🛠️ Debugging Guide: Description Textarea Not Found

This guide provides the steps to follow when the browser automation breaks because Google Photos has updated its underlying HTML structure (i.e., the CSS selector is no longer valid). The problem is almost certainly a change in the auto-generated class names like kmqzh or qURWqc.

The Fix: Rerunning the HTML Analyzer

The entire fix involves running the **dump-explorer.py** utility on a fresh HTML dump to find the new class names and updating a single line of code in browser_controller.py.

Step 1: Get a Fresh HTML Dump

If the automation is running, use your existing BrowserController functions to capture the latest HTML:

1. Use the browser application interface (if running in debug mode) or manually execute the command to dump the current page's HTML to a new file (e.g., latest_dump.html).
    - *If you are running the AssistantUI in debug mode, press the **DUMP HTML** button and save the resulting file.*
2. Ensure this dump is a single photo's detail page where the description field is visible.

Step 2: Analyze the New HTML Dump

Run the **dump-explorer.py** script against the new HTML file you just saved.
Example command using the dump file
./dump-explorer.py latest_dump.html

What to Look for in the Output:

The script will output a section called **Ancestor Path** and a **SUMMARY**. You are looking for the new class names that replaced kmqzh and qURWqc.
Example of the OLD Output (Reference):
[TEXTAREA 1]
  > Immediate Parent Info:
    - ID: None
    - Class: kmqzh   <-- PARENT CLASS (Most fragile)
  > Ancestor Path (Parent > Grandparent): div.kmqzh > div.qURWqc > ... <-- GRANDPARENT CLASS (More stable reference)

Step 3: Update the Selector in browser_controller.py

You only need to update the single, centralized constant in your browser_controller.py file.

1. Open **browser_controller.py**.
2. Locate the constant DESCRIPTION_TEXTAREA_SELECTOR.
3. Update the selector string using the new class names found in Step 2.

The Target File: browser_controller.py

```
"""Browser controller - extracted from inject_v3.py"""
# ... imports

class BrowserController:
    # ...
    # --- ADDED: CENTRALIZED CSS SELECTORS FOR RESILIENCE ---
    # Update this line when the UI breaks.
    # Pattern: 'div.<Grandparent_Class> > div.<Parent_Class> > textarea'
    DESCRIPTION_TEXTAREA_SELECTOR = 'div.<NEW_GRANDPARENT_CLASS> >
div.<NEW_PARENT_CLASS> > textarea'
    # For your original dump, this was: 'div.qURWqc > div.kmqzh > textarea'
    # ------------------------------------------------

    def __init__(self):
        # ... rest of __init__
```

Summary of Selector Components

Component	Example Value	Role in Resilience
Parent Div Class	kmqzh	Contains the textarea. *Most likely to change.*
Grandparent Div Class	qURWqc	Contains the Parent Div. Using this path makes the selector more specific and robust by relying on two classes instead of one.
New Selector	'div.qURWqc > div.kmqzh > textarea'	The combined, most resilient selector based on the current HTML structure.

Once you update the DESCRIPTION_TEXTAREA_SELECTOR constant in browser_controller.py with the new values and restart your application, it should be able to locate the description field again.