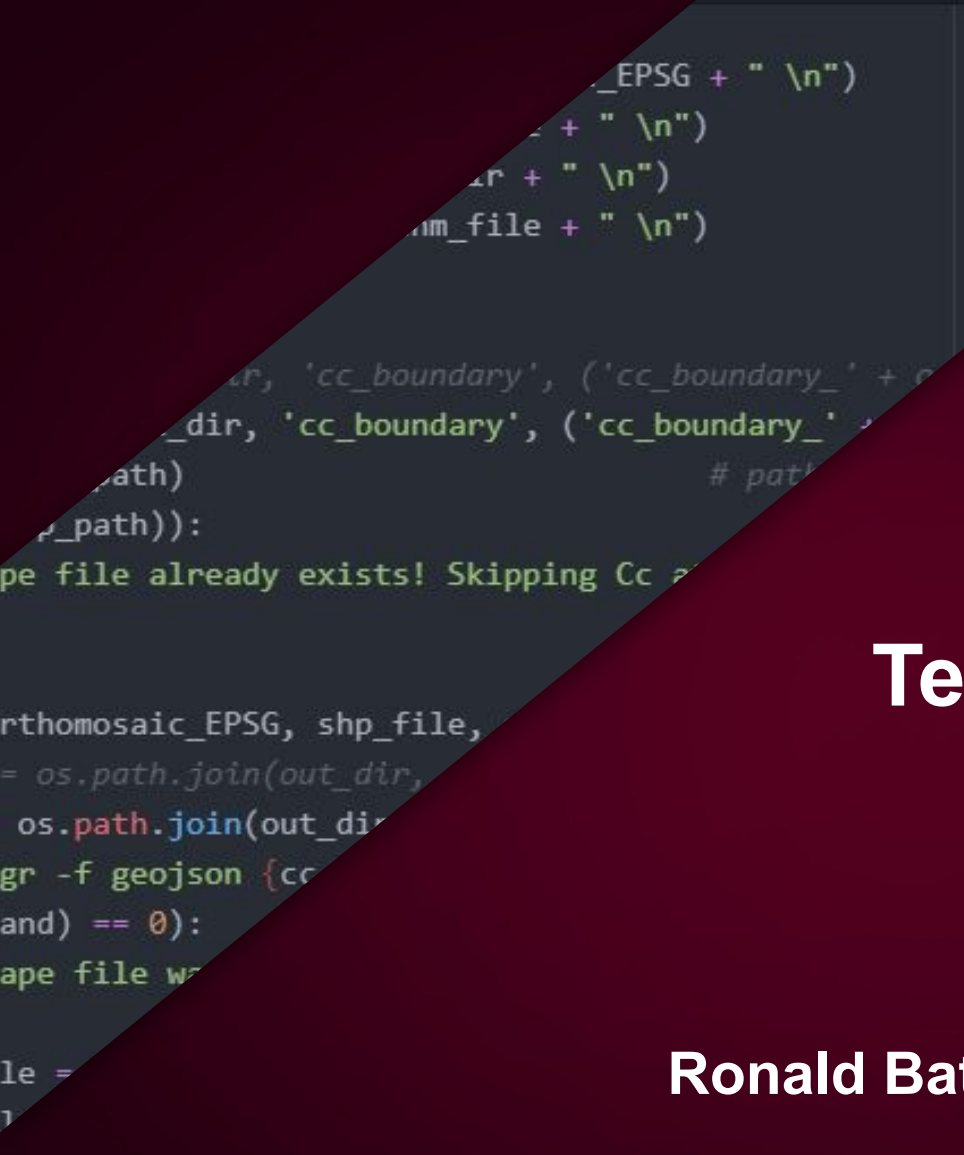


Dwight Look College of ENGINEERING
TEXAS A&M UNIVERSITY

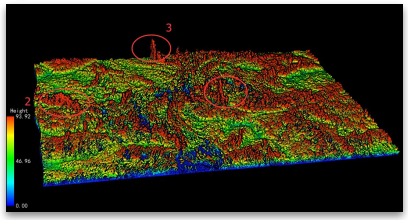
Team 25: Plant Attribute Extraction
Bi-Weekly Update 2

