HUSTLE

3 Musketeers and a Rifleman

Bennett DenBleyker: User Access and Authorization
Jacob Haight: Account Balances and Transfers
Hagen Larsen: User Experience and Design
Jade Blunt: Job Functionality and Documentation

Design Decisions and Fulfilled Requirements

Reusable templates for similar views

- 4.2.1: A Worker must be able to view available jobs
- 3.3.1: A Customer must be able to change the details of the job if it has not been claimed
- 5.2.1: An Owner must be permitted to review complaints submitted by Customers

Three states of jobs

- 3.2.1: A Customer must be permitted to create and post jobs for Workers to acquire
- 3.7: A Customer can cancel any job up to 24 hours before the start time of the job
- 4.3.1: A Worker must be able to mark accepted jobs as complete

Workers can bid on jobs

- 3.2.2: A Customer must be permitted to approve or decline Workers that have placed a bid on the Customer’s job.
- 4.2.2.3: When bidding on a job, a Worker must also define the wage that they will accept. This can be different than the proposed wage by the Customer.

Surveys of Jobs by Workers

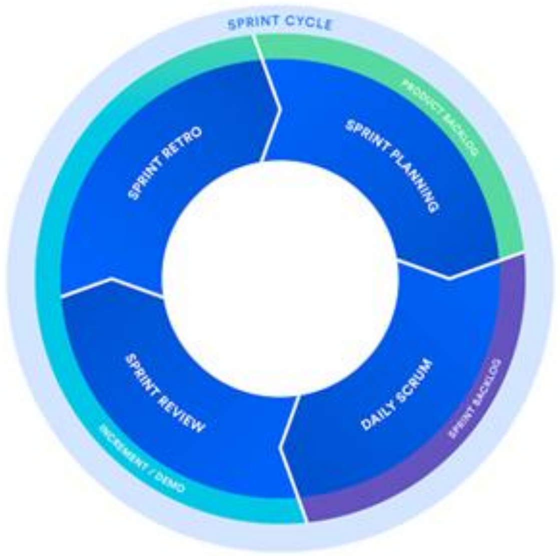
- 4.2.2: A Worker must be able to bid on jobs they would like to work
- 4.3.2.1: The survey must include the degree of complexity of the job (relative to expectation)
- 4.3.2.2: The survey must also include the duration of the job (relative to the expected duration)

