

Rentto.me Web Application

Software Engineering CSC648-848 Section 4

Section 4 Team 6

Team Leader: Vanessa Van Dinh

Frontend Lead: Alan Lo, Justin Lau, Vanessa Van Dinh

Backend Lead: Jonathan Tsegaye

Github Master: Camilla Alcartado

Scrum Master: Wilson Wu

Milestone 4

May 2, 2022

Revisions	Date
Version 1	05/02/2022

Table of Contents

Section 1: Product Summary	2
Section 2: QA Testing	3
Section 3: Code Review	4
Section 4: Self-Check	5

Section 1: Product Summary

Rentto.me is designed to give the opportunity to users to rent, try-then-buy, and rent out their tools, gears, or any other items that they need. Rentto.me website will allow users to rent and also lend a variety of tools, gears, etc. that they will need. For example, if an individual needs some tools quickly, to work on their engine on their car, they can easily access our website and browse through the available items that are needed to accomplish their goals on the car. Also, if an individual has tools that have been collecting dust, they can post their tools with the appropriate dollar amount for others to rent out. What makes rentto.me unique from other websites such as Sparetoolz, is that we provide a feature of renter reviews and lender reviews which will let users know if a potential renter has credibility and vice versa. Also, we give the opportunity to users to have the chance to try out a tool and perhaps may consider purchasing this tool in the future through our website. Our priority functions in developing this project are to allow users to register and log in with ease, the ability to post their item for others to rent, then checkout the item and have the opportunity to write a review about the rented or lended tool and have the option to comment in an appropriate manner. We are also working on functions that will allow users to add a rating system and schedule.

URL: rentto.me

Section 2: QA Testing

Unit Testing: (To be completed)

Registration.js (User Registers an Account)

Unit testing for when the user registers an account covers most of the statements. Tests for when the user inputs the list of their information and credentials are tested and passed. In addition to testing the submit button, it passes.

Login.js (User Logins into Their Account)

Unit testing for when the user logs in covers most of the statements. Tests for when the user inputs their login username and password are tested and passed.

CreatePost.js (User Creates a New Rental Post)

Unit testing for when the user creates a new rental post covers most of the statements. Test for when the user inputs the list of their information they want to post are tested and passed.

Integration Testing: (To Be Completed)

Section 3: Code Review

Coding Style:

Components of our react application are neatly organized in their own files and corresponding server/routes files and tests files. In addition, since we are implementing a react application, we are also following javascript coding style with usage of arrow functions and classes in conjunction with JSX.

Github Pull Requests:

1. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/10>
2. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/2>
3. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/4>
4. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/5>
5. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/8>
6. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/9>
7. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/11>
8. <https://github.com/CSC-648-SFSU/csc648-spring22-04-Team06/pull/12>

Section 4: Self-Check

1. **Compatibility:** Application shall be compatible and usable for both PC browsers and mobile browsers.
 - a. **ISSUE** - we are currently facing an issue regarding mobile compatibility, it is mainly on the frontend that the mobile side does not scale to.
2. **Performance:** Users shall be able to use the application without lag. **ON TRACK**
3. **Storage Space:** 2 GB **ON TRACK**
4. **Usability:** Application shall be easy to use and intuitive through the user interface, layout, etc. **ON TRACK**
5. **Security:** Application has password-encryption and tokenized login. **DONE**
6. **Storage:** Application shall be hosted and stored through AWS 2GB.
 - a. **ISSUE** - Application was switched to Digital Ocean and Droplet, as we came across an issue with AWS instance. ****Credentials to access it will be posted on next page****
7. **Availability:** Application shall be available and accessible based on our AWS instance. **ISSUE** - Application was switched to Digital Ocean and Droplet, as we came across an issue with AWS instance.
8. **Fault Tolerance:** Application shall be available based on our AWS instance.
 - a. **ISSUE** - Application was switched to Digital Ocean and Droplet, as we came across an issue with AWS instance.
9. **Development Requirements:**
 - a. Data shall be stored in AWS instance 1vCPU 2 GB ram
 - i. **ISSUE** - Application was switched to Digital Ocean and Droplet, as we came across an issue with AWS instance.
 - b. The code in the master branch of team's github repo should be well maintained, by branches, and guaranteed working at any time **ON TRACK**

****Credentials to access our SSH instance of Digital Ocean and Droplet instance****

Droplet login info:

username: root

password: D.3583qk-YvipYU

ipv4: 137.184.124.26



