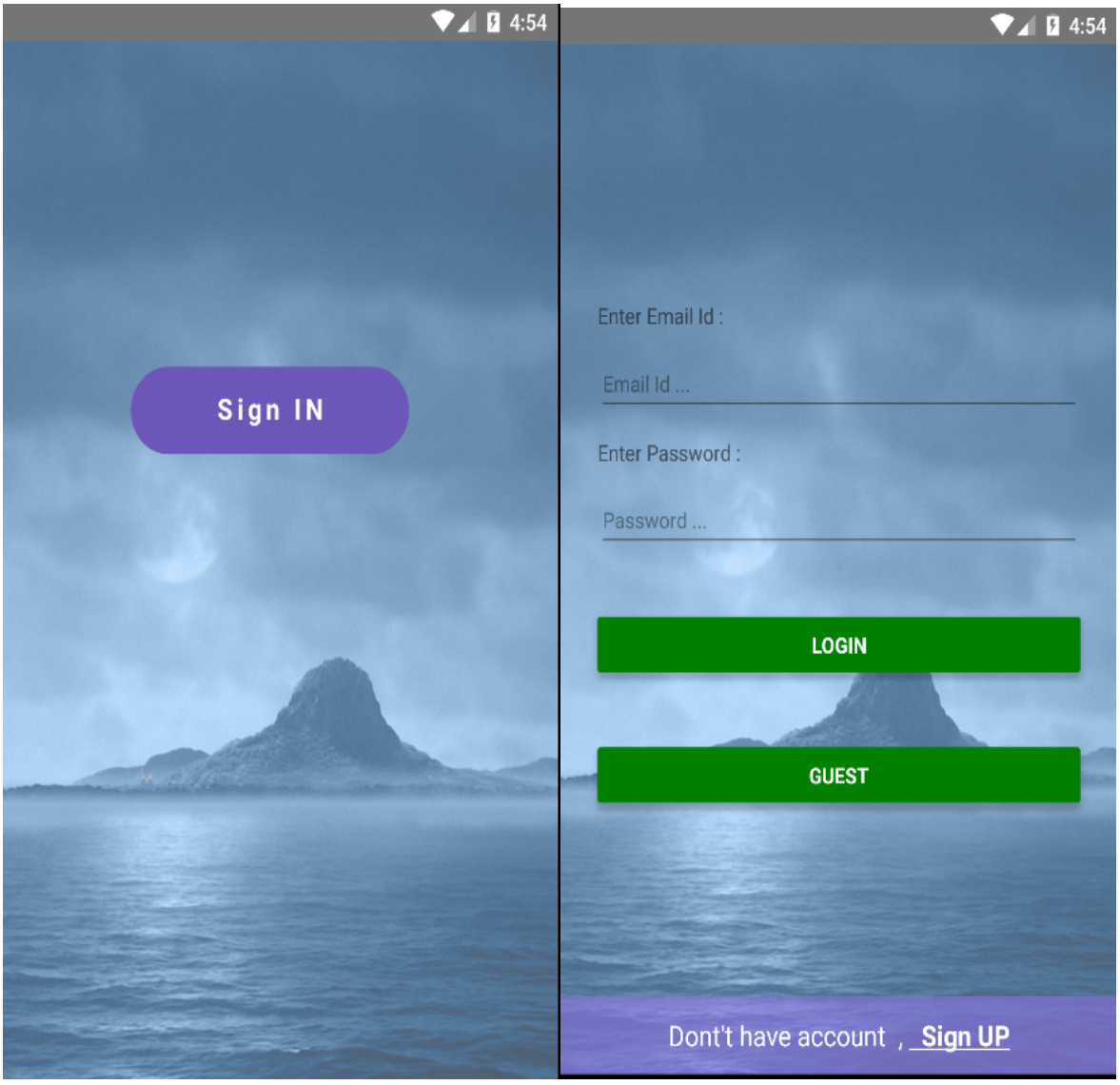
Also maximum components and api is used and the database is nosql (filebase)

