

Haus: User Documentation

Izzy Harker, Andrew Rehmann, Alex Peterson Santos, Connie Williamson, and Carter Young

Department of Computer Science: University of Oregon

CS 422: Software Methodologies

Prof. Anthony Hornof

March 13, 2024

[GitHub Link](#)

Overview

This software enables automatic chore scheduling for a group ("Haus") of users. Each user has an individual account, which allows them to:

- view other users
- see their currently assigned chores
- complete a chore
- add a new chore to the Haus
- view documentation on the github page

Users are also able to delete their accounts at any time, and new users can create an account from the login page.

At this point in time, the software runs on a local React/Flask server with only one Haus, and all users are automatically added to that Haus.

For the purposes of software testing and grading of the system, two default user logins are provided:

Username: andrew password: password

Username: Connie password: test

Of course, creating your own account is possible at any time and will be integrated into the Haus.

Here are the functions available from the home screen of the webpage after a user has logged in [Fig. 1]:

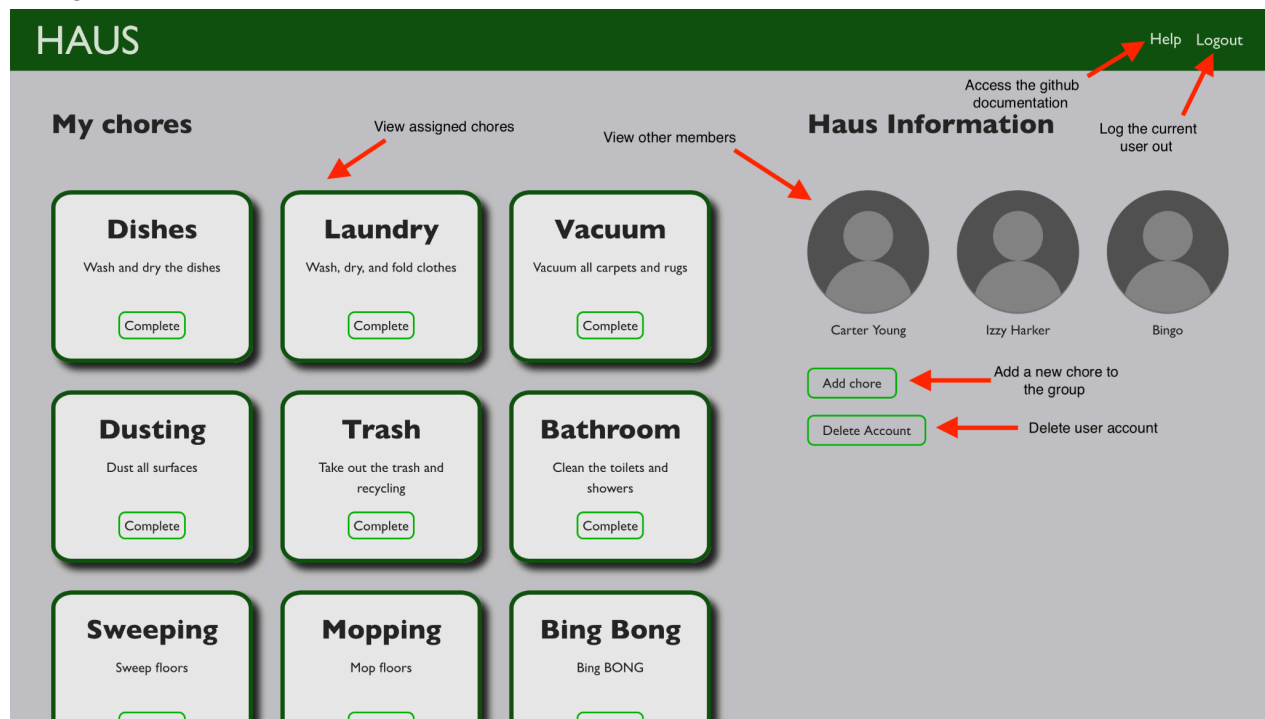


Figure 1: Home page of Haus website

Tasks

Task 0: Installing the software

This software runs on both Python and JavaScript, with local servers for both environments. Please note that this has not been tested on Linux, so we cannot guarantee success within that environment.