Ronald Batista, Campbell Motter, Rosendo Torres

Sponsor: Texas A&M AgriLife Corpus Christi
TA: Dalton Cyr

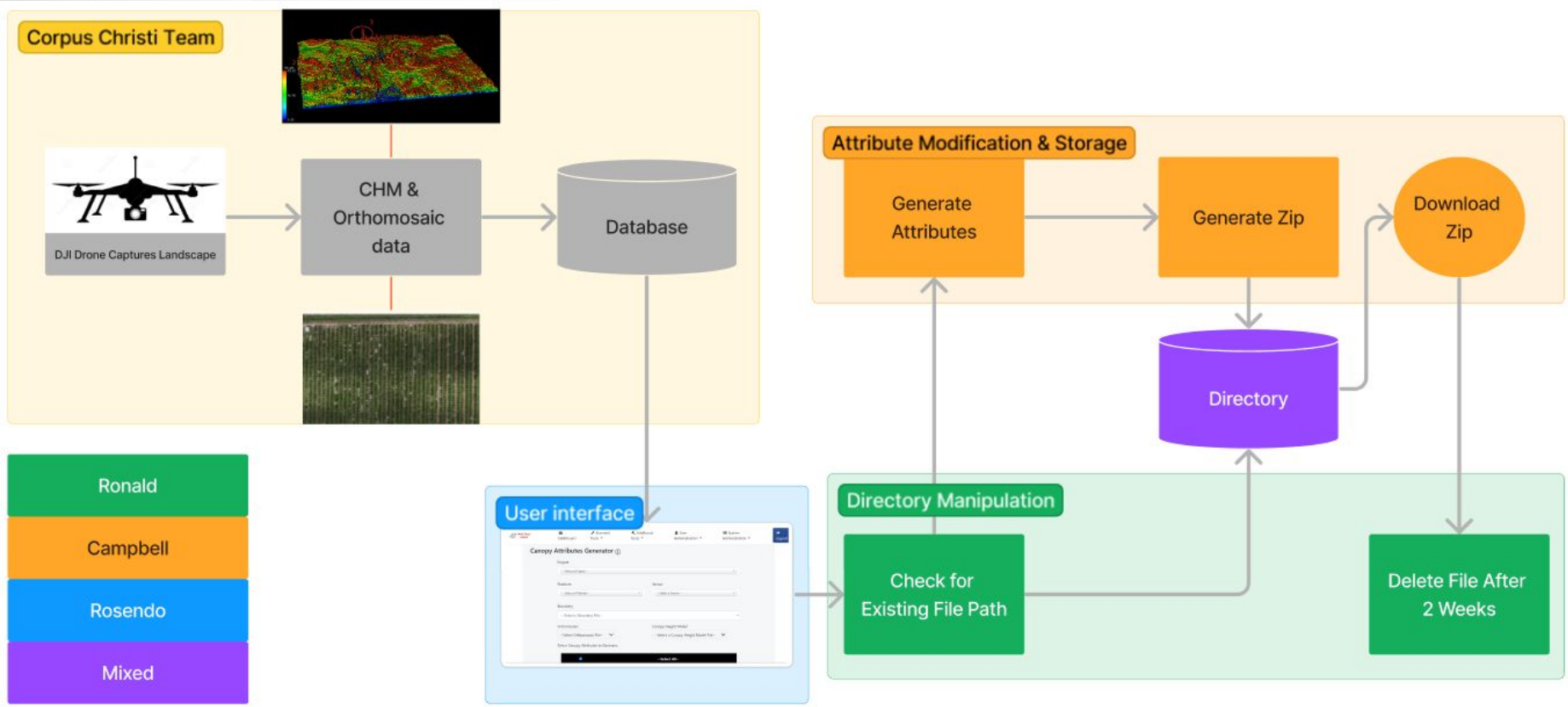
Project Summary

- The current website that receives crop imagery from a drone is inefficient in generating data and lacks the ability for large scale attribute extraction.
- We plan to manipulate the current code and website to be able to generate more data for the desired attributes based on user input.

