Backend Architecture for UCCE Data Collection System

Dylan Savage, Xiomara Quinonez, Jason Cisneros, Zach Teal

ENGR 110

Fall 2023

Dr. Jessica Kuczenski

12/07/2023

In Collaboration with UCCE - Compost Education Program

Abstract

For this project, the team collaborated with the University of California Cooperative Extension's composting education program to find a way to engage composting community members at home. These composting community members are important because composting is crucial for reducing landfill waste and enriching the soil with valuable nutrients. On a broader scale composting plays a large role in global efforts towards environmental sustainability by mitigating greenhouse gas emissions, conserving resources, and creating a society that promotes responsible waste management practices. The approach we took for this project was to provide community members with real-time data on how they are contributing to Santa Clara's effort to reduce greenhouse gas emissions through composting. In turn, we hope this creates a more interactive compost education program that motivates and excites composters to continue what they are doing and possibly encourage others to do so as well. This was done through the configuration of a Google Cloud Project to enable the Google Sheets API which would allow the website team to pull data from a compiled worksheet of historical data of composting efforts collected by UCCE.

Table of Contents

Introduction

The University of California Cooperative Extension's (UCCE) mission is to provide a way for the University of California to engage with California citizens to achieve innovation in a way that offers sustainability and science literacy. This is done through learning resources, enrichment programs, and volunteer initiatives. Specifically, at Martial Cottle Park, there are showcases to learn more about the park's farming past and different gardening initiatives (University of California Division of Agriculture and Natural Resources, 2023). We are working with a program that is focused on composting. Composting is a natural process that involves the decomposition of organic materials to create nutrient-rich compost, which can then be used to improve soil health. The process begins with the collection of organic waste such as kitchen scraps, yard trimmings, and other biodegradable items which are then broken down into a soil-like material that helps oil structure, fertility, and water retention.

The UCCE Compost Education Program is a program that is focused on educating and getting Santa Clara County residents involved in composting. This is done through community workshops and school presentations in the county to get both children and adults involved. The community workshops allow people to get hands-on practice with composting. People can even enroll in a Master Composter course where they participate in lectures, events, and field trips for several months to continue building their composting skills and knowledge (Regents of the University of California, 2023). Regarding the community partner's needs, Martial Cottle Park needs tools and resources that will allow them to continue advancing their composting initiatives in the community.

For instance, community engagement from local composters is a huge goal of our partner's. The Compost Education Program's vision is to create a multifunctional website that will do a couple of things. First off, it will engage the community by

allowing them to enter their composting data and receive immediate feedback on how their composting is contributing to the overall composting effort in Santa Clara County. It will also collect the information as community members enter it, reducing the reliance on using surveys sent out three months after community members participate in a workshop. Our partner's program is mainly run with help from volunteers, which means that the data needs to be stored in a way that makes it easy to understand and accessible to people who may not have a background in data science. This also means the data storage and website need to be self-contained, low maintenance, and easy to understand if someone must work on it.

The critical Customer of our project will mainly be the UCCE Compost Education Program (CEP) as they will be the ones using, comparing, and sharing the data collected using our backend architecture system. However, the individuals composting and contributing to the data are also critical customers as they will be receiving feedback and data themselves based on their own composting contributions. They will receive analytics showing them how big of a difference they are making and how they are doing compared to other composters, the goal is for this information to motivate them to keep doing what they are doing.

Discussion

Our first task was to determine the focus of our project and research some possible solutions that we could implement. To help us with this we decided to create and narrow down our project specifications. We agreed on four main specifications:

- Store Data - System to store required data
- Shall have an exposed API
- API should be able to handle read and write requests to the database
- API usage needs to be well-documented

Based on this, the team began working on what would become the initial design allocation through Amazon Web Services (AWS) which had most of its functionality available for free, but it had a small service fee required for any computer resource that our web framework may have used like CPU and memory utilization. The team presented the idea of using AWS to work on the database and API implementation. This would have required the use of lambda functions and more coding to implement the solution.