Mac

1. Download and install Python 3.12 from python.org. If you already have a python environment, proceed to step 2.

2. Run the following command in the Terminal to install pip (for package management). If you already have pip installed, proceed to step 3 (under "Both"):

```
python -m ensurepip --upgrade
```

Windows

1. Download and install Python 3.12 from python.org. If you already have a python environment, proceed to step 2.

2. Run the following command in the Terminal to install pip (for package management). If you already have pip installed, proceed to step 3 (under "Both"):

```
C:> py -m ensurepip --upgrade
```

Both

3. Install Flask (dependency for the system) by running the following command in the same terminal as step 2:

```
pip install flask
```

4. Download and install Node 20.11.1 from <https://nodejs.org/en>. At the moment, this is the LTS version of the software.

5. Navigate to the "CS422_Haus/Frontend/" directory of the project. Run the command

```
npm install
```

to install the required React dependencies for the project.

6. Follow instructions in Task 1 to start the system

Task 1: Starting the software

The first task a user will need to complete is to start the software. Once all dependencies and packages are installed, follow these instructions:

- 1) In a terminal window, navigate to the main directory of the project ("CS422_Haus").
- 2) Run "python flask_integration.py" to start the Flask server
- 3) Start a new terminal window.
- 4) Navigate to the "Frontend" directory within the project
- 5) Run "npm run dev" to start the React server.
- 6) Cmd + click on the displayed link to open the window in a browser.

Your final terminal windows with both servers running should look like [Fig. 2], although the top file path will be different. Detailed instructions for installation and dependencies can be found under the README.md file in the Github, or under "Task 0" in this document.

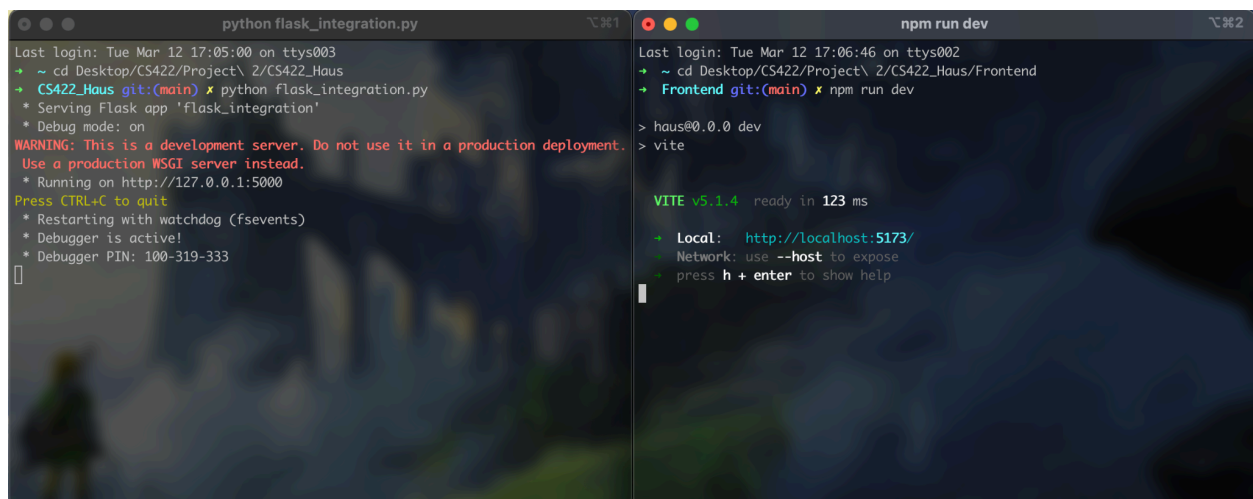


Figure 2. The two terminal windows running the Flask (left) and React (right) servers for the Haus system (pictured on Mac).