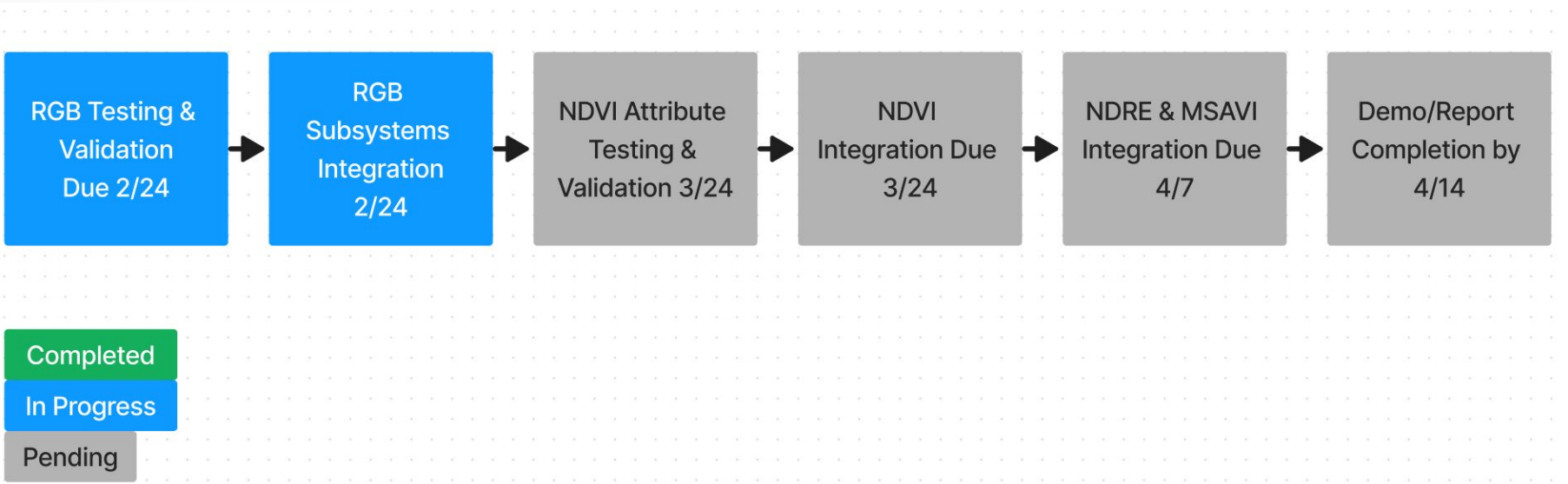


Row name	Col	FID	20220523
1	1	0	0.02056595
1	2	1	0.02549317
1	3	2	-0.0121878
1	4	3	0.06381568
1	5	4	0.17575994
1	6	5	0.06538012
1	7	6	0.52365637
1	8	7	0.28521359
1	9	8	0.42652851
1	10	9	0.39120442
1	11	10	0.18587758
1	12	11	0.38030854
1	13	12	0.18536243
1	14	13	-0.0772267
1	15	14	0.19025867

Project/Subsystem Overview



Project Timeline



Directory Management

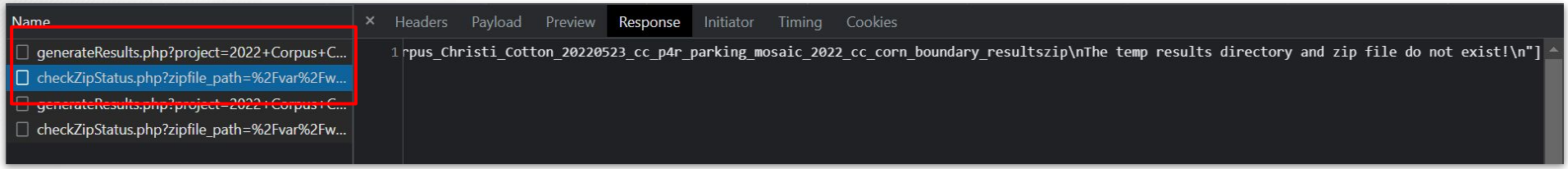
Ronald Batista

Accomplishments since Update 1 6 hrs of effort	Ongoing progress/problems and plans until the next presentation
Passed test to identify attributes being generated. Tested prompts on console and matching prompt with real result.	Testing the attribute generation and stopping the generation if file path exists Ongoing integration of deletion of the filepath. Preparation of integrating results for RGB.

