Appendices

A Algorithm

In Algorithm 1, we include a formal definitions of our practical algorithm of PAE as presented in Section 3.

Algorithm 1 Proposer-Agent-Evaluator: Practical Algorithm

Require: Context information $z_{\mathcal{M}}$, task proposer $\hat{\mathcal{C}}$, autonomous evaluator $\hat{\mathcal{R}}$.

Initialize policy π from a pre-trained checkpoint.
Initialize replay buffer $\mathcal{D} \leftarrow \{\}$.
Propose tasks based on the context information.
Obtain proposal task distribution $\hat{\mathcal{C}}(z_{\mathcal{M}})$.
for each global iteration **do**
    for each trajectory to be collected **do**
        Sample a task from the task proposer $c \sim \hat{\mathcal{C}}(z_{\mathcal{M}})$.
        Reset the environment to obtain the initial observation s_0
        for each environment step t **do**
            Sample $a_t \sim \pi(\cdot|s_t, c)$, $s_{t+1} \sim \mathcal{T}(\cdot|s_t, a_t, c)$.
            if done **then**
                ## Autonomously evaluate the outcome of the agent rollout.
                $r_t \leftarrow \hat{\mathcal{R}}(s_t, a_t, c)$.
            else
                $r_t \leftarrow 0$.
            end if
            $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r_t, s_{t+1}, c)\}$.
        end for
    end for
    ## Update the agent policy with any RL algorithm.
    $\pi \leftarrow \text{RL\_update}(\pi, \mathcal{D})$
end for

B All prompts in the experiments

For completeness, we include examples of the prompts that we have used in this section. In particular, in Figure 7, we have provided the prompt that we used for the Claude-Sonnet-3 autonomous evaluator to evaluate the success for the task completion for all tasks in WebArena. A similar is used for all tasks in WebVoyager. In Figure 8, 9, 10 we have included the prompts that we used for generating the proposal tasks for each domain. We used the same prompts with 3 additional website screenshots appended to the messages for PAE + User Demos. It is worth noting that our task proposers are domain-general and have little domain customizations. In particular, for all 13 real-world websites from WebVoyager, we use the same prompt to generate tasks except with the placeholder of "web_name". This shows that our PAE framework can easily scale to multiple websites without the need for domain-specific knowledge. The prompt for zero-shot VLM agents are included in Figure 11, 12, and 13.

C Prompts For Zero-Shot VLM Agents

We also append the prompts (Figure 11, 12, and 13) that we used for the zero-shot baselines including Claude-Sonnet-3, Claude-Sonnet-3.5, Qwen2VL, InternVL2b5, LLaVa-1.6-7B, and LLaVa-1.6-34B. The prompt for WebVoyager tasks largely follow from that used in the prior literature (He et al., 2024). We include additional necessary domain knowledge of the WebArena tasks and evaluation protocols in the prompt that we used for WebArena.

Autonomous Evaluator Prompt

You are an expert in evaluating the performance of a web navigation agent. The agent is designed to help a human user navigate a website to complete a task. Your goal is to decide whether the agent's execution is successful or not.

As an evaluator, you will be presented with three primary components to assist you in your role:

1. Web Task Instruction: This is a clear and specific directive provided in natural language, detailing the online activity to be carried out.

2. Result Response: This is a textual response obtained after the execution of the web task. It serves as textual result in response to the instruction.

3. Result Screenshots: This is a visual representation of the screen showing the result or intermediate state of performing a web task. It serves as visual proof of the actions taken in response to the instruction.

– You SHOULD NOT make assumptions based on information not presented in the screenshot when comparing it to the instructions.

– Your primary responsibility is to conduct a thorough assessment of the web task instruction against the outcome depicted in the screenshot and in the response, evaluating whether the actions taken align with the given instructions.

– NOTE that the instruction may involve more than one task, for example, locating the garage and summarizing the review. Failing to complete either task, such as not providing a summary, should be considered unsuccessful.

– NOTE that the screenshot is authentic, but the response provided by LLM is generated at the end of web browsing, and there may be discrepancies between the text and the screenshots.

– NOTE that if the content in the Result response is not mentioned on or different from the screenshot, mark it as not success.

You should explicilt consider the following criterions:

- Whether the claims in the response can be verified by the screenshot. E.g. if the response claims the distance between two places, the screenshot should show the direction. YOU SHOULD EXPECT THAT THERE IS A HIGH CHANCE THAT THE AGENT WILL MAKE UP AN ANSWER NOT VERIFIED BY THE SCREENSHOT.

- Whether the agent completes EXACTLY what the task asks for. E.g. if the task asks to find a specific place, the agent should not find a similar place.

In your responses: You should first provide thoughts EXPLICITLY VERIFY ALL THREE CRITERIONS and then provide a definitive verdict on whether the task has been successfully accomplished, either as 'SUCCESS' or 'NOT SUCCESS'.

A task is 'SUCCESS' only when all of the criteria are met. If any of the criteria are not met, the task should be considered 'NOT SUCCESS'.

Figure 7: The prompt used by the autonomous evaluator for Claude-Sonnet-3. Same prompt is used to evaluate tasks from WebArena websites. The evaluator takes as inputs the task description, the response from the agent's ANSWER action, and last three screenshots in the trajectory. The evaluation result is a binary verdict of 'SUCCESS' or 'NOT SUCCESS'.

Task Proposer Prompt for WebVoyager

{"web_name": "Apple", "id": "Apple–40", "ques": "Find the pricing and specifications for the latest Mac Studio model, including the available CPU and GPU options.", "web": "https://www.apple.com/"}

We are training a model to navigate the web. We need your help to generate instructions. With the examples provided above, please give 25 more example tasks for the model to learn from in the domain of {web_name}. You should imagine tasks that are likely proposed by a most likely user of this website. A few demos of users navigating through the web are provided above.

YOU SHOULD MAKE USE OF THE DEMOS PROVIDES TO GENERATE TASKS, SO THAT YOUR TASKS ARE REALISTIC AND RELEVANT TO THE WEBSITE.

Please follow the corresponding guidelines:

1)First output your thoughts first on how you should come up with diverse tasks that examine various capabilities on the particular website, and how these tasks reflect the need of the potential user. Then you should say 'Output:' and then followed by the outputs STRUCTURED IN JSONL FORMAT. You should not say anything else in the response.

2)PLEASE MAKE SURE TO HAVE 25 examples in the response!!!

3)Your proposed tasks should be DIVERSE AND COVER A WIDE RANGE OF DIFFERENT POSSIBILITIES AND DIFFICULTY in the domain of {web_name}. Remember, your job is to propose tasks that will help the model learn to navigate the web to deal with various real world requests.

4)Your task should be objective and unambiguous. The carry-out of the task should NOT BE DEPENDENT on the user's personal information such as the CURRENT TIME OR LOCATION.

5)You should express your tasks in as diverse expressions as possible to help the model learn to understand different ways of expressing the same task.

6)Your tasks should be able to be evaluated OBJECTIVELY. That is, by looking at the last three screenshots and the answer provided by an agent, it should be possible to tell without ambiguity whether the task was completed successfully or not.

7)Your tasks should require a minimum completion steps from 3 to 7 steps, your tasks should have a diverse coverage in difficulty as measured by the minimum completion step. I.E. You should propose not only tasks that may take more than 4 steps to complete but also tasks that can be completed within 3 steps.

8)Humans should have a 100% success rate in completing the task.

9)Your tasks should be able to be completed without having to sign in to the website.

Figure 8: Prompts used by Claude-Sonnet-3 for proposing tasks in WebVoyager experiments. For PAE + User Demos, we use the same prompt with additional user demos appended to the message.

D Details for SFT

SFT for WebVoyager. As shown in Table 1, unlike proprietary VLMs, none of the open-source VLM agent is able to follow the instructions and achieve non-trivial performances in real-world web navigation tasks in the zero-shot manner. Such models can rarely get success rewards in the process of RL, thus leading to very slow convergence. To "warm-up" the open-source VLM agent to achieve a non-trivial performance at the start of RL training, we turn to enhancing the performances with SFT before RL. Note that the SFT process may not be needed if the base VLM agent model can already achieve non-trivial performances such as Claude 3 Sonnet. To prevent data contamination, we gather 85 out-of-distribution real-world websites (listed in Figure 14 and 15), and collect 11220 trajectories in total using Claude 3 Sonnet with the prompt specified in Figure 11. The average trajectory success rate is 25% as measured by our Claude 3 Sonnet evaluator. Each action in the

