

Code

main

Go to file

> .vscode

> public

> src

> Context

> Dashboard

> Pages

> AllAuthors

> AllBooks

> AllBooksDetails

> Authentication

Loginjs

Registerjs

> Author

> CategoryDetails

> Contact

> ErrorPage

> Home

> PublicationDetails

> Subscribe

> assets

> components

> BookingForm

> PricingPlan

> SearchComponent

Documentation • Share feedback

kitabkhazanah-client-v1 / src / Pages / Authentication / Login.js

Murad501 code updated

6edb17c · last month History

Code Blame 298 lines (287 loc) · 12.4 KB

Raw Download Edit View

```
1 import React, { useContext, useEffect, useState } from "react";
2 import { useLocation, useNavigate } from "react-router-dom";
3 import { useForm } from "react-hook-form";
4 import auth from "../../firebaseInit";
5 import {
6   signInWithEmailAndPassword,
7   signInWithGoogle,
8 } from "react-firebase-hooks/auth";
9 import { toast } from "react-hot-toast";
10 import { fetchUrls } from "../../components/Shared/FetchUrls";
11 import axios from "axios";
12 import { useContext } from "../../App";
13 import LoadingButton from "../../components/Shared/LoadingButton";
14
15 const Login = () => {
16   const navigate = useNavigate();
17   const location = useLocation();
18   const from = location.state?.from?.pathname || "/";
19   const [showPassword, setShowPassword] = useState(false);
20   const [isLoading, setIsLoading] = useState(false);
21   const {
22     register,
23     handleSubmit,
24     formState: { errors },
25   } = useForm();
26   const [signInWithEmailAndPassword, user, loading, error] =
27     signInWithEmailAndPassword(auth);
28   const [signInWithGoogle, gUser] = signInWithGoogle(auth);
29   const value = useContext(dataContext);
30   const { setUpdate } = value;
31
32   const onSubmit = (data) => {
33     setIsLoading(true);
34     signInWithEmailAndPassword(data.email, data.password);
35   };
36
37   useEffect(() => {
38     if (gUser) {
39       axios
40         .post(fetchUrls("auth/register"), {
41           username: gUser?.user?.displayName,
42           email: gUser?.user?.email,
43           password: gUser?.user?.uid,
44         })
45         .then((res) => {
46           if (res.status === 200) {
47             navigate(from, { replace: true });
48             toast.success("User register Successfully!");
49             localStorage.setItem(
50               "user",
51               JSON.stringify({
52                 ...res.data?.user,
53                 photo: gUser?.user?.photoURL,
54               })
55             );
56             localStorage.setItem("token", JSON.stringify(res?.data?.token));
57             setUpdate(Math.random());
58             // localStorage.setItem("token", JSON.stringify(user?.user?.accessToken));
59           } else {
60             navigate(from, { replace: true });
61             setUpdate(Math.random());
62             localStorage.setItem(
63               "user",
64               JSON.stringify({
65                 ...res.data?.user,
66                 photo: gUser?.user?.photoURL,
67               })
68             );
69             localStorage.setItem("token", JSON.stringify(res?.data?.token));
70             toast.success("Google Login Successfully!");
71           }
72         })
73         .catch((err) => {
74           // toast.error(err.response.data.message)
75         });
76     }
77     if (user) {
78       setIsLoading(true);
79       axios
80         .post(fetchUrls("auth/register"), {
81           name: user?.user?.displayName,
82           email: user?.user?.email,
83           password: user?.user?.uid,
84         })
85         .then((res) => {
86           navigate(from, { replace: true });
87           setUpdate(Math.random());
88           localStorage.setItem("user", JSON.stringify(res.data?.user));
89           localStorage.setItem("token", JSON.stringify(res?.data?.token));
90           setIsLoading(false);
91           toast.success("Login Successfully!");
92         })
93         .catch((err) => {
94           setIsLoading(false);
95         });
96     }
97   }, [gUser, user, from, navigate, setUpdate]);
98
99   let errorMsg;
100
101   if (error) {
102     if (error.code === "auth/user-not-found") {
103       errorMsg = "User not found";
104     }
105     if (error.code === "auth/wrong-password") {
106       errorMsg = "Wrong password";
107     }
108   }
109
110   useEffect(() => {
111     if (loading) {
112       setIsLoading(true);
113     }
114   });
115 }
```

Symbols

Find definitions and references for functions and other symbols in this file by clicking a symbol below or in the code.

