UNIVERSIDAD DE GUADALAJARA
CENTRO UNIVERSITARIO DE LOS VALLES

Software Configuration Management

System for the Inspection of a Photovoltaic Park with Aerial Images in High Definition and Thermal

---

Professor: Phd. Omar Ali Zatarain Durán

Author: Raúl Rubén Romero López

SCM-SIPaF-V1.0

November 03, 2022 Ameca, Jalisco

Change control

Revision	Description	Author	Date	Version
0.1	Preliminary version	Raúl Romero	08/26/2022	SCM-SIPaF-V0.1
0.2	Design, code and test sections added	Raúl Romero	01/09/2022	SCM-SIPaF-V0.2
1.0	CR-03 applied	Raúl Romero	03/11/2022	SCM-SIPaF-V1.0

Revision

Version	Responsible	Date	Signature

Authorization

Version	Responsible	Date	Signature

Table of contents

Table of figures

1. Introduction

With the purpose of simplifying and making more efficient the process of detecting failures in solar panels within photovoltaic plants, thus testing the system within CUValles in Ameca, Jalisco; It is proposed to develop this project for the analysis of high definition images and thermal images taken from a drone.

Fault detection tracking without software for analyzing the status of solar panels within the plant is a problem because of the way the electrical interconnections of the solar panels are made. The process is time-consuming and may be subject to human error. This process is carried out as follows:

1. Select the sector of interest within the plant.
2. Send a drone to obtain images of the panels in that place.
3. Analyze the data or images obtained with the software.
4. Identify the position of the damaged panel.
5. Deliver the validation document to the team in charge of repairing the damage.

1.1 Purpose

The objective of this document is to define the specification of the functional and non-functional requirements, as well as the objectives that need to be done for the acceptance of the system that will help to inspect the state of the solar panels using aerial images taken by a drone. The design of the system, as well as some code and tests will be provided in this document.

1.2 Scope of the system

Develop a system that can analyze images taken by a drone in order to identify damaged cells using the infrared spectrum. There should be a report generated by the software with special information about the analysis. The system must have a trained model that can analyze infrared images and orthomosaic images, specifying where the damaged cells are located.

1.3 Acronyms details

Term	Definition
SIPaF	System for the Inspection of a Photovoltaic Park
SCM	Software Configuration Management
MIS	Master Degree in Software Engineering
CUValles	Centro Universitario de los Valles

1.4 Document overview

In order to guide the reader, this document is organized into the following sections:

The introduction provides background information and important factors to consider in the solar panel inspection process.

The second section offers a general description of the system, where the general factors included in the product and its requirements are described. The stakeholders to whom this section of the document is addressed are the users involved and the system development team. Users will be able to identify the functionalities of the system and developers will be able to understand the software and communications restrictions under which development will proceed.

In the third section, the specific requirements of the system and the follow-up that they will have during the development of the project are described. The elements are grouped into functional and non-functional requirements, in such detail that it facilitates the work of the development team using a natural and simple language, in order to integrate all the stakeholders of the project within the process of building the software product.

2. General description

This section specifies the factors of interest and the functional requirements of the SIPaF system. To achieve this, it will be necessary to make a detailed description of the environment where the system will be implemented, and the factors involved in the application space. In this way, this collected information will provide the guidelines for the development and implementation of this software.

2.1 Product perspective

The desktop application of System for the Inspection of a Photovoltaic Park(SIPaF) aims to be a tool to improve the current process of inspection and detection of production failures of photovoltaic panels in a solar plant for the university campus of Valleys located in Ameca, Jalisco.

2.2 Product features

The SIPaF system intends to make use of technological tools to achieve the objective set by the client, in this case, the project director, Dr. Himer Avila George. For this they are necessary: the analysis, design of the system, as well as the database and the user interface.

Broadly speaking, the product will allow the following functionalities:

1. Enter aerial images in high definition and thermal images.
2. Make an orthomosaic of the place with the two types of images.
3. Identify each of the solar panels in the orthomosaics.
4. Determine which panels have faults and give them a priority level.
5. Make a report with the data obtained.
6. Help to better visualize the data obtained from the plant in each report.