Task 2: Login/Create account

The login screen will show up on the first time starting the system, or after a user logs out. For a returning user:

- 1) Enter username and password
- 2) Click the "Login" button (shown in red) to log in

Enter HAUS

Username

Password

Login

[Create account](#)

For a new user:

- 1) Click "Create account" to take you to the account creation screen
- 2) Enter new username and password
- 3) Click the "Create account" button on this screen. This will create the new user account and log you in.
- 4) If you would like to cancel the account creation process, clicking "Cancel" will take you back to the login screen.

HAUS

Help

Join HAUS

Username

Password

Create Account

Cancel

Task 3: Deleting account

A user may want to delete their account. This can be done once they have logged in.

- 1) Click the "Delete account" button (labeled in Fig. 1).
- 2) Enter your password
- 3) Click the Delete Account button to confirm deletion.

HAUS

Help Logout

My chores

Dishes

Wash and dry the dishes

Complete

Laundry

Wash, dry, and fold clothes

Complete

Vacuum

Vacuum all carpets and rugs

Complete

Dusting

Dust all surfaces

Complete

Trash

Take out the trash and recycling

Complete

Bathtub

Clean the bathtub and showers

Complete

Sweeping

Sweep floors

Complete

Mopping

Mop floors

Complete

Bing Bong

Bing BONG

Complete

Haus Information

Current User

Gay Herber

Angie

Add chore

Delete Account

Enter password to delete account

Warning: this action is irreversible!

Password

Delete Account

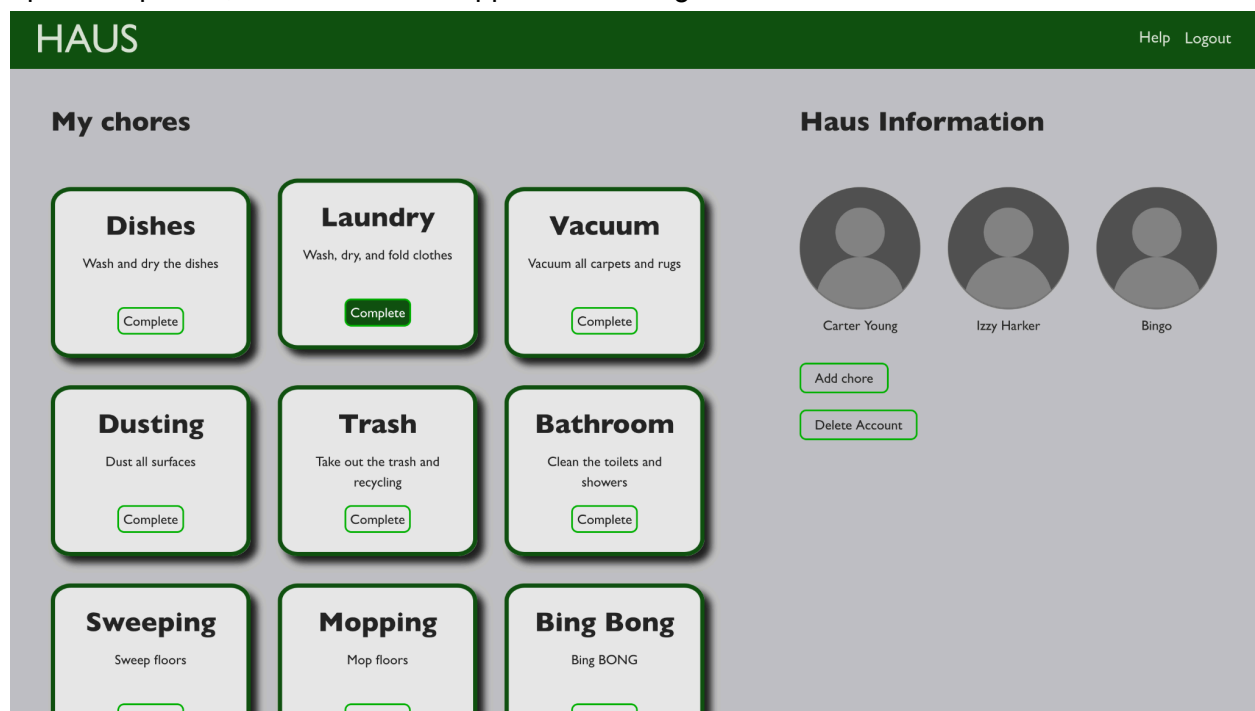
Upon successful account deletion, the user will be returned to the login screen and their account will be deleted.

