

Sprint 1 Planning Document

Team 7

Recyclr

Cory Laker (claker@purdue.edu), Geoffrey Myers (myers259@purdue.edu),
Pranav Vasudha (pvasudha@purdue.edu), Ryan Walden (waldenr@purdue.edu),
Vedant Nevetia (vnevatia@purdue.edu), Zachary Rich (richz@purdue.edu)

Sprint Overview

- The goal of our first sprint is to allow users to sign up for accounts, sign into those accounts, for companies to sign up for accounts, for them to sign into their accounts, allow users to change email, password, and other accounts settings, as well as deleting their accounts, being able to track, share, and pay for their recycling, arrange times for their recyclables to be set up, post messages about their recyclables, and allow admin accounts to ban abusive users.
- The scrum master will be Vedant Nevetia. Meetings will be on Wednesday and Sunday evenings.
- Risks and challenges include:
 - Challenge 1: Getting the team to coordinate and write clean and efficient code
 - Challenge 2: Set up a well optimized database system immune to changes
 - Challenge 3: Coordinating our schedules to be able to complete the tasks assigned to us in a timely manner.
 - Risk 1: Setting up a clean and fast testing framework to write tests on-the-go
 - Risk 2: We will be using some tools we are not all familiar with, and will have to make sure we are using correct conventions and writing good, clean code.
 -

Current Sprint Detail

1. As a user, I would like to sign up with an account on Recyclr.
 - 1.1. Create sign up page
 - 1.1.1. Create a page with a email and password form as well as the Log In With Google button to allow the user to authenticate with Recyclr

- 1.1.2. Assigned to Cory Laker
 - 1.1.3. Workload estimation: 5 hours
 - 1.2. Create route on backend
 - 1.2.1. Create a route which will allow user information to be sent from the frontend and be read on the backend to create an account in the database.
 - 1.2.2. Assigned to Zachary Rich
 - 1.2.3. Workload estimation: 5 hours
 - 1.3. Create User table in database
 - 1.3.1. Create a User table in the database to store user information and last login time.
 - 1.3.2. Assigned to Zachary Rich
 - 1.3.3. Workload estimation: 4 hours
 - 1.4. Test to ensure a user can be signed in
 - 1.4.1. Create tests to show that a user can successfully sign in to the application.
 - 1.4.2. Assigned to Geoffrey Myers
 - 1.4.3. Workload estimation: 2 hours
 - 1.5. Acceptance criteria:
 - 1.5.1. Given a user can visit the signup page, when they enter information and click submit, then it should sent the data to the backend.
 - 1.5.2. Given a user can submit the signup info, when the backend receives the information, then a user account should be created in the database.
 - 1.5.3. Given a user account is created in the database, when the account is created it should return login info to the frontend, then the frontend should update to show the newly created account.
- 2. As a user, I would like to sign in with my account on Recyclr.
 - 2.1. Create sign in page.
 - 2.1.1. Create a page with an email and password form and a submit button, to allow for authentication.
 - 2.1.2. Assigned to Cory Laker.
 - 2.1.3. Workload estimation: 5 hours.
 - 2.2. Create route on backend.
 - 2.2.1. Create a route on the backend that will handle requests to sign into the app given the user's credentials.
 - 2.2.2. Assigned to Vedant Nevetia.
 - 2.2.3. Workload estimation: 5 hours.
 - 2.3. Test to ensure sign in properly functions.
 - 2.3.1. Create tests that when users input correct account, they are logged in, and if user inputs incorrect data, they are not logged in.

