//æøå

import \* as React from 'react';

import { RouteComponentProps } from 'react-router';

import { Http } from "../utilities/http";

import { User } from "../utilities/user";

//this React component is just like login, except this registers a new user on the site

export class Register extends React.Component<RouteComponentProps<{}>, {}> {

username: HTMLInputElement | null = null;

password1: HTMLInputElement | null = null;

password2: HTMLInputElement | null = null;

constructor() {

super();

}

public render() {

return <div>

<h2>register</h2>

<span>username: </span><input ref={(input) => this.username = input} onKeyUp={e => this.loginSender(e)} type="text" /><br />

<span>password: </span><input ref={(input) => this.password1 = input} onKeyUp={e => this.loginSender(e)} type="password" /><br />

<span>password: </span><input ref={(input) => this.password2 = input} onKeyUp={e => this.loginSender(e)} type="password" /><br />

<input type="submit" onClick={() => this.submit()} /><br />

</div>

}

public loginSender(e: React.KeyboardEvent<HTMLInputElement>) {

if (e.key == "Enter") {

this.submit();

}

}

public submit() {

if (this.username != null && this.password1 != null && this.password2 != null) {

let username: HTMLInputElement = this.username as HTMLInputElement;

let password1: HTMLInputElement = this.password1 as HTMLInputElement;

let password2: HTMLInputElement = this.password2 as HTMLInputElement;

if (password2.value != password1.value) {

alert("passwords doesnt match")

}

Http.post<CheckResult>('api/UserGateway/New', {

username: username.value,

password: password1.value,

}).then(data => {

if (data.authorized) {

User.tryLogin(username.value, password1.value, (n: User) => {

this.props.history.push("home");

}, (n: string) => {

alert(n);

});

}

else {

alert(data.message);

}

});

}

}

}

interface CheckResult {

type: string;

message: string;

authorized: boolean;

cookieKey: string | null;

}