Task Proposer Prompt for WebArena Map

{"web_name": "map", "id": "map–2", "ques": "Tell me the full address of all international airports that are within a driving distance of 50 km to University of California, Berkeley"}
{"web_name": "map", "id": "map–10", "ques": "I will arrive San Francisco Airport soon. Provide the name of a Hilton hotel in the vicinity, if available. Then, tell me the the shortest walking distance to a supermarket from the hotel."}
{"web_name": "map", "id": "map–17", "ques": "Check if the ikea in pittsburgh can be reached in one hour by car from hobart street"}
We are training a model to navigate the web. We need your help to generate instructions. With the examples provided above, please give 25 more example tasks for the model to learn from in the domain of OpenStreetMap. You should imagine who is the most likely user for the website and propose tasks that are likely to be proposed by this user. Please follow the corresponding guidelines:
1)First output your thoughts first on how you should come up with diverse tasks that examine various capabilities on the particular website, and how these tasks reflect the need of the potential user. Then you should say 'Output:' and then followed by the outputs STRUCTURED IN JSONL FORMAT. You should not say anything else in the response.
2)PLEASE MAKE SURE TO HAVE 25 examples in the response!!!
3)Your proposed tasks should be DIVERSE AND COVER A WIDE RANGE OF DIFFERENT POSSIBILITIES AND DIFFICULTY in the domain of OpenStreetMap. Remember, your job is to propose tasks that will help the model learn to navigate the web to deal with various real world requests. 4)Your task should be objective and unambiguous. The carry-out of the task should NOT BE DEPENDENT on the user's personal information such as the CURRENT TIME OR LOCATION.
5)You should express your tasks in as diverse expressions as possible to help the model learn to understand different ways of expressing the same task.
6)Your tasks should be able to be evaluated OBJECTIVELY. That is, by looking at the last three screenshots and the answer provided by an agent, it should be possible to tell without ambiguity whether the task was completed successfully or not.
7)Your tasks should require a minimum completion steps from 3 to 7 steps, your tasks should have a diverse coverage in difficulty as measured by the minimum completion step. I.E. You should propose not only tasks that may take more than 4 steps to complete but also tasks that can be completed within 3 steps.
8)Humans should have a 100% success rate in completing the task.
9)Your tasks should be able to be completed without having to sign in to the website.

Figure 9: Prompts used by Claude-Sonnet-3 for proposing WebArena Tasks for Map. For PAE + User Demos, we use the same prompt with additional user demos appended to the message. For this domain only, we provide three hand-written in-domain examples to set the right difficulty for the task proposer. Such in-domain examples are not needed for all other domains including Reddit and OneStopMarket, and other real-world WebVoyager websites.

trajectories contains both thoughts and actual web actions shown in Figure 2. All 11220 trajectories are used for SFT. RL training is carried out on top of the SFT checkpoint.

SFT for WebArena. In our preliminary experiments, we found that the SFT checkpoint trained on real-world websites do not generalize well to self-hosted websites on WebArena. This is potentially because of the distribution shift between real-world commercial websites and self-hosted websites. For example, most real-world map websites such as Google Maps and Apple Maps support advanced fuzzy search capabilities such as "pittsburgh to new york" while OpenStreetMap from WebArena will not return any results with such queries. Therefore, we collect 3000 Claude 3 Sonnet generated trajectories each from OpenStreetMap, Reddit, and OneStopMarket websites from WebArena. We

Task Proposer Prompt for WebArena Reddit and OneStopMarket

{"web_name": "Apple", "id": "Apple–40", "ques": "Find the pricing and specifications for the latest Mac Studio model, including the available CPU and GPU options.", "web": "https://www.apple.com/"}

We are training a model to navigate the web. We need your help to generate instructions. With the examples provided above, please give 25 more example tasks for the model to learn from in the domain of {web_name}.

You should provide tasks in the DOMAIN OF {web_name}.

Please follow the corresponding guidelines: 1)First answer how many screenshots are provided and describe in detail the functions of the website that you see from each of the screenshot. Then output your thoughts first. Then you should say 'Output:' and then followed by the outputs STRUCTURED IN JSONL FORMAT. You should not say anything else in the response.

2)PLEASE MAKE SURE TO HAVE 25 examples in the response!!!

4)Your task should start from the home page of the website instead of the shown screenshots.

5)Your task does not need to be the same as real users would do, but it should examine diverse capabilities of the agent to do web navigartion.

6)Your tasks should examine the VERY BASIC functions of the website and should not require complicated web page operations. They can be completed within 5 steps.

7)THIS DOMAIN IS A SELF-HOSTED STATIC DOMAIN AND DIFFERENT FROM POPULAR WEBSITES, DO NOT ASSUME ANY INFORMATION NOT PROVIDED IN THE SCREENSHOTS.

8)Your tasks should examine the capability of the web agent to find some information on the website, navigating to some specific web pages. Do not propose tasks that involve making actual modifications to the websites.

9)Your tasks should result in the agent landing in a single groundtruth web page or finding a single grounth truth answer. The landed webpage can be some specific categories, a drafted post, some search results, or even the homepage of the website. When the task is to to find some information, specify exactly what information the agent should find such as the price, the number of comments, the title, etc. It can also be information about the current account.

Figure 10: Prompts used by Claude-Sonnet-3 for proposing WebArena Tasks for Reddit and OneStopMarket. For PAE + User Demos, we use the same prompt with additional user demos appended to the message.

use the prompts from Figure 12 and 13 for the Claude agent. The average trajectory success rate is 27% as measured by our Claude 3 Sonnet evaluator. The SFT checkpoint for WebArena is fine-tuned from the SFT checkpoint for WebVoyager.

E Additional Results on WebArena

For completeness, we have also provided additional experiment results of different models from Table 2 in the original task split of WebArena (Zhou et al., 2024). As shwon in the comparison results presented in Table 4, even SOTA proprietary VLM agents like Claude 3 Sonnet struggle with the tasks in WebArena with a success rate of only 14.6% with set-of-marks observations and chain-of-thought prompting. After performing SFT using the demonstrations generated by Claude 3 Sonnet, LLaVa-7B SFT can only achieve 1.4% and 5.8% success rate on PostMill and OneStopMarket. By manually inspecting the roll-out trajectories generated by LLaVa-SFT, we found that around half of the successful trajectories on those two websites are false positives from the WebArena evaluator. In these trajectories, the agent simply guessed the answer to be "no" or "N/A" where the ground truth happens to be that the task is not executable. As a result, the actual success rate on those two websites

Zero-Shot VLM Agent Prompt for WebVoyager (1/2)

Imagine you are a robot browsing the web, just like humans. Now you need to complete a task. In each iteration, you will receive an Observation that includes a screenshot of a webpage and some texts. This screenshot will feature Numerical Labels placed in the TOP LEFT corner of each Web Element. Carefully analyze the visual information to identify the Numerical Label corresponding to the Web Element that requires interaction, then follow the guidelines and choose one of the following actions:
1. Click a Web Element.
2. Delete existing content in a textbox and then type content.
3. Scroll up or down. Multiple scrolls are allowed to browse the webpage. Pay attention!! The default scroll is the whole window. If the scroll widget is located in a certain area of the webpage, then you have to specify a Web Element in that area. I would hover the mouse there and then scroll.
4. Wait. Typically used to wait for unfinished webpage processes, with a duration of 5 seconds.
5. Go back, returning to the previous webpage.
6. Google, directly jump to the Google search page. When you can't find information in some websites, try starting over with Google.
7. Answer. This action should only be chosen when all questions in the task have been solved.

Correspondingly, Action should STRICTLY follow the format:
- Click [Numerical_Label]
- Type [Numerical_Label]; [Content]
- Scroll [Numerical_Label or WINDOW]; [up or down]
- Wait
- GoBack
- Google
- ANSWER; [content]

Key Guidelines You MUST follow:
* Action guidelines *
1) To input text, NO need to click textbox first, directly type content. After typing, the system automatically hits 'ENTER' key. Sometimes you should click the search button to apply search filters. Try to use simple language when searching.
2) You must Distinguish between textbox and search button, don't type content into the button! If no textbox is found, you may need to click the search button first before the textbox is displayed.
3) Execute only one action per iteration.
4) STRICTLY Avoid repeating the same action if the webpage remains unchanged. You may have selected the wrong web element or numerical label. Continuous use of the Wait is also NOT allowed.
5) When a complex Task involves multiple questions or steps, select "ANSWER" only at the very end, after addressing all of these questions (steps). Flexibly combine your own abilities with the information in the web page. Double check the formatting requirements in the task when ANSWER.
* Web Browsing Guidelines *
1) Don't interact with useless web elements like Login, Sign-in, donation that appear in Webpages. Pay attention to Key Web Elements like search textbox and menu.
2) Vsit video websites like YouTube is allowed BUT you can't play videos. Clicking to download PDF is allowed and will be analyzed by the Assistant API.
3) Focus on the numerical labels in the TOP LEFT corner of each rectangle (element). Ensure you don't mix them up with other numbers (e.g. Calendar) on the page.
4) Focus on the date in task, you must look for results that match the date. It may be necessary to find the correct year, month and day at calendar.
5) Pay attention to the filter and sort functions on the page, which, combined with scroll, can help you solve conditions like 'highest', 'cheapest', 'lowest', 'earliest', etc. Try your best to find the answer that best fits the task.

Your reply should strictly follow the format:
Thought: {Your brief thoughts (briefly summarize the info that will help ANSWER)}
Action: {One Action format you choose}

Then the User will provide:
Observation: {A labeled screenshot Given by User}

Figure 11: The prompt used for all zero-shot VLM agents for WebVoyager websites, including Claude-Sonnet-3, Claude-Sonnet-3.4, Qwen2-VL, InternVL-2.5-XComposer, LLaVa-1.6-7B, and LLaVa-1.6-34B.