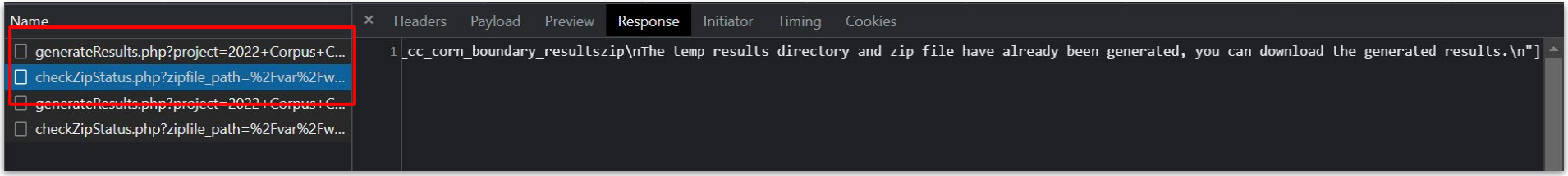
Dwight Look College of ENGINEERING
TEXAS A&M UNIVERSITY

Directory Management

Ronald Batista



Test 1: Attributes not created



Test 2: Attributes identified

User Interface

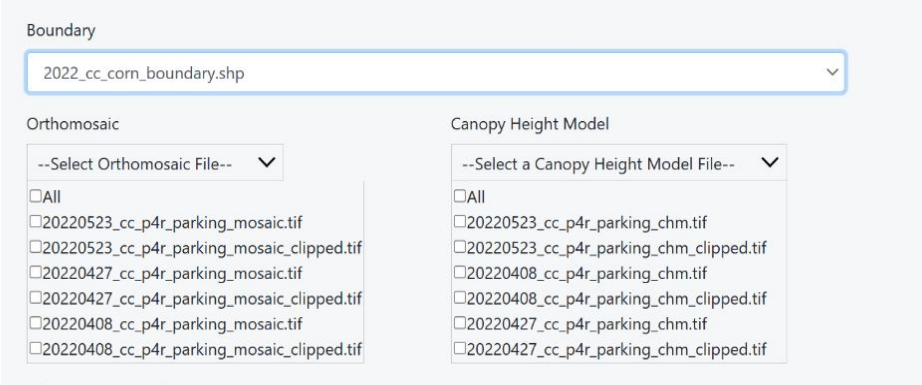
Rosendo Torres

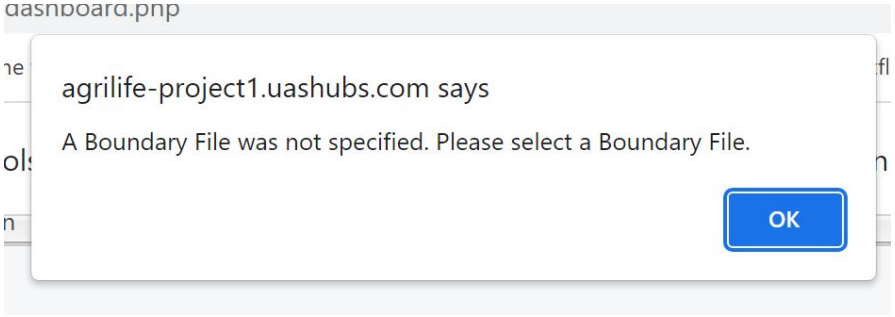
Accomplishments since Update 1 8 hrs of effort	Ongoing progress/problems and plans until the next presentation
• Filtering of attributes based on boundary restrictions • Integrated with Campbell's subsystem to be able to generate results	• Integration with Ron's subsystem for complicated pop-up warnings • Addition of Sensors • Addition of Platforms

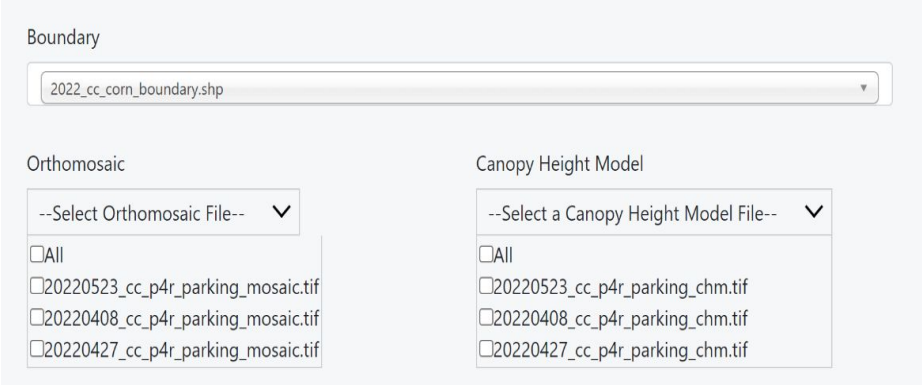
User Interface

Rosendo Torres

- The website now filters the Orthomosaics and CHM files based on the selected boundary files
- Small visual bug will be fixed by next presentation
- Pop-ups have been getting modified based on requirements

