**Y10 Endstand Tracker Website (mobile viewing will be prioritized for finished product)**

[**GitHub Repo Link**](https://github.com/mdsmith577/Y10_Execution_Tracker.git)

**Idea Recap:**

* On all pages in the NavBar at the top:
  + Walgreens logo and Gallo logo with title: “May-June 2019 WAG Y10 Execution Tracker” in the center
  + Hamburger menu button on the left
  + In the top right, have a statement that says “Updated 6/11/2019”
* When page is loaded, have it load the “home” page which will have Total Program Recap table
  + Same thing as the “recap” tab in the excel tracker that I currently update weekly
  + Also, at the bottom, have a “Notes:” section that briefly describes the how void/executed is determined and also: how the tracker site can be navigated and used most effectively
* Have the home page load with the total program recap table at the top (pretty much static)
* Then as you scroll down, have the leaflet.js map, then the total store table, then the progress chart
* Then in separate pages that can be access through the hamburger menu, have pages for: 1) Store Search, 2) Contacts, 3) Submit Comments/Help

**Project Description:**

1. Your task is to tell a story through data visualizations.
   1. My story: create a visual representation of which Walgreens stores have and have not executed the current program – organized by: area, region, state, and distributor
2. Focus on providing users an interactive means to explore data themselves.
   1. I will use leaflet.js and interactive tables
3. Prepare a 10-minute presentation that lays out your theme, coding approach, data munging techniques, and final visualization.
   1. Theme: A mobile and on-the-go tool that helps all Gallo and distributor partners to execute the programs that have been laid out by Walgreens by store
   2. Coding approach:
      1. Backend Python code to read in csv files into sqlite database
      2. Frontend JavaScript to create leaflet.js map and interactive tables
      3. CSS to style the HTML
   3. Data munging techniques: 2 csv files, use sqlite to clear the database table, then import the first csv file (last 2 weeks of previous month, named “1.csv”) and then import the second csv file (program period depletions, named “2.csv”), remove blanks, sum depletions by store #, filter on only UPCs that need to be tracked (which maybe comes from a list in a separate csv file that lists the UPCs and the plano IDs they correspond to named “program\_reference.csv”)
   4. Final visualization: described above
4. You may choose a project of any theme, but we encourage you to think broadly.
5. You will have ample time in class to work with your group, but expect to put in hours outside of class as well.

**Project Requirements: I’ve highlighted components which I’ll be using**

1. Your visualization must include a Python Flask–powered RESTful API, HTML/CSS, JavaScript, and at least one database (SQL, MongoDB, SQLite, etc.).
2. Your project should fall into one of the below four tracks:
   1. A custom “creative” D3.js project (ex: a nonstandard graph or chart)
   2. A combination of web scraping and Leaflet or plotly
   3. A dashboard page with multiple charts that update from the same data
   4. A “thick” server that performs multiple manipulations on data in a database prior to visualization **(must be approved)**
3. Your project should include at least one JS library that we did not cover. React? Victory.js? Nivo.js?
4. Your project must be powered by a data set with at least 100 records. No problem
5. Your project must include some level of user-driven interaction (e.g., menus, dropdowns, textboxes).
6. Your final visualization should ideally include at least three views. Map, interactive table, interactive bar chart

**Additional Notes:**

* Focus on mobile first
  + My company doesn’t really have a lot to offer in terms of mobile tools that are user friendly and smooth
* This tool will provide just the same features as the excel tracker that I prepare and send out weekly, however here’s a full list of features:
  + A made-for-mobile tracker website that communicates Y10 execution (end stand)
  + Tracking will be done by store using depletions data and will be compared to the goal set for each store
  + First, at the top of the tracker site, will be a hamburger menu
    - Although I’m not sure exactly what options I will have in that menu
      * Maybe: 1) Home, 2) Total Program Recap, 3) Map, 4) Store Table, 5) Progress Chart 6) Store Search, 6) Contacts, 7) Submit Comments/Help
  + Then, at the top center of the site, will be a leaflet.js map of the country that visualizes program execution (all brands included in 1 execution #) by: area, region, state, distributor
    - When the site loads, it will show the national view by Area
    - When it loads with the national view by area, I think a heatmap would be most appropriate. Also, I think a heatmap will be best for the regional and state levels too.
      * When the heatmap views (area, region, and state) are engaged, maybe instead of a heatmap maybe I should assign a color to each geography for that view and then in the center of the geography have 1) Geography Name, 2) Geography’s Execution % (so as not to get confused about where one geography begins and where one ends), and 3) Geography’s case opportunity
      * But, once the user gets down to the distributor level, I then want the leaflet.js map to show individual pins on the map for each store
        + I want each pin to have a tooltip when hovered over (or clicked on for mobile) to show: 1) Store #, 2) Store Address, 3) Status: Executed or void, 4) # of Cases needed by brand, 5) # of cases shipped, 6) Case opportunity 7) Link for user to click on that will open a popup window (with an “x” to close when done viewing) that shows the image of the endstand that was assigned to the store by Walgreens before the program period began
    - But I will need to offer drop down menus that will allow the user to zoom in beyond just the Area view, to get down to the region, state, and distributor levels
  + As the user makes selections of geography – the map will update accordingly
  + Then, as the user scrolls down the page, I want to have a table that will update the same way the map did, to show the list of stores by geography that was filtered by the user
    - I’m thinking this table should show the total store list (not just those that haven’t executed) so that the distributor can see each row for each store that is in red (for voids) or green (for executed stores)
    - I want this table to be able to be sorted
      * I would like the headers to be able to be clicked on such that the entire table’s contents are sorted up/down by that particular header
    - Also, I need to make sure there is a “reset” button that will allow the user to reset the filters back to the default of national perspective by area
  + Also, somewhere in here I need to make sure to have the total recap roll-up by area, region, and state for each brand (which is the “recap” tab of my current Y10 excel tracker)
    - I’m not entirely sure how this will be done on mobile – this could be difficult
      * On mobile, I think the same table can be used but will definitely have to be in a different format that will allow the user to see a largely formatted table that might require them to scroll
    - Also, I’m not exactly sure where this should go, I think maybe it should have its own page, instead of being on the homepage
  + Then also, I want to have a page for the “progress chart” which will allow the user to again filter by geography and will read in a csv file that contains a bar chart with % executed on the y-axis and dates updated on the x-axis
    - This would show the execution progress made over time by the geography that has been selected by the user
  + Lastly, if I have time, I would like to be able to integrate the user’s current location so that the leaflet.js map would zoom in on the area surrounding the user’s current location so they can see which stores are nearby