Aerial-Imagery (Crop-Health Surveillance) Drones

1) Documentation

In this Project, we use an Aerial imagery Drones to inspect Crop-health to save farmers times by giving them a bird's eye view of crops so they can get information about crops so our **Drone component:**

- Drone:

  Our Drone is suitable for cover a long-range farm.

- Sensors:

  We depend on some sensors to fly and to analysis like (speed and distance Sensor- Infrared and thermal sensors – Image sensors – Lidar -Weather Sensor)

- Camera:

  Camera with bird's eye view

- Storage:

  Solid state drive 240 GB

- Battery

  LiPo Battery with self-charging

Requirements:

- Drone must be charged to perform navigation.
- Drone starts capture the images at range from 100 m to 150 m above the ground.
- Weather must be fine and may fly on rainy weather not on windy and storm weather.
- Mission must be assigned to drone “Scheduled or on demand.”
- Connect drone with phone or laptop to get analysis and photos.

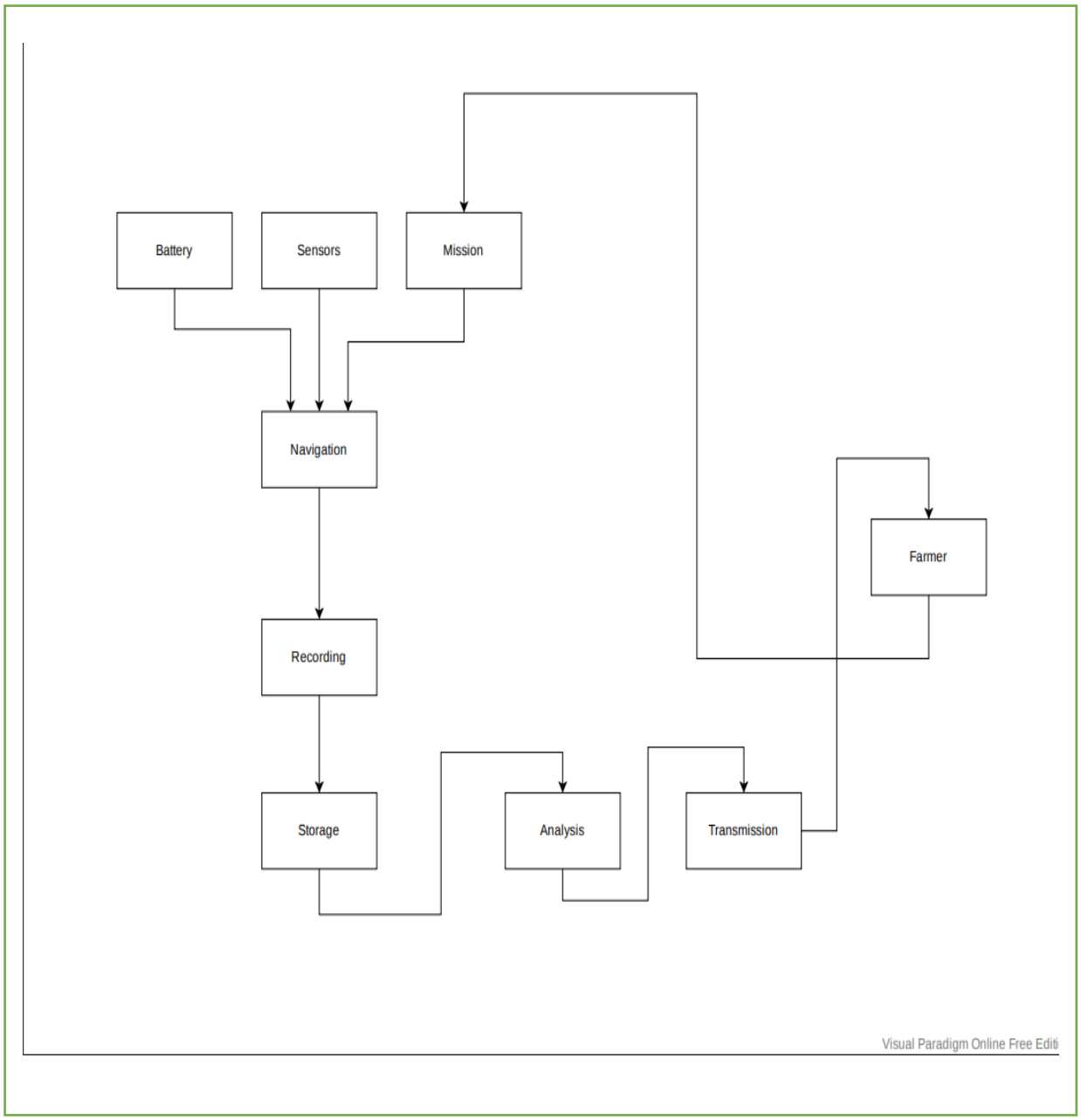
Time Constraints:

	t-min	t-max
Sensors	2ms	5ms
Motor	1µs	2µs
Take off	1s	2s
Recording	1ms	2ms
Analysis	300ms	900ms
Transmission	1s	3s
Charging	60 s	300 s
Total	62.3 s	306 s