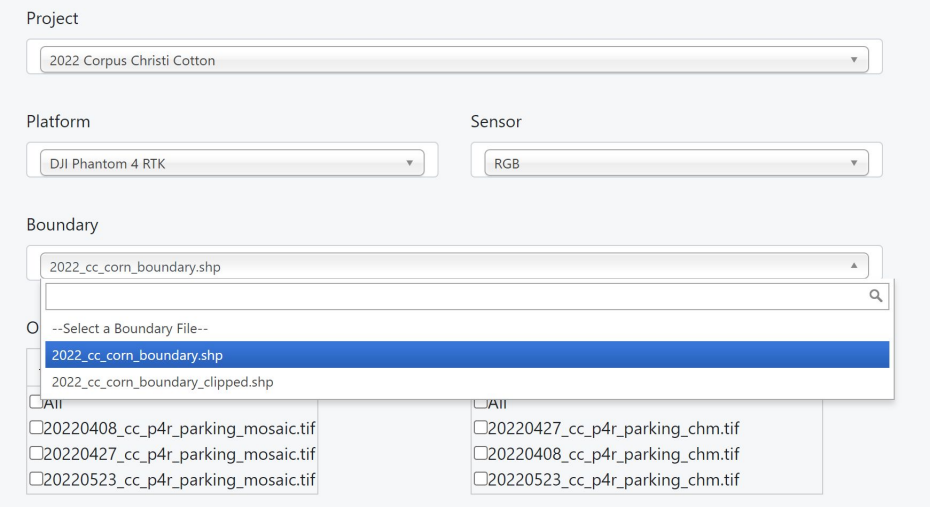


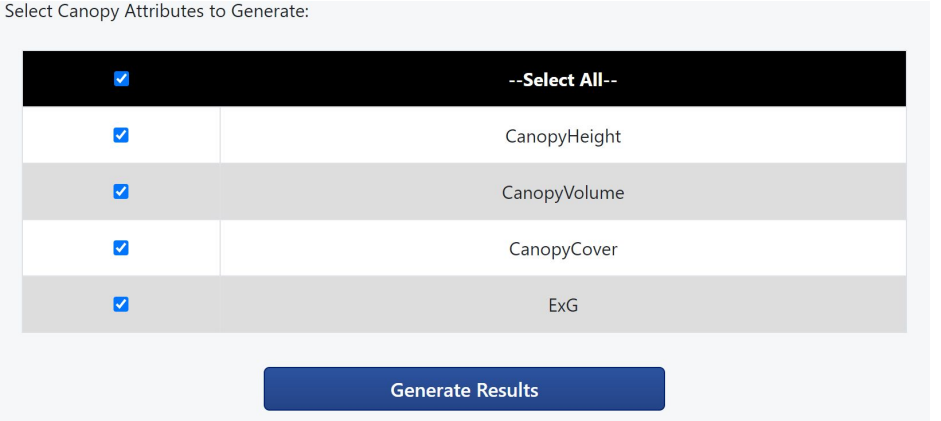




Future Work

- To add more file types to the following dropdown menus:
  - Platform
  - Sensor
  - Boundary
- Based on additional files tables will have to change





Attribute modification & Storage

Campbell Motter

Accomplishments since Update 1 10 hrs of effort	Ongoing progress/problems and plans until the next presentation
• Completed the file grouping & zipping based on attributes. (verified in PUTTY) • Fixed the ability to generate Canopy Height and Volume without a Canopy Height Model. (verified with website testing)	• Refocus on finalizing CSV file merging and verifying it. • Begin the SHP file merging. • Learn how to open and analyze SHP files in QGIS in order to verify SHP file merging.

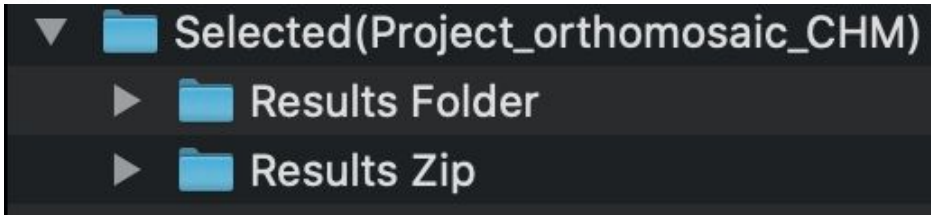
Notes:

- Verify the merging of the CSV files with Excel, and verify the merging of the SHP files with QGIS LTR(software for displaying SHP files).

