

## CVWO Assignment Submission

Name: Tiu Wee Han  
Matriculation no: A0190278X  
Heroku link: <https://todo-list-wh.herokuapp.com/>  
Github link: <https://github.com/Williamstwh/CVWO-assignment>

### 1. What I feel about my accomplishments

Before the working on this assignment, I had zero experience with back-end web development. So, being able to build a simple website from scratch (albeit on many layers of abstraction) and to deploy it online was a very rewarding experience for me. In addition to just building the application, I learnt a lot about the underlying concepts behind rails (and webdev in general) and the underlying rationale behind them.

For example, understanding the MVC framework allowed me to visualize the internal processes that were happening step-by-step, and I was able to write code in a much more structured manner. I also came to appreciate convention over configuration as it minimizes the amount of code written (and hence bugs) and improves overall readability.

Through this project, I was also able to grasp the concept behind RESTful APIs, which I found to be confusing before. I was able to implement some AJAX to load content dynamically, and through the process I was able to appreciate the difference in performance and user friendliness as compared to loading static pages for every request.

Overall, I felt that working on this project was very fruitful now that I realize how fundamentally important these concepts are for web developers, extending even beyond rails. It has allowed to pick up ExpressJS much faster than if I did not learn Rails. Frankly, I really enjoyed the learning process and I hope to be able to continue learning more through the CVWO program.

## **2. Short User manual**

### Logging in & Creating a new account:

1. When users first access the website, they will be prompted to log in with their email and password. Alternatively, they can create a new account by clicking the 'Create new account' link at the bottom.
2. If they were to create a new account, they are asked to provide their name, email and password. For the purpose of this assignment, they are also able to create an admin account by specifying true/false under the 'Admin?' section. Creating an account automatically logs you in.
3. Logging in allows a new session to be created, and the user will be redirected to the my/account page.

### Navigation Bar:

4. At the top of the page is the navigation bar. From there, users can access their 'account' page, 'todo-list' page and also log out. Furthermore, admins can access an 'index' page.
5. From the 'account' page (my/account), users are able to manage their details (name, email, password and admin status).
6. Under the 'todo-list' page (my/todos), users are able to manage and query their todo list.
7. The logout button destroys the current session, causing the user to be logged out.
8. Under the 'index' page (admin/users) strictly for admin access, admins can show, edit and delete users.

### Account:

9. Under the 'account' page (my/account), which is actually a 'show' page, users are able to view their details (Name, Email, Password and Admin)
10. Upon clicking the 'edit details' link at the bottom, they are redirected to the 'edit' page (my/account/edit) where they are able to edit and update their user details

### Todolist:

11. Under the 'Todo-List' page (my/todos), users are able to view a table with all their todo-items. Each todo-items contain a title, description and deadline. The page also contains a search bar just below the navbar, and a small form just below that for adding new todo-items.

12. The search bar allows users to search and sort the todo-items in the table. They have the option to search by 'Title' and date range (by default it is from a week before) and sort the displayed results by alphabetical and chronological order.
13. The 'Enter new todo item (AJAX)' bar allows users to create new todo items which are added dynamically to the table (i.e. without reloading the page)
14. The table consists of rows of todo-items, each with their own title, description and deadline. For each item, there is a link to its 'show' page (my/todos/:id), 'edit' page (my/todos/:id/edit) page and a 'delete' button (removes the item dynamically using AJAX).
15. The link at the bottom of the page 'New Todo item (non-AJAX)' allows users to create a new todo-item by redirecting the user to the 'new' page (my/todos/new).

#### Index:

16. The 'index' page (admin/users) on the navigation bar is only accessible to admins, and it contains a table of all the users in the database and their details.
17. Admins are able to show, edit and delete the details of all users with the exception of their own.

#### Logging out:

18. Users are able to log out by clicking the log out button on the navigation bar. This destroys the session that they are currently in, and the user is redirected to the login page.