**Trash Collector – Django Web Application – 85 points**

You have been brought on to a team building an application for a business specializing in trash collection. The application includes a fully-functional customer side that allows users to register as customers and utilize the application to start their service. Using the application, customers are able to:

* Choose a day of the week to get their regular trash picked up
* Choose a date for an additional, one-time pickup
* Temporarily suspend their service between two chosen dates
* Track their current account balance

It is your task to build out the employee-side of this application to allow employees to access relevant customer data to service their area of responsibility.

**(5 points)**: As a developer, I want to make good, consistent commits.

**(5 points)**: As a developer, I want to utilize Django Auth Groups for this application.

**(5 points)**: As a developer, I want to adhere to the naming conventions & best practices established in the starter code so that my code will seamlessly integrate to the existing project landscape.

**(10 points)**: As a newly-registered User, I want to complete the registration process and **create** my Employee profile.

**(5 points)**: As a registered employee, I want to be able to **edit** my employee information to change my name and/or zip code.

**(20 points)**: As a registered employee, I want my **index** view to be a list of today’s customers who meet ALL the following criteria:

* Customers in my zip code
* Pickup day is today’s day of week OR One-time pickup date that falls on today
* Non-suspended accounts
* Trash has not yet been picked up today

**(10 points)**: As a registered employee, I want a button/link displayed with each pickup in my daily list that I can click to “confirm” a pickup.

**(5 points)**: As a registered employee, I want all confirmed pickups to have a charge of $20 applied to the customer.

**(10 points)**: As a registered employee, I want to be able to choose a day of the week to filter by, and see all customers who gets a weekly pickup on the day selected. (One time pickups do NOT need to be displayed)

**(5 points)**: As an employee, I want to utilize an ‘employee\_base.html’ parent template that includes a navbar to direct me to links for my default daily view, profile edit, and any other pages needed.

**(5 points)**: As a developer, I want to utilize the Bootstrap Content Delivery Network (CDN) **on** the base.html parent template to stylize my templates in an attractive manner.

**Bonus User Stories** (bonus points only earned if all base user stories attempted):

**(5 points)**: As an employee, I want to be able to select a customer profile and see their address with a pin on a map (Google Maps API, Google Geocoding API).

**(5 points)**: As a developer, I want to integrate a payment API to allow customers to make payments on the application.

**(5 points)**: As an employee, I want to be able to see all of my customers on a map for delivery (multiple pins displayed on one map, unique by day).

**(5 points)**: As a customer, I want an attractive, styled front end experience.

\*You will be presenting this project to the instructors and classmates on the day it is due. The presentation is not graded.

**IMPORTANT NOTES:**

‘accounts’ and ‘customers’ apps are complete and will not need to be changed to achieve MVP.

**For migrations:**

Migration files include a time stamp, so even two partners who migrate the same exact same model properties will get a merge conflict from the timestamp differing.

Have one person in charge of migrations. They change the model, enter the command ‘python manage.py makemigrations’, test it is good by running ‘python manage.py migrate’. If the migration works, they make a new commit to push the created migration file to the repo.

Collaborators will then receive the migration file next time they make a commit and pull. After receiving the file, they will need to run ‘python manage.py migrate’ to have the migration applied to their database.

**If you re-clone the repo:**

local\_settings.py will be gone, be sure to save the one from previous version, or have partner send it over slack if you lost it.