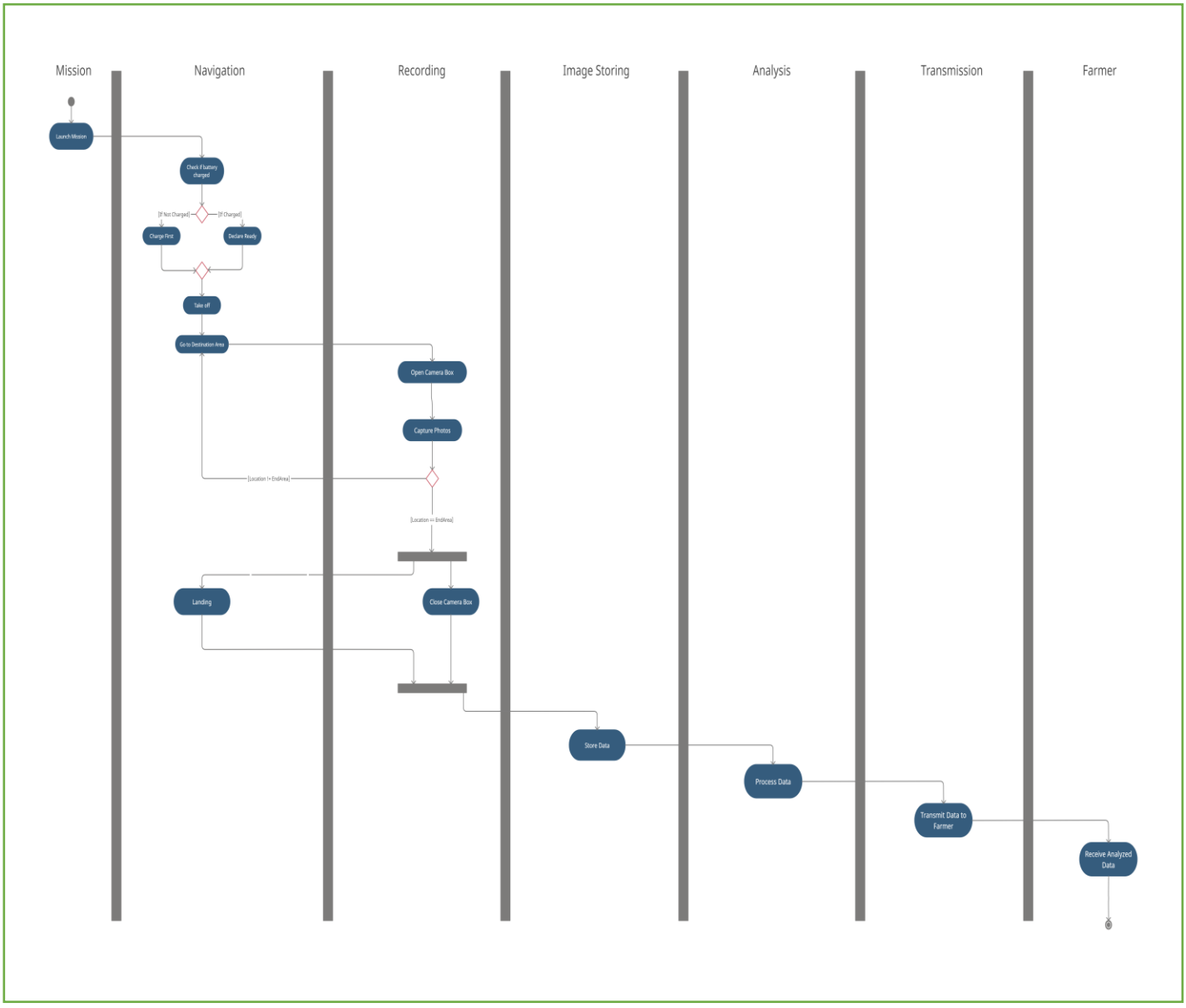
So minimum and maximum time responds of whole system approximately from 1 min to 5 min.

2)Requirements modelling:

- Block Diagram

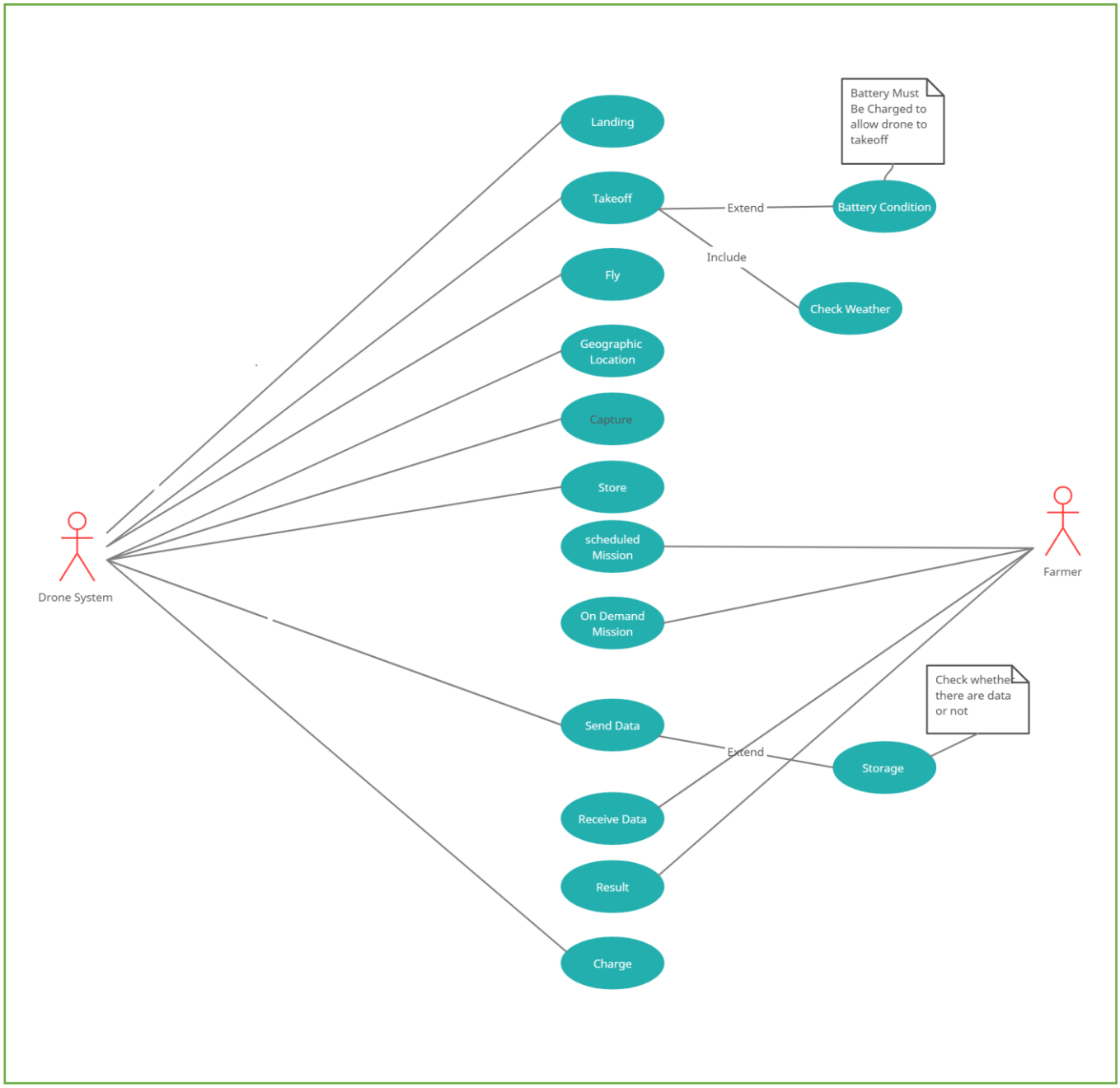


-Activity Diagram



-Use Case Diagram(Using COMET UML profile)

-**Requirements Modelling:**



Identifier and name	Take-off
Summary	In the beginning, the drone starts to fly under some conditions
Dependency	1-Battery Condition 2-Weather
Actors	Drone System
Precondition	The drone must be charged.
Postcondition	The drone will take-off and will start to fly.
Main success Scenario	**1. the drone system will check battery condition and if battery is full so it will take off otherwise it won't.** **2. The drone must check weather.**
Extensions	1.a-Battery is low. 1.a.1- The Drone will start charging. 2.a- the weather is not good. 2.a.1- the drone will not take off.
Non-Functional	• performance requirement: The drone must take off in 5 seconds.

Identifier and name	Fly
Summary	The Drone is flying after checking many conditions
Dependency	None
Actors	Drone System
Precondition	Take off
Postcondition	Drone is flying
Main success Scenario	**1. After taking off, the drone is flying**
alternative	None
Non-Functional	availability requirement: The drone must be operational all the time of the mission.

Identifier and name	Capture
Summary	Drone captures some images to inspects crop health and gathers crop stress data
Dependency	Camera and Storage
Actors	Drone System
Precondition	The drone must be in the specified location.
Postcondition	The drone captures some images and store it.
Main success Scenario	1. The drone is in the specified location. 2. It covers a specified area. 3. It captures images for the crops. 4. It stores the images.
alternative	None
Non-Functional	1. performance requirement: the images must be clear with details.

Identifier and name	Store
Summary	A function which stores the captured images to send it later to the farmer.
Dependency	None.
Actors	Drone System
Precondition	Some images must be captured.
Postcondition	Captured images will be stored.
Main success Scenario	1. Drone will capture some images. 2. It will store it to send it later.
alternative	None
Non-Functional	None

Identifier and name	Send Data
Summary	The drone sends captured images to the farmer to analyse it.
Dependency	Whether there are data or not(Storage).
Actors	Drone System
Precondition	The drone must capture images for crops
Postcondition	The drone will send those images of the crops to the farmer
Main success Scenario	1. The drone will send captured images to the farmer
alternative	There are no data(images) to be send.
Non-Functional	None

Identifier and name	Receive Data
Summary	The drone will capture some images and send it to the farmer so we will use this function to receive these data
Dependency	Storage
Actors	Farmer
Precondition	Images will be stored in drone system
Postcondition	The drone system will send those images to the farmer
Main success Scenario	1. The Drone will store data. 2. It will send these data to the farmer. 3. The farmer will receive these data to inspects crop health and gathers crop stress data.
alternative	Images are not sent
Non-Functional	None.