New Grouping & Zipping

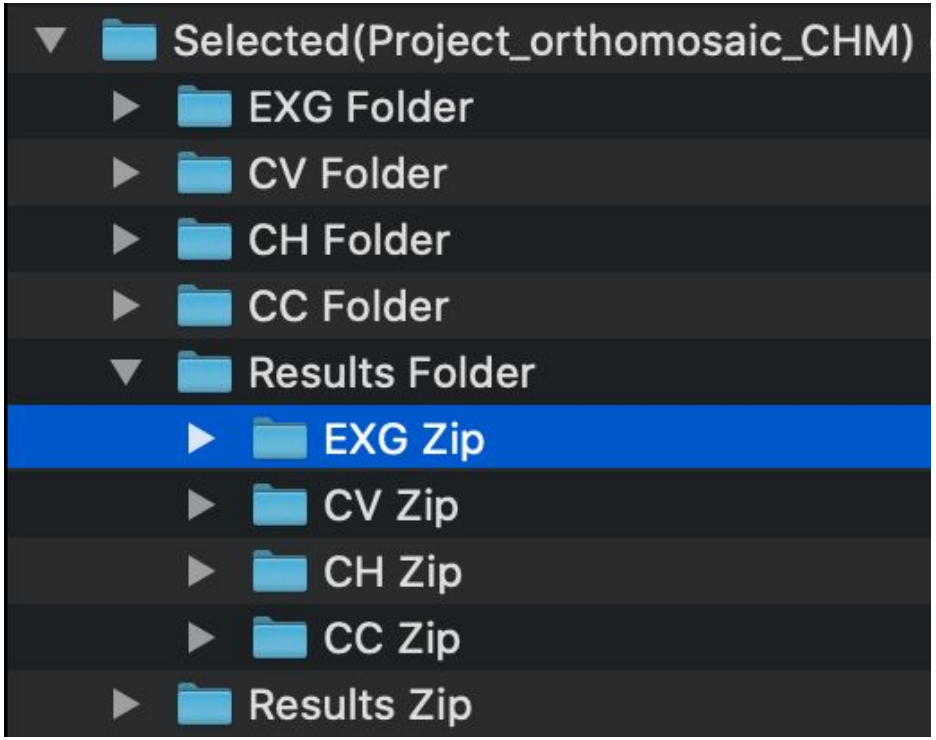
Old:

- Unorganized results folder.
- Zip contains all files.
  (A single merged file for CSV and SHP)



New:

- Organized by generated attributes.
- Final zip file contains zipped attribute folders.
  (Contains merged files for each attribute)



Execution Plan

Case	Ownership	Due	1/27/23	2/10/23	2/24/23	3/10/23	3/24/23	4/7/23	4/14/23
Generating attributes with the selection of multiple files for RGB data.		1/27/23							
Files successfully downloaded from the website in a zip file.		1/27/23							
Implement & test more specific grouping and file zipping based around attributes.		2/10/23							
Test function that will zip together all of the seperate attribute zip files.		2/10/23							
Implement & test merging together CSV files for RGB data.		2/24/23							
Implement & test merging together SHP files for RGB data.		2/24/23							
Implement & test NDVI attribute generation and storage.		3/10/23							
Implement & test NDRE attribute generation and storage.		3/24/23							
Implement & test MSAVI attribute generation and storage.		3/24/23							
Finish validating subsystem from 403		1/27/23							
Attribute restriction setups		3/10/23							
Population of dropdown menus with new requirements		3/10/23							
Validation of results table and download table		3/24/23							
Testing functionality and use of checkZipStatus and deleteTempResults for RGB		2/24/23							
Initialization of testing directory manipulation of multispectral data.		3/10/23							
Testing checkZipStatus and deleteTempResults for NDVI attribute.		3/24/23							
Integration of respective subsystems for RGB attributes.		2/24/23							
Integration of respective subsystems for NDVI attribute.		3/24/23							
Integration of respective subsystems for NDRE attribute.		4/7/23							
Integration of respective subsystems for MSAVI attribute.		4/7/23							

