Discussion Box Component

- Repository: https://github.com/etenlab/eil-discussion-box

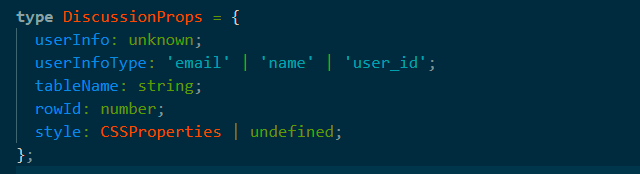
- npm: https://www.npmjs.com/package/@eten-lab/discussion-box

install:

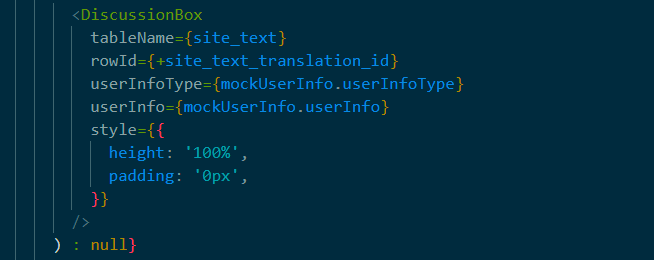
npm install --save @etenlab/discussion-box

usage:

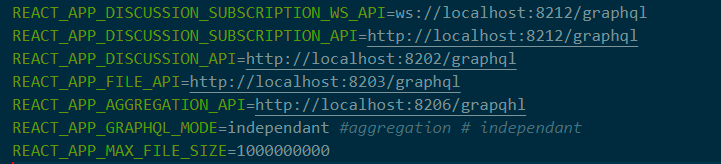
This component allows you to open discussions with specific table\_name and row\_id.



Example



.env



This component interacts with 'discussion-api', 'discussion-subscription-api', 'file-api' and 'aggregation-api'



Example

https://github.com/etenlab/translation-app/src/components/Discussion.tsx

import { useState } from 'react';

import { IonContent, IonIcon } from "@ionic/react";

import { arrowBack } from "ionicons/icons";

import { useHistory, useParams, useLocation } from "react-router";

import queryString from "query-string";

import Title from "../common/Title";

import { Discussion as DiscussionBox, MockLoginForm } from '@eten-lab/discussion-box';

export default **function** Discussion() {

**const** { site\_text, site\_text\_translation\_id } = useParams<{

site\_text: string;

site\_text\_translation\_id: string;

}>();

**const** [mockUserInfo, setMockUserInfo] = useState<{

userInfo: unknown;

userInfoType: 'email' | 'name' | 'user\_id';

} | null>(null);

**const** history = useHistory();

**const** { search } = useLocation();

**const** params = queryString.parse(search);

**const** backHandler = () **=>** {

if (params.site\_text\_id) {

history.push({

pathname: `/translation-app/site\_texts`,

search: `site\_text\_id=${params.site\_text\_id}`,

})

} else {

history.goBack();

}

}

return (

<IonContent>

<div style={{ padding: "60px 20px 60px 20px", height: "100vh" }}>

<div style={{ display: 'flex', flexDirection: "row", alignItems: "center" }}>

<IonIcon

className="back"

icon={arrowBack}

onClick={backHandler}

/>

<Title title={`${site\_text} #${site\_text\_translation\_id}`} />

</div>

{!mockUserInfo && (

<MockLoginForm

setMockUserInfo={({ userInfo, userInfoType }: {

userInfo: unknown;

userInfoType: 'email' | 'name' | 'user\_id';

}) **=>** setMockUserInfo({ userInfo, userInfoType })}

/>

)}

{(mockUserInfo && site\_text.trim() !== "" && site\_text\_translation\_id) ? (

<DiscussionBox

tableName={site\_text}

rowId={+site\_text\_translation\_id}

userInfoType={mockUserInfo.userInfoType}

userInfo={mockUserInfo.userInfo}

style={{

height: '100%',

padding: '0px',

}}

/>

) : null}

</div>

</IonContent>

);

}

In the above example, I used the MockLoginForm component to get a userId.

Now, we are using keycloak for SSO, and it is still developing. So I used this component temporarily.