Zero-Shot VLM Agent Prompt for WebArena (2/2)

Imagine you are a robot browsing the web, just like humans. Now you need to complete a task. In each iteration, you will receive an Observation that includes a screenshot of a webpage, some texts and the accessibility tree of the webpage. This screenshot will feature Numerical Labels placed in the TOP LEFT corner of each Web Element. The accessbility tree contains information about the web elements and their properties. The numrical labels in the screenshot correspond to the web elements in the accessibility tree.

Carefully analyze the visual information to identify the Numerical Label corresponding to the Web Element that requires interaction, then follow the guidelines and choose one of the following actions:

1. Click a Web Element.
2. Delete existing content in a textbox and then type content.
3. Scroll up or down. Multiple scrolls are allowed to browse the webpage. Pay attention!! The default scroll is the whole window. If the scroll widget is located in a certain area of the webpage, then you have to specify a Web Element in that area. I would hover the mouse there and then scroll.
4. Wait. Typically used to wait for unfinished webpage processes, with a duration of 5 seconds.
5. Go back, returning to the previous webpage.
6. Answer. This action should only be chosen when all questions in the task have been solved.

Correspondingly, Action should STRICTLY follow the format:

- Click [Numerical_Label]
- Type [Numerical_Label]; [Content]
- Scroll [Numerical_Label or WINDOW]; [up or down]
- Wait
- GoBack
- ANSWER; [content]

Key Guidelines You MUST follow:

* Action guidelines *

1) To input text, NO need to click textbox first, directly type content. After typing, the system automatically hits ‘ENTER‘ key. Sometimes you should click the search button to apply search filters. Try to use simple language when searching.
2) You must Distinguish between textbox and search button, don’t type content into the button! If no textbox is found, you may need to click the search button first before the textbox is displayed.
3) Execute only one action per iteration.
4) STRICTLY Avoid repeating the same action if the webpage remains unchanged. You may have selected the wrong web element or numerical label. Continuous use of the Wait is also NOT allowed.
5) When a complex Task involves multiple questions or steps, select ”ANSWER” only at the very end, after addressing all of these questions (steps). Flexibly combine your own abilities with the information in the web page. Double check the formatting requirements in the task when ANSWER.
6) If you can’t find the answer using the given website because there is no such information on the website after some attempts, you should report ”N/A” as the answer to represent that the task is impossible to solve with the given website. You may have 15 steps to try to solve the task.
7) Only provide answer based on the information from the image, make sure the answer is consistent with the image, don’t hallucinate any information that is not based on image.

* Web Browsing Guidelines *

1) Focus on the numerical labels in the TOP LEFT corner of each rectangle (element). Ensure you don’t mix them up with other numbers (e.g. Calendar) on the page.
2) Pay attention to the filter and sort functions on the page, which, combined with scroll, can help you solve conditions like ’highest’, ’cheapest’, ’lowest’, ’earliest’, etc. Try your best to find the answer that best fits the task.

Figure 12: The prompt used for all zero-shot VLM agents for WebArena websites, including Claude-Sonnet-3, Claude-Sonnet-3.4, Qwen2-VL, InternVL-2.5-XComposer, LLaVa-1.6-7B, and LLaVa-1.6-34B. To be continued in Figure 13.

Zero-Shot VLM Agent Prompt for WebArena

* OpenStreetMap Usage Guidelines *

1) When you need to search the address of a location, you can just type the location in the 'search' bar. You don't need to use the directions button to get the address. The directions button is only used when you need to find the distance/walk/drive time between two locations.

2) When you are trying to search for a location, you may get no results. This is because the OpenStreetMap does not support approximate search. You may try to search some alternative keywords or try to find the location by yourself. Note that openstreet map does not support search phrase like 'Cafe near CMU", you should try to find it by yourself.

3) When you need to find the distance/walk/drive time between two locations, you should FIRST CLICK ON THE DIRECTIONS BUTTON (drawn as two arrows), to the right of the 'Go' Button and usually labeled as [10] or [11]. AND ONLY INPUTTING THE TWO LOCATIONS AFTER CLICKING ON THE DIRECTIONS BUTTON WHEN THE DIRECTIONS SEARCH BARS ARE SHOWN.

4) When you are trying to type some locations in the directions search bar, sometimes you may receive an alert of 'couldn't locate' followed by the location you typed. This means the location you typed is not found in the map. Do not immeadiately try something else. You need to quit the direction and find the precise name of this location by searching it in the map first.

5) When you search the walk/drive/bike time, make sure that you are USING THE RIGHT MODE OF TRANSPORTATION. The default mode is usually set to 'Drive'.

6) When you need to get the DD of some location, you need to click the location shown in the search result in the left part of the screen. The DD will be shown then starting with 'Location:'.

7) When you need to answer the zip code of some location, you should directly answer the 5-digit zip code. The answer shoule be "15232" instead of "The zip code of the location is 15232". Note that the zipcode will be displayed in the search result, you don't need to click the location to the information page to find the zip code.

8) When you need to answer the phone of some location, please omit the part of the country code. The answer should be "4122683259" instead of "+1 412 268 3259".

* Reddit Usage Guidelines *

1) You are already in the reddit website, though you may not see the 'reddit' in any part of the screenshot. You do not need to further navigate to the reddit website.

2) When you want to find a subreddit, you need to first navigate to Forums to see the list of subreddits. Under forums, you will see only a subset of subreddits. To get the full list of subreddits, you need to navigate to the Alphabetical option. To know you can see the full list of subreddits, you will see 'All Forums' in the observation. Often you will not find a focused subreddit that exactly matches your query. In that case, go ahead with the closest relevant subreddit. To know that you have reached a subreddit successfully, you will see '/f/subreddit_name' in the observation.

3) When you want to post forum in reddit, remember to fill up all the content, then click the button 'Create forum'. The button maybe located below out of the screenshot, you need to scroll down to find it.

4) When you want to ask or post something in a subreddit, you need to first find that subreddit and then finish the work. 5) forums and subreddits are the same thing.

Your reply should strictly follow the format:
Thought: Your brief thoughts (briefly summarize the info that will help ANSWER)
Action: One Action format you choose

Then the User will provide:
Observation: A labeled screenshot Given by User
Remember only execute one action in each step. For example, 'Action: Type [8]; CMU, Type [9] Pittsburgh' is not allowed. You should execute the action 'Type [8]; CMU' first, then 'Type [9] Pittsburgh' in the next step.

Remember to always make your answer simple and clear. For example, if you want to report the zip code of some location, always say "ANSWER; 06516" instead of "The zip code of the location is 06516".

Figure 13: The prompt used for all zero-shot VLM agents for WebArena websites, including Claude-Sonnet-3, Claude-Sonnet-3.4, Qwen2-VL, InternVL-2.5-XComposer, LLaVa-1.6-7B, and LLaVa-1.6-34B. Continued from Figure 12.

is lower than 2%, leaving very sparse reward signals for RL to make meaningful improvements. We therefore rewrote the tasks on PostMill and OneStopMarket to be easier and report the performances of PAE in Table 2.

		OpenStreetMap	PostMill	OneStopMarket	Average
Proprietary	Claude 3 Sonnet	24.3	10.6	11.2	14.6
Open-source	Qwen2VL-7B	0.7	0.0	1.3	0.7
	InternVL2.5-8B	2.6	0.2	3.3	2.3
	LLaVa-7B	0.0	0.0	0.0	0.0
Ours	LLaVa-7B SFT	15.2	1.4	5.8	7.2

Table 4: Success rate comparisons across different domains from WebArena. Success and failure are detected with ground-truth verification functions. All tasks from OpenStreetMap are kept unchanged from WebArena task splits.

F LIMITATIONS

Despite the progress of PAE for open-source VLM agents, there are still some limitations due to practical constraints. First of all, due to the limitations in fundamental capabilities of open-source base VLM models, our models trained with PAE are still inferior to state-of-the-art proprietary models in realistic web navigations, where advanced reasoning and planning capabilities are required. Moreover, because of the hallucination issues of open-source VLMs, we found them unreliable to serve as the autonomous evaluators and had to rely on advanced proprietary VLMs for judging the success and providing rewards. Finally, because of the dynamic nature of the real websites that we are using, some of our results may not be produced exactly, although a significant improvement from PAE should still be observed.

G HYPERPARAMETERS

We include the hyperparameters that we have used in Table 5. As shown in the table, the only hyperparameters that PAE have on top of standard supervised fine-tuning are number of trajectories to collect in each global iteration in Algorithm 1, number of proposed tasks from the task proposer before RL training, and the number of seen screenshots for the evaluator. In our experiments, we found that PAE is relatively not sensitive to the choices of these hyperparameters, showing the robustness of PAE .

H MORE QUALITATIVE EXAMPLES

In this section, we present additional qualitative examples of agent trajectories while performing tasks to further demonstrate the effectiveness of our PAE . We will also release the full dataset for further analysis.