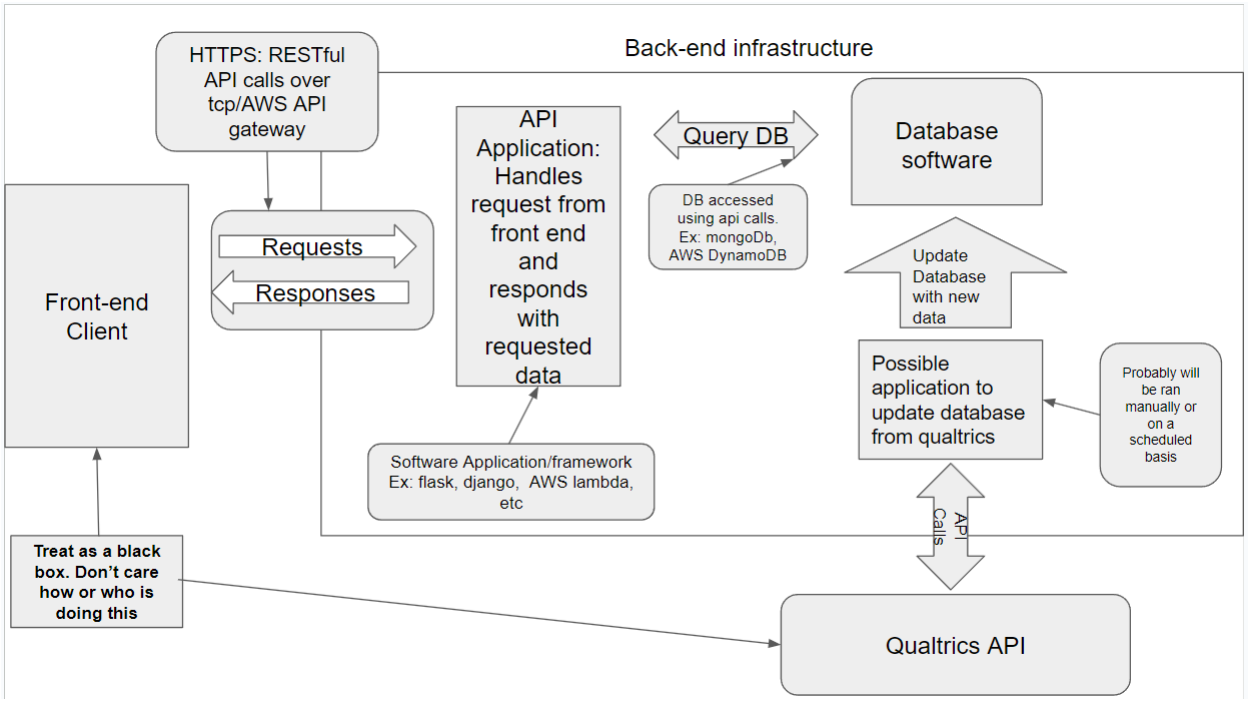

Figure 1: AWS Architecture

However, due to the possible complexity of coding and costs, this would have become an issue for UCCE. So the UCCE suggested that we look into other solutions such as Airtable and Google Sheets APIs. Upon researching Airtable, we found that it would not be very helpful in accomplishing what we are trying to achieve due to its focus on other things like user experience. After researching Google Sheets, we found that it would make the process easier to configure the API and make it accessible for UCCE in a way that is still understandable for people with limited coding knowledge.

In our project, there are a few key organizations that play a pivotal role in addressing public policy issues that arise. One such source of data that will initially populate our database has been collected by the UCCE CEP. Utilizing this data, we must verify with the program whether it can be stored in a publicly accessible Google spreadsheet, or if we need to implement security measures to protect the data. Additionally, if we opt to use a Google Form to collect updated data, we need to ensure compliance with UCCE's privacy and data collection guidelines, as most UCCE sites include a privacy statement (University of California Agriculture and Natural Resources). Furthermore, the California Privacy Protection Agency, which assists in "implementing and enforcing" privacy laws (California Privacy Agency), may offer guidance on regulations that could impact our data collection methods not already covered by UCCE policies.