Task 4: Completing a chore

Perhaps the most common task a user will want to perform is notifying the system that they have completed an assigned chore.

- 1) Log in.
- 2) Locate the desired chore on the grid of cards (under "My chores").
- 3) Click the "Complete" button (shown under "Laundry" in the figure below) to complete the chore.

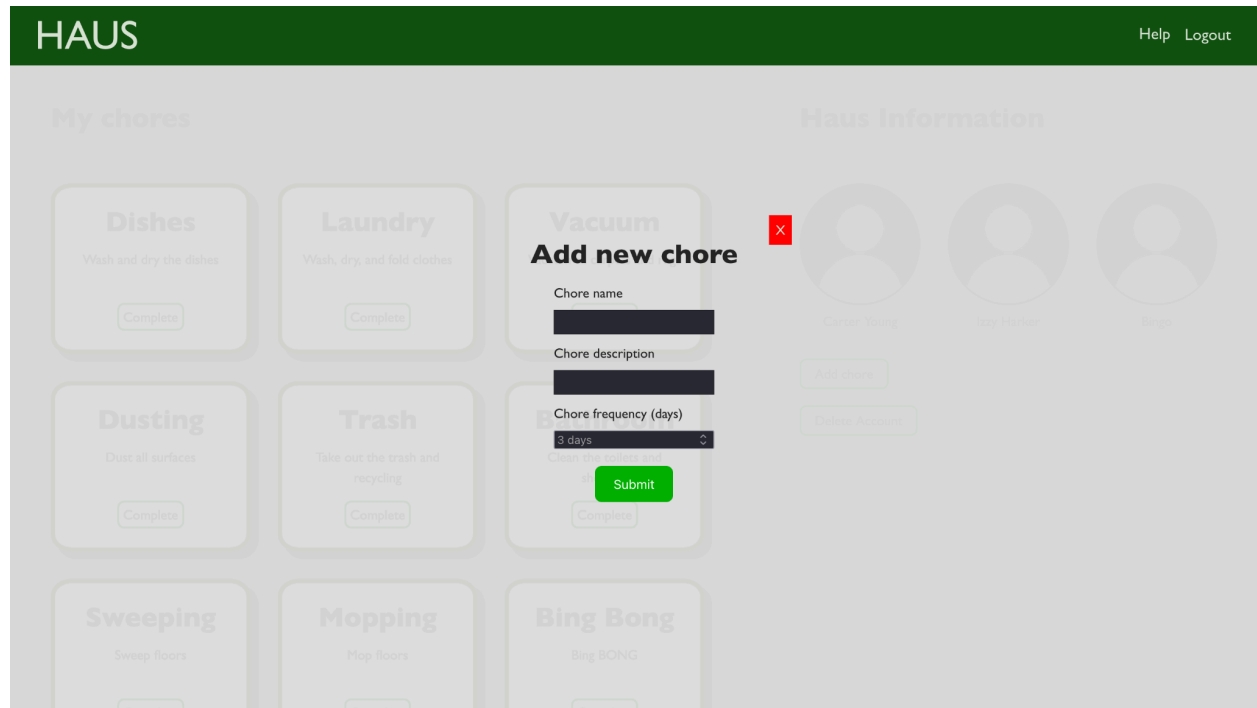
Upon completion, the chore will disappear from the grid.