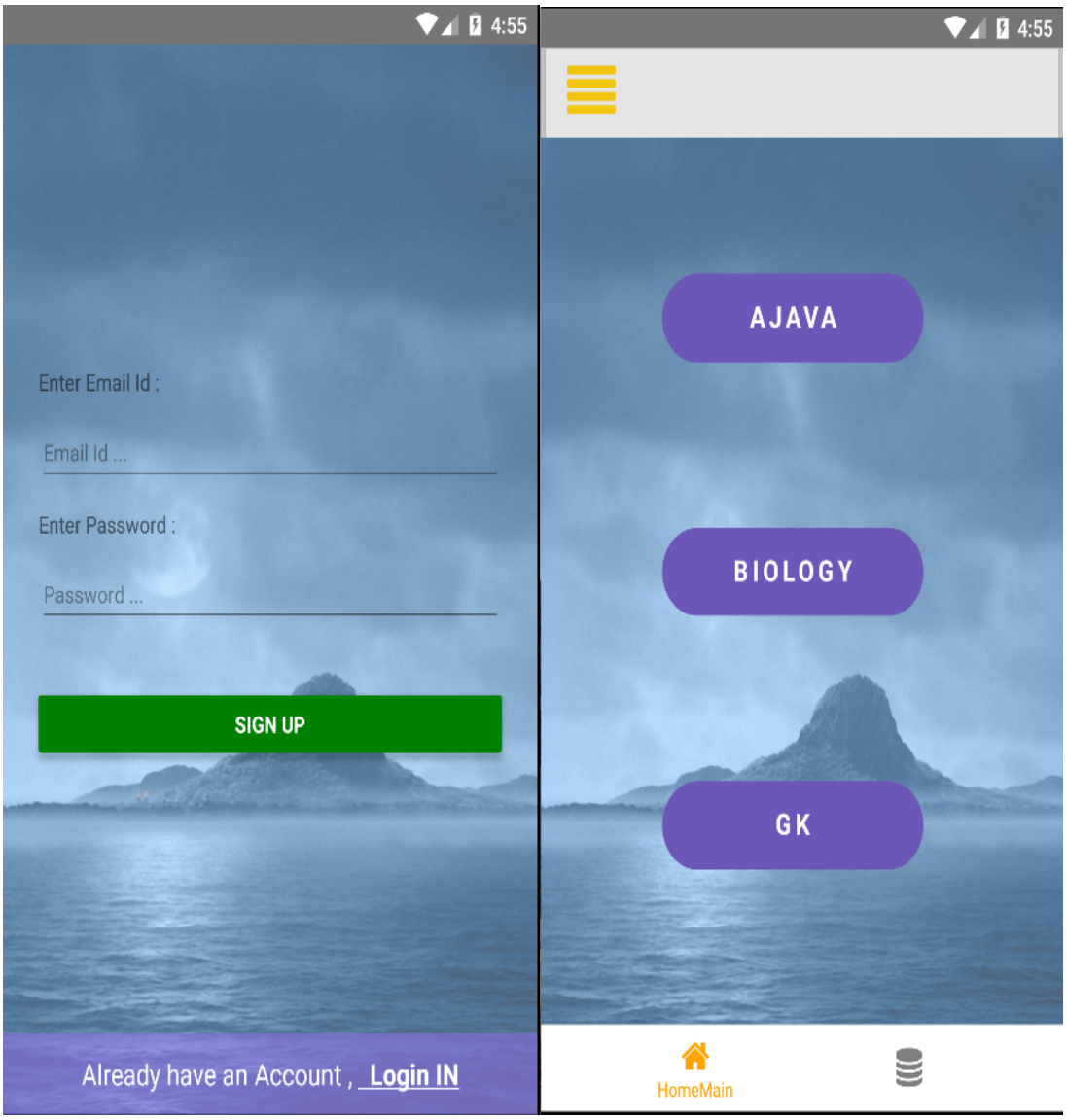
And we have finally used react-redux to manage the api states to store and retrieve it from temporary OR cache memory

