SENG 696- Assignment 2 (Group 9)

(Ahmad Elkholy, Falak Koli, Sparsh Mehta, Yash Mishra)

Project Objective:

Classification of user fed images into categories using a Machine learning model and storing the information into a Data Base, all done on a computer with weak processing power and a powerful backend.

Agents:

There will be 4 Agents performing specific roles in the multi agent software system-

- I.A. Agent (Image Acquisition Agent)
- F.V. Agent (File Verification Agent)
- I.P Agent (Image processing Agent)
- Data Base Agent

Workflow:

The system works with the 4 agents communicating with specific protocols and permissions as discussed below (according to GAIA methodology and SPADE as the platform)-

- The Graphical User Interface (GUI) will be the first point of contact in this multi agent software system. This is on a weak machine with no capabilities of running a full-fledged ML algorithm.
- The GUI will determine if the user wants to just process an image or wants to generate a data base report (needs admin credentials).
- User will enter an image into the GUI. The F.V. Agent will then verify the file type, and if acceptable, I.A. Agent will encode and send the image (in the form of a string) to the I.P. Agent for processing through a robust information sharing protocol.
- The second (I.P.) and the third (D.B.) Agents are on the more capable machine that runs the ML models easily.
- The I.P. Agent receives that string, decodes it back to an image and sends back an acknowledgement. It then feeds the picture to the Machine Learning model and retrieves the results.
- The results are then communicated back to the I.A. Agent for display to the user via the GUI.
- A copy of the results is sent to the Data Base Agent- for storage into a tabular format.
- The data base agent also is responsible for generating a report for when a user with the Administrator credentials requests it through the GUI.

Role Model:

Role Schema		Interaction with user for Image Acquisition
Description		Gets a user input file from the user through a GUI, sends it for verification and processing
Protocols and Activities		GetImage, RequestFileVeri., RequestLabeledImage, SendImageData, EncodeImage
Permissions		Is the master agent; Writes ImageData to DB; Reads Image, ImageReport, Status from DB
Responsibilities	Liveness	ENCODE_IMAGE = GetImage.Encode GET_INPUT_IMAGE = GetImage.VerifyImage.DisplayRespond DISPLAY_REPORT = GetImage.RequestGenerateReport.DisplayReport
	Safety	GUI service established and running

Role Schema		Verification of user file for an image file
Description		Checks the user provided file for image extensions (viz. .jpeg, .png etc)
Protocols and Activities		SendVeri.Report
Permissions		Is a slave agent; Reads RequestFileVeri and responds with SendVeri.Report
Responsibilities	Liveness	CHECK_FILE = GetFile.VerifyImage VERI_REPORT = GetFile.RequestGenerateReport.DisplayReport
	Safety	File should fall under the extensions of an image file

Role Schema		Image Processing by object detection and image labeling
Description		Verified image is feeded into the ML model and results are reported back
Protocols and Activities		SendLabeledImage, DecodeImage
Permissions		Is a slave agent; Reads GetImage.SendImageResize
Responsibilities	Liveness	DECODE_IMAGE = SendImage.Decode ENCODE_LABELED_IMAGE = LabeledImage.Encode
	Safety	ML model is trained and tested Successful connection with the Image Acquisition