- 2.3.2. Assigned to Geoffrey Myers.
 - 2.3.3. Workload Estimation: 2 hours.
- 2.4. Acceptance criteria
 - 2.4.1. Given the sign in page exists, when the user inputs correct account data, then the user is logged in.
 - 2.4.2. Given the sign in page exists, when the user inputs incorrect account data, then the user is prompted about incorrect data.
 - 2.4.3. Given that the user inputs correct account information, when the backend authenticates the user, then they are taken back to the home page.
- 3. As a user, I would like sign up option using Google.
 - 3.1. Add sign up with Google.
 - 3.1.1. On the sign in page, integrate the Google sign in API to allow users to sign up with their Google account.
 - 3.1.2. Assigned to Cory Laker
 - 3.1.3. Workload Estimation: 6 hours
 - 3.2. Create route on backend.
 - 3.2.1. Create a route which will allow user information to be sent from the frontend and be read on the backend to create an account in the database.
 - 3.2.2. Assigned to Zachary Rich
 - 3.2.3. Workload Estimation: 5 hours
 - 3.3. Add user information onto the user table
 - 3.3.1. Add the information given from the Google sign up to the User table in the database.
 - 3.3.2. Assigned to Vedant Nevetia
 - 3.3.3. Workload estimation: 5 hours
 - 3.4. Test to ensure that a user can sign up with Google.
 - 3.4.1. Create tests to show that a user can successfully sign up to the application with Google and that information is correctly stored in the database.
 - 3.4.2. Assigned to Ryan Walden
 - 3.4.3. Workload Estimation: 4 hours
 - 3.5. Acceptance criteria:
 - 3.5.1. Given that the Google sign up is implemented correctly, when a user tries to sign up with Google, then the information will be correctly stored in the database.

- 3.5.2. Given that the Google sign up is implemented correctly, when the backend receives the information, then a user account should be created in the database.
 - 3.5.3. Given that the Google sign up is implemented correctly, when the account is created, then it should return login info to the frontend, then the frontend should update to show the newly created account.
- 4. As a user, I would like sign in/sign out option using Google.
 - 4.1. Add a "sign in with Google" button to the signin page
 - 4.1.1. Create a button on the signin page to allow someone to sign in using Google authentication, rather than our own account creation process.
 - 4.1.2. Assigned to Cory Laker
 - 4.1.3. Workload estimation: 2 hours.
 - 4.2. Add the functionality to sign out from your Google account with regular sign out functionality.
 - 4.2.1. Add the correct functionality needed to make a call to sign someone out of their Google account.
 - 4.2.2. Assigned to Ryan Walden.
 - 4.2.3. Workload estimation: 4 hours.
 - 4.3. Test to ensure that an account can properly be signed in/out of their Google account
 - 4.3.1. Write tests to check whether an account can be properly signed into their Google account, and addition tests to check that a user can properly be signed out of their Google account on the application.
 - 4.3.2. Assigned to Geoffrey Myers
 - 4.3.3. Workload estimation: 2 hours.
 - 4.4. Acceptance criteria:
 - 4.4.1. Given that someone wants to sign in with their Google account, when they click the sign in with Google Button, then it should use Google authentication to sign in the user.
 - 4.4.2. Given that someone is using Google authentication, when they sign in it will use the specific calls to log in with Google authentication, then they will be signed in successfully using Google's authentication API.
 - 4.4.3. Given that someone wants to sign out of their user account, when the click the sign out button, then it should sign them out of their

Google account.

5. As a user, I would like to sign out of my account.
 - 5.1. Create a button on the navbar,
 - 5.1.1. Create a Button, which when clicked, allows the user to sign out of Recyclr
 - 5.1.2. Assigned to Pranav Vasudha
 - 5.1.3. Workload Estimation 3 hours
 - 5.2. Test the functionality for users to be able to properly sign out of their account
 - 5.2.1. Create tests to ensure that there is no error when users are trying to log out of their account.
 - 5.2.2. Assigned to Geoffrey Myers.
 - 5.2.3. Workload Estimation: 3 hours.
 - 5.3. Acceptance Criteria:
 - 5.3.1. Given that the user previously signed in, when they click on the sign out button in the navigation bar they should successfully be signed out.
 - 5.3.2. Given that the user wants to sign out, when they click the signout button, then they should not lose any data they saved previous to signing out.
 - 5.3.3. Given that the the user wants to sign out, when they sign out, then they should be able to sign back in without any problem.
6. As a company, I would like to be able to sign up with an account on Recyclr.
 - 6.1. Create a button to signify a different account type
 - 6.1.1. Create a specific button on the sign up page that allows the new account to be considered a company account and not a basic user account.
 - 6.1.2. Assigned to Geoffrey Myers.
 - 6.1.3. Workload Estimation: 2 hours.
 - 6.2. Create route on backend.
 - 6.2.1. Create a route that will allow for new company information to be sent to the server so that the account may be created.
 - 6.2.2. Assigned to Zachary Rich.
 - 6.2.3. Workload Estimation: 5 hours.
 - 6.3. Create Company Table in the database
 - 6.3.1. Create a new table in the database in order to store the new company's information and credentials.