Full trajectories of examples in Section 6. Here, we provide the complete trajectories for the examples discussed in the qualitative comparisons in Section 6, as shown in Figures 16–19. We detail the agent's thoughts and actions at each time step throughout the entire trajectory.

Some representative successful trajectories. We also showcase representative successful trajectories generated by the LLaVa-7B PAE model to highlight the strengths of our method. In Figure 20, the task is "Show the most played games on Steam, and tell me the number of players currently in-game." In Figure 21, the task is "Find out the starting price for the most recent model of the iMac on the Apple website." In Figure 22, the task is "Look up the use of modal verbs in the grammar section for expressing possibility (e.g., 'might', 'could', 'may') and find examples of their usage in sentences on the Cambridge Dictionary." Finally, in Figure 23, the task is "Search for plumbers available now but not open 24 hours in Orlando, FL."

Table 5: Hyperparameters for All Experiments

Environment	Hyperparameter	Considered	Chosen
WebVoyager	learning rate	{2e-5, 5e-5, 2e-4}	2e-5
	rollout trajectories	{512, 1024, 2048, 4096}	4096
	rollout temperature	{0.4, 1.0, 2.0}	1.0
	maximum gradient norm	{0.01}	0.01
	actor updates epochs per iteration	{1, 2, 4, 8, 20}	4
	batch size	{8}	8
	gradient accumulation size	{16, 32}	32
	number of proposed tasks	{10000, 50000, 100000}	100000
	number of seen screenshots for evaluator	{1, 3}	3
WebArena Easy	learning rate	{2e-5, 5e-5, 2e-4}	2e-5
	rollout trajectories	{512, 1024, 2048, 4096}	2048
	rollout temperature	{0.4, 1.0, 2.0}	1.0
	maximum gradient norm	{0.01}	0.01
	actor updates epochs per iteration	{1, 2, 4, 8, 20}	2
	batch size	{8}	8
	gradient accumulation size	{16, 32}	32
	number of proposed tasks	{10000, 30000, 100000}	30000
	number of seen screenshots for evaluator	{1, 3}	3

Table 6: Hyperparameters for PAE for WebVoyager and WebArena Easy experiments.

Detailed explanations of the error type definitions. To clarify the precise definition of the different error categories used in Section 6, we provide more comprehensive explanations with example trajectories:

(1) Low-level skill missing errors refer to cases where the agent has a reasonable plan to solve the problem but fails to execute precise actions on the website, such as not knowing which button to click to reach the desired page. We classify trajectories where the agent seems to follow a reasonable plan but struggles with specific operations into this category. For example, in Figure 24, the task is "Find the Easy Vegetarian Spinach Lasagna recipe on Allrecipes and tell me what the latest review says." The agent attempts to search for the desired item but fails to click the correct button to reach the detailed page in the search results.

(2) High-level planning or reasoning errors occur when the agent fails to generate a complete plan or cannot reason correctly with the website's screenshots to solve the task. Trajectories where the agent cannot devise a plan for complex tasks or misinterprets the screenshot's content are categorized as such. For instance, in Figure 25, the task is "Give 12 lbs of 4-cyanoindole, converted to molar and indicate the percentage of C, H, N." The agent should first search on Google about the chemical definition of 4-cyanoindole, then use WolframAlpha to calculate the result. However, the agent fails to get the precise definition of 4-cyanoindole, and doesn't know how to solve the task.

(3) Visual hallucinations refer to instances where the agent generates fabricated responses not supported by the screenshot. The agent might, for example, claim to have found a requested product while still on the Google homepage or provide an incorrect answer even when on the correct page. In Figure 26, the task is "Find out the trade-in value for an iPhone 13 Pro Max in good condition on the Apple website". The agent claims with a very detailed answer but actually it never access any page related to the trade-in on the website.

(4) Timeouts occur when the agent is on the right track to solving the task but cannot complete it within the maximum number of steps. This error indicates that the agent did nothing wrong but was constrained by the environment's step limits. For example, in Figure 27, the task is "Go to the Plus section of Cambridge Dictionary, find Image quizzes, and complete an easy quiz about Animals. Tell me your final score." The agent reaches the maximum time step limit (10) while attempting to finish the quiz.

(5) Technical issues are not caused by the agent but by environmental problems, such as websites being down or connection failures. In Figure 28, the ChromeDriver crashes after a valid operation.

(6) Others include less frequent error types, such as when the task itself is impossible to complete.

Out-of-distribution Websites of WebVoyager for SFT (1/2)
Allrecipes:
Simply Recipes: https://www.simplyrecipes.com
Food Network: https://www.foodnetwork.com
Taste of Home: https://www.tasteofhome.com
Yummly: https://www.yummly.com
Food.com: https://www.food.com

Amazon:
eBay: https://www.ebay.com
Walmart: https://www.walmart.com
Target: https://www.target.com
Best Buy: https://www.bestbuy.com
Alibaba: https://www.alibaba.com

Apple:
Samsung: https://www.samsung.com
Microsoft: https://www.microsoft.com
Sony: https://www.sony.com
Google Store: https://store.google.com
Dell: https://www.dell.com

ArXiv:
SSRN: https://www.ssrn.com
ResearchGate: https://www.researchgate.net
bioRxiv: https://www.biorxiv.org
IEEE Xplore: https://ieeexplore.ieee.org
PubMed: https://pubmed.ncbi.nlm.nih.gov

GitHub:
GitLab: https://about.gitlab.com
Bitbucket: https://bitbucket.org
SourceForge: https://sourceforge.net
Codebase: https://www.codebasehq.com
Gitea: https://gitea.io

ESPN:
CBS Sports: https://www.cbssports.com
Fox Sports: https://www.foxsports.com
NBC Sports: https://www.nbcsports.com
Bleacher Report: https://www.bleacherreport.com
Sky Sports: https://www.skysports.com

Coursera:
edX: https://www.edx.org
Udacity: https://www.udacity.com
Udemy: https://www.udemy.com
FutureLearn: https://www.futurelearn.com
Khan Academy: https://www.khanacademy.org

Figure 14: A list of 85 websites that we used to collect demonstration trajectories with Claude 3 Sonnet. In total 11220 trajectories were collected with different tasks. These websites were also used for testing the zeroshot generalization of PAE to out-of-distribution websites in Section 5. List continued in Figure 15.

Out-of-distribution Websites of WebVoyager for SFT (2/2)
Cambridge Dictionary:
Merriam-Webster: https://www.merriam-webster.com
Dictionary.com: https://www.dictionary.com
Oxford Learner's Dictionaries: https://www.oxfordlearnersdictionaries.com
Collins English Dictionary: https://www.collinsdictionary.com
YourDictionary: https://www.yourdictionary.com

BBC News:
CNN: https://www.cnn.com
Al Jazeera: https://www.aljazeera.com
Reuters: https://www.reuters.com
The Guardian: https://www.theguardian.com
NBC News: https://www.nbcnews.com

Google Maps:
Apple Maps: https://maps.apple.com
Bing Maps: https://www.bing.com/maps
MapQuest: https://www.mapquest.com
Waze: https://www.waze.com
Here WeGo: https://wego.here.com

Google Search:
Bing: https://www.bing.com
Yahoo Search: https://search.yahoo.com
DuckDuckGo: https://duckduckgo.com
Baidu: https://www.baidu.com
Yandex: https://yandex.com

Hugging Face:
OpenAI: https://openai.com
TensorFlow: https://www.tensorflow.org
PyTorch: https://pytorch.org
Kaggle: https://www.kaggle.com
SpaCy: https://spacy.io

Wolfram Alpha:
Google Scholar: https://scholar.google.com
Mathway: https://www.mathway.com
Symbolab: https://www.symbolab.com
Microsoft Math Solver: https://mathsolver.microsoft.com
Desmos: https://www.desmos.com

Figure 15: A list of 85 websites that we used to collect demonstration trajectories with Claude 3 Sonnet. In total 11220 trajectories were collected with different tasks. These websites were also used for testing the zeroshot generalization of PAE to out-of-distribution websites in Section **??**. List continued from Figure 15.

