

Exerciseopgave05-06 – Navigator og tabbar

React-navigation

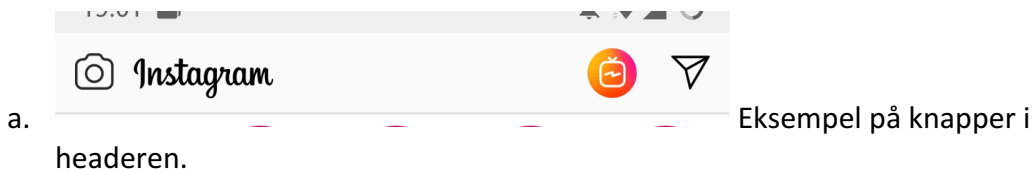
1. Åben terminal terminalen og naviger til dit workspace. Opret et nyt projekt og naviger ind i den.
2. Installer react-navigation ved at køre kommandoen "npm install --save react-navigation" (der er to bindestreger før save)
3. Hvis du har lavet et nyt projekt, så opret *components* mappen. Inde i denne kan du oprette en mappe ved navn *pages* for at strukturere projektet lidt.
4. Opret to components (pages) med eksempelvis navnene HomeScreen.js og SettingsScreen.js
5. Opbyg disse components (pages) med deres imports, klasse-deklarering der extender React.Component og render() funktionen der returnerer et view med noget text i hver. I princippet præcist som App.js er bygget.
6. Slet alt kode i App.js og importer dine to pages. Importer også createBottomTabNavigator fra react-navigation.
7. Deklarer createBottomTabNavigator som set i Slide 9 for dagens time, sådan så du har to routes kaldet Home og Settings, der hver tager imod sin page.
 - a. En "route" er en 'rute' din app kan tage. Her kan din app altså tage en 'rute' til Home-skærmen eller tage en rute til Settings-skærmen
8. Appen burde nu kunne navigere mellem de to views.

Stack-navigator

Du skal nu oprette såkaldte *stacks* af pages. Dette gøres for at fortælle appen, at når du er inde på din HomeScreen, kan det eksempelvis være at man kan trykke på noget indhold, hvor man åbner indholdet op på en ny side for at vise nogle detaljer. Eksempelvis når du er på Instagrams "Udforsk" side, og du trykker på et opslag, der åbner billedet/videoen op. På den måde er du kommet et niveau "dybere" i appen, og skal kunne navigere tilbage. Dette sørger Stack-navigatoren for.

9. Opret et nyt component inde i *pages* mappen kaldet DetailsScreen.js og sæt det op ligesom du gjorde med HomeScreen og SettingsScreen. Få den til at vise noget tekst der indikerer at det er DetailsScreen inde i render() funktionen.
10. Opret en SettingsStack med createStackNavigator som du importerer fra react-navigation. Stacken indeholder Settings og Details som routes. Se slide 10.

11. I din `createBottomTabNavigator` skal du ændre din Settings route til at tage i mod din `SettingsStack` i stedet for dit `SettingsScreen`.
12. I `SettingsScreen` skal du oprette en `Button` (husk import) med en eventhandler som en separat funktion (husk at tilføj `.bind(this).`) Eventhandleren skal navigere til Details routen. Kan ses i slide 10 på billedet der viser `HomeScreen`.
13. Appen burde nu kunne navigere mellem Home og Settings samt kunne navigere ind i og ud af Details, når du er under Settings med hjælp af `Button` og en tilbageknap, der opstår i headeren.
14. Optional: Giv de forskellige pages en header og en titel ligesom set på linje 5-7 på højre billede i slide 10. *For at tilføje knapper og anden funktionalitet til headeren til jeres eget projekt, kan I undersøge dette link <https://reactnavigation.org/docs/en/header-buttons.html>.*



Tab bar icons

15. Tilføj `navigationOptions` inde i `App.js` til din `createBottomTabNavigator`. Indeni denne skal du skrive en funktion for `tabBarIcon`, der forsøger at matche `routeName` med den route, der benyttes når der eksempelvis navigeres fra Home til Settings. Når der er et match, kan du vælge et passende icon [her](#) og sætte variablen `iconName` lig dets navn.
16. Importer `Ionicons` fra `@expo/vector-icons`.
17. Få funktionen du har oprettet i `tabBarIcon` til at returnere et `<Ionicons>` tag, der indeholder følgende properties: `name`, `size` og `color`. Benyt dig af `iconName` og `tintColor`. Hvad sker der når du justerer `size`?
18. Tilføj nogle `tabBarOptions`, der indikerer hvad farve et icon i tab-baren skal have når det er aktivt og når det er inaktivt.

Ekstraopgaver

19. Opret et nyt page og tilføj den til tab baren, så man kan navigere til den. Husk at tilføje en route.
20. Kan du gøre så man kan navigere fra dette nye page til din `HomeScreen` uden at benytte sig af tab-baren?