2.3 User

This section describes the type of user that will make up the system. A single user is contemplated who can share this information with other people through emails.

Type of users	Description	Privileges	Technical Experience	Frequency of use
User	User who has permissions to create projects, upload images, make reports and view information. Also, he must be able to share the information using an email. The images cannot be shared, but the results of the reports can.	Access all the functionalities of the system.	Advanced Basic system administration skills with significant application knowledge are required.	Whenever required.

2.4 Restrictions

- The SIPaF system will be designed in python and as a desktop application.
- The application's operating system will be Windows 11 64-bit and its code will be in Python with version 3.9.7.
- Users will need to have the program installed on their hard drive.
- The response speed is determined by the computer's processor and graphics card.
- Access to the system will not be restricted, but a password will be requested when registering an email for sending files.

2.5 Assumptions and dependencies

- The system requirements can change during the development of the application only if they follow the SCM policies.
- Availability of development team work.
- Time availability of project stakeholders.
- It will be necessary to validate the operation of the software with an expert for its full implementation.

3. Specific requirements

In this section you will see all the client's needs to accept the product as a quality product.

3.1 External interfaces

3.1.1 User interfaces

The user interface will have a set of windows with buttons, lists and text fields. The user interface will be displayed when the program is run. To access the interface, you must have the program installed on your hard drive.

3.1.2 Software Interfaces

- Desktop Application: Windows 11 64-bit.

3.2 Functional requirements

This section will describe the system requirements in detail as shown in the following tables:

1. User: Create project
2. User: Delete project
3. User: Modify project
4. User: Search file
5. User: Create report
6. User: Delete report
7. User: Set email account
8. User: Change email account
9. User: View reports
10. User: Create orthomosaics
11. User: Share obtained reports
12. User: Print reports
13. User: View data from project
14. User: Save the graphs created
15. System: Identify the faults from the images
16. User: Open project
17. User: Create a monthly report of the drone usage

3.2.1 Functional requirements tables

Functional Requirements Specification			
Code	Name		Degree of need
RF-01	Create project		High
Description	The user needs to create a project in which he can have the information obtained from that analysis.		
Inputs	Source	Outputs	Restrictions
+Project name +Address +Images for processing	User	>Project display in the organization bar.	*If the project already exists, a notification will be sent to the user to know if they want to replace it.

Process	New project must be selected in the file menu or in the project display area. Then you must choose the location to save the project. Next is to select the images for analysis.

Functional Requirement Specification			
Code	Name		Degree of need
RF-02	Delete project		High
Description	The user needs to delete the files created by the system of a project already created so that it does not appear in the system.		
Inputs	Source	Outputs	Restrictions
+Project	User	>Deletion of the project from the organization bar.	*Choose a project that has been created.
Process	Must select an already created project and then select the delete option. The image analysis file will be deleted, but the created reports will not be deleted.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-03	Modify project		Medium
Description	The user can select an existing project and remake the process changing the images used.		
Inputs	Source	Outputs	Restrictions
+Project || Parameters	User	>Redoes the project replacing all the information it had.	*Requires a finished or failed project.
Process	Must select an already created project and then select the start option after checking the parameters for the process are correct. The products of the project will be deleted to create the new ones.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-04	Search file		Medium
Description	The user searches for a project or report in the display bar where the files of the working directory are displayed.		
Inputs	Source	Outputs	Restrictions
|| Project name || Report name	User	>Displays the project that matches with the search and is found in the project folder.	*The project must have the characters specified on their name.
Process	Must select the search bar above the display bar and type the characters of the name that are on the project.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-05	Create report		High
Description	The user can create a report of the information gathered by the project in a PDF format.		
Inputs	Source	Outputs	Restrictions

+Project || Name of the report.	User	>PDF file of the report on the display bar.	*The report will be created in the same directory as the project..
Process	Must select a finished project and then select the make report option. The display bar is updated when the process finishes and a message is displayed..		