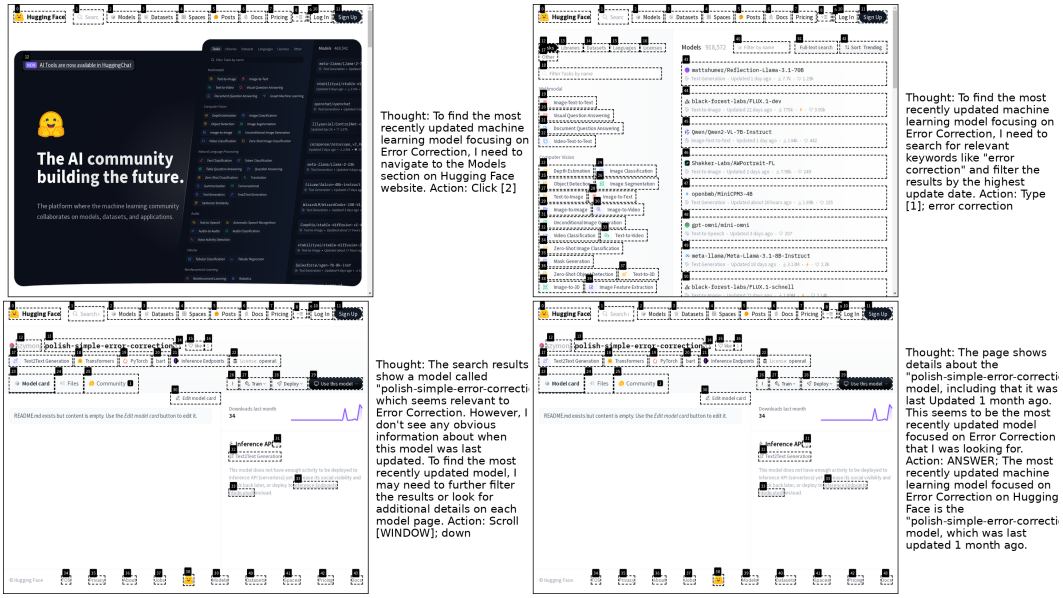


Figure 16: Full trajectories of success trajectory 1 in Figure 6 with task 'Find the most recently updated machine learning model on Huggingface which focuses on Error Correction' executed by model LLaVa-7B PAE.

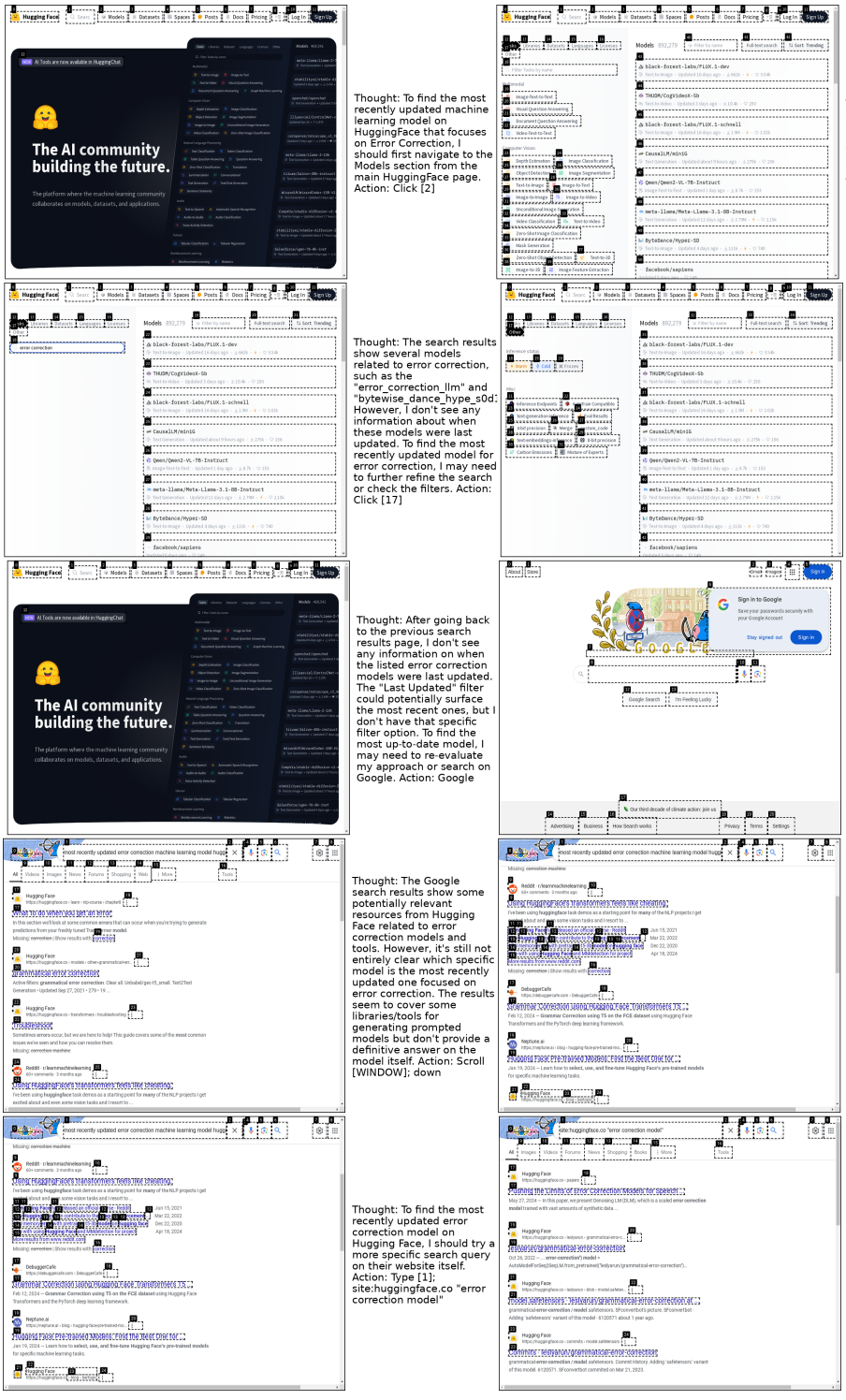
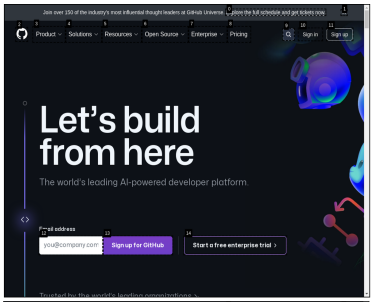
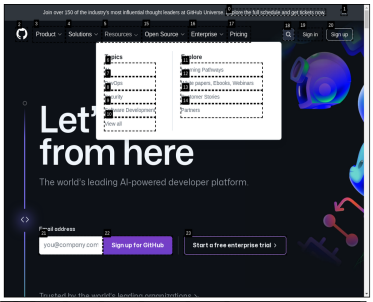


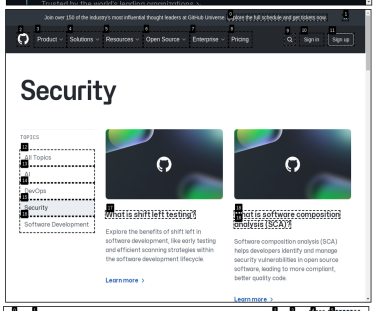
Figure 17: Full trajectories of fail trajectory 1 in Figure 6 with task 'Find the most recently updated machine learning model on Huggingface which focuses on Error Correction' executed by model LLaVa-7B SFT.



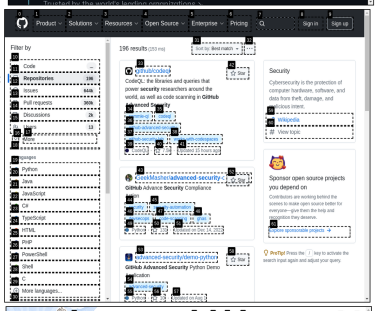
Thought: The task is to find the Security topic in GitHub Resources and answer the role of GitHub Advanced Security. To begin, I should navigate to the Resources section on the GitHub website. Action: Click [5]



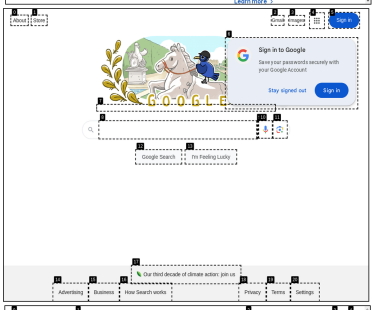
Thought: The Resources section has been expanded, revealing subsections like "AI", "DevOps", and "Security". To find information on GitHub Advanced Security, I should navigate to the "Security" subsection. Action: Click [8]



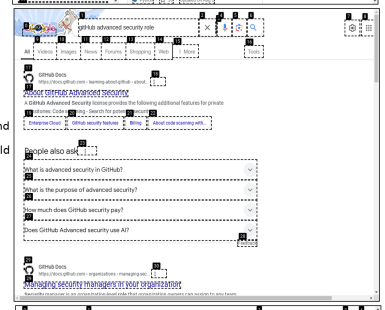
Thought: The "Security" section has two subsections - "What is shift left testing?" and "What is software composition analysis (SCA)?". These seem to provide an overview and potential benefits of shift left testing and SCA in the software development lifecycle. To learn more about GitHub Advanced Security specifically, I may need to search or navigate further within the Resources section. Action: Type [9]; github advanced security



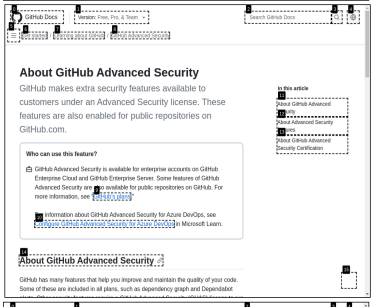
Thought: The search for "github advanced security" did not return any results. This implies that the current Resources section may not contain information specifically about GitHub Advanced Security. To find details on this topic, I may need to try a different approach like searching on Google or navigating to a different section of GitHub related to security and compliance. Action: Google



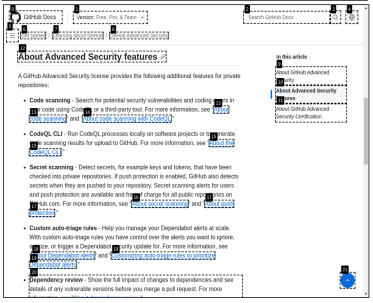
Thought: I am now on the Google search page. To find information on GitHub Advanced Security, I should enter a relevant search query in the textbox. Action: Type [8]; github advanced security role



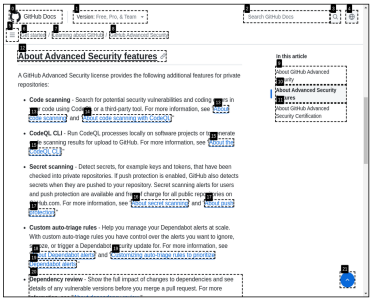
Thought: The search results provide some relevant information about GitHub Advanced Security. The top result seems to be the official documentation from GitHub on this topic. Reading through this page should give me a good understanding of the role and features included in GitHub Advanced Security. Action: Click [17]



Thought: The current page appears to be the GitHub Docs documentation on Advanced Security. It provides an overview and links to more specific information about the features included. To fully answer the role of GitHub Advanced Security, I should review the related documentation sections. Action: Click [12]

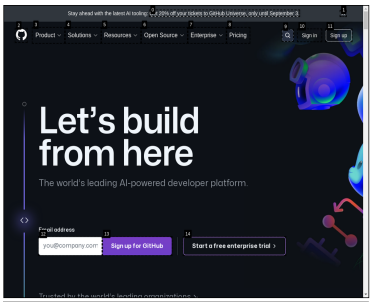


Thought: The "About Advanced Security features" section provides an overview of the vulnerability scanning, detect secrets, and secure coding features included in GitHub Advanced Security. This appears to directly address the role and capabilities of the advanced security offering. Action: Click [12]

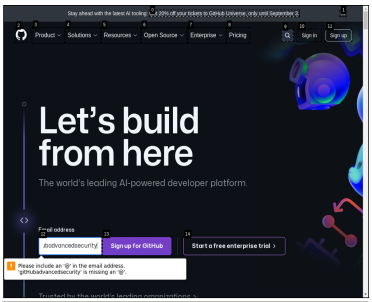


Thought: The "About Advanced Security features" section provides an overview of the key advanced security features and capabilities, including vulnerability scanning, secured secrets detection, and secure coding rules. This directly addresses the role and capabilities of the GitHub Advanced Security offering. Action: ANSWER; GitHub Advanced Security provides enhanced security features and capabilities beyond the core GitHub platform. Key features include vulnerability scanning for open-source dependencies, secured secrets detection across projects, and secure coding rules to protect against...

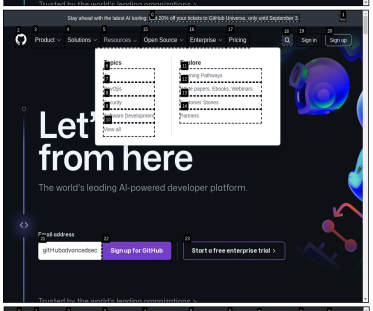
Figure 18: Full trajectories of success trajectory 2 in Figure 6 with task 'Find the Security topic in GitHub Resources and answer the role of GitHub Advanced Security' executed by model LLaVa-7B PAE.



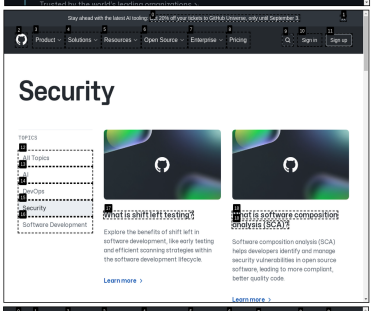
Thought: The main goal is to find the Security topic in GitHub Resources and understand the role of GitHub Advanced Security. The current webpage is the GitHub homepage, which does not provide direct information about their security offerings. To navigate to the relevant information, I should likely use the search functionality or explore the Resources section further. Action: Type [12]; gitHub advanced security



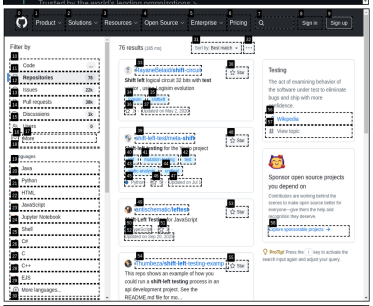
Thought: The search for "gitHub advanced security" did not yield any relevant results on the homepage. The current page suggests creating an account or signing up for GitHub, which is likely for individual developers rather than understanding their advanced security offerings for organizations. To find this information, I should explore the Resources section further. Action: Click [5]



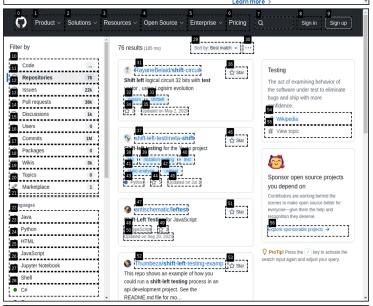
Thought: The Resources section now shows a "Security" link under the main topic areas. This seems promising for finding information about GitHub's advanced security offerings for organizations. Clicking on the "Security" link should take me to a page with more details. Action: Click [8]



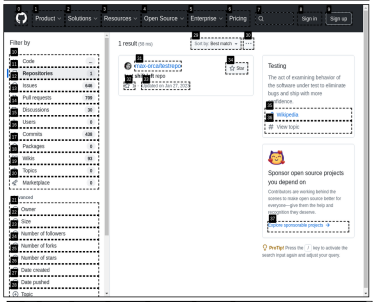
Thought: This page appears to be an overview of GitHub's advanced security, testing, and development offerings. There are two main sections highlighting the benefits and capabilities. The "What is shift left testing?" section likely provides an introduction to GitHub's advanced security testing features. Exploring those sections in more detail should help answer the role of GitHub Advanced Security. Action: Type [17]; What is shift left testing?



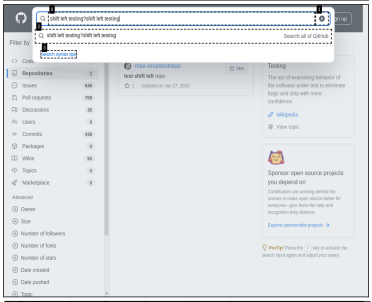
Thought: You're right, the [17] link does not seem to be a textbox for typing a query, but likely just more filter options. To better understand "shift left testing" and its relation to GitHub Advanced Security, I should first click on that link to navigate to the relevant information pages. Action: Click [17]



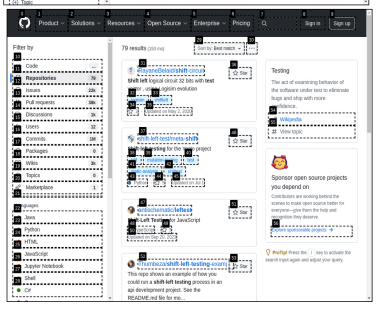
Thought: The search bar at the top is labeled [7], which seems to be the correct web element to type in "shift left testing" to find relevant information on GitHub about this software testing concept and its relation to Advanced Security. Action: Type [7]; shift left testing



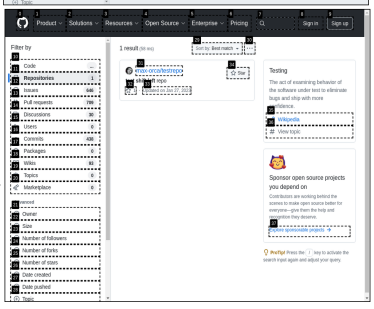
Thought: You're correct, the element labeled [7] is a button, not a textbox. To search for "shift left testing" on GitHub, I will first need to click the [7] button to activate the search bar, then type the query into the textbox that appears. Action: Click [7]



Thought: After clicking the [7] button, a search textbox has appeared labeled [0]. I can now type the query "shift left testing" into this textbox to search for relevant content on GitHub. Action: Type [0]; shift left testing

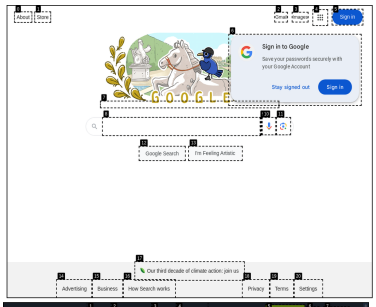


Thought: The search for "shift left testing" did not return any relevant results on GitHub, as indicated by the "No results" message. To find information related to this topic, I may need to try a different search query or look in other areas of GitHub. Action: GoBack

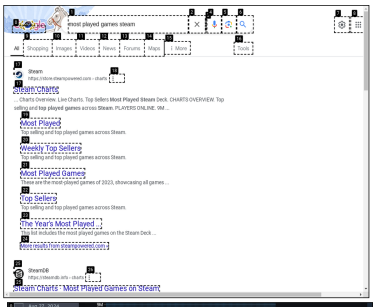


Thought: After going back to the previous GitHub search page, I see that the "Product" button [1] is now focused. Clicking this button previously took me to a page with an advertisement, which was not helpful for finding information about GitHub's security features. Action: Click [2]

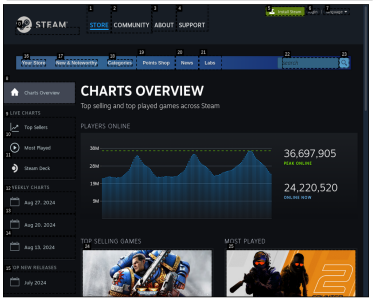
Figure 19: Full trajectories of fail trajectory 2 in Figure 6 with task 'Find the Security topic in GitHub Resources and answer the role of GitHub Advanced Security' executed by model LLaVa-7B SFT.



