Aero

A Simple & Clean Flight Tracker & Manager

Aero Server - Front End

How do i run this locally?

Terminal

Open the terminal and run the following from `/`

```
 cd server
yarn install
yarn start
```

Documentation

> :warning: **We have over 50 differnet services, components, and pages. Only important ones are documented.

Hooks

`useAuth`

Once a user logs in, his id is stored securely in local storage. Whenever a page with the useAuth hook is visited, it checks wether his id, exists and then makes a get request `/user/get/:id. Then returns a object of functions that allow to check user role, and information.

`useScreenSize`

Basic hook that checks the current width of the window and returns a corresponding 2 character code that components use to properly align and scale themselves.

Services

`user.service`

Dashboard page uses this to retreive role specific data that belongs to each user. you can thing of this as a view route but on client side.

Pages

`SignIn`

Form takes only email and password, it makes an http request to `/user/signin`, which will return an id that is stored aswell as a role. On sucessful SignIn the user is redirected to the /home page.

`SignUp`

This is a dynamic form, that first asks wether you are a customer, agent or staff. It then uses the corresponding sign up function in the `user.service`. If a user is sucessfully registered, it automatically then called `SignIn` which will store the users session on the client.

`Dashboard`

Uses the `useAuth` hook to determine the user type, then will dynamically render one of these three components: `agentView`, `staffView`, `customerView`. The dashboard uses the `user.service` to get the correct data to use depending on the usrrs role.

`Flights`

Gets all the fights via http request from `/flights`. This page also has a flight search component which hits the following route `/flights/startdate/enddate/departurelocation/arrival/location`. If a user enters nothing into the search it will just return all the flights. Same goes for not entering any dates, it will then completely ignore dates. This page also uses the `useAuth` hook to determine the user type, if the user is a staff, then he can also create a new flight, but only one that is under his airline.

`Transaction`

The transaction page is meant takes everything it needs from local storage. When a user or a agent clicks buy on a flight, it sets the flight id in local storage, now it redirects you to `/transaction` where the data is extracted from the query parameters and you are allowed to put in your credit card information. If everything about the transaction goes through you should be redirected to your dashboard.

`Airports & Airlines & Destinations`

These are three simple pages that use the same generic list component. Airports and Airlines are only available to staff. and there they have the option to create a new airport/airline.

There are more components that arent documented.