SCRUM Practices

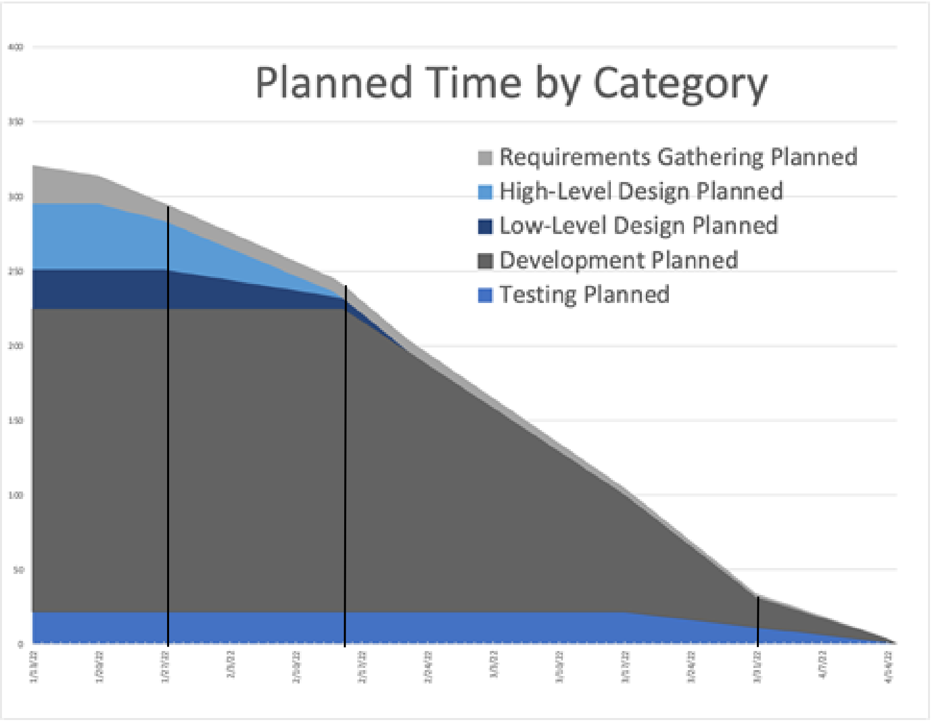
Sprints

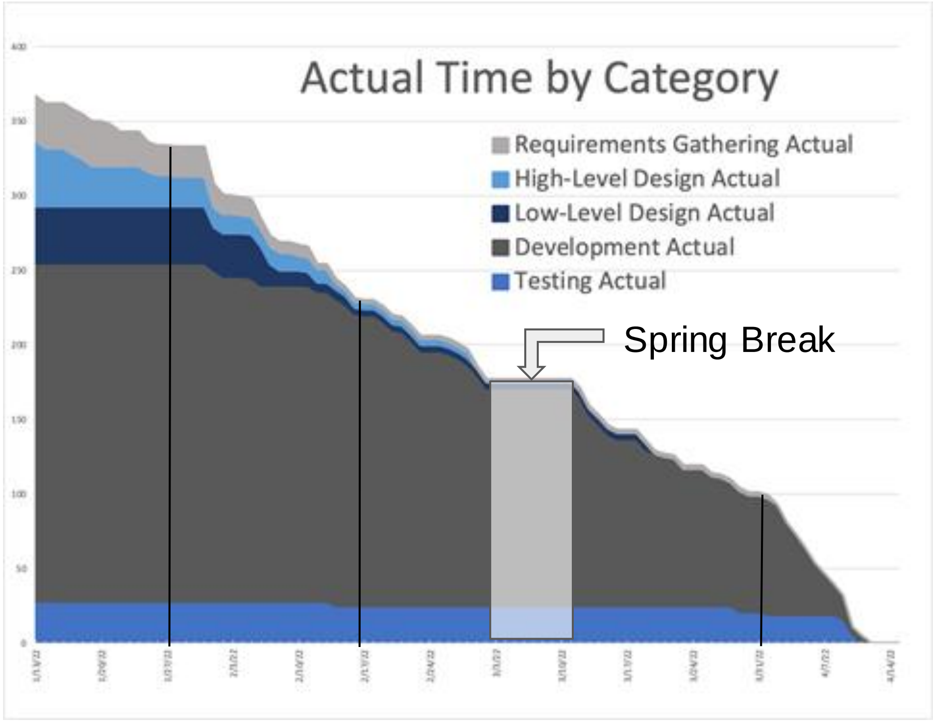
- Two-week periods of time
- Started with a planning meeting
- Multiple standup meetings throughout
- Retrospective at the end
- All tasks logged and tracked in the backlog in Gith
- Scrum Master:
  - In charge of maintaining the backlog and keeping the sprint on task
  - Alternated each sprint

All sprint documents are located in the git repository in docs/planning/



Burndown





Planned Velocity for each Milestone (team hours/day)

1	2	3	4
1.8 h/d	2.8 h/d	4.7 h/d	2.4 h/d

Actual Velocity for each Milestone (team hours/day)

1	2	3	4
2.4 h/d	4.9 h/d	3.1 h/d	7.2 h/d

Testing Procedures

Milestone 3: We created unit testing. These tests can be ran by running:

```
python manage.py test
```

after completing all the build instructions. Extended instructions can be found in the README.md document.

Milestone 4: We ran system testing by testing the app as a whole. Each requirement from our requirements definition document was checked. Every use case was ran through.

Correctness of Code

We are confident that our code is correct and professional. We know that our app does what we intended it to do. It is simple and easy to read. It follows the ACM code of ethics. Our testing procedures have helped prove this.