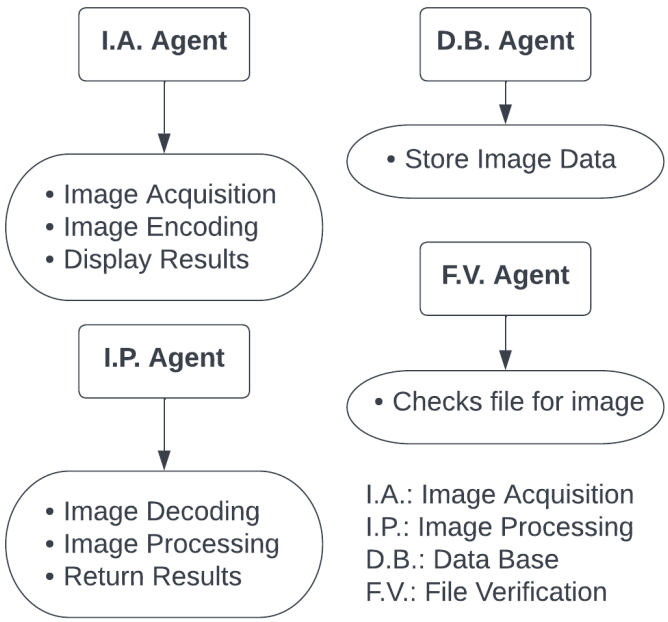
Role Schema		Storage of Image data into a CSV data base
Description		Inputs the data obtained after image processing into a CSV file
Protocols and Activities		SendStatus, SendImageData
Permissions		Is a slave agent; Reads ImageData, LabeledImageReport; Writes CSV
Responsibilities	Liveness	FEED_IMAGE_DATA = RequestImageReport.Update
	Safety	Successful connection with the Image Acquisition

Interaction Model:

Protocol	RequestFileVeri.	RequestLabeledImage
Purpose/Parameters	Asks for verification for the file type	Sends the user image for processing and labeling
Initiators(s)	Image Acquisition through the GUI	Image Acquisition after verification
Receiver(s)	File Verification	Image Processing
Processing	It will check whether or not the provided file falls in the list of extensions for an image file (E.g., JPEG, PNG, RAW etc.)	Verified image file is encoded and handed over to the Image Processing service

Protocol	RequestStoreData
Purpose/Parameters	Requests the DB to store the results of the ImageData
Initiators(s)	Image Processing after processing and labeling
Receiver(s)	Data Base
Processing	Stores the image data into a tabular format (CSV)

Agent Model:

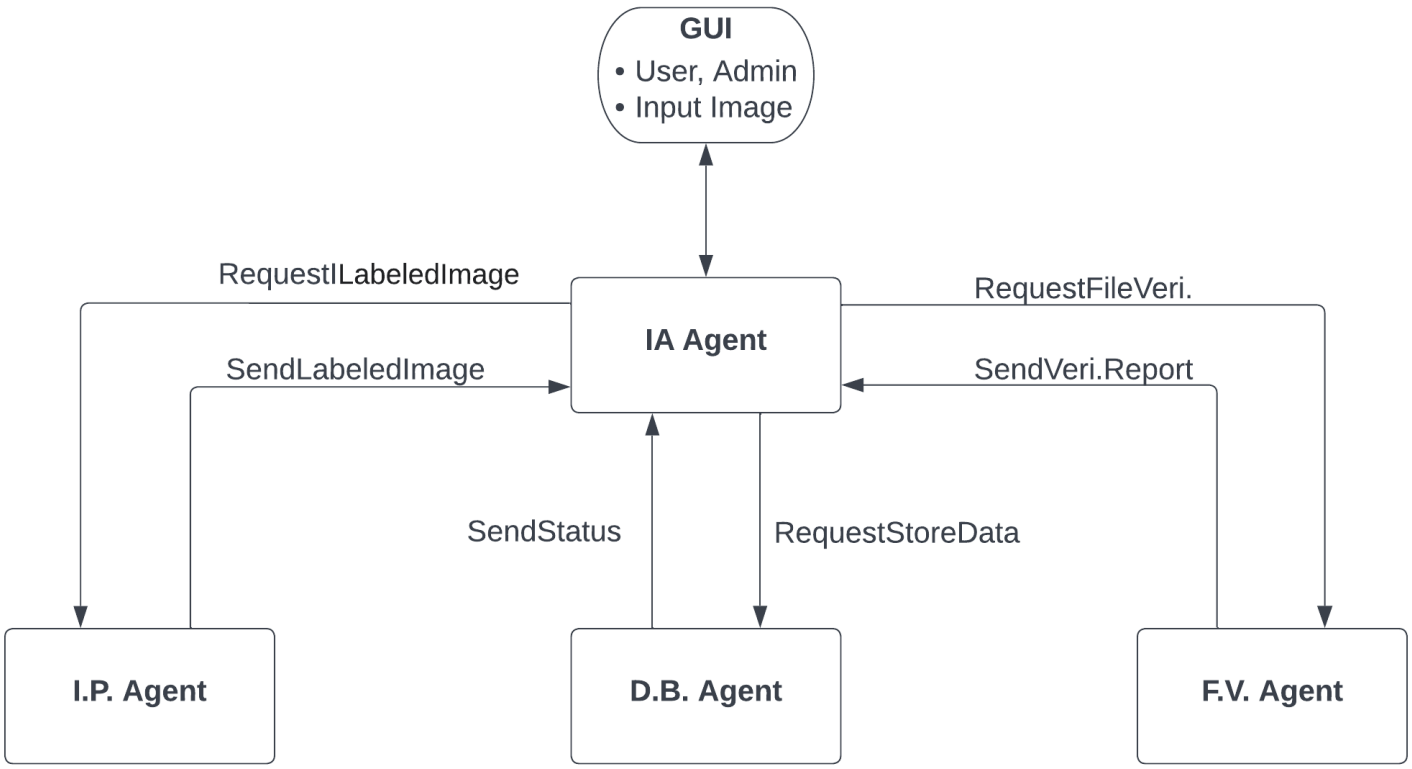


Service Model:

Service	Inputs	Outputs
Interaction with user for Image Acquisition	User Image	Encoded Image
Verification of user file for an image file	User File	Verification of file
Image Processing by object detection and image labeling	Encoded & Unprocessed Image	Encoded Labeled Image
Storage of Image data into a CSV data base	Labeled Image Data	Storage of the data

Service	Pre-Conditions	Post-Conditions
Interaction with user for Image Acquisition	GUI is launched and working	Established connection with GUI and all slave services
Verification of user file for an image file	File available	Successful connection with Image Acquisition service
Image Processing by object detection and image labeling	ML algorithm is up and running	Successful connection with Image Acquisition service
Storage of Image data into a CSV data base	CSV file exists	Secure connection with Image Acquisition Service

Acquaintance Model:



Communication Protocol:

Agent	ID
IA Agent	1
IP Agent	2
FV Agent	3
DB Agent	4

Error Code	ID
Success	0
Failure	1

Command Format			
Sender Agent ID	Receiver Agent ID	Error Code	Data
1Char	1Char	1Char	X Char