Identifier and name	Scheduled Mission
Summary	The farmer may schedule mission for the drone system.
Dependency	The farmer
Actors	Farmer
Precondition	None
Postcondition	A scheduled mission will be added to the drone system
Main success Scenario	The farmer will schedule a mission.
alternative	None
Non-Functional	None

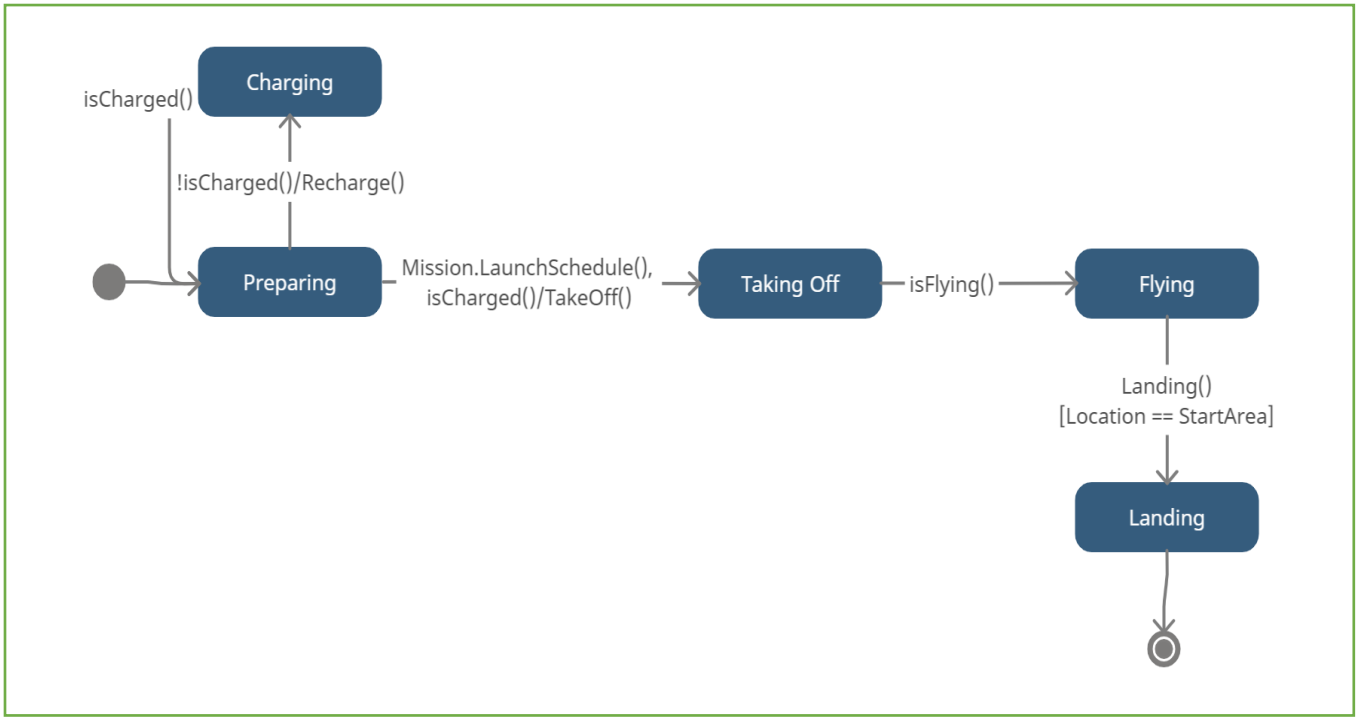
Identifier and name	On Demand Mission
Summary	The farmer may make a “On Demand Mission” for the drone system
Dependency	Farmer
Actors	Farmer
Precondition	None
Postcondition	“On Demand Mission” will be made for the drone system
Main success Scenario	The farmer will make a “On Demand Mission” for the drone system
alternative	None
Non-Functional	None

Identifier and name	Result
Summary	It is the result of the analysis which the farmer will make it.
Dependency	Drone images and the analysis.
Actors	Farmer
Precondition	Analysis.
Postcondition	The farmer will be able to know the reasons for the stress which decrease crop yields.
Main success Scenario	4. Analysis 5. Result of this analysis. 6. The farmer will make actions.
alternative	None
Non-Functional	performance requirement: High Accuracy

Identifier and name	Landing
Summary	Function to land the drone after finishing its work.
Dependency	Scheduled mission or On Demand missions.
Actors	Drone System.
Precondition	Scanning the area, capturing images and sending them to the farmer.
Postcondition	Turn off
Main success Scenario	1.fininshing its work. 2.Returns to its box.
alternative	None
Non-Functional	Performance requirements: Time Accuracy.

-**State Machine Diagram** & **Stimuli/response** Identification

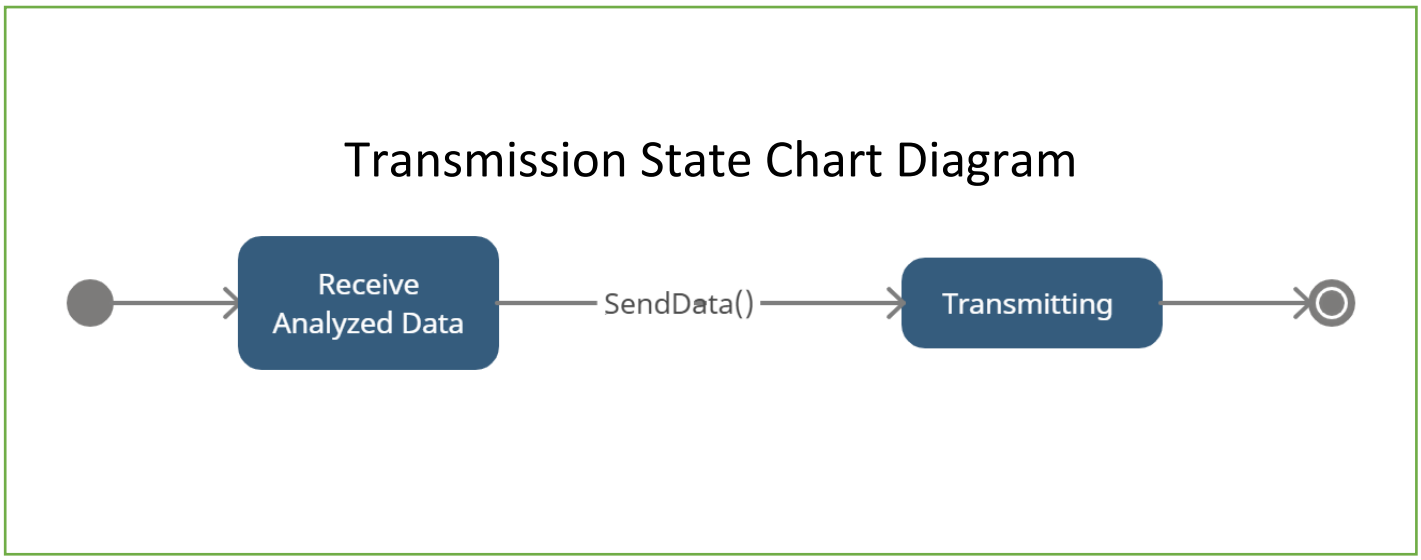
Navigation State chart diagram



Stimuli: Flying Drone

Response: Charging, Flying , Landing

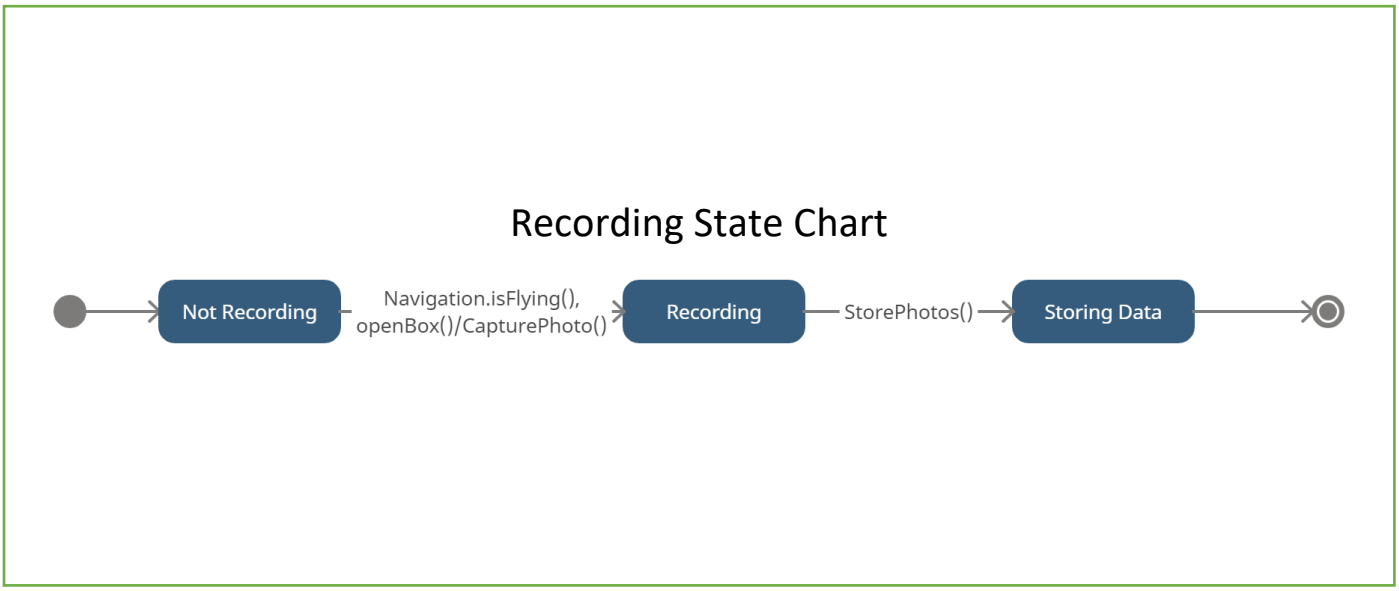
State(current)/State(next)	Preparing	Charging	Taking off	Flying	Landing
Preparing	———	Recharge() /Charging	Mission Lunched/ Takeoff	———	———
Charging	Is Charged=true / Prepare	———	———	———	———
Taking off	———	———	———	Flying/fly	———
Flying	———	———	———	———	Landing/land
Landing	———	———	———	———	———



Stimuli: Transfer image and analyzed data

Response: transmit data to farmer

State(current)/State(next)	Receive Analyzed Data	Transmitting
Receive Analyzed Data	———	send Data/ data transmitted to farmer
Transmitting	———	———



Stimuli: Open Camera and recording

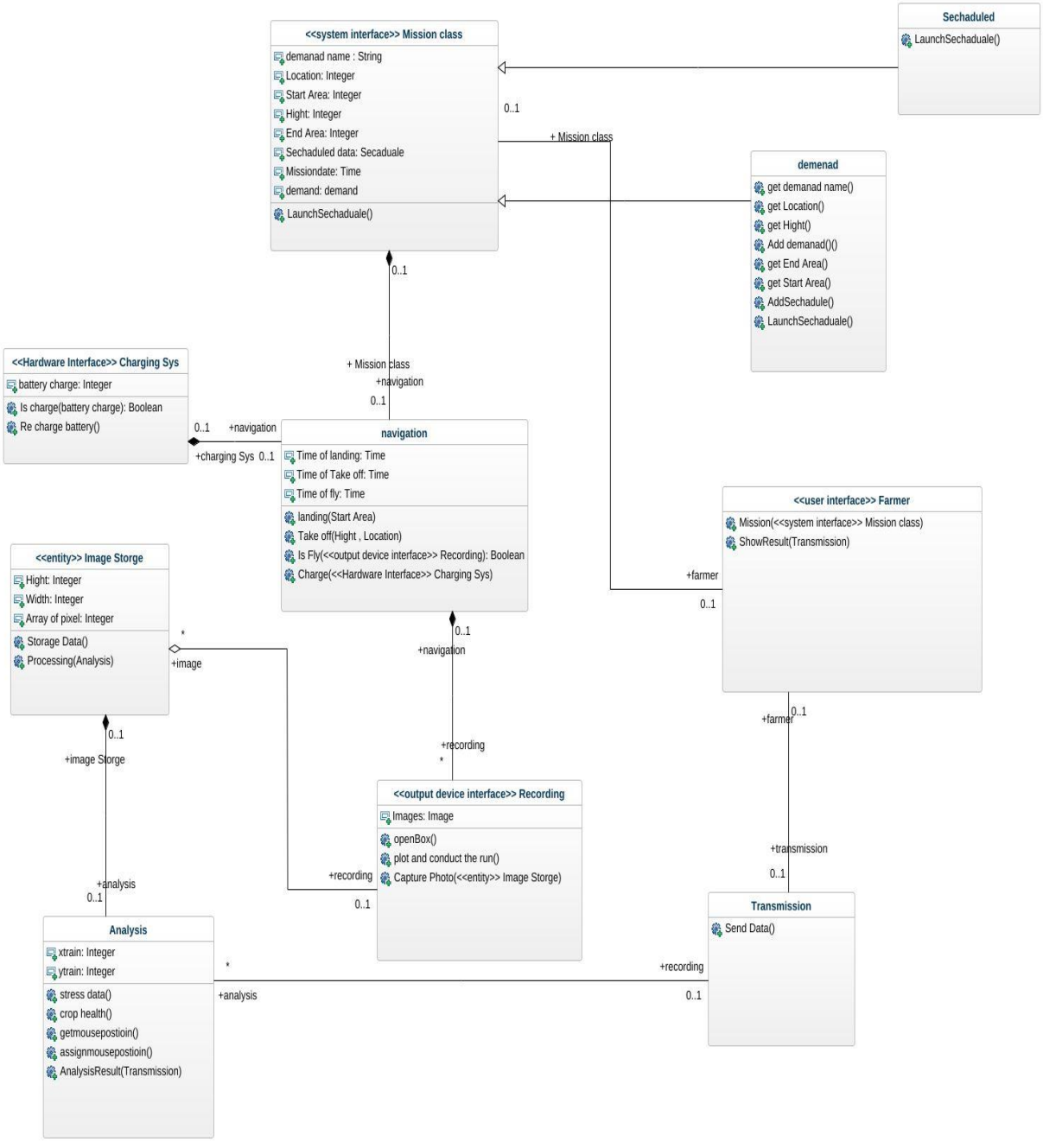
Response: Recording and store images

State(current)/State(next)	Not Recording	Recording	Storing Date
Not Recording	_____	Is flying==true / Open Camera box and record	_____
Recording	_____	_____	Store Date/save data in storage
Storing Date	_____	_____	_____