Deployment

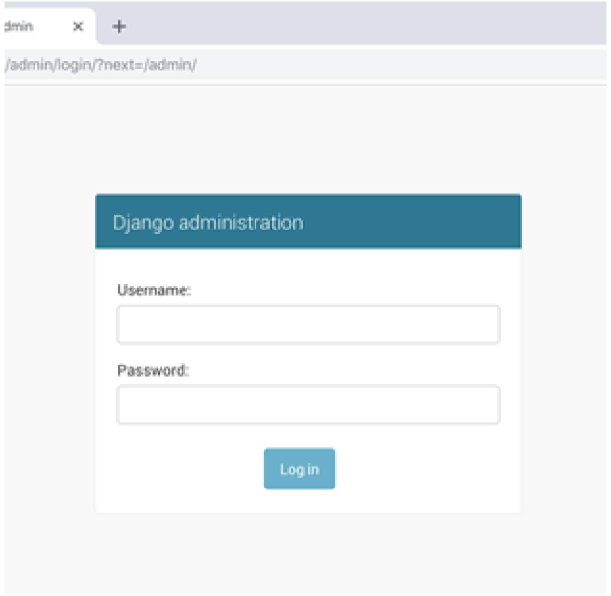
Our code is ready for deployment. We have tested the code and verified its correctness. In the context of this course, deployment means submitting the project and turning it in. However, if we were planning to actually deploy this, we believe it would be ready.

User Registration - Requirement Definition

From Requirements Definition

1.1. A user must authenticate using an email address and password before gaining access to the system.

- **1.1.1.** A user must be able to create a new account by providing an email address and password. The user should also provide:
  - **1.1.1.1.** A preferred contact method (Email, Phone)
  - **1.1.1.2.** A phone number (if phone preferred)
  - **1.1.1.3.** For Customers - An address
- **1.1.2.** For future logins, if a user enters an incorrect email address/password combination, they must be allowed to attempt another login.



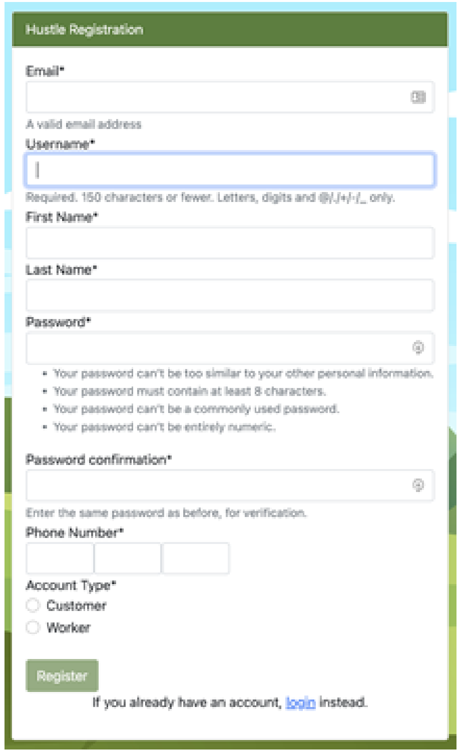
User Registration - Design Choices

Django Built-In Authentication

- Pros
  - Username, Secure Password
  - Users could be affiliated with Groups
- Cons
  - No email, address, phone number, etc.
  - No way for users to register themselves

Leveraged existing functionality,

modified to meet our needs.


Hustle Registration

Email*

A valid email address

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

First Name*

Last Name*

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation*

Enter the same password as before, for verification.