Functional Requirements Specification			
Code	Name		Degree of need
RF-06	Delete report		Medium
Description	The user can delete from the project's folder a report created.		
Inputs	Source	Outputs	Restrictions
+Report	User	>PDF file of the report on the display bar is removed.	*The report will no longer exist..
Process	Must select a report and then select the delete report option. The display bar is updated when the process finishes and a message is displayed..		

Functional Requirements Specification			
Code	Name		Degree of need
RF-07	Set email account		High
Description	Configure an email on the system.		
Inputs	Source	Outputs	Restrictions
+Email +Password	User	>Message displaying the status.	*Can only have one email configured at the time.
Process	Provide a valid account and wait for the system to verify it.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-08	Change email account		High
Description	Change to another email account.		
Inputs	Source	Outputs	Restrictions
+Email +Password	User	>Message displaying the status.	*Can only have one email configured at the time.
Process	Must select the settings and then change the email account option. Then provide the last email and password account. If valid, must provide a new valid account and wait for the system to verify it.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-09	View reports		Medium
Description	Open a report created by selecting it.		
Inputs	Source	Outputs	Restrictions

+Report	User	>Opens the report on another application that visualizes PDFs.	*Won't be able to see it on the system.
Process	Must select a report and select the option to open it. Then a PDF reader application will be open with the report.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-10	Create orthomosaics		High
Description	Create an orthomosaic if selected with the images for the analysis.		
Inputs	Source	Outputs	Restrictions
+Images || RGB || Thermal	User	>Shows the orthomosaic on the display bar.	*Can only do it if selected during the analysis..
Process	Must select the option when creating a project and start the project. The orthomosaic will be made in the RGB and Infrared images if selected. They will be displayed on the view tab.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-11	Share obtained reports		High
Description	The reports obtained can be shared through email.		
Inputs	Source	Outputs	Restrictions
+Report +Email account(s)	User	>Displays an image after sending the report.	*The system will not check if the email was received. *The system won't verify if the destination account is valid. *A valid email must be active on the system.
Process	Must select a report and select the option to share it. Then the destination emails must be entered.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-12	Print reports		Medium
Description	Be able to send the report to a printer.		
Inputs	Source	Outputs	Restrictions
+Report +Printer	User	>Displays a message after sending it to the printer.	*Won't verify if it was printed already.
Process	Must select a report and select the option to print it. Then a printer must be selected and send it print.		

Functional Requirements Specification		
Code	Name	Degree of need
RF-13	View data from project	High

Description	Let the user select multiple projects to compare the data from them and visualize them.		
Inputs	Source	Outputs	Restrictions
+Projects	User	>Displays a graph with the data obtained on the data bar.	*There will be a limit of 20 projects.
Process	Must select multiple projects and select the option to compare their data. The graph will be displayed on the data bar and some options will be available to modify some parameters of the graph.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-14	Save the graphs created		High
Description	Let the user save the graph created.		
Inputs	Source	Outputs	Restrictions
+Graph +Address	User	>Displays a message after saving the graph.	*The graph won't be linked to a project.
Process	Must select a graph and select the option to save it. The graph will be saved on the address specified and can be opened from there.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-15	Identify the faults from the images		High
Description	Must be able to locate the faults in the images selected from the user.		
Inputs	Source	Outputs	Restrictions
+Images +Trained model	User	>Displays all the data of the process using images and text labels in the system.	*The data must be displayed in a good way in the system.
Process	After creating a project and starting the process, the system must be able to identify the faults within the images presented and display them in an organized form within the system screen.		

Functional Requirements Specification			
Code	Name		Degree of need
RF-16	Open project		High
Description	Must be able to load a file of a project already created to watch the information.		
Inputs	Source	Outputs	Restrictions
+Project	User	>Displays all the data of the project using images and text labels in the system.	*The data must be displayed in a good way in the system.
Process	After selecting a project the information about it should be uploaded and see it .		

