

CVWO Mid-Assignment

Todo App

Use cases:

UC 01 - Register

1. User requests to register
 2. System adds the new user
- Use case ends

UC 02 - Login

1. User requests to login
 2. System logs the user in
- Use case ends

UC 03 - Logout

1. User requests to logout
 2. System logs the user out
- Use case ends

UC 04 - Create Todo

1. User requests to add a todo
 2. System adds the new todo
- Use case ends

UC 05 - Edit Todo Information

1. User requests to edit a todo information
 2. System edits the todo
- Use case ends

UC 06 - Mark Todo as done

1. User requests to mark a todo as done
 2. System marks the todo as done
- Use case ends

UC 07 - Delete Todo

1. User requests to delete a todo
 2. System deletes the todo
- Use case ends

UC 08 - Tag Todo

1. User requests to tag a todo

2. System tags the todo
Use case ends

UC 09 - Search Todo

1. User requests to search todos based on some attributes
2. System shows the todos
Use case ends

Execution plan:

- a. Use iterative process
- b. Deploy backend apis using Heroku, and frontend pages using Netlify
- c. Flow in each iterations:
 1. Create the backend apis using Ruby on Rails and test them using Postman
 2. Create/modify the frontend pages using React.js and Redux
 3. Deploy/push to the deployed website
- d. Iterations:
 1. Basic features (CRUD)
 2. More features (tagging, categorising by tag/due time)
 3. Account-related features (login, logout, register)