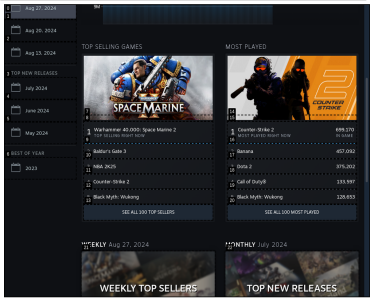
Thought: To find the most played games on Steam and the number of players in In game at this time, I need to search for "most played games steam" and "number of players in In game at this time" on Google. Action: Type [8]: most played games steam



Thought: The search results show a website called Steam Charts that provides information on the most played games on Steam, including top sold games and most played games across Steam. This seems promising to find the answer. Action: Click [17]

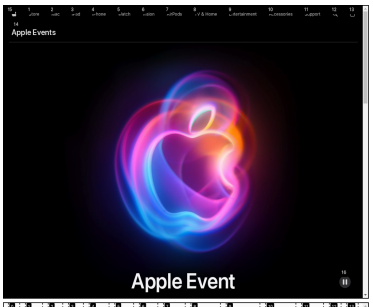


Thought: The Charts Overview page provides a line graph showing the number of players (in-game at a specific time) for the top 10 most played games on Steam over the last month. The graph shows the peak online hours for each day. To get more details on the specific games and their player counts, I may need to explore further options like selecting different time periods orcategories. Action: Scroll [WINDOW]; down



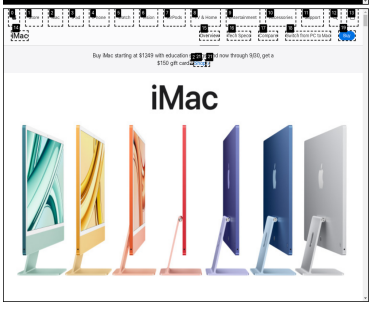
Thought: The "Most played" section shows the top 5 most played games on Steam right now, including their names, player counts in-game, and the percentage change compared to the previous month. This directly answers the task of showing the most played games on Steam and their current player counts. Action: ANSWER; The top 5 most played games on Steam right now are: 1. Counter-Strike 2 - 699,170 players (69% increase from last month) 2. Black Myth: Wukong - 128,653 players (n/a) 3. Dota 2 - 375,202 players (10% decrease) 4. Team Fortress 2 - 133,597 players (2% increase) 5....

Figure 20: Extra full trajectories of successful trajectory 1 with task 'Show most played games in Steam. And tell me the number of players in In game at this time' executed by model LLaVa-7B PAE.



Thought: The current webpage is the Apple homepage, which does not contain information about the starting price for the most recent iMac model. To find this, I will likely need to navigate to the Mac section of the website. Action: Click [2]

Thought: The "Mac" section now shows different Mac models, including iMac. However, the starting price is not explicitly mentioned here. To find this, I may need to click on the iMac link or compare the models. Action: Click [16]



Thought: The starting price for the most recent iMac model is now displayed - $1299 with education discount, getting through $930 of the $150 gift card. This answers the original question about the starting price for the latest iMac. Action: ANSWER; The starting price for the most recent iMac model on the Apple website is $1299 with education discount.

Figure 21: Extra full trajectories of successful trajectory 2 with task 'Find out the starting price for the most recent model of the iMac on the Apple website' executed by model LLaVa-7B PAE.

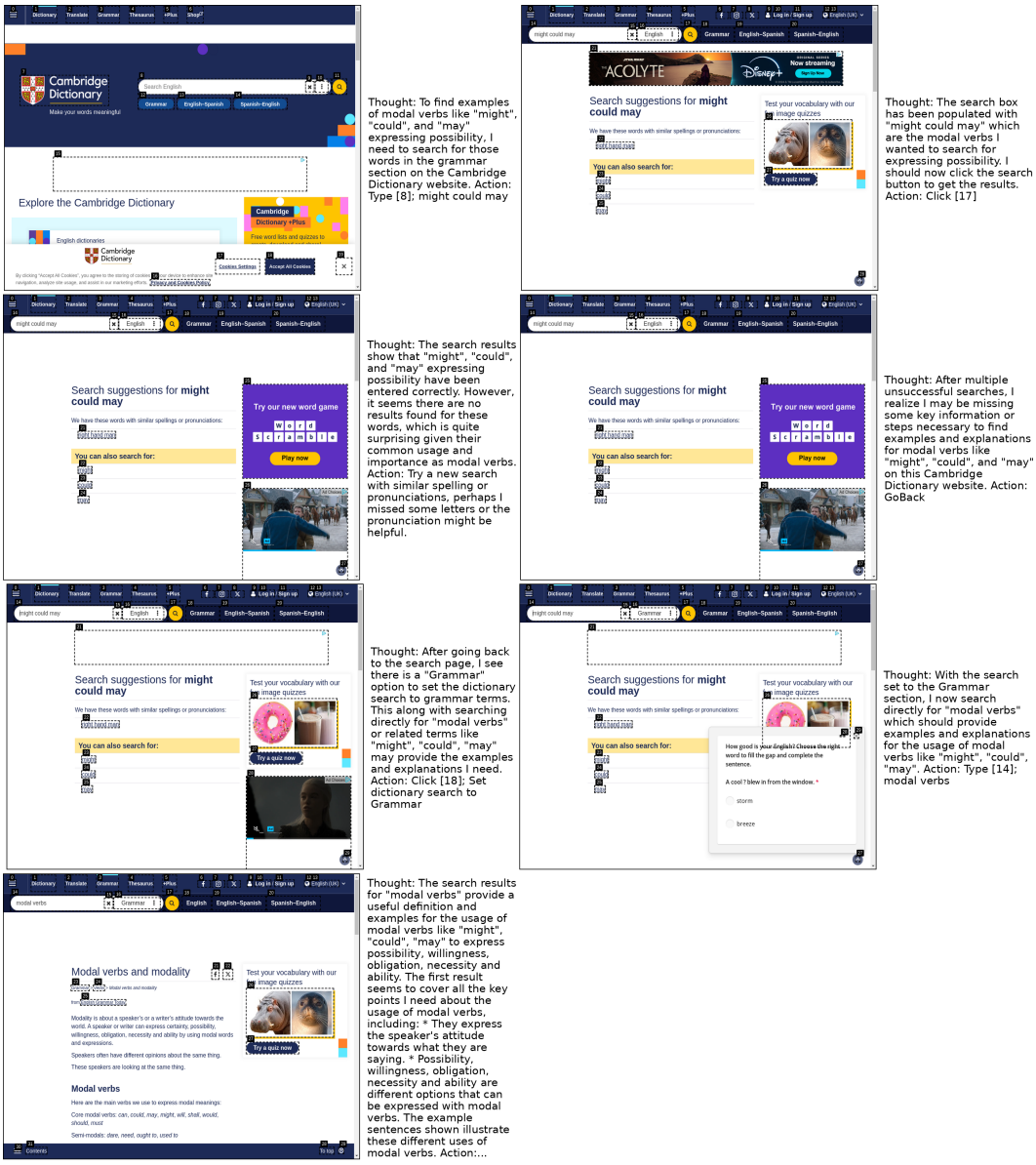


Figure 22: Extra full trajectories of successful trajectory 3 with task 'Look up the use of modal verbs in grammar section for expressing possibility (e.g., 'might', 'could', 'may') and find examples of their usage in sentences on the Cambridge Dictionary' executed by model LLaVa-7B PAE.