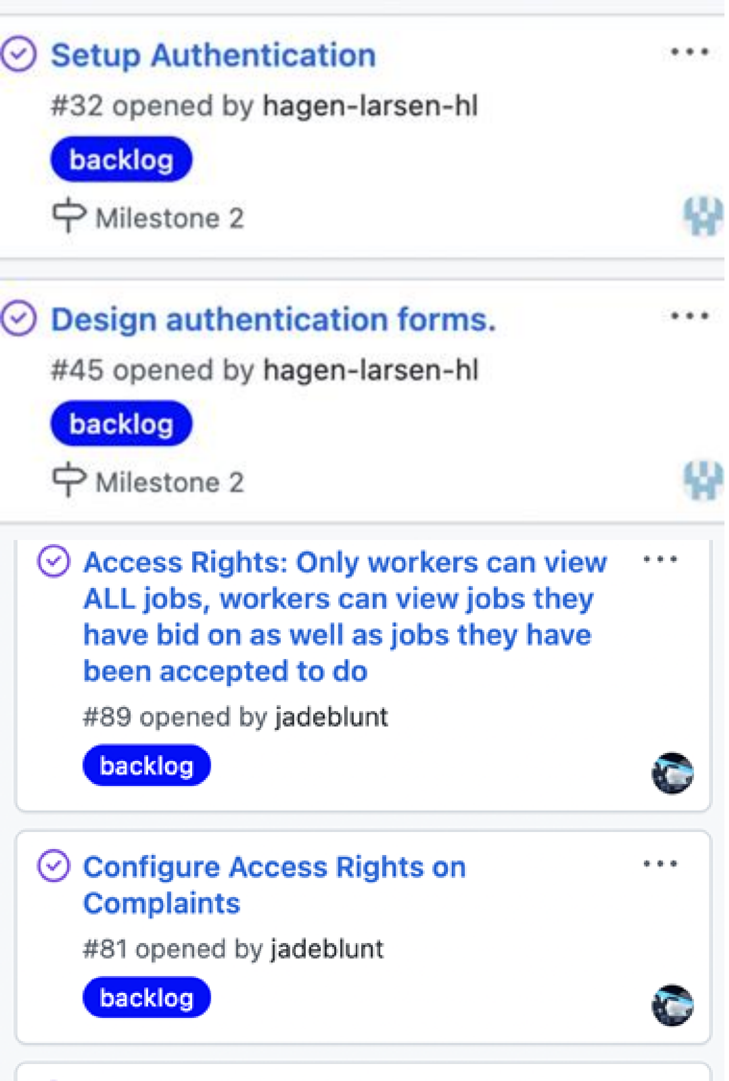
Phone Number*

Account Type*

Customer

Worker

If you already have an account, login instead.


Setup Authentication

#32 opened by hagen-larsen-hl

backlog

Milestone 2

Design authentication forms.

#45 opened by hagen-larsen-hl

backlog

Milestone 2

Access Rights: Only workers can view ALL jobs, workers can view jobs they have bid on as well as jobs they have been accepted to do

#89 opened by jadeblunt

backlog

Configure Access Rights on Complaints

#81 opened by jadeblunt

backlog

User Registration - Tasks

Milestone 2

- 32 - Setup Authentication - Hagen
- 45 - Design Authentication Forms - Hagen

Milestone 3

- Tasks 81, 88, 89 - Cleanup/Simplification - Bennett
- Task 121 - Access Rights - Bennett

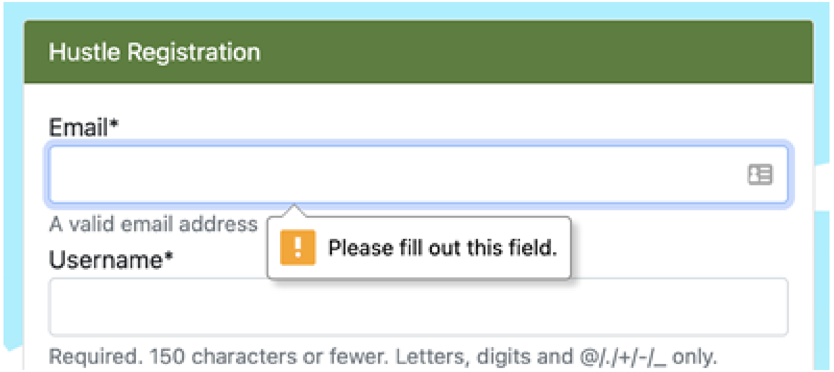
User Registration - Testing, Dependencies

Dependencies

- This part of the app didn't depend on anything else (other than the project being created), however most everything else in the app depended on authentication!
- It was imperative to get this done early

Testing:

- Testing for this requirement was done by executing our use cases while monitoring the data being stored through the Django Admin portal


Hustle Registration

Email*

A valid email address

Please fill out this field.