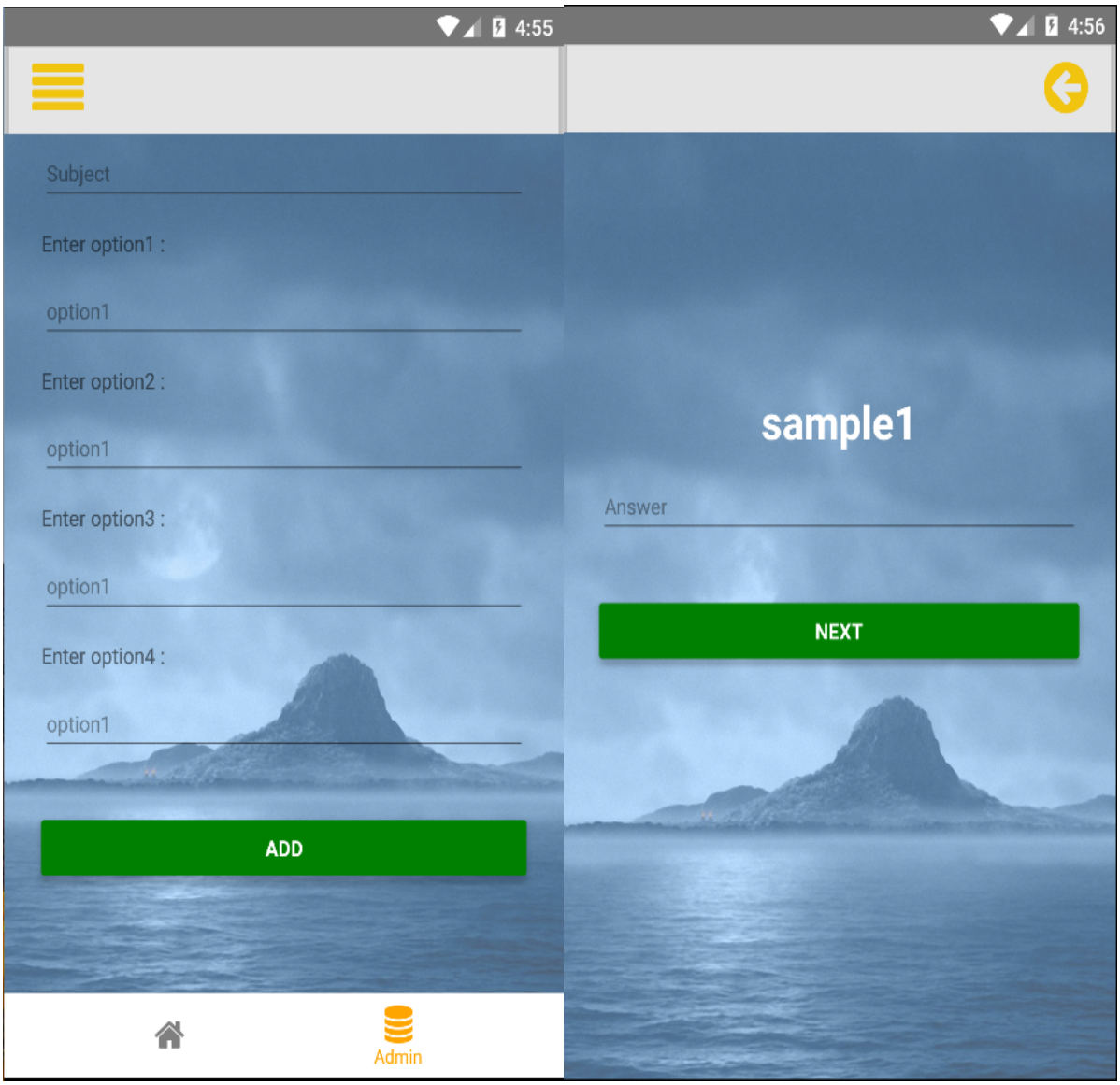
➢ **GITHUB : -**

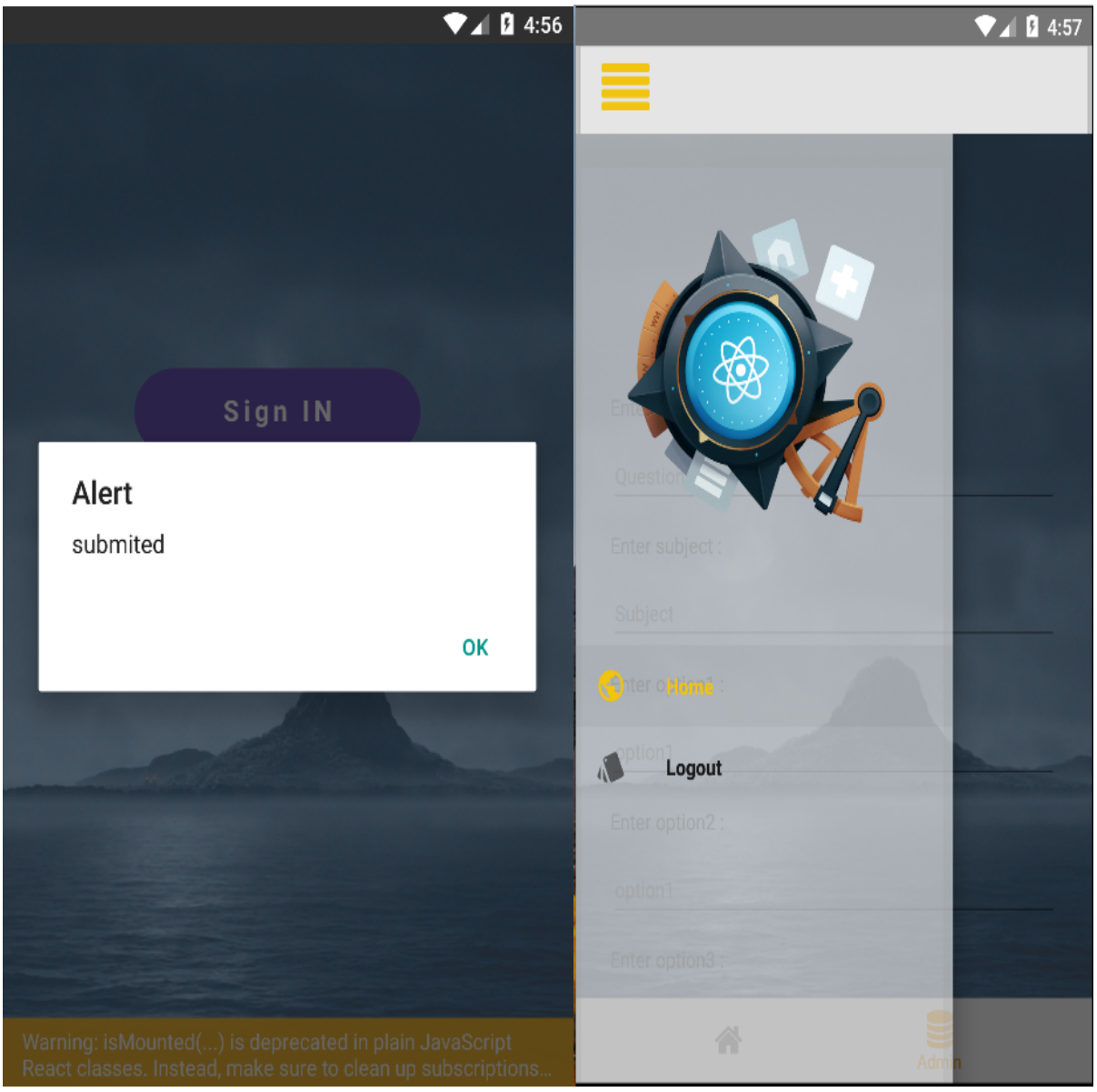
https://github.com/affan00733/react-native-industrial-training.git