Filter symbols

func Login

func onSubmit

```

113     } else {
114         setIsLoading(false);
115     }
116 }, [loading]);
117
118 return (
119     <div>
120         <div className="bg-gradient-to-tl from-green-400 to-indigo-900 w-full pt-10 px-4 h-[93vh]">
121             <div className="flex flex-col items-center justify-center">
122                 <div className="bg-white shadow rounded lg:w-1/3 md:w-1/2 w-full p-10 mt-5 ">
123                     <p>
124                         tabIndex={0}
125                         aria-label="Login to your account"
126                         className="text-2xl font-extrabold leading-6 text-gray-800"
127                     >
128                         Login to your account
129                     </p>
130                     <p>
131                         onClick={() => navigate("/register")}
132                         className="text-sm mt-4 font-medium leading-none text-gray-500 "
133                     >
134                         Don't have an account?{" "}
135                         <span>
136                             tabIndex={0}
137                             role="link"
138                             aria-label="Sign up here"
139                             className="text-sm font-medium leading-none underline text-gray-800 cursor-pointer hover:text-primary"
140                         >
141                             {" "}
142                             Register here
143                         </span>
144                     </p>
145
146                     <form onSubmit={handleSubmit(onSubmit)}>
147                         <div className="mt-4">
148                             <label className="text-sm font-medium leading-none text-gray-800">
149                                 Email
150                             </label>
151                             <input
152                                 aria-label="enter email address"
153                                 type="email"
154                                 className="bg-gray-200 border rounded focus:outline-none text-xs font-medium leading-none text-gray-800 py-3 w-full pl-3 mt-2"
155                                 {...register("email", {
156                                     required: {
157                                         value: true,
158                                         message: "Email is Required",
159                                     },
160                                     pattern: {
161                                         value: /^[a-z0-9]+@[a-z]+\.[a-z]{2,3}/,
162                                         message: "Provide a valid Email",
163                                     },
164                                 })}
165                             />
166                             <label className="label">
167                                 {errors.email?.type === "required" && (
168                                     <span className="label-text-alt text-red-500">
169                                         {errors.email.message}
170                                     </span>
171                                 )}
172                                 {errors.email?.type === "pattern" && (
173                                     <span className="label-text-alt text-red-500">
174                                         {errors.email.message}
175                                     </span>
176                                 )}
177                             </label>
178                         </div>
179                         <div className="mt-6 w-full">
180                             <label className="text-sm font-medium leading-none text-gray-800">
181                                 Password
182                             </label>
183                             <div className="relative flex items-center justify-center">
184                                 <input
185                                     aria-label="enter Password"
186                                     type={showPassword ? "text" : "password"}
187                                     className="bg-gray-200 border rounded focus:outline-none text-xs font-medium leading-none text-gray-800 py-3 w-full pl-3 mt-2"
188                                     {...register("password", {
189                                         required: {
190                                             value: true,
191                                             message: "Password is Required",
192                                         },
193                                         minLength: {
194                                             value: 6,
195                                             message: "Must be 6 characters or longer",
196                                         },
197                                     })}
198                                 />
199                                 <div>
200                                     onClick={() => setShowPassword(!showPassword)}
201                                     className="absolute right-0 mt-2 mr-3 cursor-pointer"
202                                 >
203                                     <svg
204                                         width={16}
205                                         height={16}
206                                         viewBox="0 0 16 16"
207                                         fill="none"
208                                         xmlns="http://www.w3.org/2000/svg"
209                                     >
210                                         <path
211                                             d="M7.99978 2C11.5944 2 14.5851 4.58667 15.2124 8C14.5858 11.4133 11.5944 14 7.99978 14C4.40511 14 1.41444 11.4133 0.787109 8C1.41378 4.58667 4.40511 2 7.99978 2"
212                                             fill="#71717A"
213                                         />
214                                     </svg>
215                                 </div>
216                             </div>
217                             <label className="label">
218                                 {errors.password?.type === "required" && (
219                                     <span className="label-text-alt text-red-500">
220                                         {errors.password.message}
221                                     </span>
222                                 )}
223                                 {errors.password?.type === "minlength" && (
224                                     <span className="label-text-alt text-red-500">
225                                         {errors.password.message}
226                                     </span>
227                                 )}
228                                 {error && (
229                                     <span className="label-text-alt text-red-500">
230                                         {error?.msg}
231                                     </span>
232                                 )}
233                             </label>
234                         </div>
235                         <div className="mt-8">
236                             <button
237                                 type="submit"

