Plant Attribute Extraction

Ronald Batista
Campbell Motter
Rosendo Torres

FUNCTIONAL SYSTEM REQUIREMENTS

REVISION – 1
25 January 2018

Functional System Requirements
FOR
Plant Attribute Extraction

Prepared by:

_____
Author Date

Approved by:

_____
Project Leader Date

_____
John Lusher, P.E. Date

_____
T/A Date

Change Record

Rev.	Date	Originator	Approvals	Description
-	10/3/22	Plant Attribute Extraction		Draft
1	11/28/22	Plant Attribute Extraction		Revision 1
2	4/28/23	Plant Attribute Extraction		Revision 2

Table of Contents

List of Tables

No table of figures entries found.

List of Figures

1. Introduction

1.1. Purpose and Scope

To properly develop the system needed in order for our sponsor to efficiently submit and receive data, we shall be optimizing the current code being used for their data manipulation. The requirements needed to optimize their code as per their requirements are as follows:

- To merge data taken on different dates into one single table for all data types in RGB.
- To reconfigure the UI to be more user friendly and apply error cases to the website if the user makes mistakes.
- To evaluate and manipulate the generation of data to save and delete generated data.
- Creating an overall more efficient system for users.
- To read the EPSG of the project to project the map correctly in the QGIS software.

Figure 1. Plant Attribute Extraction Process

1.2. Responsibility and Change Authority

During the design and implementation process, Ronald Batista will ensure that the requirements set by the client and that each team member is caught up with the deliverables that are set. Should any changes be made, Ronald and the clients, Jose L. Landivar Scott and Mahendra Bhandari, will lead any changes made to the project in terms of the deliverables set in place or the parameters of the project. If any changes are to be made, they will be updated in future documents and presentations.

2. Applicable and Reference Documents

2.1. Applicable Documents

The following documents, of the exact issue and revision shown, form a part of this specification to the extent specified herein:

Document Number	Revision/Release Date	Document Title
IEEE SA - P2992	5/21/2021	Recommended Practice for Data Expression, Exchange, and Processing in Smart Agriculutre

2.2. Reference Documents

The following documents are reference documents utilized in the development of this specification. These documents do not form a part of this specification and are not controlled by their reference herein.

Document Number	Revision/Release Date	Document Title
N/A	7/14/2021	*What is an orthomosaic map? how these maps are helping catch bad guys, grow crops, and keep people safe*.
N/A	2013	*High-Res Digital Surface Modeling using Fixed-Wing UAV-based Photogrammetry.*
N/A	8/6/2019	*Relational-databases*.
N/A	8/31/2020	*How to create and manipulate SQL databases with python*.
N/A	3/8/2022	*Relational Database Management Systems (RDBMS): All you need to know*.
N/A	10/28/2021	*How to connect HTML to database with mysql using php?*

2.3. Order of Precedence

In the event of a conflict between the text of this specification and an applicable document cited herein, the text of this specification takes precedence without any exceptions.

All specifications, standards, exhibits, drawings or other documents that are invoked as “applicable” in this specification are incorporated as cited. All documents that are referred to within an applicable report are considered to be for guidance and information only, except ICDs that have their relevant documents considered to be incorporated as cited.

3. Requirements

3.1. System Definition

Our project was created in support of an existing Texas A&M AgriLife project. We were tasked with creating more efficiency to the existing website, as well as implementing new capabilities with data formatting and merging. Some of these new capabilities include selecting multiple Canopy Height Models and Orthomosaics from the website. This process will include work on the GUI(Graphical User Interface) as well as data compilation on the python file (the back end development). Another feature to be added is the ability to store compiled data in the directory for two weeks, then deleting the data afterwards in order to manage space on the database. This feature will fall under the directory management sub-system. Our project is broken into three sub-systems, front-end development, back-end development, and directory management. Starting with the front end we'll be using html and javascript in order to complete the GUI (Graphical User Interface). some of which include expanding the capabilities such as allowing the user to select and generate results for multiple orthomosaics. The front end is the system that the users will actually interact with. Our website UI will let them choose their specific parameters to generate results and allow them to pick their desired attributes and file types. Next, we have the back end development which will consist of Javascript, Python, and PHP. The back end will be in charge of generating the plant attributes, generating the result zip packages, and creating merged csv and shp files if multiple orthomosaic files are selected. The Javascript will pass all of the user inputs to the PHP file, which will then make a python command and execute it on a linux shell. The directory data will then be updated and the python response will then be sent back through PHP back to the javascript to continue with the process. Next we have the directory management which will consist of the same languages as the back end; Javascript, Python, and PHP. This subsystems main role will be making sure to delete old results and check to see if results exist for what the user has input. Therefore, if the users input already exists in the directory, then you can immediately download and skip the lengthy generation. These are the three subsystems in the process of Plant Attribute Extraction.

