MAST5112

POE

PART 1

ST10435344

Tinyiko Thubelihle Ndlovu

## PERSONALISED CULINARY EXPERIENCE

Christoffel is a private chef renowned for his personalised culinary experience he needs a mobile application for his business. The app will be an easy way to swiftly change up his menu for clients, this will ensure that the clients always have access to the latest offerings, enhancing their dining experience. The app will serve as a digital menu

Youtube video link

[](https://www.youtube.com/embed/en5yAODjsRI?feature=oembed)

Github link

[menuApp/menu-app (5).zip at main · ModelSlick/menuApp (github.com)](https://github.com/ModelSlick/menuApp/blob/main/menu-app%20(5).zip)

Table of Contents

[PERSONALISED CULINARY EXPERIENCE 1](#_Toc175161895)

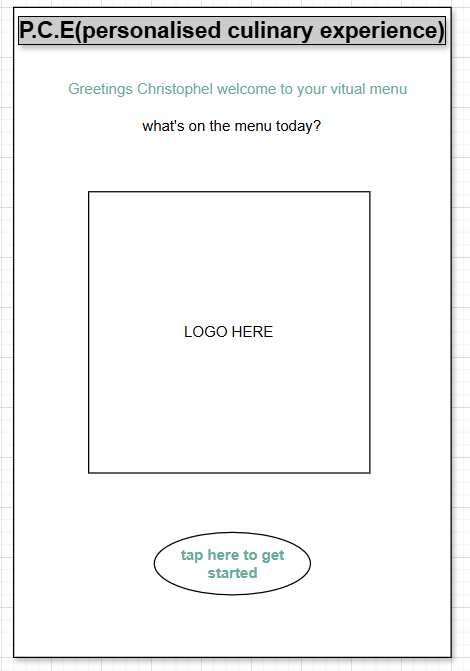
[Screen designs 3](#_Toc175161896)

[HOW I WILL IMPLEMENT THE ABOVE ON REACT NATIVE STEP BY STEP 18](#_Toc175161897)

[Reference 19](#_Toc175161898)

## Screen designs

1st screen



Explanation

\*On my first screen this is where the user will be welcomed to the page, on this screen the business name will be displayed at the top using the text import, followed by a warm welcome to the user, then after the logo will be displayed, to add the logo on react native, I will save the image I will be using as a “.png” or “jpg” image and store it in a folder I would have created on my app called the “images” folder from there I will import it to the component I want to display it and code it into my app.

\*From this screen the user will press a button “press here to get started” to be taken to the next screen. Adding the button is easy I must import it then code it in there.

2nd screen

A screenshot of a menu

Description automatically generated

Explanation

The chefs selected menu for the day will be displayed here categorising the vegan and non-vegan dishes separately in a list using the View text import. the chef will see his menu. Then using the import buttons “Vegan options” and “non-vegan options” users will be redirected to the selected course page.

3rd screen

A screenshot of a menu

Description automatically generated

EXPLANATION

In this screen the chef can unput and correct his menu for the day, he chooses how it is going to be displayed, at the top is the Text displaying “Enter todays menu Christoffel” Then below will be text inputs for the chef to enter todays menu, the chef then has to go back to the home screen to view his menu, by clicking on the “home” button, the clear button will be coded in a way that when clicked upon the input texts will disappear and the chef can make corrections, or retype the menu.

4TH screen

A screenshot of a computer screen

Description automatically generated

Explanation

\*The second screen I would have already imported image, so all I must do is save the chefs picture in my images folder and code it in the right component, then after write a little information about the chef. After seeing that the user will see a button at the bottom for them to click on and be taken to the next screen, which I also have to code in separately.

6th screen

A screenshot of a phone

Description automatically generated

Explanation

\*If the user clicks on the vegan menu button, they will be taken to the next page where “vegan menu” will be displayed at the top using the “Text” import, including the type of course they selected, which in this case is the “Starters” Under will be a checkbox of all the starters that are available for the clients and the users can tick which ones they would prefer, they can select more than one as they are not limited.

