Here's the updated outline of the Python code architecture for your AI agent application, including the additional features and modules:

gui: Handles the graphical user interface, layout, user input, and display of output messages.

create\_main\_window

create\_output\_text\_boxes

create\_input\_fields

bind\_events

update\_output

create\_agent\_selection

display\_agent\_info

prelaunch\_check: Performs prelaunch checks, such as validating API keys, checking the OS, and setting up the environment.

check\_env\_files

validate\_openai\_key

validate\_google\_key

get\_chatgpt\_model

check\_os

memory\_management: Manages the AI agent's short-term and long-term memory using the SQLite database.

create\_db\_connection

create\_memory\_tables

add\_memory\_item

get\_memory\_item

update\_memory\_item

delete\_memory\_item

git\_integration: Handles interactions with Git for version control.

clone\_repository

init\_repository

stage\_changes

commit\_changes

pull\_changes

push\_changes

create\_branch

switch\_branch

merge\_branch

web\_browsing: Handles web browsing using the Google API key.

search\_web

get\_web\_content

parse\_web\_content

file\_operations: Handles file reading and writing operations on the native machine.

read\_file

write\_file

delete\_file

ai\_agent: Core AI agent class that integrates all the above modules and manages the agent's behavior.

init

del

handle\_user\_input

generate\_response

reflect\_on\_response

execute\_command

update\_memory

generate\_agent\_id

set\_agent\_environment

get\_agent\_resources\_path

agent\_manager: Manages multiple AI agents, their storage, and environments.

list\_agents

get\_agent\_size

create\_agent

load\_agent

delete\_agent

switch\_agent\_environment

settings: Manages user settings and preferences.

load\_settings

save\_settings

apply\_settings

logging: Handles application logging.

setup\_logging

log\_event

log\_error

update\_management: Manages application and AI agent updates.

check\_for\_updates

download\_updates

apply\_updates

help: Provides built-in help and documentation resources.

display\_help

search\_documentation

performance\_monitoring: Monitors AI agent performance and resource usage.

monitor\_performance

display\_performance\_metrics

security: Handles the secure storage and handling of user data, API keys, and sensitive information.

encrypt\_data

decrypt\_data

secure\_storage

This updated outline incorporates the additional features and modules discussed earlier, creating a comprehensive architecture for your AI agent application. This design provides a solid foundation for building a user-friendly, secure, and feature-rich application that can be easily maintained and extended in the future.