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

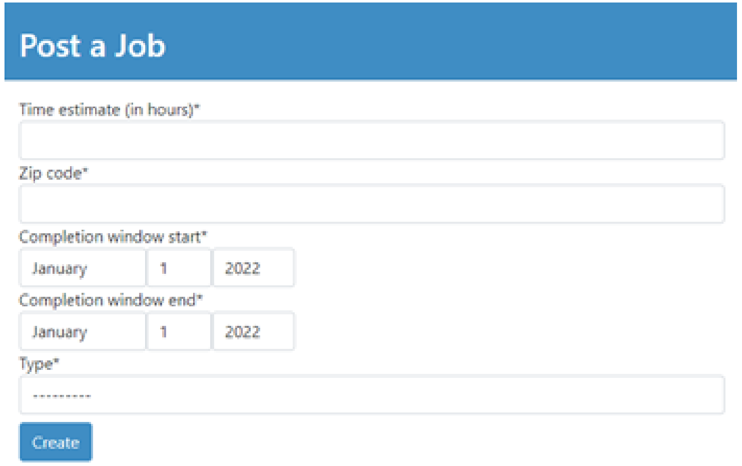
Job Posting - Requirement Definition

From Requirements Definition

3.2. Post Jobs

- **3.2.1.** A Customer must be permitted to create and post jobs for Workers to acquire.
  - **3.2.1.1.** A job must be categorized
  - **3.2.1.2.** A job must include an expected time to completion for the job.
  - **3.2.1.3.** A job must include a desired completion window for when the job can be performed.
- **3.2.2.** A Customer must be permitted to approve or decline Workers that have placed a bid on the Customer's job.
  - **3.2.2.1.** Payment must happen from Customer to Owner at this time.

/jobs/create


Post a Job

Time estimate (in hours)*

Zip code*

Completion window start*

January 1 2022

Completion window end*

January 1 2022

Type*

Job Posting - Design Choices

Selection Method: Bidding

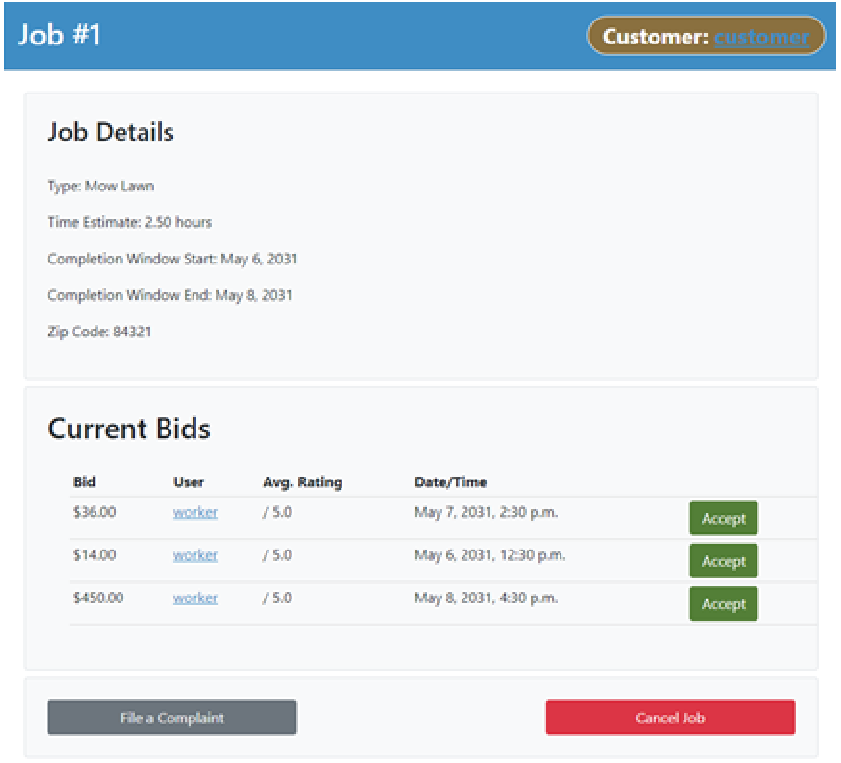
Pros

- Who you pick is your fault
- Creates competition between workers

Cons

- Requires several stages of bidding/acceptance
- Requires some way to value individual workers