Our project primarily focuses on civic issues at the community level, with a strong emphasis on community engagement and motivating residents to compost more. By accurately tracking food waste, the project aims to foster a sense of responsibility and environmental consciousness, effectively tackling the civic challenge of sustainable waste management. Encouraging composting mitigates the environmental impact of food waste and promotes a sense of ownership and involvement in sustainability efforts, contributing to a more environmentally mindful and proactive community. This initiative enables community members to actively engage with a crucial civic issue—waste reduction—while also advocating for environmentally responsible practices for the collective well-being of the community and its surroundings.

The policies we must adhere to involve using personal information (like name and location) for limited purposes. For instance, in California, businesses are required to comply with the Consumer Privacy Act when handling personal data, such as geolocation information. When a consumer opts out of data usage, businesses must

ensure that sensitive information is used for restricted purposes only (Rob Bonta, 2023). Additionally, we must navigate several regulations since we handle data from Santa Clara County and the University of California. This data must be adequately protected. Consequently, we must be vigilant about adhering to data privacy laws to ensure that personal data, specifically from individuals completing the UCCE survey for our project, is not compromised.

Results and Analysis

- Product specifications:
    - The project has two main facets: the Google sheet worksheet and the Google Cloud Project (GCP) API configuration.

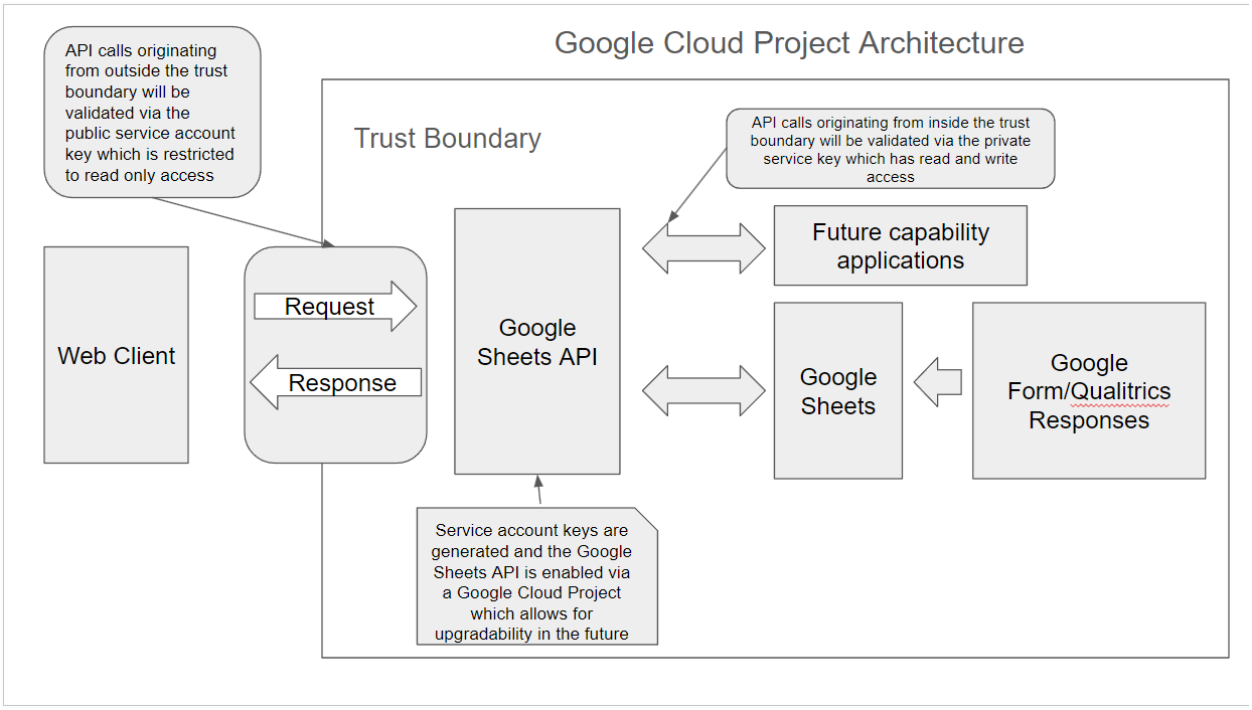

Figure 2: GCP Architecture

- The Google sheet worksheet works as a data store that can take raw data collected from the UCCE via Qualtrics (possible import from our made Google form) and calculates different statistics for the

data, storing the statistics in specific cells that can then be accessed by the website team and other users of the data API (todo: need to create a table of what the raw data inputs are and then the calculations with formulas. Then create a table of spreadsheet IDs and cell ranges for specific statistics to be pulled. Reference and explain in this section).

Sheet	Data Source	Column A (IDs)	Column B (Food Waste)	Column C (Yard Waste)	Total Food Compost	Total Yard Compost	Total Compost	Form Link
