

Angular

Capabilities covered

KO observation 1:

Ability to create an Angular application using Typescript & Angular CLI

KO observation 2:

Ability to use and create angular modules, components, controllers

KO observation 3:

Ability to use Routing (module, component), apply route guards, lazy loading of components

Problem Statement

➔ Design a simple Admin page where admin has given access to browse the pages to restricted MIDs.

Assignment

→ Create an Angular App named as “Orchard-Admin” using Angular CLI in the Visual Studio Code IDE.

→ Create a fixed header comprising of various links like “Lead”, “Campus Minds” and “Capabilities”.

→ Design a login Page having field as shown below:-

MID

Password

Login

Reset

→ Using Angular route guard concepts design to execute following use cases:-

#Use Case 1:- If the logged in user MID role is **Lead**, then on clicking “**Lead**” link on the header, user can be able to access the Lead page which contains all the Campus Minds List and also has access to “**Capabilities**” link to see the list of capabilities.

Note:- Create a JSON file named as **CampusMinds.json** which contains details of campus minds.

Create a JSON file named as **Capabilities.json** which contains list of capabilities.

#Use Case 2:- If the user logged in user MID role is **CampusMind**, then on clicking “**Campus Mind**” link on the header, user can only access the Campus minds page and “**Capabilities**” link to see the list of capabilities. If the Campus mind user clicks “Lead” link, the user should be able to see “**Access Denied!! You are not allowed to visit this page.**”

Note:-

- i) You can use necessary lifecycle hooks and apply some Angular Material Design, Bootstrap to beautify the website to cover the Non Kos.
- ii) No need of database storage is required. Use necessary JSON files to read the data.

*****End of Document*****