- 6.3.2. Assigned to Vedant.
 - 6.3.3. Workload Estimation: 4 hours.
- 6.4. Create Tests to ensure new accounts are created properly.
 - 6.4.1. Create tests to ensure new company accounts are properly created, and that a new account can not be created if it already exists.
 - 6.4.2. Assigned to Geoffrey Myers.
 - 6.4.3. Workload Estimation: 2 hours.
- 6.5. Acceptance Criteria:
 - 6.5.1. Given that a button exists, when a company creates an account then that account will be created as a company account.
 - 6.5.2. Given that the user fill out the form, when the button is pressed, then the backend will handle the creation of the account and the additions to the database.
 - 6.5.3. Given that the users successfully creates a company account, when the company logs in to Recyclr, then they are redirected to a proper landing page.
- 7. As a company, I would like to be able to sign into my account on Recyclr.
 - 7.1. Create a row in the sign in page with a link for companies to use to sign in
 - 7.1.1. Create a link that will allow companies to go to a seperate sign in page meant only for accounts of the company type.
 - 7.1.2. Assigned to Pranav Vasudha
 - 7.1.3. Workload Estimation: 4 hours
 - 7.2. Create route on backend.
 - 7.2.1. Create a route that will allow for the company information to be sent to the server so that the account may be signed in.
 - 7.2.2. Assigned to Vedant Nevetia
 - 7.2.3. Workload Estimation: 4 hours
 - 7.3. Add entry to company's row in database with updated sign in time.
 - 7.3.1. Assigned to Zachary Rich
 - 7.3.2. Workload estimation: 4 hours
 - 7.4. Test that the company is able to sign in
 - 7.4.1. Write tests that will ensure that a company is able to sign into their account successfully.
 - 7.4.2. Assigned to Geoffrey Myers
 - 7.4.3. Workload estimation: 2 hours
 - 7.5. Acceptance Criteria:

- 7.5.1. Given that the company has previously created an account, when the company attempts to log in with the correct information, then they will be successfully signed in.
 - 7.5.2. Given that the company has previously created an account, when the company attempts to login with the incorrect information, then they will be prompted to re-enter the correct information.
 - 7.5.3. Given that the company tries to sign in with an account that does not exist, when the company tries to sign in, then they will be prompted that the information does not correlate to an existing account.
- 8. As a company, I would like to be able to sign out of my account.
 - 8.1. Create the ability to sign out.
 - 8.1.1. Create a sign out button on the navigation bar for users.
 - 8.1.2. Assigned to Geoffrey Myers
 - 8.1.3. Work Estimation: 2 hours
 - 8.2. Test the sign out button.
 - 8.2.1. Check to make sure that the users is successfully signed out of their account when the button is clicked.
 - 8.2.2. Assigned to Geoffrey Myers
 - 8.2.3. Work Estimation: 2 hours
 - 8.3. Acceptance Criteria:
 - 8.3.1. Given that the company previously signed in, when they click on the sign out button in the navigation bar, then they should successfully be signed out.
 - 8.3.2. Given that the company wants to sign out, when they click the sign out button, then they should not lose any data they saved previous to signing out.
 - 8.3.3. Given that the the company signs out, when they try to sign in again, then they should be able to sign back in without any problem.
- 9. As a user, I would like to be able to change my email and password.
 - 9.1. On Preferences/Settings page add functionality to update old email and password with verification for both to ensure that they are valid new emails and passwords.
 - 9.1.1. Assigned to Pranav Vasudha
 - 9.1.2. Work Estimation: 5 hours
 - 9.2. Test that this feature works properly.

