**142. Redirecting and Wildcard Routes**

* -: Whoa, you already learned a lot about routing.
* We're nearing the end, but did you ever wonder what happens if we enter something here which we don't cover?

Graphical user interface, text, application, email

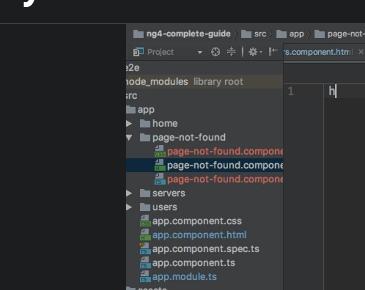
Description automatically generated

* Like, let's say it's just slash 'something' we get an error, because clearly we don't have a route with slash 'something' here.

Graphical user interface, application

Description automatically generated

* So, that is something we probably want to improve.
* We want to do some 404 error handling.
* We maybe want to redirect the user to a specific page whenever he tries to visit a page we don't have.
* These are the topics I'll focus on next.
* Setting up a route which handles all routes we don't have set up, and redirecting requests.
* Not necessarily two topics you use together, but also not that uncommon.
* So, let's start with redirecting.
* Let's say we add a new component to our application, and I will use the CLI 4 that's with the GC for generate component command.
* And I'll name it 'page not found' the page not found component, which was added here in our app folder.



* In this new page not found folder I'll get rid of this back file.
* And here I will simply say in the template this page was not found.
* Something like this.
* Graphical user interface, text

  Description automatically generated
* Now I want to render this page in, let's say, the case that we visit a certain route.

Text

Description automatically generated

* I'll add it here at the bottom where the path is, for now, 'something.
* ' And in the 'something' case here I want to load my page not found component.
* Of course, make sure that you added it to declarations and so on because we're still using it in our app, of course.
* So with this in place, if we now save this, go back to our application and visit slash 'something' we of course see this page was not found.
* Now let's change 'something' to 'not found.

Graphical user interface, text, application, email

Description automatically generated

* ' And let's make sure that once we add another new route, which then is 'something' let's say, we don't also have to add component here, but we redirect to another route, to the not found route, maybe.
* For this, we get another property we can add to any route configure.
* **RedIrect:**
* So, of course, also to any of the other routes if you don't want to specify a component to load, you can say 'redirect to.
* ' Now this is an alternative to component.
* Component loads the specific component.
* Redirect to redirects to another path.
* So here you can specify any path, and the path maybe now is slash not found.

Text

Description automatically generated

* So, now, if you do this, you see that it automatically redirected.

Graphical user interface, text, application

Description automatically generated

* So, let me show you this again by typing slash 'something' watch the URL, it redirects to slash 'not found' and all the other routes of course still work.
* So this is how we can use redirection.

**WILDCARDS:**

* Now, of course, it's not very convenient to define slash 'something' here because you probably can't define all possible wrong URLs the user can enter.
* So, a more convenient way to catch all routes which are not covered by your app is to use the double asterisk route here.
* This is the wildcard route, which means catch all paths you don't know.
* *And the order is super important here.*
* ***Make sure that this very generic route is the last one in your array of routes, because your routes get parsed from top to bottom here.***
* So, if this was actually at the beginning, you would always get redirected, as I can quickly show you by saving this like this.
* Now, no matter where you click, you always are redirected to not found.
* So make sure this is the last route here in your setup.

Text

Description automatically generated

* And with this, now all the other routes work.
* But if you enter anything invalid here, you always get redirected to 'not found' because now you're using this wildcard route, catching all routes which are unknown to angler at this point of time, once it reaches this line here and which then redirects to the not found route, which displays the page not found page or component.

Graphical user interface, text, application, chat or text message, email

Description automatically generated

* This is how you can redirect and make sure you catch routes you're not covering right