Functional Requirements Specification		
Code	Name	Degree of need

RF-17	Create a monthly report of the drone		High
Description	Every information flight made by the drone should be retrieved in order to make a monthly report containing the information of the flying hours, zones in which the drone flew during the month and purpose for each flight.		
Inputs	Source	Outputs	Restrictions
+Flight information	System	>Report of information with the flights made by the drone.	*The purpose should be specified each time a project is made. *The process can be made automatic or manual.
Process	The process should be programmed and specify the folder where the projects of a drone should be taken. Every month the process would take the projects of the last 30 days made by the drone and		

3.3 Nonfunctional Requirements

3.3.1 Performance requirements

The system must have a database manager.
The system must have an optimal response time.

3.3.2 Reliability Requirements

The system must be reliable.
The system must warn against possible erroneous operations or actions.

3.3.3 Availability Requirements

The system must be available 100% of the time.
The system must be able to identify the projects created and displaying them if they are on the same working directory.

3.3.4 Maintainability Requirements

The system must have parameterizable characteristics to allow future maintenance (source code).
The system should be created in such a way that modules or updates can be added in the future.

3.3.5 Portability Requirements

The system must be able to be installed and used on any Windows 11 system.

3.3.6 Design Restrictions

The system must be intuitive.
The system must have a logo.
The system must have text-type aids in the forms.

3.3.7 System Attributes

The system must be able to send emails.

3.3.8 Other requirements

The system does not interact with another external system.
The system must have an intellectual property and be registered in INDAutor.

4. Design

In this section, the design of the software is specified using UML diagrams to represent the whole system. First we will have some general diagrams to identify the components and the scenarios that are going to be addressed by the software, after that, we will explore in detail the modules of the system.

4.1 General design

In Fig. 1, the general modules of the system and their relationship between them.

