**GUI Interaction Sample1:**

1. Run program and display initial display GUI image
2. “Open Image” button is clicked, and a uncoded image is uploaded

3. A new text message is entered, and “Encode Image” button is clicked

4. Reset Image/Save Image

4.1 Reset Image

4.2 Save Image

**GUI Interaction Sample2:**

1. Run program and display initial display GUI image
2. “Open Image” button is clicked, and a coded image is uploaded

3. Text message is modified, and “Encode Image” button is clicked

4. Reset Image/Save Image

4.1 Reset Image

4.2 Save Image

**GUI Interaction Sample1:**

1. **Run program and display initial display GUI image.**

Access File/Input data:

images file: “welcome\_image.jpeg” – String

Graphical user interface, application

Description automatically generated

(Current input/

Maximum input)

Button disabled

Default Welcome Image

Placeholder

1. **“Open Image” button is clicked, and a uncoded image is uploaded.**

Call following functions and get input data for GUI:

1. def open\_image\_file (): “uncoded\_image\_1.jpeg” – String
2. def decode\_secret\_message (): decoded text message from uploaded file – String
3. def pixelator (): using pixel data list get maximum words capacity for the text field – int

Variables/Return data from GUI:

1. textfield \_widget.text: input text in the text field – String
2. textfield\_widget.text \_count: Current input words count – int

Display message “Note: No message is encoded in the uploaded image”:

1. def decode\_secret\_message (): decoded text message from uploaded file is empty String

Graphical user interface, application

Description automatically generated

Button disabled

Input secret data

(Current input/

Maximum input)

Display message

Uploaded Image

1. **A new text message is entered, and “Encode Image” button is clicked.**

Call following functions and get input data for GUI:

1. def open\_image\_file (): “uncoded\_image\_1.jpeg” – String

Display message “Note: The secret message is successfully encoded!”:

1. def encode\_secret\_message (): No error occurs in the function call.

Graphical user interface, text, application

Description automatically generated

Uploaded Image

(Current input/

Maximum input)

Display message

Button enabled

1. **Reset Image/Save Image**
   1. **“Reset Image” button is clicked.**

Call following functions and get input data for GUI:

1. def open\_image\_file (): “uncoded\_image\_1.jpeg” – String

Display message “Note: The image has been reset!”:

1. def reset\_image (): No error occurs in the function call.

**Graphical user interface, application

Description automatically generated**

Reset count

(Current input/

Maximum input)

Delete input data,

display Placeholder

Display message

Button disabled

Uploaded Image

**4.2 “Save Image” button is clicked.**

Access File/Input data:

1. images file: “welcome\_image.jpeg” – String

Display message “Note: The encoded image has been successfully saved!”:

1. def save\_image\_file (): No error occurs in the function call

Graphical user interface, application

Description automatically generated

Reset count

(Current input/

Maximum input)

Button disabled

Display

Placeholder

Display message

Default Welcome Image

**GUI Interaction Sample2:**

1. **Run program and display initial display GUI image.**

Access File/Input data:

images file: “welcome\_image.jpeg” – String

Graphical user interface, application

Description automatically generated

(Current input/

Maximum input)

Button disabled

Default Welcome Image

Placeholder

1. **“Open Image” button is clicked, and a coded image is uploaded.**

Call following functions and get input data for GUI:

1. def open\_image\_file (): “coded\_image\_1.jpeg” – String
2. def decode\_secret\_message (): decoded text message from uploaded file – String
3. def pixelator (): using pixel data list get maximum words capacity for the text field – int

Variables/Return data from GUI:

1. textfield \_widget.text: input text in the text field – String
2. textfield\_widget.text \_count: Current input words count – int

Display message “Note: Below is the secret message encoded in the uploaded image.”:

1. def decode\_secret\_message (): decoded text message from uploaded file is not empty.

Graphical user interface, text, application

Description automatically generated

(Current input/

Maximum input)

Button disabled

Decoded secret data

Display message

Uploaded Image

1. **Text message is modified, and “Encode Image” button is clicked.**

Call following functions and get input data for GUI:

1. def open\_image\_file (): “coded\_image\_1.jpeg” – String

Display message “Note: The secret message is successfully encoded!”:

1. def encode\_secret\_message (): No error occurs in the function call.

Graphical user interface, application

Description automatically generated

Uploaded Image

Modified

Secret data

(Current input/

Maximum input)

Button enabled

Display message

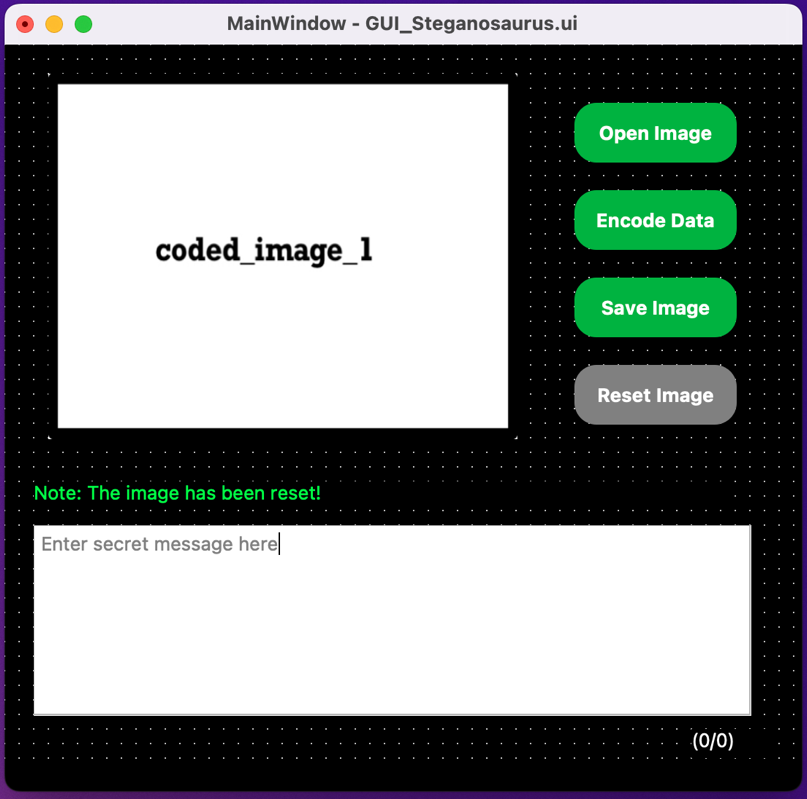
1. **Reset Image/Save Image**
   1. **“Reset Image” button is clicked.**

Call following functions and get input data for GUI:

1. def open\_image\_file (): “coded\_image\_1.jpeg” – String

Display message “Note: The image has been reset!”:

1. def reset\_image (): No error occurs in the function call.



Reset count

(Current input/

Maximum input)

Button disabled

Delete input data,

display Placeholder

Display message

Uploaded Image

**4.2 “Save Image” button is clicked.**

Access File/Input data:

1. images file: “welcome\_image.jpeg” – String

Display message “Note: The encoded image has been successfully saved!”:

1. def save\_image\_file (): No error occurs in the function call.

Graphical user interface, application

Description automatically generated

Reset count

(Current input/

Maximum input)

Button disabled

Display

Placeholder

Display message

Default Welcome Image