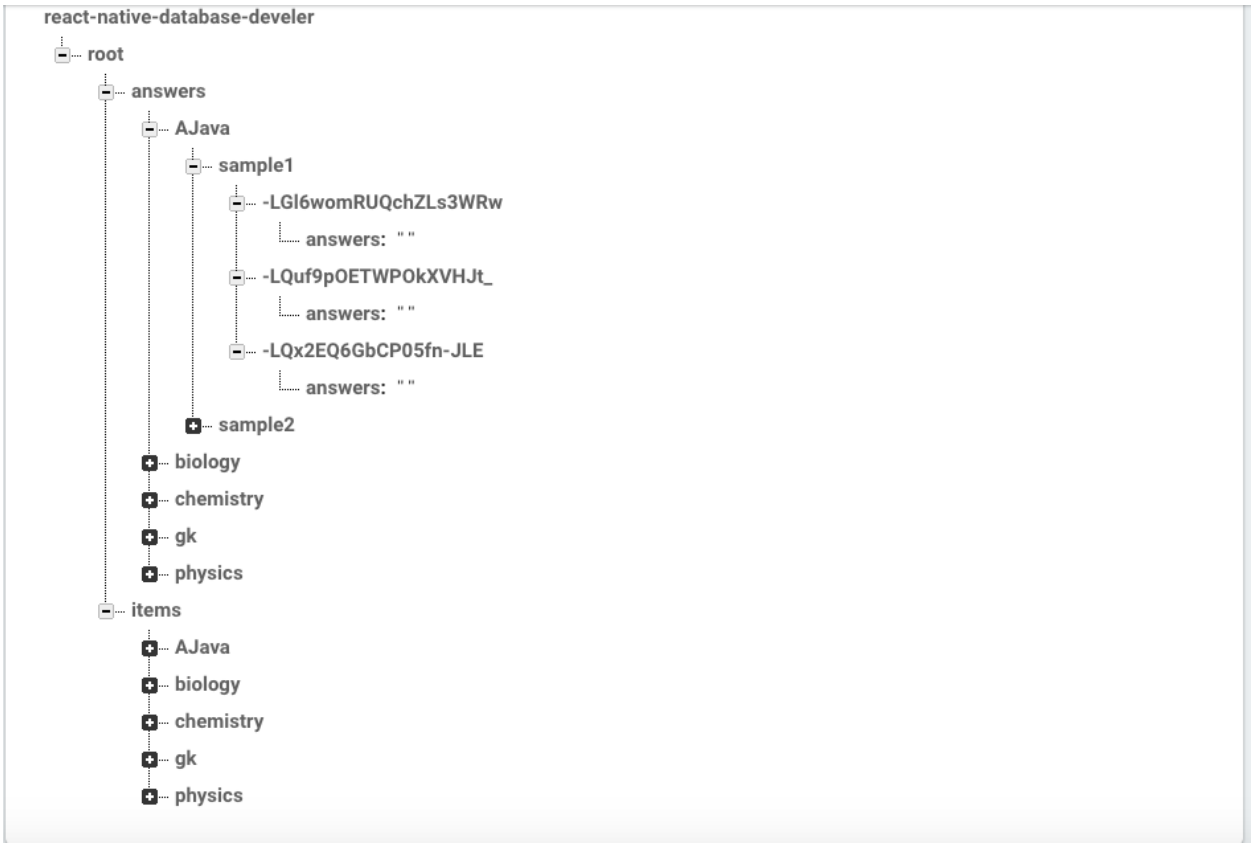
➢ **SCREENSHOT : -**




4:55
Enter Email Id :
Email Id ...
Enter Password :
Password ...
SIGN UP
Already have an Account , Login IN
4:55
AJAVA
BIOLOGY
GK
HomeMain


4:55
Subject
Enter option1 :
option1
Enter option2 :
option1
Enter option3 :
option1
Enter option4 :
option1
ADD
Admin
4:56
sample1
Answer
NEXT


4:56
Sign IN
Alert
submited
OK
Warning: isMounted(...) is deprecated in plain JavaScript
React classes. Instead, make sure to clean up subscriptions...
4:57
Enter subject :
Subject
Home
Logout
Enter option2 :
option1
Enter option3 :
Admin