Completed
In Progress
Pending
Behind

Ownership Legend
Ronald
Rosendo
Campbell
All

Validation

FSR Section	Test Name	Success Criteria	Methodology	Staus	Progress
3.2.1.1	Testing	Making sure the current code in place does not have any hidden errors as well as understanding the current code used by AgriLife.	Run Python code and analyze the database for any changes and read through any comments to better understand the process.	TESTED	COMPLETED
3.2.1.2	EPSG Calculation	The EPSG value will be implemented within the PHP code to satisfy the Python code.	PHP variables will be implemented in PHP files that correlate to the corresponding Python files that use the EPSG variable in their code to make a more efficient setup.	TESTED	COMPLETED
3.2.1.3	Zip File Path Identification	The code created will identify if there is an existing file path that was generated by the user, and depending on the result, will either let the generation continue, or stop the generation and notify the user a file path exists.	Using Python and SQL code to identify filepaths and send a printed response to the console. A boolean function will be created to pass a binary 1 or 0 to let the main.js know whether to cancel or continue with the generation.	TESTED	IN PROGRESS
3.2.1.4	File Path Deletion	Once the data has been generated and it has been 2 weeks since the generation, the filepath and the contents in the path will be deleted.	Using Python and SQL code to implement a timer in the background of the website to keep the generated file path for 2 weeks. Using a similar structure to identifying file paths, the code used for deletion will generate after the attribute is generated.	TESTED	IN PROGRESS
3.2.2.1	Browsers	The website can work on all Internet Explorer, Chrome and Firefox.	Run website URL, data generation and file downloads on all stated browsers.	TESTED	WORKS
3.2.2.2	Filtering of Attributes	Based on requirements (file types, data restrictions, directory manipulation) the data generation step is restricted.	Generate data on website.	TESTED	IN PROGRESS
3.2.2.3	Menu Population	The data populates the dropdown menus correctly based on selected criteria.	Run website with all possible combinations of platforms, sensors, boundaries and projects.	TESTED	IN PROGRESS
3.2.2.4	Table Configuration	The tables for both generating results and downloading data will include all required data types and file types.	Select different configuration of platforms and sensors to validate that the tables displayed change data and file types based on selected platforms and sensors.	NOT TESTED	IN PROGRESS
3.2.3.1	Multiple File Selections	Attributes are generated for all file selections within one run of the program.	Testing that the array contains all selected files, and that the attribute generation is run the correct amount of times. Furthermore, checking the directory to see that all attributes for each file were generated. Final tests are done through the website.	TESTED	COMPLETED
3.2.3.2	Zip File is Downloadable	A zip file is downloaded from the website containing all of the cooresponding attribute data.	Ensuring that all refrences to the results Zip file path are correct and referring to the same file and location. Furthermore, checking the directory to see the generated Zip file. Final tests are done through the website.	TESTED	COMPLETED
3.2.3.3	Attribute zipping within Results Zip	Within the results zip file, there should be seperate zip files cooresponding to the attributes selected.	Ensuring that all refrences to the attribute Zip file paths are correct and referring to the same file and location. Furthermore, checking the directory to see the generated attribute Zip files within the results folder to be zipped. Final tests are done through the website.	TESTED	COMPLETED
3.2.3.4	Merged CSV data (RGB)	Within the attribute zip files, there is a single CSV file containing the merged data sets of the individual data sets for a specific attribute.	Examining the generated CSV files individaully and verifying that the data contained in the merged CSV file is correct and doesnt have any overlapping or missing data. This is done through Excel.	TESTED	IN PROGRESS
3.2.3.5	Merged SHP data (RGB)	Within the attribute zip files, there is a single SHP file containing the merged data sets of the individual data sets for a specific attribute.	Examining the generated SHP files individaully and verifying that the data contained in the merged SHP file is correct and doesnt have any overlapping or missing data. This is done through QGIS LTR.	NOT TESTED	IN PROGRESS
3.2.3.6	Merged CSV data (multispectral)	Within the multispectral zip files, there is a single CSV file containing the merged data sets of the individual data sets for a specific multipectral attribute.	Examining the generated CSV files individaully and verifying that the data contained in the merged CSV file is correct and doesnt have any overlapping or missing data. This is done through Excel.	NOT TESTED	PENDING
3.2.3.7	Merged SHP data (multispectral)	Within the multispectral zip files, there is a single SHP file containing the merged data sets of the individual data sets for a specific multipectral attribute.	Examining the generated SHP files individaully and verifying that the data contained in the merged SHP file is correct and doesnt have any overlapping or missing data. This is done through QGIS LTR.	NOT TESTED	PENDING

Thank you!
Questions?