- 9.2.1. Check to make sure that the email and password are properly updated and a user can login with their updated information.
 - 9.2.2. Assigned to Pranav Vasudha
 - 9.2.3. Workload estimation 4 hours
- 9.3. Acceptance Criteria:
 - 9.3.1. Given that a user visits the profile page, when they update either the email or password or both, their new email and password in the backend should be updated
 - 9.3.2. Given that a user visits the update page, when they make changes but do not click the save button or redirect to another page without saving, no changes to their email or password should be made
 - 9.3.3. Given that this feature is properly implemented, when a user tries to change their password to the same password, then the user will be prompted that the new password is the same as the old password.
- 10. As a user, I would like to be able to change my account settings.
 - 10.1. Create Preferences/Settings page.
 - 10.1.1. Create a preferences page that is accessible via the navbar which takes the user/company to their account page.
 - 10.1.2. Assigned to Pranav Vasudha.
 - 10.1.3. Workload Estimation: 6 hours.
 - 10.2. Create backend route.
 - 10.2.1. Create a backend route for updating settings for the account.
 - 10.2.2. Assigned to Vedant Nevetia.
 - 10.2.3. Workload Estimation: 6 hours.
 - 10.3. Update database.
 - 10.3.1. Update the database to reflect new user preferences/settings.
 - 10.3.2. Assigned to Vedant Nevetia.
 - 10.3.3. Workload Estimation: 5 hours.
 - 10.4. Test that users can successfully change their account settings.
 - 10.4.1. Write tests that can ensure that when a user tries to update their account settings that the information is correctly saved and updated.
 - 10.4.2. Assigned to Pranav Vasudha
 - 10.4.3. Workload estimation: 4 hours
 - 10.5. Acceptance Criteria:

- 10.5.1. Given that the user wants to change a setting, when they click on the submit button, then the changes must reflect on their instance of our web application.
 - 10.5.2. Given that the user wants to change a particular setting, or a few settings, when they click submit, then the database should be updated to reflect the new setting(s).
 - 10.5.3. Given that the user does not click save, when they leave the page, then no user accounts settings are updated.
11. As a user, I would like to be able to delete my account.
- 11.1. Create a delete account button.
 - 11.1.1. On Preferences/Settings page create a button that allows the user to delete their account.
 - 11.1.2. Assigned to Pranav Vasudha
 - 11.1.3. Workload Estimation: 5 hours
 - 11.2. Test that users are able to delete their account.
 - 11.2.1. Test the account details not being available anymore after the user deletes their account.
 - 11.2.2. Assigned to Pranav Vasudha
 - 11.2.3. Workload estimation: 5 hours
 - 11.3. Acceptance Criteria:
 - 11.3.1. Given that a user wants to delete their account, when they click delete account, then they should be prompted to confirm if they want to delete the account or not.
 - 11.3.2. Given that the user confirms that they want to delete their account, their account should be deleted.
 - 11.3.3. Given that the user wants to delete their account, when they don't confirm that they want to delete their account, then their account should not be deleted.
12. As a user, I would like to be able to track my own recycling progress.
- 12.1. Create a Progress page
 - 12.1.1. Create a progress page that is accessible via the navbar or the Profile page.
 - 12.1.2. Assigned to Ryan Walden.
 - 12.1.3. Workload Estimation: 8 hours.
 - 12.2. Create route on backend
 - 12.2.1. Create a route on the backend to fetch progress information.
 - 12.2.2. Assigned to Zachary Rich.

- 12.2.3. Workload Estimation: 5 hours.
- 12.3. Test the ability to track recycling progress.
 - 12.3.1. Test that recycling progress is updated after a recycling transaction was made.
 - 12.3.2. Assigned to Ryan Walden.
 - 12.3.3. Workload estimation: 4 hours.
- 12.4. Acceptance Criteria:
 - 12.4.1. Given that a user visits the profile page, when they click on progress, they should be able to see how many recycling transactions they have made.
 - 12.4.2. Given that a user visits the profile page, when they click on progress, they should be able to see the total weight of all their recycled material.
 - 12.4.3. Given that a user visits the profile page, when they click on progress, they should be able to narrow down their progress to a given year, month or week.
- 13. As a user, I would like to be able to share my recycling progress on social media.
 - 13.1. Social media buttons.
 - 13.1.1. On your recycling process view there should be different social media buttons that share your progress with those respective social media platforms
 - 13.1.2. Assigned to Cory Laker
 - 13.1.3. Workload Estimation: 3 hours
 - 13.2. Testing this feature.
 - 13.2.1. Write tests to ensure that the social media buttons redirect to the right social media page and follow the intended templates
 - 13.2.2. Assigned to Cory Laker
 - 13.2.3. Workload Estimation: 3 hours
 - 13.3. Acceptance Criteria
 - 13.3.1. Given that this feature is implemented correct, when a user presses the share button, then the progress will be shared to the correct social media platform.
 - 13.3.2. Given that the user clicks on the share my recycling progress, the progress shared should be correct and up to date with their actual recycling progress.
 - 13.3.3. Given that the user clicks on the correct social media link,when it redirects them to the correct social media page, then there should

