

CRITERION A: PLANNING

Scenario

My client, Mr. Sreedhar Kukunooru, is the president of the Washington Association of Telangana (WATG), a 401(c) non-profit dedicated to promoting "Telangana culture and identity, and contribute towards [the] development of Telangana.". WATG does this by conducting several charity raising events throughout the year, the biggest one of which is Dhoom Dhaam.

Dhoom Dhaam is a competitive talent show held every December that attracts hundreds of people from around the Seattle area that usually turns into the biggest charity-raising event of the year. However, because there are so many people either performing or coming to the event, the WATG board finds it difficult to manually take care of tedious elements like notifying people of their performance times.

Therefore, I decided to create a two-part program:

- A web form that allows people to register for their selected event and records the data onto an easily readable spreadsheet
- A notifier program using that notifies people 10 and 5 minutes before their performance begins

Rationale

I have decided to use a combination of HTML/CSS/JS and Python -- the former for the web-form and the latter for the notifier program. I chose to not put everything into an app because I do not want to spend unnecessary time learning the Android/Swift SDK and/or Flutter -- instead, I can utilize HTML/CSS/JS to have a platform accessible to almost all devices. Additionally, the platform the programs perform on should serve the best interests of the person using them: the web form and the subsequent recording of the data must be on-line, since the clients are directly using them, but the notifier can be run from my own machine quite easily.

The libraries I chose for this project are the Google Sheets API and the Twilio API. I chose the Google Sheets API because it is at the moment the most popular (and therefore maintained) spreadsheet API. I had considered using some easy method like APISpreadsheets, which does everything for you at the click of a button, but decided on this instead. I chose the Twilio API for mostly the same

reasons -- it is an extremely popular software used by companies like Lyft, Airbnb, and Shopify to do actions from maintaining an entire call center to notifying their customers of a discount. Additionally, my client has some experience with Twilio, so it would be easy to converse with him about the workings of the program.

This is a good IA project for me because it will help me manipulate different languages (and APIs) at the same time and discover the connections between them, something that is very important for a programmer to do considering all the up-and-coming programming languages like Go and Dart. Finding the connections between programming languages and APIs will be an essential skill to have when I enter the workforce.

Requirements

- Allow for people to register for the talent show and record their team name, a summary of what they're performing, their email address, and their phone numbers in a form and click submit
 - If the form data they entered is wrong, notify the users that they need to fill out the form again.
 - Have the data from that form directly transfer to a spreadsheet to more easily facilitate manipulations of the data by the organizers.
 - This is already in place with a Google Form, but my client would like to have everything on one page and not link to external services. For more information on this topic, please refer to Appendix A.
-
- After the organizers determine times for each of the performers in the spreadsheet and put the times next to each performer, have an algorithm that calls the data in the spreadsheet and formats it into an appropriate dictionary with nested lists.
 - I don't want to use arrays since Python doesn't support that without me importing another library and I already am importing too many libraries. Even though performance doesn't matter, I would still like to keep the programs' memory use as low as possible. (I realize using a low-level language would help with that, but I do not have the time to learn C.)
 - Have an algorithm that checks the current time and compares it with the times in the formatted dictionary to determine whether to start the notifier program.
 - After checking them in on the day of the event, show the participants what times they are performing and provide them with two reminders to get ready for the performance.

- In previous years, my client had to spend hundreds on pagers to each team and remind them manually when to prepare for the performance, leaving less time for organization of other parts of the event. For more information about our conversation, please refer to Appendix A.