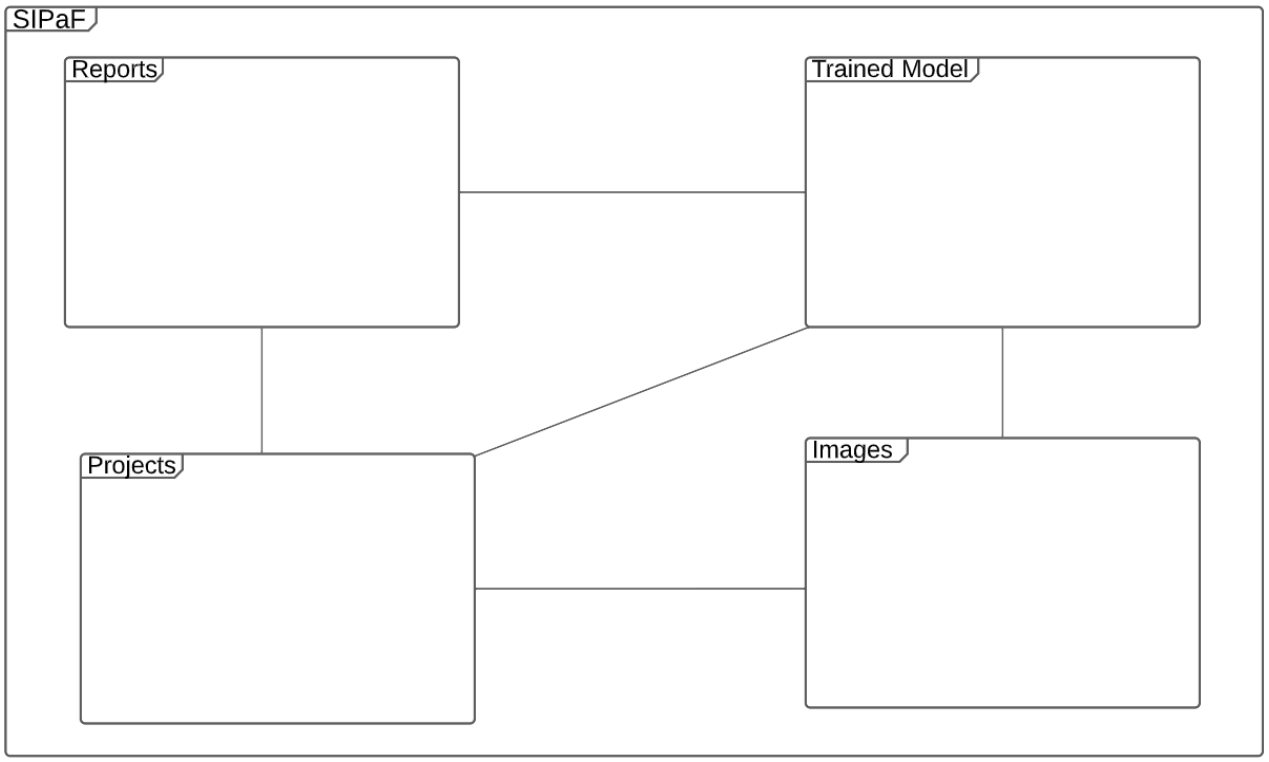

Fig. 1 General view of the system components.

In Fig. 2, the use case for the module more important of the system is shown, the project module. As we can see, this module is the one that contains the majority of the requirements. The second one with other requirements will be the reports module, and besides having less functional requirements, the trained model module is the second one more important because it is the one that will gather the information that will be displayed on the images and on the reports.

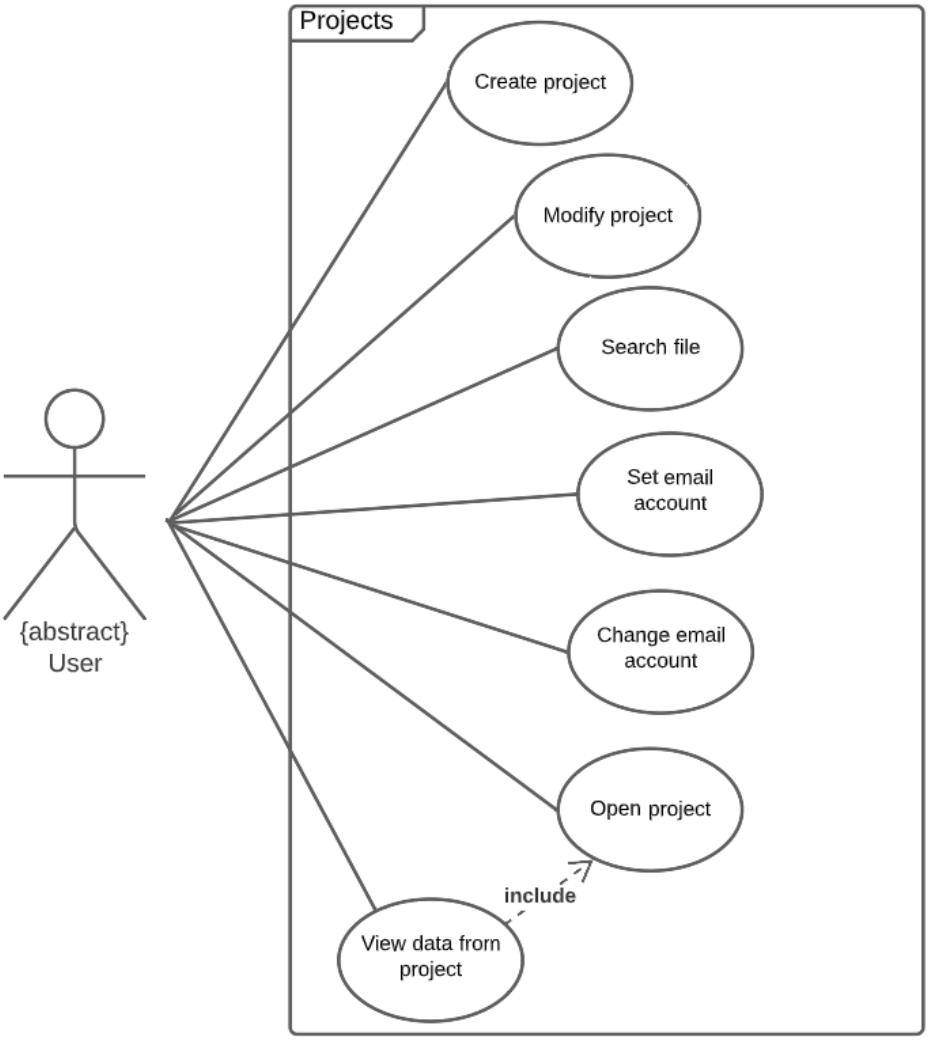

Fig. 2 Use case users with the projects module.