`/jobs/view/1`



Job #1 — Customer: customer

Job Details

Type: Mow Lawn

Time Estimate: 2.50 hours

Completion Window Start: May 6, 2031

Completion Window End: May 8, 2031

Zip Code: 84321

Current Bids

Bid	User	Avg. Rating	Date/Time	
$36.00	worker	/ 5.0	May 7, 2031, 2:30 p.m.	Accept
$14.00	worker	/ 5.0	May 6, 2031, 12:30 p.m.	Accept
$450.00	worker	/ 5.0	May 8, 2031, 4:30 p.m.	Accept

Job Posting - Tasks

From Requirements Definition

3.2. Post Jobs

3.2.1. A Customer must be permitted to create and post jobs for Workers to acquire. *(/jobs/create)*

3.2.1.1. A job must be categorized *(JobType Model)*

3.2.1.2. A job must include an expected time to completion for the job. *(Time Estimate field)*

3.2.1.3. A job must include a desired completion window for when the job can be performed. *(Completion Window fields)*

3.2.2. A Customer must be permitted to approve or decline Workers that have placed a bid on the Customer's job. *(Accept button)*

3.2.2.1. Payment must happen from Customer to Owner at this time. *(It does :D)*

Job Form: #55 (Jade)
JobType Model: #57 (Jade)
Job Model: #58 (Jade)
Job View: #72 (Jade)

- Include Bids: #85 (Jade)

Bid Functionality: #84 (Jade)
Bid View: #85 (Jade)
Bid Model: #87 (Jacob)
Access Rights / Ownership: #88 (Bennett)
Access Rights / Roles: #102 (Bennett)
Adding Job States: #106 (Jade)
Styling: #118 (Hagen)
Styling: #158 (Jade)

9 Commits to the Model Form
10 Commits to the Create Template
23 Commits to the View Template
24 Commits to the Python Handler

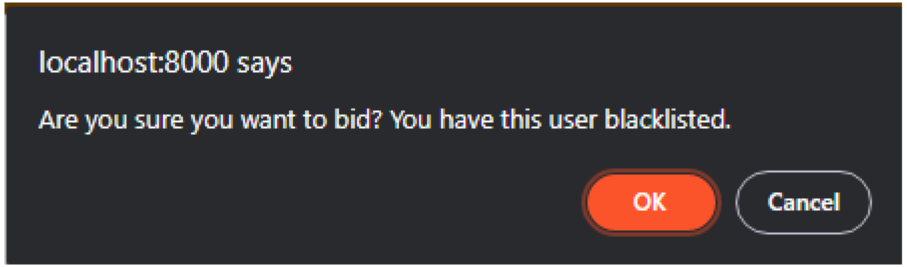
Job Posting - Testing, Dependencies

Dependencies

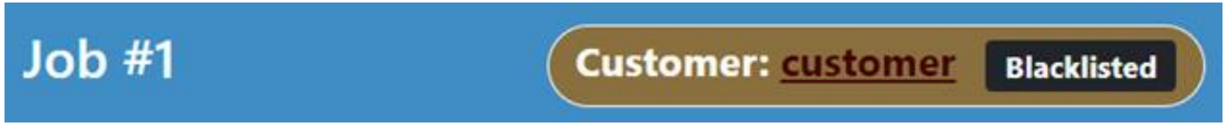
- Job Model
  - User Model
  - JobType Model
- Bid Model

Testing

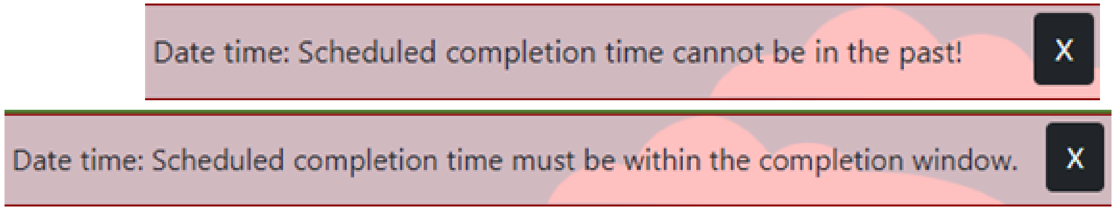
- Use cases
- Django Shell
- A keen eye


localhost:8000 says

Are you sure you want to bid? You have this user blacklisted.