UCCE SCC Home Composting	QualtrixRawData	A2:A2000	B2:B2000	C2:C2000	=sum(B2:B2000)	=sum(C2:C2000)	=sum(D2:E2)	N/A
UCCE SCC Home Composting	GoogleFormRawData	A2:A2000	B2:B2000	C2:C2000	=sum(B2:B2000)	=sum(C2:C2000)	=sum(D5:E5)	F1: Form Link

Figure 3: Sheets Cell Documentation

- The Google Cloud project (GCP) is configured to enable access to the Google Sheets data store via the built-in GCP APIs. To enable access to the Google sheet data store, two different APIs must be enabled, one for Google Drive and the other for Google Sheets. Once both are enabled we created two service account credential keys under the Google Sheets API. The service account's main function within our project is to make authorized API calls with the appropriate level of access (Google Cloud 2023). The reason behind the two service accounts is to allow one to have read-only access and the other to allow read/modify permissions, each having distinct use cases. The read-only account is to allow the

authentication and service account information to be stored in an insecure environment like the GitHub pages the web team will be hosting their static website from. The read-only permissions safeguard from malicious actors attempting to manipulate the stored data as the key they would have would have insufficient permissions. The read/modify key is to be used by internal applications that run on secure and trusted networks. This key is provided to allow for potential future teams to reuse the Google sheet and GCP and iterate a fully functional web application and website over. (todo: create a figure showing the data flow between the web client, GCP, and the Google sheet). This key should be kept guarded and if at any time the key is made publicly available, the service account should be deleted from the GCP. (reference a figure or doc outlining how to do this)

- Budget: From the project onset our partner did not disclose an exact budget limit for this project. The only hard limits discussed were from our engineering 110 budget pool which was allocated at one hundred US dollars per team. With that in mind, our team looked for no or low-cost solutions to hosting a database and possibly a web API framework. In the end, we determined GCP with Google Sheets to be our path forward which was a fully free option.

- We used Python and the Google API client python package (Python Software Foundation, 2023) to test that the service account keys worked to access our prototype Google sheet. This also allowed us to ensure that the permissions set for the read-only service account were properly set

