

Test Project

<skill name>

Submitted by:

Contents

| | |
|--|----------|
| Introduction | 3 |
| Description of project and tasks..... | 3 |
| Data models..... | 3 |
| Guest site..... | 3 |
| Admin site..... | 4 |
| Default data | 4 |
| Instructions to the Competitor..... | 5 |

Introduction

In this module, you need to implement a knowledge-sharing website called "NeedToKnow" in the node.js environment, based on the provided front-end page templates. Guests can browse published knowledge in the Guest site and filter relevant content based on their needs. The website administrator can manage the knowledge content on the website through the Admin site.

Description of project and tasks

Data model:

The following describes the data models that must be used in this project. You can modify these models as needed.

- Admin
 - Name
 - Password
- Knowledge
 - Title
 - Excerpt
 - Content
 - Publish Date
 - Fields that represent the association of this data with user data *author*
- Image
 - Fields that represent the association of this data with knowledge data
 - Url
- Tag
 - Name
- Intermediate data between Tag and Knowledge:
 - Fields that represent the association of this data with knowledge data
 - Fields that represent the association of this data with Tag data

Guest site:

The site has two pages, Home page and Knowledge details page. Guests can access all content on the site without logging in.

- Home page requirements:
 - Filter by tag(s): List all tags related to Knowledge. Guests can click on a tag to filter the knowledge list.
 - Filter by publish date: Filter knowledge data published after the selected date.

- Filter by title: Filter knowledge data that matches the entered title. The entered content does not need to be an exact match to the title, and is case-insensitive.
- Reset button: Clear all filter conditions when clicked.
- Knowledge list: Display knowledge data that meets the filter conditions. If no filter conditions are added, display all knowledge data. The list should be an infinite scroll list. By default, some knowledge data is displayed. As the user scrolls down, more knowledge data is loaded. Each knowledge item in the list contains an image, title, excerpt, publish date, tags, and a detail button.

Knowledge Detail page requirements:

- This page is displayed when the detail button on the home page is clicked.
- Display the corresponding knowledge's title, all images, content, publish date, author, and all tags on this page.
- Back to home button: Display the Home page when clicked.

Admin site:

Pages within this site require authentication through the Login page to access. The access address for the Login page is: http://{host}:{port}/{xx_need_to_know}/admin/login. http://{host}:{port}/{xx_module_c}/admin/login.

Upon successful login, the user will be automatically redirected to the Dashboard page of the Admin site.

Dashboard page requirements:

- Knowledge List: Display all knowledge data. Each knowledge item contains all images, content, publish date, author, all tags, and a delete button.
- When the delete button is clicked, the corresponding knowledge data is deleted.

Knowledge Create page requirements:

- Image selection area: Select at least one image and at most three images. The selected images can be previewed.
- Tags selection area: Select at least one tag or create a new tag.
- Title input field, excerpt input field, content input field, and publish date picker are all required fields.
- Create button: When clicked, the knowledge data is created and the knowledge list page is displayed.

Default Data:

You need to provide 4 admin users with the following information:

Admin 1:

Name: admin

Password: admin

Admin 2:

Name: ntw_001

Password: ntw_001_pwd

Admin 3:

Name: ntw_002

Password: ntw_002_pwd

Admin 4:

Name: ntw_003

Password: ntw_003_pwd

Instructions to the Competitor

- Project should be accessible via /XX_module_c/ where XX is the workstation number.
- All page transitions, data fetching, and data submission operations within both sites are asynchronous and will not cause page refreshes.
- The HTTP server port for this project is 3000.
- For testing convenience, please add at least 5 knowledge entries for each user.
- All assessment is done on server. No assessment process in workstation.
- Please consider easy-to-maintain for the code and images creation.