Job #1

Customer: customer Blacklisted

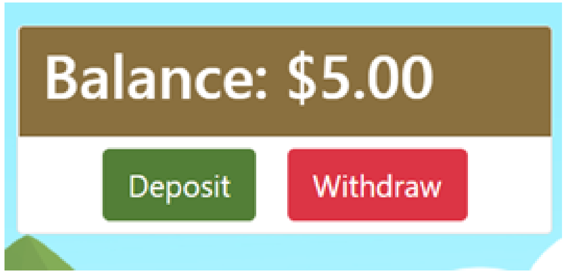

Date time: Scheduled completion time cannot be in the past!

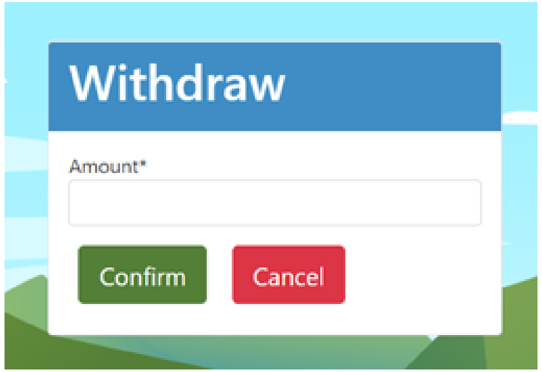
Date time: Scheduled completion time must be within the completion window.

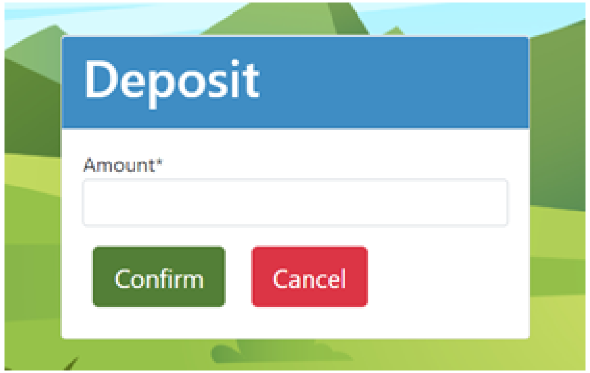
Account Balance - Requirements Definition

2.2. Account Balance

- **2.2.1.** Any User must be allowed to view the balance of their account.
- **2.2.2.** Any User must be allowed to deposit money into their account.
- **2.2.3.** Any User must be allowed to withdraw money from their account.







Account Balance - Design Choices

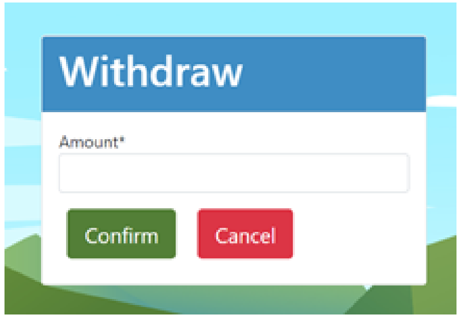
Withdrawing and Depositing

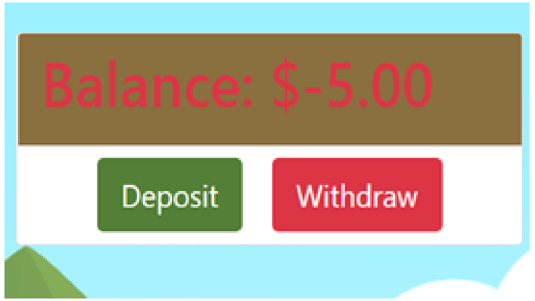
Pros

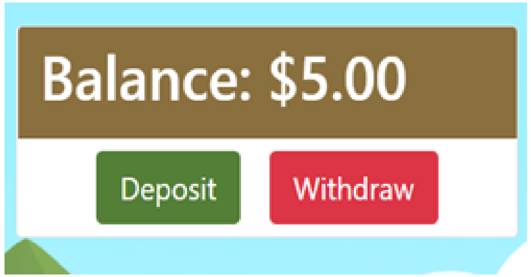
- Pop up is makes the total UI less cluttered
- Users can deposit money and withdraw money