Message Sequence Chart

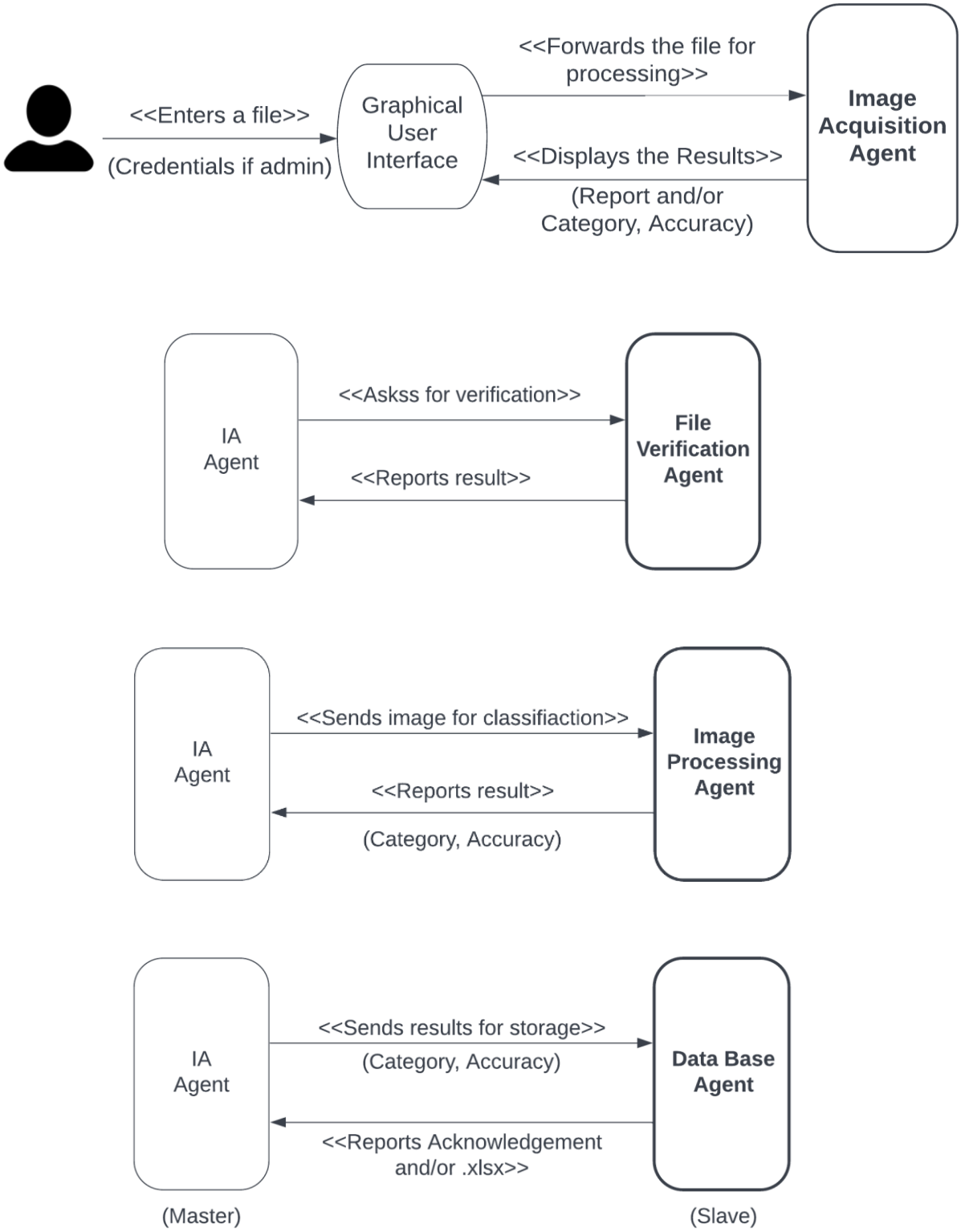
Command	Sender Agent ID	Receiver Agent ID	Error Code	Data	Description
RequestFileVeri.	"1"	"3"	"0"	"FileName"	Checking file for image
SendVeri.Report	"3"	"1"	"0" or "1"	"0" or "1"	Reporting file type
					0=True, 1=False
RequestLabelledImage	"1"	"2"	"0"	"File_String"	Requesting image processing
SendLabeledImage	"2"	"1"	"0" or "1"	"Classification:Accuracy"	Image Report
					Accuracy Format- XXX.XX%
RequestStoreData	"1"	"4"	"0"	"Image_ID:Classification:Accuracy"	Requesting DB Update
					Accuracy Format- XXX.XX%
SendStatus	"4"	"1"	"0" or "1"	"0" or "1"	Acknowledge data storage
					0=Success, 1=Failure

Applications of said system:

There are 3 major use cases for this system-

1. One, where there is a low capability machine wanting to achieve image characterization.
2. This system can also be useful for educating toddlers or preschoolers about the types of animals when provided with an image in an interactive manner.
3. Data base analysis.

Use Case (Agent wise):



Code Flow Chart: