

Nama : Muksin Bagaskara

NIM : 119140167

Kelas : RC

## LAPORAN UTS PAM

### A. Link Apk :

<https://drive.google.com/file/d/1e7KMgcv6z5xMdnCSRnJtQrNaHVRoYs-M/view?usp=sharing>

### B. Link GitHub :

[https://github.com/mbagas/uts\\_kapalzy.git](https://github.com/mbagas/uts_kapalzy.git)

### C. Link Video Demo :

<https://drive.google.com/file/d/1y5AHjQrXwZpE6wMMPRj7otBV3j9Z88EO/view?usp=sharing>

### D. Penjelasan Aplikasi :

#### 1. Data

Terdapat beberapa data seperti pelabuhan dan layanan yang nantinya akan berhubung dengan data tiket. Dengan menyocokkan id data.

```
pam > src > utils > .js Datajs > @01TIKET > @01LAYANAN
1 export const PELABUHAN = [
2   {
3     pelabuhan_id: 'p1',
4     pelabuhan_nama: 'merak',
5     pelabuhan_lokasi: 'jakarta',
6   },
7   {
8     pelabuhan_id: 'p2',
9     pelabuhan_nama: 'bakauheni',
10    pelabuhan_lokasi: 'lampung',
11  },
12  {
13    pelabuhan_id: 'p3',
14    pelabuhan_nama: 'Pearl Harbour',
15    pelabuhan_lokasi: 'panjang',
16  },
17  {
18    pelabuhan_id: 'p4',
19    pelabuhan_nama: 'seaport',
20    pelabuhan_lokasi: 'teluk',
21  },
22  {
23    pelabuhan_id: 'p5',
24    pelabuhan_nama: 'tanjung priuk',
25    pelabuhan_lokasi: 'indonesia',
26  },
27 ];
28
29 export const LAYANAN = [
30   {
31     layanan_id: 'l1',
32     layanan_nama: 'Express',
33     layanan_harga: 'Rp 65.000',
34   },
35   {
36     layanan_id: 'l2',
37     layanan_nama: 'VIP',
38     layanan_harga: 'Rp 75.000',
39   },
40   {
41     layanan_id: 'l3',
42     layanan_nama: 'Business',
43     layanan_harga: 'Rp 85.000',
44   },
45   {
46     layanan_id: 'l4',
47     layanan_nama: 'Executive',
48     layanan_harga: 'Rp 150.000',
49   },
50 ];
51
52 export let TIKET = [
53   {
54     tiket_id: '1',
55     pelabuhan_awal_id: 'p1',
56     pelabuhan_tujuan_id: 'p5',
57     layanan_id: 'l3',
58     tgl_masuk: 'Saturday, 24 03 2021',
59     jam_masuk: '19:40',
60     nama_lengkap: 'Hendri',
61     identitas: 'laki',
62     umur: '21',
63     status: 'aktif',
64   },
65   {
66     tiket_id: '2',
67     pelabuhan_awal_id: 'p3',
68     pelabuhan_tujuan_id: 'p2',
69     layanan_id: 'l1',
70     tgl_masuk: 'friday, 23 03 2022',
71     jam_masuk: '19:40',
72     nama_lengkap: 'bagas',
73     identitas: 'laki',
74     umur: '21',
75     status: 'aktif',
76   },
77   {
78     tiket_id: '3',
79     pelabuhan_awal_id: 'p1',
80     pelabuhan_tujuan_id: 'p2',
81     layanan_id: 'l2',
82     tgl_masuk: 'friday, 23 03 2022',
83     jam_masuk: '19:40',
84     nama_lengkap: 'bagas',
85     identitas: 'laki',
86     umur: '21',
87     status: 'batal',
88   },
89 ];
```

## 2. Halaman Home

Pada Landing Page atau halaman utama terdapat home yang berisikan form pengisian jadwal pelayaran dengan data pelabuhan awal, tujuan, kelas layanan, tgl masuk dan jam masuk. Proses yang terjadi untuk menampilkan halaman ini adalah dengan menggunakan initial route pada bottomtabs yang mengarah ke stack navigation dengan initial route home.

```
4 import Stack from './Stack';
5 import Tickets from './Tickets';
6 import Cancellations from './Cancellations';
7 import Other from './Other';
8 import FontAwesome from 'react-native-vector-icons/FontAwesome5';
9 const Tab = createBottomTabNavigator();
10
11 const MyTabs = () => {
12   return (
13     <Tab.Navigator initialRouteName='HomePage'>
14       <Tab.Screen
15         name="HomePage"
16         component={Stack}
17         tabBarLabel="HomePage"
18         color="#11468F"
19         options={{
20           tabBarIcon: ({focused}) => (
21             <FontAwesome name="home" size={22} color="#11468F" />
22           ),
23           tabBarLabelStyle: {
24             fontSize: 13,
25           },
26           headerShown: false,
27         }}
28       />
29     </Tab.Navigator>
30   );
31 }
```

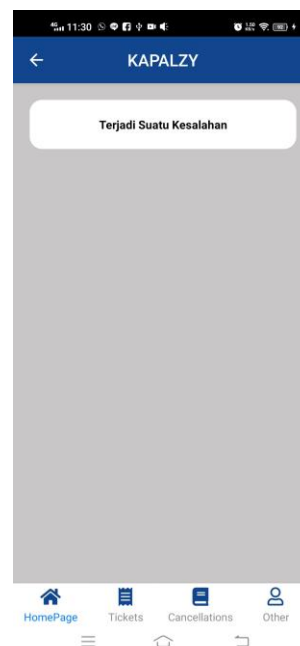
```
4 import Home from './Home';
5 import Confirm from './Confirm';
6 import Booking from './Booking';
7 import TicketDetail from './TicketDetail';
8
9 const Stack = createNativeStackNavigator();
10
11 const ScreenStack = () => {
12   return (
13     <Stack.Navigator
14       initialRouteName='Home'
15       screenOptions={{
16         headerShown: true,
17         headerTitleAlign: 'center',
18         headerTintColor: 'white',
19         headerStyle: {
20           backgroundColor: '#11468F',
21         },
22       }}
23     >
24       <Stack.Screen
25         name="Home"
26         component={Home}
27         options={{
28           title: 'KAPALZY',
29         }}
30       />
31       <Stack.Screen
32         name="Confirm"
33         component={Confirm}
34         options={{
35           title: 'KAPALZY',
36         }}
37       />
38     </Stack.Navigator>
39   );
40 }
```

Lalu menginisialisasi state yang nantinya data tersebut akan digunakan untuk di pass ke halaman lain

```
17 const Home = ({navigation}) => {
18   const [awal, setAwal] = useState('');
19   const [tujuan, setTujuan] = useState('');
20   const [layanan, setLayanan] = useState('');
21   const [tgl, setTgl] = useState('Tanggal Masuk');
22   const [jam, setJam] = useState('Jam Masuk');
23
24   const [date, setDate] = useState(new Date());
25   const [open, setOpen] = useState(false);
26   const [time, setTime] = useState(new Date());
27   const [timeOpen, setTimeOpen] = useState(false);
28
29   const [isDatePickerVisible, setDatePickerVisibility] = useState(false);
30   const [isTimePickerVisible, setTimePickerVisibility] = useState(false);
31
32   const showDatePicker = () => {
33     setDatePickerVisibility(true);
34   };
35
36   const hideDatePicker = () => {
37     setDatePickerVisibility(false);
38   };
39
40   const handleConfirm = date => {
41     setTgl(Moment(date).format('dddd, DD MM YYYY').toString());
42     // console.warn('A date has been picked: ', date, tgl);
43     hideDatePicker();
44   };
45 }
```