Cons

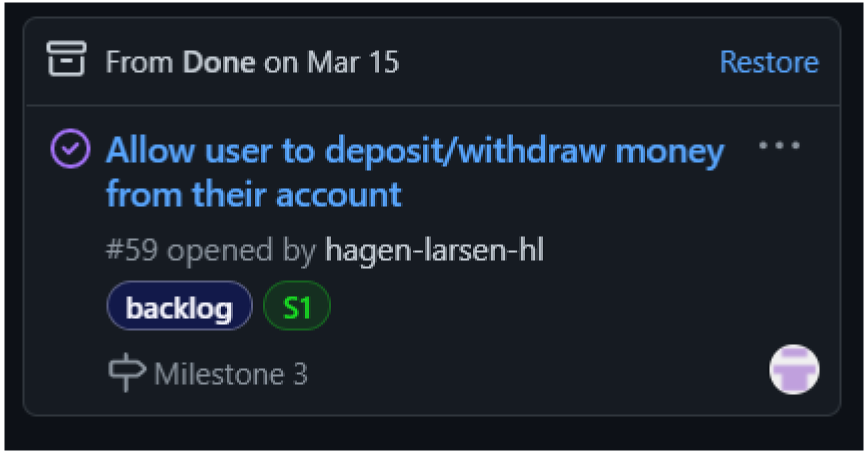
- Popup is in a weird place







Account Balance - Tasks


From **Done** on Mar 15 Restore

Allow user to deposit/withdraw money from their account

#59 opened by hagen-larsen-hl

backlog S1

Milestone 3

Completed in Sprint 1 by Jacob Haight

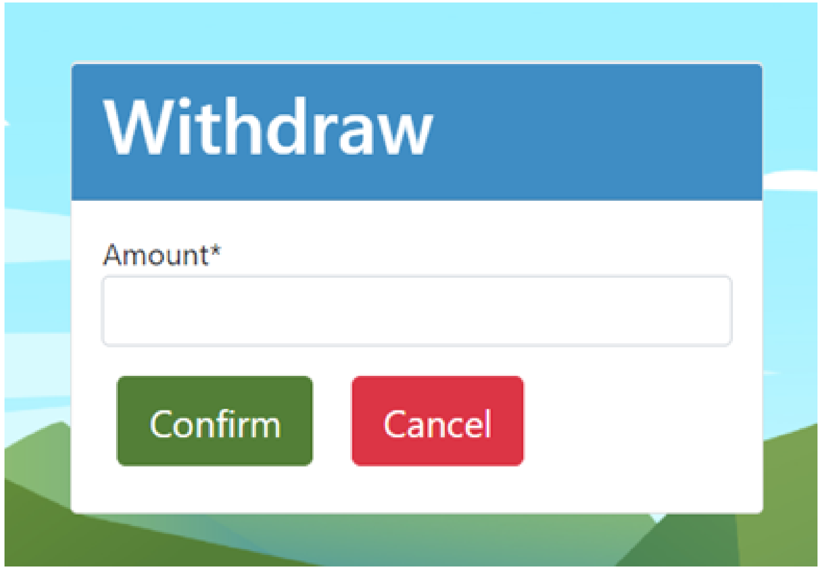
Account Balance - Testing, Dependencies

Dependencies

- This only depended on the user data, because that is where the users money amount is stored
- Other process are dependent on this to be able to access the users money

Testing:

- This is tested in the unit tests to make sure that the user can add and subtract money from the account and that it will save



Videos

Project Resources

Atlassian, & Drumond, C. D. (2022). *What is Scrum?* Atlassian. https://www.atlassian.com/agile/scrum

The web framework for perfectionists with deadlines | *Django*. (2022). Django. https://www.djangoproject.com

Otto, M. J. T. (2022). *Bootstrap*. Bootstrap. https://getbootstrap.com

Github. (2022). *Github Docs*. Github Documentation. https://docs.github.com/en

Atlassian. (2022). *Learn burndown charts with Jira Software*. Burndown Charts.

https://www.atlassian.com/agile/tutorials/burndown-charts

Questions?