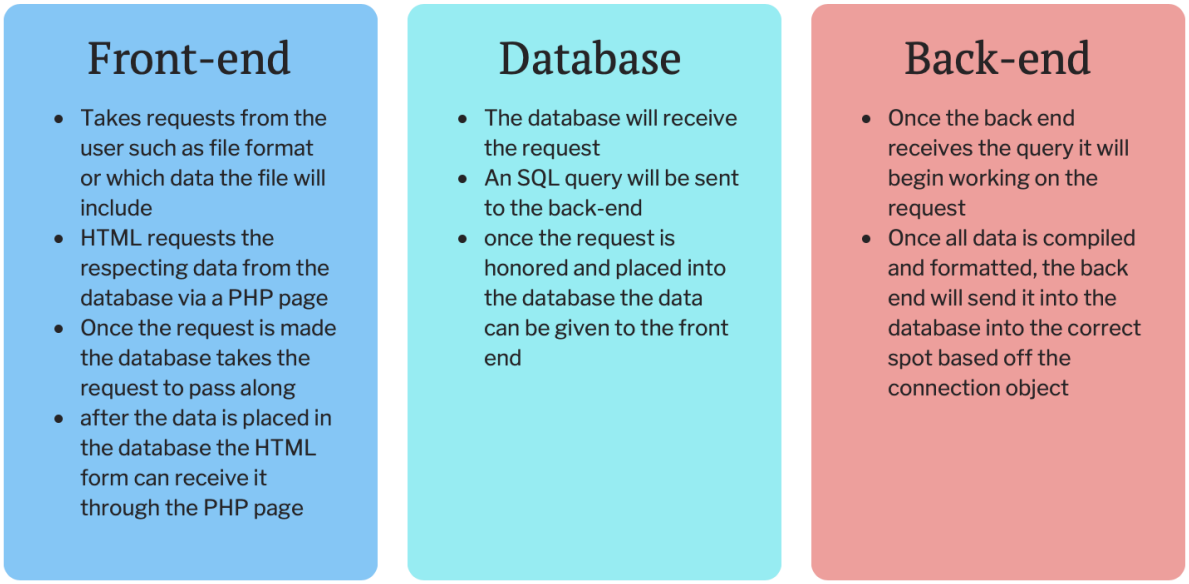

Figure 2. Software Partition

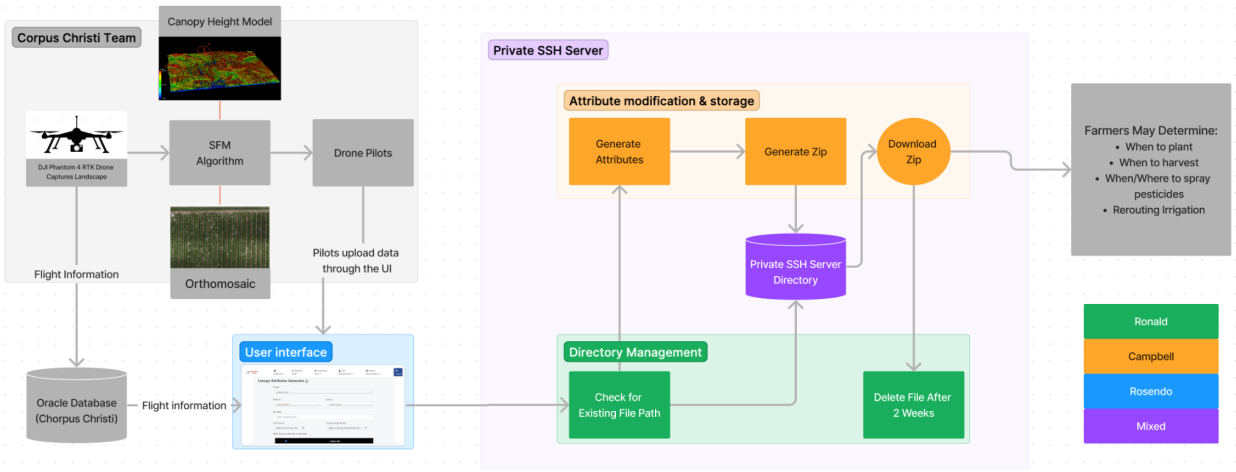

Figure 3. Block Diagram of System

Our project will consist of three major systems; the User Interface, the back-end directory management system, and the attribute modification and storage. The UI will be the front-end section of what the user will see first hand. This section will aid the user in informative sections and multiple error cases to be able to help the user completely generate the data they would need to acquire. The directory management system will run in the background while the user is using the website to be able to analyze the directories created by the user to either inform the user of generated directories or delete old directories to not fill up space in the SSH server. Finally, the attribute modification and storage system will make data generation work differently than before. This will allow merged sets of data to be generated to be able to compare data taken from different days in

the same file in all the data types the user can download and all the different attributes the website can generate.

3.2. Characteristics

3.2.1. Directory Management

3.2.1.1. Zip File Path Identification

Any files that have been generated and saved within the SSH server directory will be kept in the directory. If the user is to generate the same data and attributes at another time, the website will identify if the directory still exists. If the directory still exists, it will prompt the user that the data has already been generated and allow them to download the necessary files they would need. This is advantageous to cut down the time of generation in the website by minutes to provide the data at a faster pace.

3.2.1.2. EPSG Calculation

The European Petroleum Survey Group (EPSG) is similar to a zip code in how it references a section of a state, but in this case it references a certain longitude and latitude range of the world. When the user downloads the data and uses the Shapefile on the QGIS software, it should be able to identify the EPSG of the file and be able to correctly project the satellite image of the land, and line up to the crops with the boundary generated.