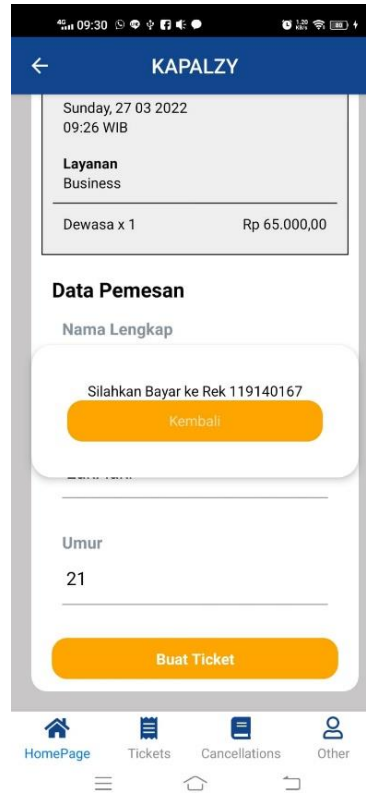
### 3. Halaman konfirmasi

Setelah mengisi form pada halaman home, data tersebut akan dikirim ke halaman konfirmasi dengan menggunakan react native navigation. Jika data yang dikirimkan tidak ada atau tidak sesuai, maka akan menampilkan informasi bahwa terjadi suatu kesalahan



#### 4. Halaman pemesanan tiket

Setelah mengkonfirmasi tiket, dilanjutkan dengan mengisi data diri pemesan dengan data nama lengkap, jenis kelamin dan umur. Setelah mengisi data diri dan menekan tombol “buat tiket” akan muncul modal/pop up yang menampilkan rekening pembayaran. Setelah itu akan dikembalikan ke halaman home dan data pesanan akan ditambahkan ke dalam data Json TIKET dengan status aktif.



## 5. Halaman Tickets

Pada halaman ini dapat diakses melalui bottomTabs “tickets”. Halaman ini berisi informasi terkait ticket yang sudah dipesan dan berstatus aktif. Untuk melihat perubahan data (penambahan/pengurangan) diperlukan merefresh halaman dengan menarik layar dari atas kebawah.



```
<FlatList
  data={TIKET.filter(x => {
    return x.status == 'aktif';
  })}
  renderItem={renderItem}
  keyExtractor={item => item.tiket_id}
  onRefresh={onRefresh}
  refreshing={isFetching}
  extraData={TIKET}
/>
```

```
const sleep = ms => new Promise(resolve => setTimeout(resolve, ms));
const [isFetching, setIsFetching] = React.useState(false);
const onRefresh = async () => {
  setIsFetching(true);
  await sleep(1000);
  setIsFetching(false);
};
```

## 6. Halaman Detail Ticket

Pada Halaman ini dapat diakses dengan menekan salah satu tiket di halaman tickets. Halaman ini berfungsi untuk melihat detail ticket dan membatalkan tiket dengan mengubah statusnya menjadi tidak aktif. Setelah membatalkan ticket, akan kembali ke halaman tickets. Untuk melihat perubahan data (penambahan/pengurangan) diperlukan merefresh halaman dengan menarik layar dari atas kebawah.



## 7. Halaman Pembatalan Tiket

Pada Halaman ini dapat diakses melalui bottomTabs “cancellations”. Halaman ini berisi informasi terkait ticket yang sudah dibatalkan dan bertatus batal.



```
<FlatList
  data={TIKET.filter(x => {
    return x.status == 'batal';
  })}
  renderItem={renderItem}
  keyExtractor={item => item.tiket_id}
  onRefresh={onRefresh}
  refreshing={isFetching}
  extraData={TIKET}
/>
```

## 8. Halaman Lainnya

Pada halaman ini berisikan informasi mengenai pengembang yaitu Muksin Bagaskara yang dapat diakses dari bottomTabs “others”.