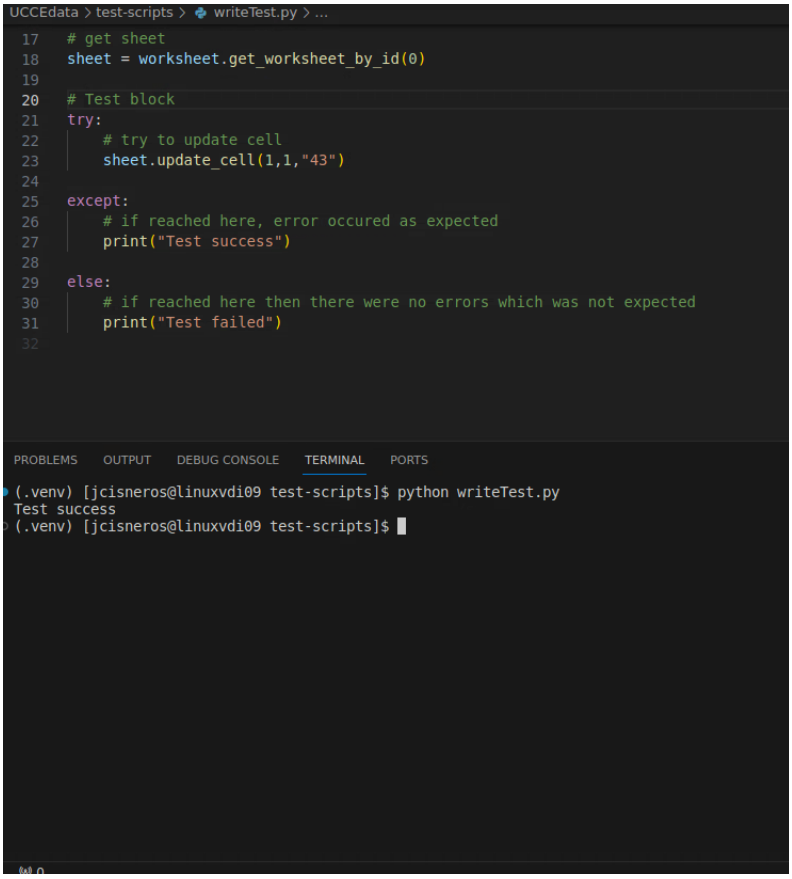

Figure 4: Insufficient permissions test

```
#####################
## check whether a given cell range is valid for reading or writing data in Google Sheets
#####################

import gspread

# path to authentication key json file
key_path = '/DCNFS/users/student/jcisneros/ENGR110/UCCEdata/public-key/public-key.json'

#access api
gc = gspread.service_account(key_path)

#open worksheet
worksheet = gc.open('UCCE SCC Home Composting').sheet1

#tests specific cells
cell_range = "B2:B"

#try to get data from specific range
try:
    data = worksheet.get(cell_range)
    #print message if successful/valid
    print("Cell range '{}' is valid.".format(cell_range))
except gspread.exceptions.APIError as e:
    print("Cell range '{}' is not valid. Error: {}".format(cell_range, str(e)))
```

Figure 5: Cell Range Test

```
####################
## test for worksheet existence using private key
####################

import gspread

#assign path to key_path to be used later
key_path = '/DCNFS/users/student/jcisneros/ENGR110/UCCEdata/private-key/private-key.json'

gc = gspread.service_account(key_path)

#pass in worksheet name
worksheet_name = 'UCCE SCC Home Composting'

try:
    #attempt to open worksheet based on specific name passed in
    worksheet = gc.open(worksheet_name).sheet1
    #print message if successful
    print("Worksheet '{}' exists.".format(worksheet_name))

except gspread.exceptions.SpreadsheetNotFound:
    #print another message if not successful
    print("Worksheet '{}' not found.".format(worksheet_name))
```

Figure 6: Worksheet Existence Test

```
####################
## check how errors are handled
####################

import gspread

# path to authentication key json file
key_path = '/DCNFS/users/student/jcisneros/ENGR110/UCCEdata/public-key/public-key.json'

#access api
gc = gspread.service_account(key_path)

try:
    # Intentionally use an invalid method to trigger an error
    gc.invalid_method()
    print("Error handling test failed")

except gspread.exceptions.APIError as e:
    print("Error handling test success. Error: {}".format(str(e)))
```