In Fig. 3 we will see the representation of the interface it is expected to be designed.

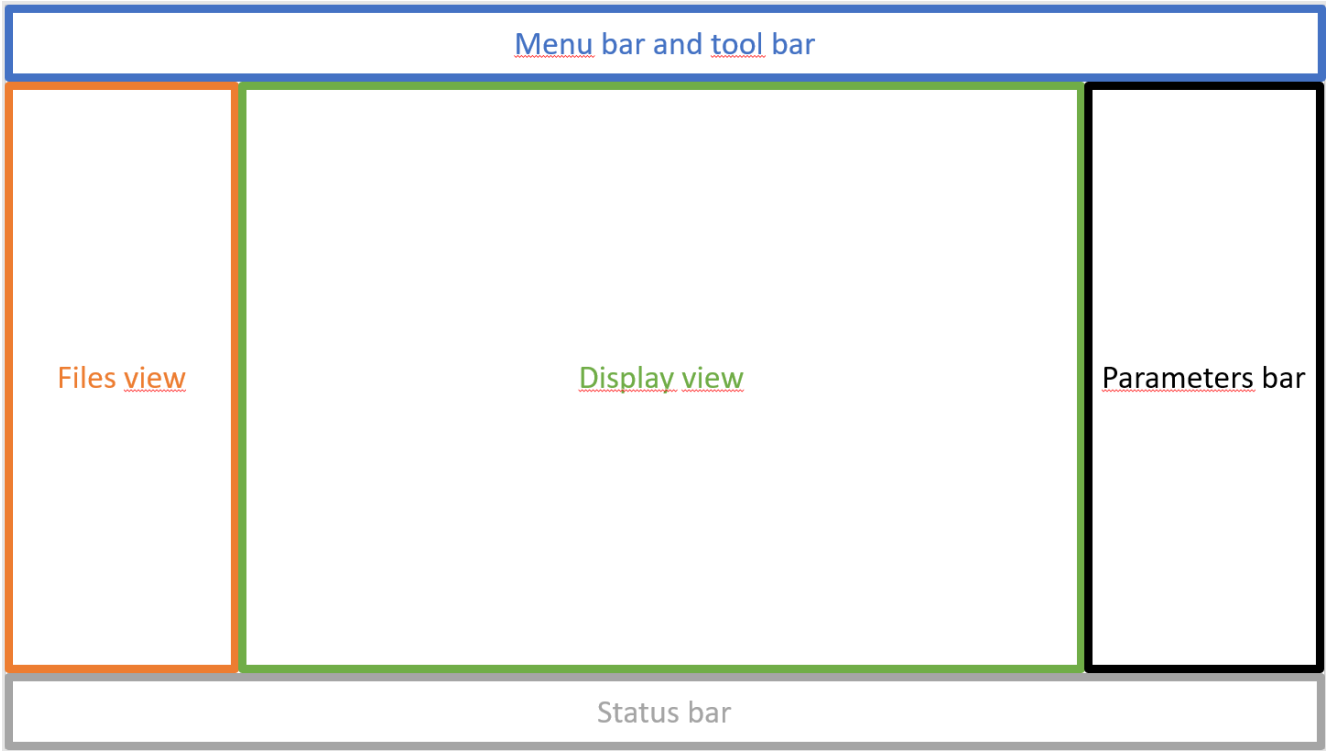

Fig. 3 Interface organization.