be a way to track if the user actually shared the post or not so as to notify likewise on Recyclr.

14. As a user, I would like to be able to pay for my recycling to be collected at my home.

14.1. Location and payment selection

14.1.1. The user should have an option to be able to select location and payment

14.1.2. Assigned to Geoffrey Myers

14.1.3. Workload Estimation: 8 hours

14.2. Testing the payment option

14.2.1. Testing the transaction details of the payment option to ensure that it performs transactions properly.

14.2.2. Assigned to Geoffrey Myers

14.2.3. Workload Estimation: 4 hours

14.3. Acceptance Criteria:

14.3.1. Given that this feature is implemented correctly, when a users want to pay for their recycling collection, then they should have the option to do so

14.3.2. Given that the user if provides correct transaction and account details, the payment should be successful.

14.3.3. Given that the user provides incorrect transaction and account details, the payment should be unsuccessful.

15. As a user, I would like to be able to use convenient payment methods like PayPal.

15.1. Implement payment method using PayPal

15.1.1. The payment method will be simple and easy to use with the transaction details abstracted away

15.1.2. Assigned to Zachary Rich

15.1.3. Workload Estimation: 10 hours

15.2. Test the payment methods

15.2.1. The PayPal API calls should be tested to ensure that the correct values are being passed in and a valid response is given back.

15.2.2. Assigned to Ryan Walden

15.2.3. Workload estimation: 4 hours

15.3. Acceptance Criteria:

- 15.3.1. Given that the user wants to pay, when they use valid payment details, then the transaction should go through successfully.
 - 15.3.2. Given that the user wants to pay, when they use invalid payment details, then the transaction should not go through successfully.
 - 15.3.3. Given that the user wants to pay, when they try to pay without payment details, then the transaction should not go through successfully.
- 16. As a user, I would like to be able to arrange times and locations for my recyclables to be picked up.
 - 16.1. A calendar needs to be created that users and companies will use to schedule times
 - 16.1.1. Show dates and times in calendar format that users can select from; this may be set to a weekly basis
 - 16.1.2. Assigned to Ryan Walden
 - 16.1.3. Workload Estimation: 9 hours
 - 16.2. Test calendar is visible to both user and company
 - 16.2.1. Calendar dates should be tested to ensure that both the user and company can see the correct arranged time and location
 - 16.2.2. Assigned to Ryan Walden
 - 16.2.3. Workload estimation: 4 hours
 - 16.3. Acceptance Criteria:
 - 16.3.1. Given that the user wants to have recycling picked up, when they select a time and location, they will be notified with confirmation
 - 16.3.2. Given that a company wants to pick up recycling, they should be able to offer time slots to users when they are available to pick-up, and will be alerted when a user selects a time
 - 16.3.3. Given that a user wants to have a weekly recycling schedule with a company, they should be able to select an option to create a weekly schedule with the same time each week
- 17. As a user, I would like to post information about my recyclable material
 - 17.1. Create a page to host listing details for every listing a user creates on Recyclr
 - 17.1.1. Create a page with images, a title and description about the items the user is putting up, as well as a button for a particular company to freeze the listing
 - 17.1.2. Assigned to: Cory Laker
 - 17.1.3. Workload Estimation: 10 hours

