

# CVWO Final Submission

Chaitanya Baranwal (A0184716X)

January 26, 2019

## 1 Accomplishments

- Created a todo-list with a tagging system, where a task can be tagged as belonging to multiple categories
- Earlier I was using Devise for managing the entire authentication process, but I ended up creating an authentication system from scratch (it has its limitations, but I got to learn more about users and sessions).
- Learnt more about the MVC-structure, and how an idea of how objects in your application will interact makes it much easier to actually implement the app.
- Learnt more about HAML, a templating system for HTML which makes reading and linting of HTML easier.
- Used Bootstrap and AJAX for the first time, learning more about different front-end frameworks and JQuery.
- Database creation and modifying, and management of simple databases using SQL. Also deployed the app on Heroku.
- Learnt Rails from scratch, and implemented a decent project on it in a short period of time!

## 2 User Manual

### 2.1 Login page

As soon as the app is started, a user login page asks the user to enter his credentials regardless of where he wants to go to. This is because there are only a few components in the app, and both should be accessible only after the user is logged in, so the app automatically redirects to login page.

Some seed data is already added into the application, for proof of working purposes. A dummy user can be accessed with the credentials `bruce@gmail.com` and `password1234`.

### 2.2 Navigation bar

A Bootstrap navigation bar has been implemented, which has links to tasks, categories and a user dropdown list. The dropdown list contains signup + login links no one is logged in, else it contains an edit profile + logout links.

### 2.3 Tasks page

The first, and probably the most important page of the application is the tasks page, which lists down all the tasks present in the todo-list. The **New Task** option can be used to add tasks, which opens an AJAX modal form to input the task data. Saving the task closes the modal and adds the task to the main list. The modal form has a `bootstrap-datepicker` to make adding due-date much user-friendlier.

A search bar is also implemented, which can filter tasks based on task-name. It can be extended to filter tasks not only by task-name, but all of its other properties. Changing the checkbox value beside task name dynamically modifies the completed option for task.

### 2.4 Categories page

The categories page is very similar to the tasks page in terms of feel and corresponding components, however, it does not contain components like checkbox which are applicable only to tasks. The category list shows all the

properties of a particular category, and if category description is too long, it just overflows using an ellipsis. Hovering over the category description gives the entire description of a category.

## 2.5 Edit profile page

For users to be able to edit (and delete) their profiles, an edit profile page is implemented where user credentials like email and password can be changed. However, editing the user will only work out after further authentication using a password check.

## 3 Potential improvements

- Reset password option. I tried implementing this but couldn't get the sending emails part to work out for some reason.
- Deleting the completed tasks. This is something trivial which I only thought of towards the end, but a **Delete completed tasks** option beside the **New Task** option can help in reducing clutter.
- An alert banner to show that a task is marked as completed/pending after clicking the checkbox.
- Automatic generation of new categories in case input category during new task is not present.

**Note:** A working copy (several improvements yet to be implemented) of the todo-app has been deployed to Heroku, and can be found at: <http://gentle-coast-15254.herokuapp.com>.