4.2 Modules

This system has 4 modules, the biggest and most important module is the project one. The next important module is the trained model module, that module contains the network that will be the one with the trained parameters to analyze the images. The third module is the one in charge to check the images and help process them to make the orthomosaics. Finally, the reports module will be able to save the information in a way they can be seen in a pdf document.

4.2.1 Project

This module will be the one that will interact with the other modules and should be the one to define the relationships with the other modules. It needs to be able to create a file that will contain all the information of the project. Here is where the process begins and where it ends. This module will be the one that will have the most interaction with the user. The other models will depend on the information this has so it must have all the information and methods they need in order to complete their tasks.

Besides, the project needs to receive the input of the zone, the flying hours, which drone was used and the purpose of the flight.

4.2.2 Trained model

This module needs to have pretrained neural networks to work with the different types of images and solar panels. But the first model must be able to find the hotspots of the solar panels that are in CUValles. There should be hyper parameters as some of their entrances depending on the type of solar panels to be analyzed on the images.

4.2.3 Images

The images module will be in charge of creating the orthomosaics and to have the information of the images that are being used. It will also have the directory where the images are saved.

4.2.4 Reports

This module will be the one that has to have the information and methods to create a report with all the data from the project. It will also be able to create the pdf file and be capable of sending it through email or downloading it.

Additionally, this module will have more information, apart from the information found by the trained model, which needs to be used to create another report each month.

5. Code

In this section, we will have some of the most important functions by each module.
The new variables added to the project class are:
report_day: int # The day of the month to make the reports.
flight_purpose: String # The reason for doing this flight.
drone_number: int # The identifier for that drone.
flight_zone: String # Describe where the zone flew
flying_start: day, hour # Specify when the drone started flying
flying_end; day, hour # Specify when the drone stop flying

The new method added to the class report is:

create_flying_report(report_day): Report # Create the flying report and save it.

6.Test

Finally, this section will present the tests that must be performed for the software in order to assure the quality of the product. We first specify the hardware and software characteristics for the tests and then there will be some described tests per module.

6.1 Hardware

Specification for the equipment where the tests will be taken:

Characteristic	Specification
Device name	LAPTOP-6OBLSTJL
Processor	Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz 2.59 GHz
RAM	32.0 GB
Operating system	Windows 11 Home Single Language, 64 bits

6.2 Software

Specification for the software where the tests will be taken:

Name	Version
Pytest	7.1.2
Python	3.10.5

6.3 Version control

Version control will be using GitHub, this lets us share the code through the internet and is easy to implement on the software of PyCharm, where the code will be developed. And we will also be able to have the reports saved and baselines of the project to have a version control of all the documents during the development of the project.

6.4 Test cases

In this subsection the intention is to present the module and the test cases that need to be tested for each module.

Module	Requirement	Test
Projects	1- Create project	1- Create project successful
Projects	2- Delete project	2- Delete project successful

Projects	3- Modify project	3- Modify the projects name
Projects	3- Modify project	3- Modify the projects images associated
Projects	4- Search file	4- Search a file using its name
Reports	5- Create report	5- Check that all the information of a report was displayed
Reports	5- Create report	6- Check that all the information created was saved
Projects	7- Set email account	7- Save the email account
Projects	7- Set email account	8- Use the email account to share a report
Projects	8- Change email account	9- Change the email account saved
Reports	6- Delete report	10- Check the reports can be deleted
Reports	9- View reports	11- Check that the report can be opened
Images	10- Create orthomosaics	12- Check that the orthomosaic image is created
Reports	11- Share obtained reports	13- Check that the reports can be send in a pdf file through email
Reports	12- Print reports	14- Check that the report can be send to the printer
Projects	13- View data from project	15- Be able to see the information of the selected project on the program
Projects	14- Save the graphs created	16- Check that the graphs can be downloaded as images
Model trained	15- Identify the faults from the images	17- Check that the program is able to identify the faults on the images and display them on the images as masks
Projects	16- Open project	18- Check that a project is loaded correctly
Reports	17- Create a monthly report	19- Check the report is done
Projects	17- Create a monthly report	20- Verified flying information is asked and saved