\*After selecting the user can click on the import button “Next” to go to the next menu or simply press on “back” if they have made a mistake and want to go correct it on the previous page.

7th screen

A screenshot of a cell phone

Description automatically generated

Explanation

\*The main course page is similar to the Starter page, just that on this page they are selecting the main course they want, if they don’t want any they can go back or press next to see the next screen.

8th screen

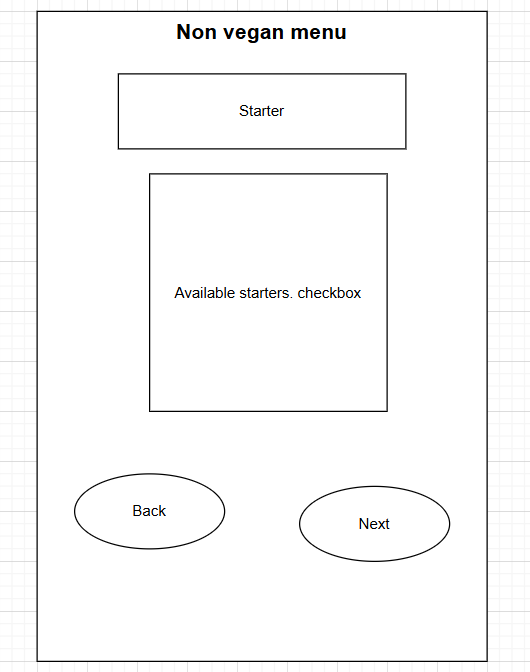
A screen shot of a computer screen

Description automatically generated

Explanation

\*The desert page is also like the previous two pages just that here the user will be choosing the desert they want. And can press back or next at any time.

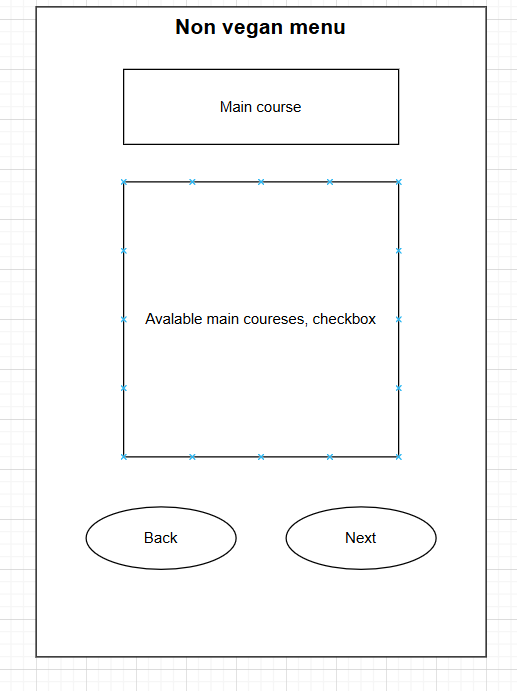
9th screen



Explanation

\*Should the user have selected the “non-vegan” option they would have been taken directly to this page, which works the same way as the vegan screen, at the top “non-vegan” is displayed using the “Text import” followed by the course name and a checkbox of available things the user can choose from, which is unlimited so the user can check on many boxes, then press “Back” or the “next” button for further information.

10th screen



Explanation

\*This screen is similar to the previous screens “main course” is displayed using the “Text” import, and available main courses are displayed in a checkbox option which is unlimited. And the user can either go back or press next. Using the button imports.

11th screen

A screenshot of a grid

Description automatically generated

Explanation

\*The deserts screen is also similar to the previous pages, just that here the user is choosing the deserts they would like to have. And just like the other screens press beck or next.