Figure 7: Error Handling Test

- With the use of a cell range test, we pass in the key path and access the API based on that key path. The specific worksheet name is then opened and tested for specific cell ranges, which in this case is B2:B. However, the cell range can be modified to any other cell range. Data is then attempted to be retrieved from the specific cell range. If successful, the script outputs that the test was successful. The use of an existence test allows for an accurate test of whether the worksheet attempting to be accessed exists. By passing in the exact worksheet name, the script attempts to open the worksheet based on that name instead of other parameters. If successful, the script outputs that the specific worksheet, based on the worksheet name, exists. If unsuccessful, the script outputs that the worksheet does not exist. With the error handling test, we can check how errors are handled with both the private and public keys. We pass in the key path and access the API based on that key path, which then uses an invalid method to intentionally create an error and test how the error is being handled. Having tests like these is critical in building our project's reliability and allowing us to know its limits. These tests act as safety nets that UCCE can leverage when modifying and adding to the project in the future. With different read, write, and range accesses, we can begin the project with a more stable and better-quality of integration.
- When we handed off the service account key to the website team to test access. It was discovered that service account keys cannot support client-side JavaScript, which is how the website team wrote their website. This made the keys incompatible with the web. With this discovery, we had to work with the web team for a workaround that involved publishing

the Google Sheet with the permission of the partner to allow it to be publicly accessible via an implementation done by the web team as a stopgap.

- (todo: outline Google Forms testing and implement if we have time)

Recommendations:

With the above-mentioned oversight of the incompatibility of the service account API keys and the web team's choice of implementation, we recommended that another team continue development with the Google Cloud Project. They would need to create a proxy API that would redirect the traffic of the website from the Google Sheets API to the proxy API, which would in turn query the Google Sheets API based on the request using the provided private development key. This would allow the Google Sheet to become private again and would allow for more security as the proxy API can do more authentication and security. Since the GCP is already configured, we also recommend the proxy API be hosted within the GCP as each GCP is provided with a free E2 virtual machine with compute engine service (Google 2023).

An additional recommendation is to automate the importation of Qualtrics data using Python scripting and the development API key in order to reduce the need to have someone manually import new data into the Google spreadsheet. This can also leverage the GCP compute engine service possibly.

Bibliography

Google Cloud (2023) Compute Engine. Google, https://cloud.google.com/compute/#pricing Accessed 6 December 2023

Google Cloud (2023, October 30). Service accounts overview. Google, https://cloud.google.com/iam/docs/service-account-overview Accessed 2 Nov. 2023

PyPI (2023, November 7). google-api-python-client 2.107.0. https://pypi.org/project/google-api-python-client/ Accessed 14 Nov. 2023

Regents of the University of California. (2023). Composting Education Program. UCCE Santa Clara County; UCANR. https://cesantaclara.ucanr.edu/Home_Composting_Education/. Accessed 24 Sep. 2023.

Shydlowski, M. (2023, September 19). UC Cooperative Extension Composting Education Program. September 19, 2023. https://drive.google.com/file/d/1q8VEODud2NH023byXphfWiVui2oSjKdY/view. Accessed 24 Sep. 2023.

University of California Division of Agriculture and Natural Resources. (2023). Martial Cottle Park Demonstration Garden and Community Education Center. UC Master Gardeners, Santa Clara County, CA; UCANR. https://mgsantaclara.ucanr.edu/demonstration-gardens/martial-cottle-park/. Accessed 24 Sep. 2023.

Appendices

Project Overview:

- Store Data - System to store required data
- Shall have an exposed API
- API should be able to handle read and write requests to the database

Remaining Needs/Challenges:

- Handing the project over to UCCE
- Documentation of how to maintain the project
- Finding out any safety/privacy restrictions that UCCE may have
- Replacing Qualtrics
    - With our Google Form, we could replace and simplify their current system