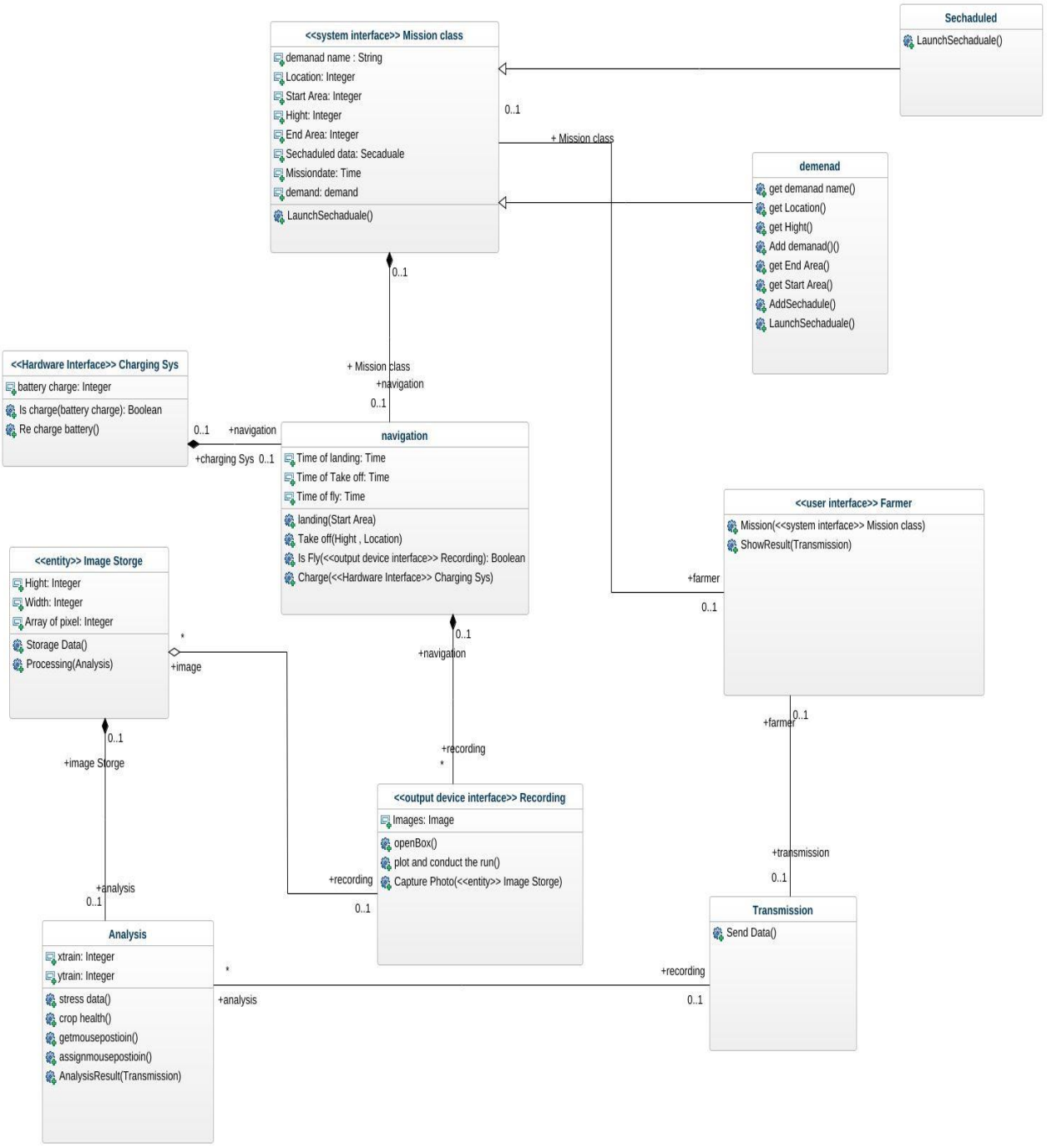
B-Static Analysis

-class Diagram

Class Diagram Version 1



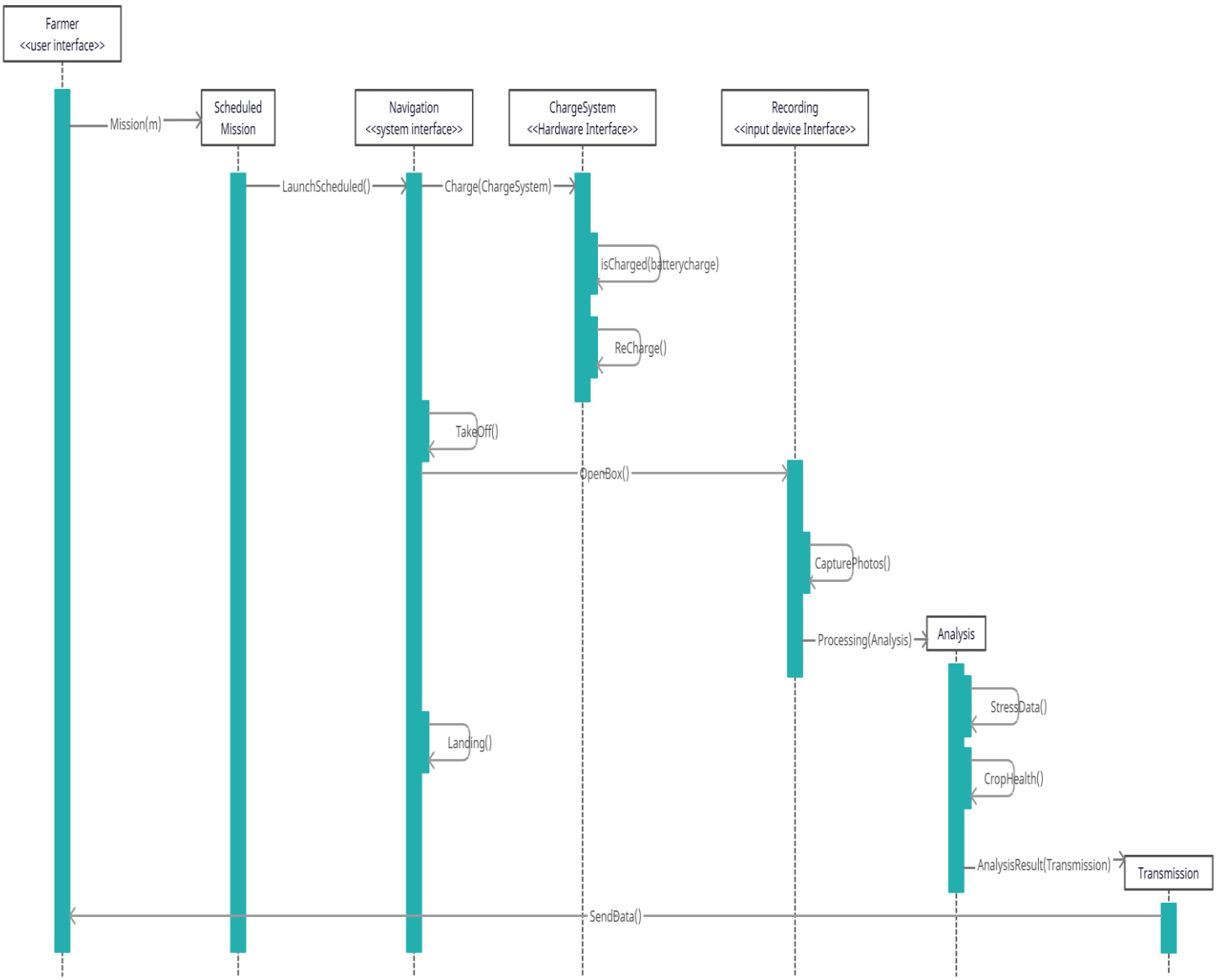