- 17.2. Create a route in the backend to store and fetch listing details
 - 17.2.1. We need to create a route into the backend, one POST and one GET to update and fetch listing details respectively.
 - 17.2.2. Assigned to: Vedant Nevetia
 - 17.2.3. Workload Estimation: 6 hours
- 17.3. Create a database entry for storing relevant listing information so it can be fetched by the backend
 - 17.3.1. Have a database table for each listing so that we can fetch all the details when a user/company visits a particular listing page
 - 17.3.2. Assigned to: Vedant Nevetia
 - 17.3.3. Workload Estimation: 5 hours
- 17.4. Test that information is correctly posted
 - 17.4.1. Write tests that ensure that information is correctly sent to the backend, stored in the database, and posted successfully
 - 17.4.2. Assigned to Geoffrey Myers
 - 17.4.3. Workload Estimation: 4 hours
- 17.5. Acceptance criteria:
 - 17.5.1. Given that a user wants to post information about recyclable material, when they go to the information page, then there should be forms for users to input information.
 - 17.5.2. Given that a user fills in information in the forms, when they click a submit button, then the information should be sent to the backend.
 - 17.5.3. Given that information is sent to the backend, when it is received, then it should be stored successfully and success information should be sent to the frontend.
- 18. As a admin, I would like to ban abusive users
 - 18.1. Create admin page
 - 18.1.1. Create a page where only accounts with admin privileges are allowed to visit.
 - 18.1.2. Assigned to Geoffrey Myers
 - 18.1.3. Workload Estimation: 4 hours
 - 18.2. Create options to ban users
 - 18.2.1. Create the option to ban users on the admin page
 - 18.2.2. Assigned to Pranav Vasudha
 - 18.2.3. Workload Estimation: 4 hours
 - 18.3. Test that banning users works

- 18.3.1. Write tests to ensure that an account is properly banned when clicking the ban options
- 18.3.2. Assigned to Cory Laker
- 18.3.3. Workload estimation: 5 hours
- 18.4. Acceptance Criteria:
 - 18.4.1. Given that an admin page exists, when an account with admin privileges tries to visit that page, then they are allowed access while non-admin accounts are not allowed access.
 - 18.4.2. Given that an admin is signed into the admin page, when they click on the ban option, then the account they are trying to ban should be banned.
 - 18.4.3. Given that an account should be banned, when they try to login to their account they should be denied, then the frontend should display that their account is banned and they cannot access the application.

Backlog

1. As a company, I would like to be able to view users who have put up recyclable items for pickup in a given vicinity.
2. As a company, I would like to be able to contact the user whose items I am picking up with information about arrival time, etc.
3. As a user, I would like to be able to track the progress of my recyclable items.
4. As a user, I would like to have an option to be incentivized upon completing a certain number of recycles.
5. As a user, I would like to be able to track my incentivized progress.
6. As a user, I would like to be able to view my invoice for previous transactions.
7. As a company, I would like to be able to charge the user based on the weight of the amount of recyclable material.
8. As a user, I would like to be given subsidized costs when the weight of my recyclable items exceeds a minimum.
9. As a user, I would like to be able to see how available companies are in the area
10. As a company, I would also like to be able to view my transaction history.
11. As a user, I would like to be given a summary of the total weight of recyclable items I have sold, as well as how much space I have saved in a landfill.
12. As a company, I would like to know the total weight of recyclable items I have bought and recycled.
13. As a user, I would like to be able to learn more about recycling.
14. As a user, I would like to be able to sell compost material (if time permits).

15. As a user, I would like to be able to buy compost material (if time permits).
16. As a company, I would like to be able to pick up compost material (if time permits).
17. As a user, I would like to be able to see the current prices of recyclable raw materials.
18. As a user, I would like to be able to rate a company that picked up my recyclable material.
19. As a user, I would like to be able to view a company's rating.
20. As a developer, I would like my code to well documented.
21. As a developer, I would like a streamlined deployment pipeline.
22. As a developer, I would like my application to well tested.

Summary

- Hours for Zachary Rich - 38 hours
- Hours for Vedant Nevetia - 42 hours
- Hours for Cory Laker - 38 hours
- Hours for Geoffrey Myers - 40 hours
- Hours for Pranav Vasudha - 40 hours
- Hours for Ryan Walden - 37 hours