Task 5: Adding a chore

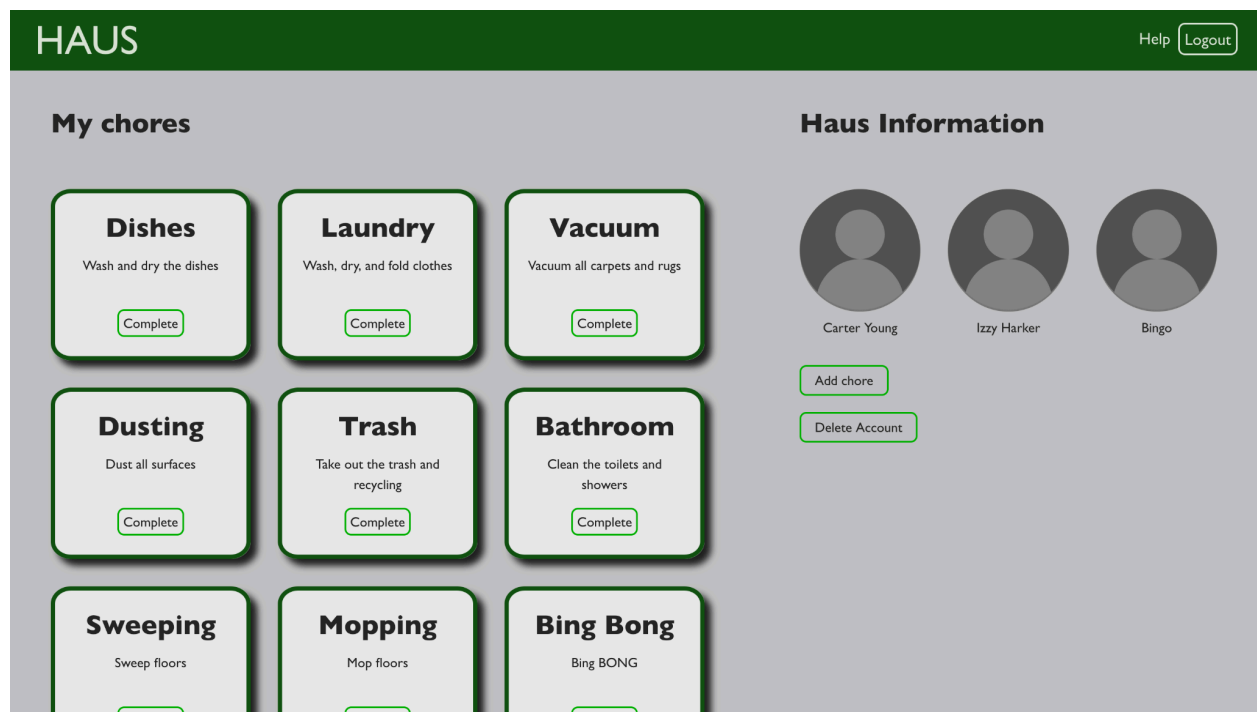
Any user can add a chore to the household by clicking the "Add chore" button. A form will pop up to fill out.

- 1) Input the name, description, frequency, and length of the new chore.
- 2) Click "Submit" to add the new chore to the Haus.



Task 6: Logging out

The system maintains login status across restarts and reloads within the same browser. So, a user will want to logout of the system to protect their privacy. When a user is logged in, they will see a "Logout" button in the top right corner of the screen, which will return them to the login screen when clicked.



Task 7: Finding documentation

The user may want to access the source code or documentation. On both the login screen and main webpage screen, a "Help" button is displayed in the top right. The user can click on this button to open a new tab with the github link to the project.

HAUS Help

Join HAUS

Username

Password

Create Account

[Cancel](#)

https://github.com/lzzyharker/CS422_Haus

Troubleshooting

It is possible that no chores will show up on the home screen, especially for a new user. This most likely only indicates that the user has no current assigned chores, and is not a cause for alarm.

If actions do not work as described in this document and the programmer's documentation, please double check that both the Flask server and React (Vite) server are actively running.

The login is standardized across all browser windows using local storage, so if you would like to test the system with multiple, either log in and out or use separate browsers.

Acknowledgements

The structure of this document is based on "EyeDrawInstructions_V8.pdf", provided to the students of CS422, Winter 2024 by Prof. Anthony Hornof.