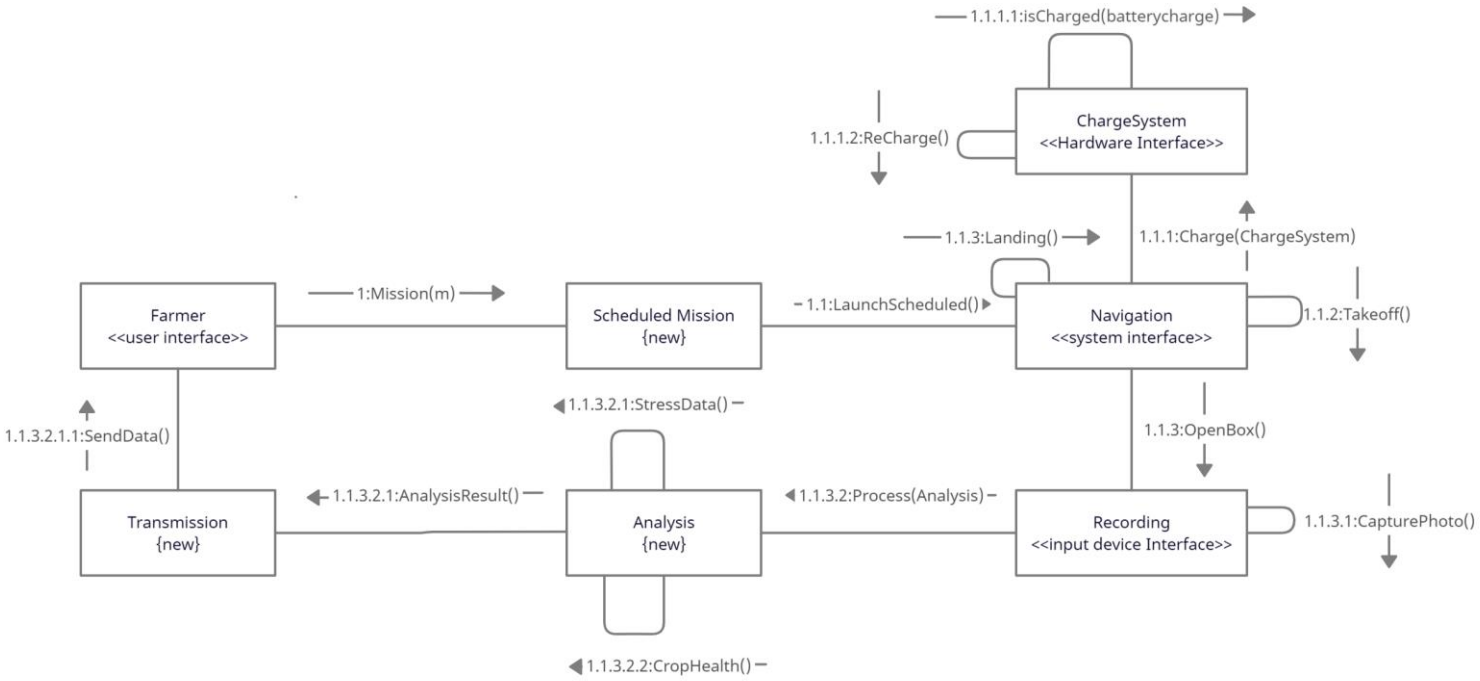
Class Diagram Version 2



C- Dynamic Analysis

Note: State Chart Diagram is before Class diagram

-Sequence Diagram



-Collaboration Diagram