

CVWO AY2018/19

Final Submission Writeup

Na Nazhou

User Manual

Sign up

For new users, you can register an account with your username, email and password. Each email can only be used for one account. After signing up, an activation email will be sent to you within 5 minutes to activate your account. By clicking on the link in your activation email, you will then be redirected to an empty to-do list and your account is then successfully activated. You can start using your to-do list now!

Log in & log out

For users who have already signed up, you can log in with your email and password and log out your account by clicking on 'log out' on the navigation bar. While logging in, if you tick the checkbox with the label 'Remember me on this computer', you will not be required to log in again when you want to access your to-do list next time!

Password & username reset

For users who have already logged in, you can reset your username and/or your password by clicking on 'Settings' on the navigation bar. However, please note that your email address cannot be reset.

For users who have forgot their password and are unable to log in, you can click the 'forgot password?' link on the login page. An email containing instructions on how to reset your password will then be sent within 5 minutes. By clicking on the link in the email, you will then be redirected to password reset page.

Adding new task

A new task can be created by selecting 'add new task' and entering relevant information such as title(*), description(optional), deadline(optional) and associated tag names(optional).

Showing the details of an existing task

The details of an existing task (including the description of the task) can be shown to the user by clicking on the '👁' button.

Editing an existing task

An existing task can be edited and updated by selecting '✎' either on the main page or the task details page.

Deleting a task

An existing task can be deleted from the to-do list by clicking on the '🗑' button.

Completing (aka. archiving) a task

An existing task can be marked complete by clicking on the '✓' sign. After a task is marked complete, it will be moved to the list of completed tasks. If you have unintentionally completed a task, clicking on the '+ ' sign adds the task back to the list containing ongoing tasks.

Tagging

There is an optional 'tag' field to be entered when creating a task. You can add new tag by clicking the '+ ' button and remove tags by clicking the '- ' button.



The screenshot shows a form field labeled "Tag(s):" with a blue "+" button next to it. Below the label is a text input field containing the text "This is a tag". To the right of the input field is a small box with a "-" button. Below the input field is another empty text input field, also with a "-" button to its right.

Searching (filtering) by tag name

There is a search bar on the main page and all the tasks containing this tag will be displayed after entering a valid tag name and pressing 'Enter'. If no such tag exists, an warning message will be displayed. If you want to clear the filter, please click the '✕ ' sign.



The screenshot shows a search bar with the text "Search by tag..." and a "✕" button to clear the search. To the right of the search bar is a magnifying glass icon. Below the search bar is a button labeled "Sort by deadline" with a double arrow icon.

Tag(s)



The screenshot shows two tags in the Tag(s) field: "my first tag" and "my second tag".

Furthermore, you can also get all tasks with a particular tag name by clicking the tag name on either the index page or show page. The tag name will be automatically filled into the search bar and the list of tasks with the particular tag will be displayed.

Sorting by deadline

Below the search bar, there is a 'sort by deadline' button. Clicking on it reorders the tasks according to the deadline, with the most urgent task on top.

Learning points

My key takeaway from the CVWO assignment is that I learnt a lot of web development skills.

Ruby on Rails

Initially I found the Rails Guide extremely difficult to understand even after reading on MVC patterns, relational database as suggested in the assignment. It is only after I started to apply and code then I slowly got how it works. Apart from the Rails

Guide, I also briefly read the Ruby on Rails Tutorial book and I learnt how to implement a complete user account system from the book as well.

HTML, CSS, Bootstrap, simple JavaScript and jQuery

I learnt these front-end tools to make the app more aesthetically appealing and user-friendly. As I am not an arts person, I also took a look at various websites for better web design and I made use of many Bootstrap components as well. I used JavaScript and the jQuery library for my sorting, adding tag and deleting tag features. In fact, I felt a lot more features could have been implemented using JavaScript.

Git

I have never used Git before and I only started to use it for this assignment. I found it really useful and I will continue using it in my future projects.

Although my to-do app is still far from perfect, I gained a lot of experience through this assignment.

Areas of improvement

Abstraction and modularisation

I think my code is not very well organized and as the project developed it became increasingly messy. Although 'the Rails way' encourages 'Do Not Repeat Yourself', I feel it is quite hard to structure my code in a neat manner. Especially as I have more ideas and more features, I regretted that I did not properly refactor my code in the early stage. But I tried to use some partials to prevent repetition.

Other cool features & better implementation

As I am completely new to web programming, I have spent a lot of time on learning Rails. Unfortunately I did not have enough time to delve into how to use AJAX in the end. Even for the part using JavaScript and jQuery, I spent quite a long time Googling how to use them.

Test Driven Development (TDD)

After reading the Ruby on Rails Tutorial book, I realised that I could have written tests to facilitate the development process so that I do not need to repeatedly test my app manually. By writing tests before coding, I can also get a clearer picture of what I am trying to implement.

Note

Limitation posed by the SendGrid mailer addon provided by Heroku

As I am using a free Heroku account, the mailing speed is unsatisfactory. It takes on average 5 minutes for users to receive the emails.

Admin User

Use this account to manage users:

Email: admin@cvwo.com

Password: admin123