12th screen

A screenshot of a checkbox

Description automatically generated

Explanation

\*The tenths screen applies for both vegan and non-vegan users, after having selected their courses, they can choose a beverage to go with it. They can go back to fix their order or press the view order button to see their order in a list

13th screen

A screenshot of a computer screen

Description automatically generated

Explanation

\*The same business logo we used in the beginning will be coded in, which makes it easier because its already in our images folder and imported so we just have to code it in. “Your order has been successfully placed” Text will be displayed, and under it The user can choose to exit or rate the chef by pressing the “rate chef” button.

14th screen

A screenshot of a menu

Description automatically generated

Explanation

\*The business name will be displayed using the text import. At the top the selected menu will be displayed in categories “starter”,” main”,” desert” and whether its vegan or non-vegan will be displayed. Then the app will calculate and display the average price of the courses to choose from and the total cost of the food ordered. The user can then go back or process their order by pressing the ‘process order” button.

15th screen

A screenshot of a screen

Description automatically generated

Explanation

\*If the user pressed the “rate chef" button they would have been taken to this page, and at the top the text import would be placed to display the “rate Christoffel” message, then a checkbox of 1 to 5 stars will be displayed so the user can choose how many stars they want to rate the chef, and a add comment section will be below that, so their voice can be heard, which may help Christoffel perfect himself. Then they can click on the “Send” button to submit they response.

## HOW I WILL IMPLEMENT THE ABOVE ON REACT NATIVE STEP BY STEP

To create a React Native app for my virtual menu that allows my chef, Christoffel to modify the menu based on preferences, I will follow these steps:   
  
1. I will setup my React Native Project  
- I will start by establishing a new React Native project using the command 'npx react-native init VirtualMenuApp'.   
- Then Install any required dependencies, such as React Navigation for screen management and a state management library, such as Redux or Context API, as needed.   
  
2.I will develop an App Structure  
screens: I will need at least three major screens:   
a. The home Screen that will show the menu's overall results (Main, Starter, Dessert).   
b. The edit menu screen where the chef can alter the menu items.   
c. The category screen which will separate screens for the Main, Starter, and Dessert categories.   
(I will configure navigation between these screens with React Navigation (e.g., Stack Navigator.)

3. I will design the components  
a. Menu Item Component: A reusable component that represents a single menu item, with properties like as 'name', 'description', 'price', and 'category'.  
b. Menu list component: A list component that navigates through and displays menu items.  
  
4: I will build the Main Screen  
The 'HomeScreen' will show the entire menu, divided into Main, Starter, and Dessert. I will use a flat list or section list to display items sorted by category. Then add a button to navigate to the 'EditMenuScreen'.  
  
5. I will set up the Edit Menu Screen  
The 'EditMenuScreen' enables the chef to add, update, and delete menu items. I will use a form that includes fields for 'name', 'description', 'price', and 'category'. Then use the Context API or Redux to store menu data in a global state.  
E.g '''jsx  
var [menu, setMenu] = UseState([]);  
const addMenuItem = (item)=> {  
setMenu([...menu, item]);

};  
const UpdateMenuItem = (index, updatedItem) => {  
const newMenu = menu.map((item, i) => (i === index | updatedItem: item));  
setMenu(newMenu);  
  
};  
const deleteMenuItem = (index)=> {  
const newMenu = menu.filter((\_, i) => i!= index);  
setMenu(newMenu);  
  
};  
  
6. I will categorise the screens  
Each category (Main, Starter, Dessert) will have its own screen, the cook will be allowed to simply move between categories.  
  
7. I will do a few testing’s and Debugging’s  
By running the app on several devices/emulators. And ensuring that the navigation and data flow between screens are working properly.

## Reference

\*Ranisavljević, T., Karabašević, D., Brzaković, M., Popović, G. and Stanujkić, D., 2022. REACT NATIVE: A BRIEF INTRODUCTION TO MODERN CROSS-PLATFORM MOBILE APPLICATION DEVELOPMENT. *Quaestus*, (21), pp.120-136.

\*Boduch, A. and Derks, R., 2020. *React and React Native: A complete hands-on guide to modern web and mobile development with React. js*. Packt Publishing Ltd.

\*Van, H., 2020. Building a universal application with React and React Native.