```

```

239         aria-label="create my account"
240         className="focus:ring-indigo-700 text-sm font-semibold leading-none text-white focus:outline-none bg-indigo-700 border rounded hover:bg-indigo-600 py-4"
241     >
242         Login
243     </button>
244     ) : (
245         <LoadingButton customClass={"w-full py-3"}></LoadingButton>
246     )}
247 </div>
248 </form>
249
250 <div className="w-full flex items-center justify-between py-5">
251   <hr className="w-full bg-gray-400" />
252   <p className="text-base font-medium leading-4 px<2.5 text-gray-400">
253     OR
254   </p>
255   <hr className="w-full bg-gray-400 " />
256 </div>
257
258 <button
259   onClick={() => signInWithGoogle()}
260   aria-label="Continue with google"
261   className="focus:outline-none py-3.5 px-4 border rounded-lg border-gray-700 flex items-center w-full hover:bg-primary hover:text-white duration-300"
262 >
263   <svg
264     width={19}
265     height={20}
266     viewBox="0 0 19 20"
267     fill="none"
268     xmlns="http://www.w3.org/2000/svg"
269   >
270     <path
271       d="M18.9892 10.1871C18.9892 9.36767 18.9246 8.76973 18.7847 8.14966H9.68848V11.848H15.0277C14.9201 12.767 14.3388 14.1512 13.047 15.0812L13.0289 15.2051L15.2051 15.2051"
272       fill="#4285F4"
273     />
274     <path
275       d="M9.68813 19.9314C12.3039 19.9314 14.4999 19.0455 16.1839 17.5174L13.0467 15.0813C12.2286 15.6682 11.1306 16.0779 9.68813 16.0779C7.12612 16.0779 4.9514 16.0779 3.04857 16.0779"
276       fill="#34A853"
277     />
278     <path
279       d="M4.17667 11.9366C3.97215 11.3165 3.85378 10.6521 3.85378 9.96562C3.85378 9.27905 3.97215 8.6147 4.16591 7.99463L4.1605 7.86257L1.13246 5.44363L1.03339 5.44363"
280       fill="#FB8C00"
281     />
282     <path
283       d="M9.68807 3.85336C11.5073 3.85336 12.7344 4.66168 13.4342 5.33718L16.1684 2.59107C14.4892 0.985496 12.3039 0 9.68807 0C5.89885 0 2.62637 2.23672 1.0332 2.23672"
284       fill="#E84335"
285     />
286   </svg>
287   <p className="text-base font-medium ml-4 ">
288     Continue with Google
289   </p>
290 </button>
291 </div>
292 </div>
293 </div>
294 </div>
295 );
296 };
297
298 export default Login;

```