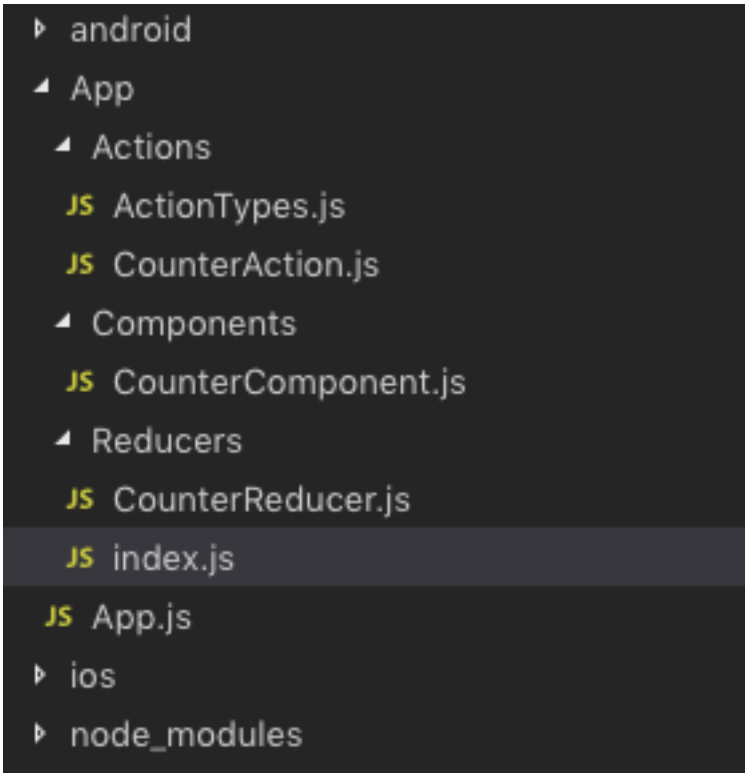
Identifier	Providers	Created	Signed In	User UID
(anonymous)		Jun 20, 2018	Jun 20, 2018	1FO1YuDThSeoXCm1GyPVyZDNV...
(anonymous)		Jun 22, 2018	Jun 22, 2018	2TDR6LPz1NMngW5GP1I5GrhN0...
(anonymous)		Jul 7, 2018	Jul 7, 2018	3tAmF9ZfrjhV9SYKx1lMvyUKJiM2
(anonymous)		Jun 20, 2018	Jun 20, 2018	5n57ApGzSFNAF142NXPfiI05uFi1
(anonymous)		Jun 22, 2018	Jun 22, 2018	6Glh3xljwLYIlsvO1seDP6S3yix2
(anonymous)		Nov 10, 2018	Nov 10, 2018	6Y0Sx3mEzxR1vDEoUklYZydE6LG2
(anonymous)		Jun 22, 2018	Jun 22, 2018	7D6k1L7XKsNknbe0gnt9DaciVL12
ads@gmail.com		Jul 7, 2018	Jul 7, 2018	8T7FbAn8EZMU7KiWu0hlz7PGxY...
(anonymous)		Jun 22, 2018	Jun 22, 2018	8qoHQWM2S4WG9C3T8FdTUu7c...
(anonymous)		Nov 10, 2018	Nov 10, 2018	9sjmcD4zbwXbg2QVKu8FI8efwA13
(anonymous)		Jun 22, 2018	Jun 22, 2018	A0pT8pXQvGSkO4ZYlfE0ZHjD3u02
(anonymous)		Jun 27, 2018	Jun 27, 2018	BZ5iaM5llhdci0y7zOi1UfITWg93

2. COUNTER STATE (REACT-REDUX) APP

In this project we used redux api to save the state.

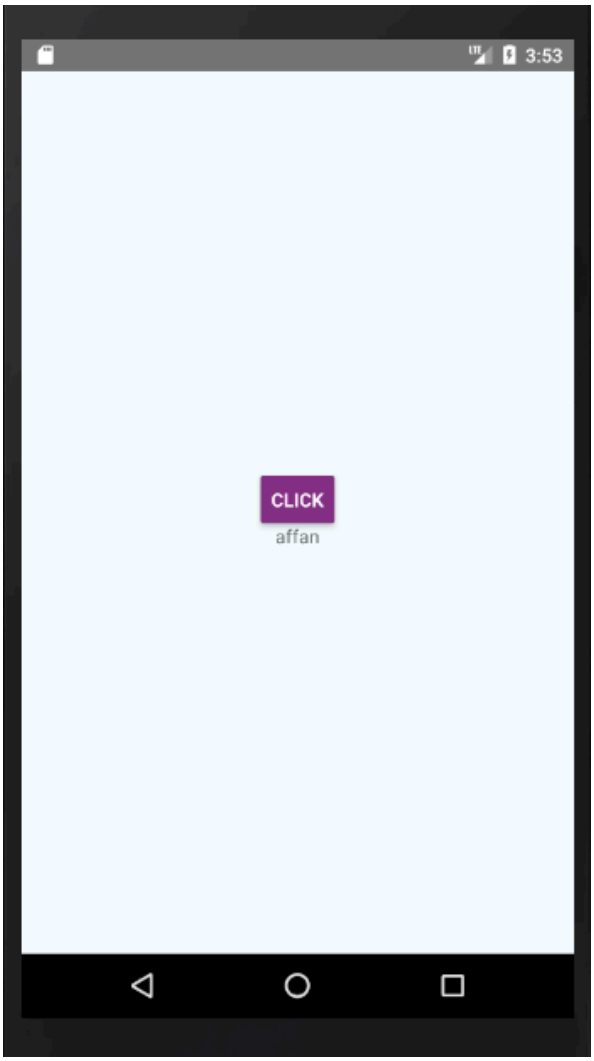
And when it goes to new state it's previous state is saved in it .