3.2.1.3. Zip File Path Deletion

When the data has existed in the SSH server database for 2 weeks, the data is to be deleted. This is done for two reasons:

1. To be able to save the data in case the data needs to be regenerated by another user (see section 3.2.1.1 for more details).
2. To not leave the directory in the database indefinitely and run out of space within the server.

This makes it ideal to be able to setup deletion in the server to allow for better manipulation of the generated directories as well as properly managing the server without the need for a user to do it manually.

3.2.2. User Interface

3.2.2.1. User Friendliness

The website was updated to become as user friendly as possible in order for the less knowledgeable users to be able to access and run through without having too many issues. Many tests were run with people of different backgrounds and knowledge to accurately test the usability of the website.

3.2.2.2. UI Restrictions

Depending on what Project, Platform, Sensor and boundary files are selected those are the data sets that you can generate. The website has the ability to filter the

orthomosaics and CHMs based on what is wanted. In addition, the attributes generated are both for the individual data set but have a merged set for comparison purposes.

3.2.2.3. Warnings and Popups

Every fail case has a warning or pop up on the website that allows the user to see what exactly they need to do in order for the generation to happen. It explains the error that happened and how to fix it effectively. In addition to just letting the user know what happened these pop ups are able to stop generation from happening in order for the database not to have an overflow on storage.

3.2.3. Attribute Modification & Storage

3.2.3.1. Multiple Files

The user will be able to select multiple orthomosaic and canopy height model files for plant attribute extraction. As opposed to the previously existing process where only one file at a time was permitted. The result package will contain the results for each of the selected orthomosaics organized by the selected attributes.

3.2.3.2. Merged Files

When a user selects multiple orthomosaic files, a merged file will be created for CSV and SHP file types. These files will contain all of the attribute data for each selected orthomosaic file. Each attribute will have its own separate zip file and each attribute zip file will contain a merged data file for CSV and SHP file types.

3.2.3.3. Special Cases

Special cases have been created for specific computations. The first pertains to two of the plant attributes. The Canopy Cover(CC) and Excess Greeness(EXG) attributes do not need a Canopy Height Model(CHM) in order to generate the results. Therefore we allowed the user to generate CC and EXG attributes without the input of a CHM file. The other special case refers to when the number of orthomosaic files inputted and the number of CHM files inputted aren't the same. We refer to this as the mismatch test case. We allow users to input less CHM files than orthomosaic files in order to calculate the canopy height and volume attributes for only the orthomosaic files that have an accompanying CHM file.

3.2.3.4. Zip files

The results are downloaded to the user's device as a zip file. This parent zip file is called the resultsZip folder. The attributes selected will each contain their own zip folder. The attribute zip folder will contain the results for one or many orthomosaic files for that specific attribute, and the parent result zip folder will contain a zip file of each selected attribute from the user.

4. Support Requirements

On top of completing the program and the project as a whole, the people who will eventually be utilizing the system will need an understanding of how the system works. When creating the code, comments shall be implemented to explain any new additions to the system that way it is easily understandable to anyone who is unfamiliar with the code and is needing to look at it or plans to update further in the future. We will also be providing a user manual to help with understanding the complete process of inputting data and what to do with the output data once it is calculated. The manual shall be in the form of a word document that presents a step-by-step format that the user can follow throughout the data input/output process.

Appendix A: Acronyms and Abbreviations

CSV	Comma Separated Value
SHP	Shapefile
XLSX	Excel Spreadsheet File
DSM	Digital Surface Model
ICD	Interface Control Document
PHP	Hypertext Preprocessor
RMDBS	Relational Database Management System
SQL	Structured Query Language
UAS	Unmanned Aircraft Systems
EPSG	European Petroleum Survey Group
CHM	Canopy Height Model

Appendix B: Definition of Terms

Orthomosaic - An image with a high resolution that combines miniature photos to create the eventual image

GeoJSON - A file representing geographic features

Boundary - A plot of landscape in the image used to calculate the canopy height and volume.

Canopy - The crops located in the boundary used from the orthomosaic image.

Appendix C: Execution Table

FSR Section	Test Name	Success Criteria	Methodology	Ownership	Staus
3.2.1.1	Zip File Path Identification (RGB)	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. Can run for multiple cases for accessing different types of data and attributes.	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.	RONALD	VALIDATED
3.2.1.2	EPSG Calculation (RGB)	The code created will pass the EPSG from the created project added and be used to properly project the shapefile to the orthomosaic when viewing the data	Similar to how the RGB code has been setup, but with the different types of multispectral data instead and testing for each attribute and for multiple files.	RONALD	VALIDATED
3.2.1.3	File Path Deletion (RGB)	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. Can run for all attributes and for any number of files.	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.	RONALD	VALIDATED
3.2.2.1	User Friendliness (RGB)	The tested user is able to go through the website relatively easily and with minimal confusion.	I will have friends try out the website to see how they perform	ROSENDO	VALIDATED
3.2.2.2	UI Restrictions (BOTH)	The UI is able to successfully display files based on restrictions and selected files.	Select different files and constraints to see if the UI successfully updates based on selected files	ROSENDO	VALIDATED
3.2.2.3	Warnings (RGB)	Based on what type of error occurs the user is notified through popups and warnings.	Run through all possible errors that can occur and make sure each one has a pop up that lets the user know what went wrong.	ROSENDO	VALIDATED
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 individually 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.	CAMPBELL	VALIDATED
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 individually 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.	CAMPBELL	VALIDATED
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 individually 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.	CAMPBELL	NOT TESTED
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 individually 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.	CAMPBELL	NOT TESTED

Appendix D: Validation Table

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 and Validation of Multispectral UI		4/7/23							
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							

Legend
Ownership
Ronald
Rosendo
Campbell
All
Progression
Completed
In Progress
Pending
Dropped