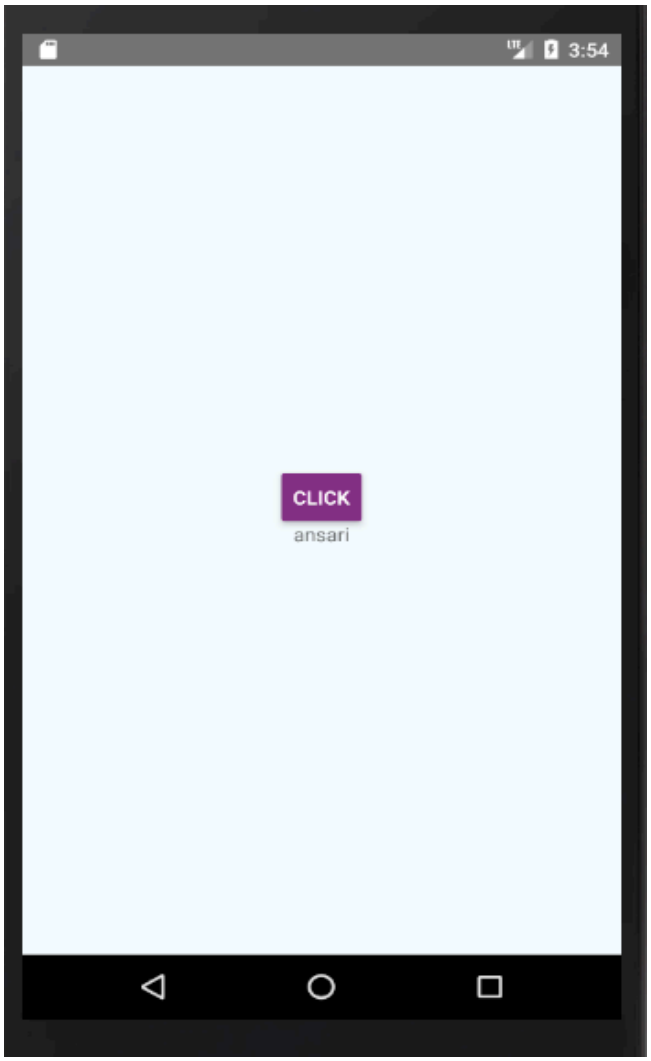
By using reducer , combineReducer and many more method's .



➢ **GITHUB : -**
https://github.com/affan00733/React-redux.git

➢ **SCREENSHOT : -**





CHAPTER 7 : CONCLUSION

In this industrial training we have learned the react -native which is hybrid development framework . And its implementation in the real world . Aso the javascript , other node packages which we can use with react - native . And we have made the small projects with the every concept of react - native and the a project which covers the every concept , css is also important for styling and managing components

And in every project we have used react-redux to manage the states

Note : - as react - native work for both android and ios it reduces work for developer

CHAPTER 8 : REFERENCES

https://facebook.github.io/react-native/

https://facebook.github.io/react-native/docs/tutorial.html

https://www.tutorialspoint.com/react_native/

https://nativebase.io/

https://console.firebase.google.com/u/0/project/react-native-database-develer/database/data

https://rnfirebase.io/docs/v5.x.x/getting-started

https://github.com/vhpoet/react-native-styling-cheat-sheet

https://drive.google.com/open?id=0B6IzhS9BSk2FSG1hMTd5ckVPbjgyenlEV0plbjZDdnA0NG9z

https://drive.google.com/open?id=0B6IzhS9BSk2FbDQxNlREQmVORm9DWEpySlhsQWVvQ3h2LWxV

https://reactnavigation.org/en/

https://github.com/affan00733/React-redux.git