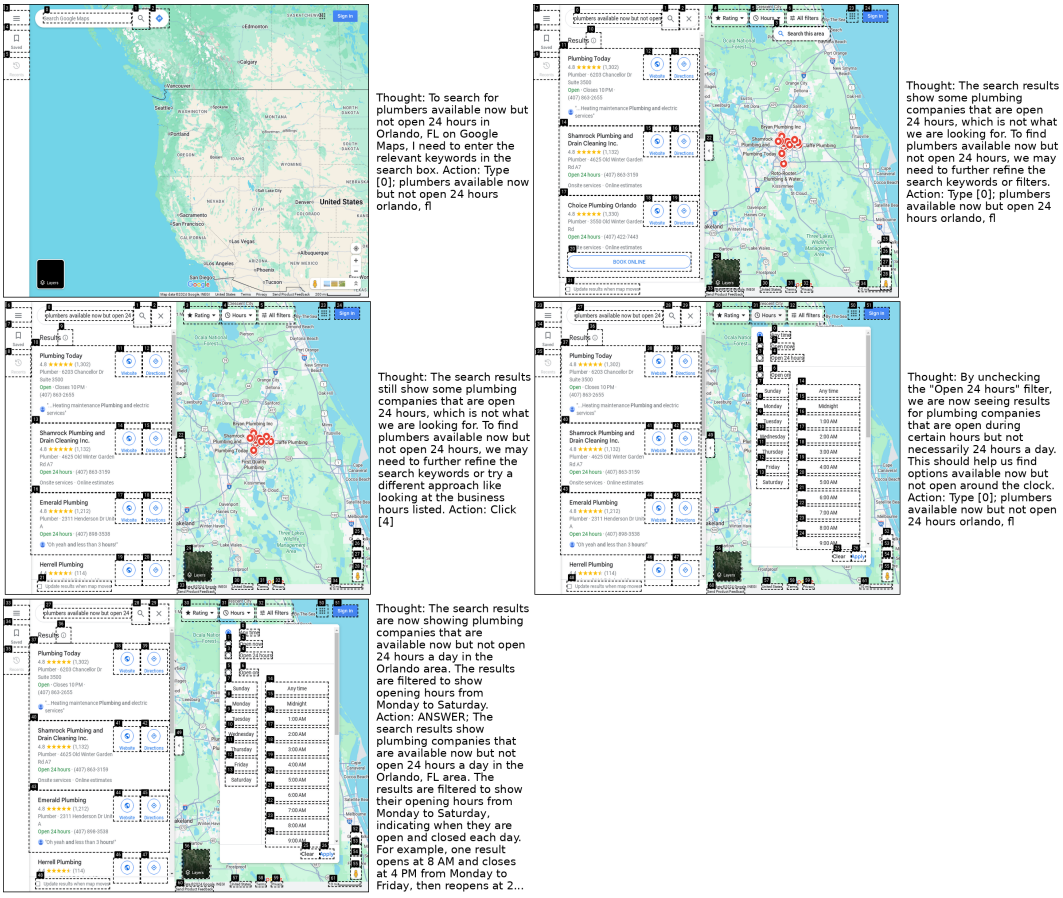


Figure 23: Extra full trajectories of successful trajectory 4 with task 'Search for plumbers available now but not open 24 hours in Orlando, FL' executed by model LLaVa-7B PAE.

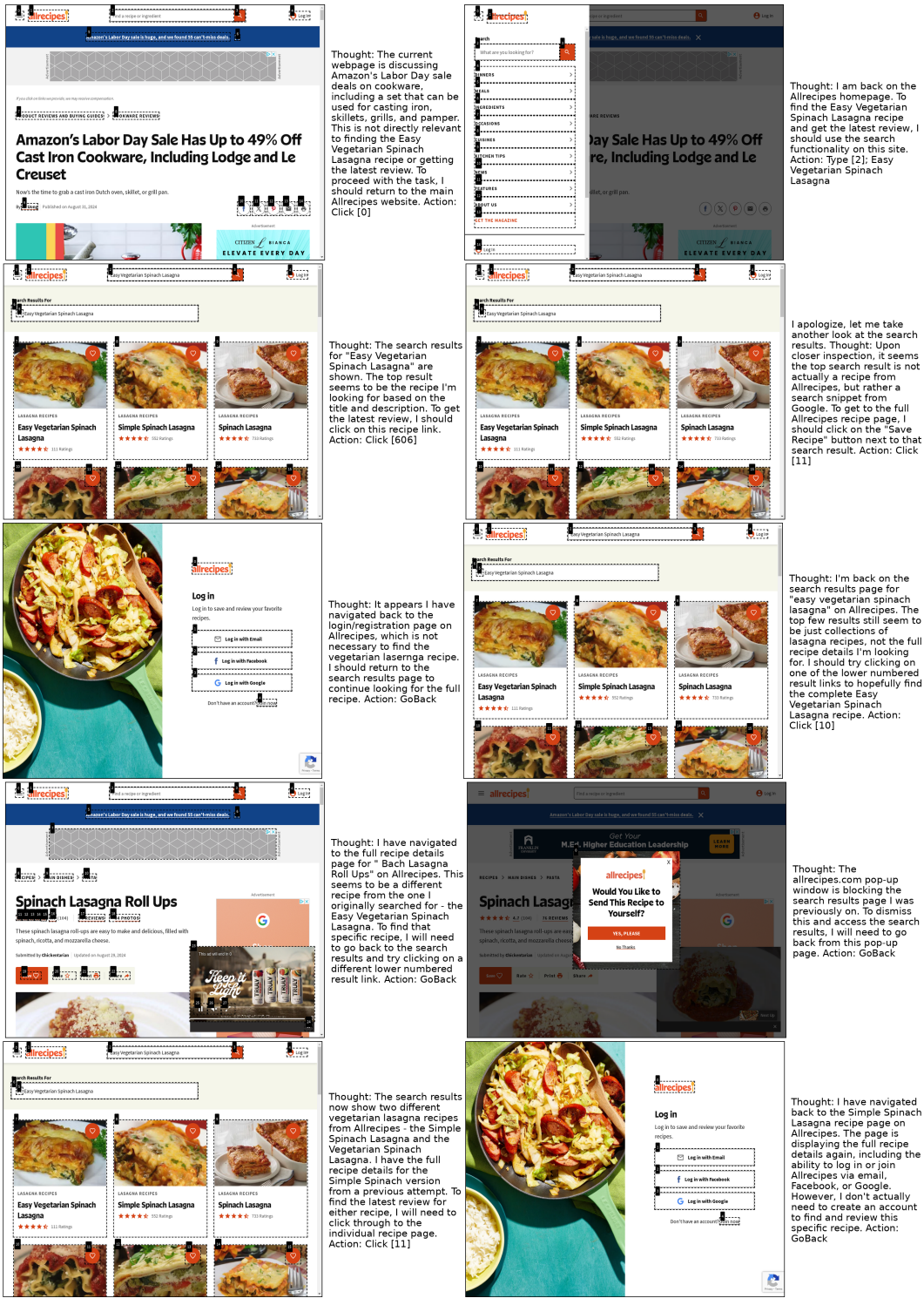


Figure 24: Extra full trajectories of fail trajectory 1, with error type **Low-level Operational error**, executed by model LLaVa-7B SFT. The task is 'Find the Easy Vegetarian Spinach Lasagna recipe on Allrecipes and tell me what the latest review says'.

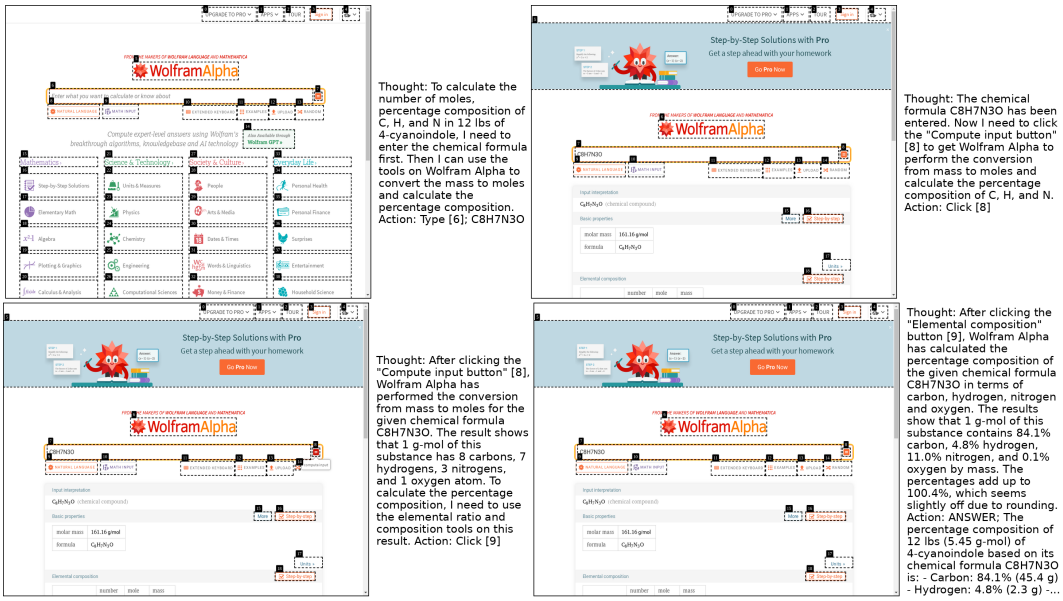


Figure 25: Extra full trajectories of fail trajectory 2, with error type **Planning or Reasoning error**, executed by model LLaVa-7B PAE. The task is 'Give 12 lbs of 4-cyanoindole, converted to molar and indicate the percentage of C, H, N'.

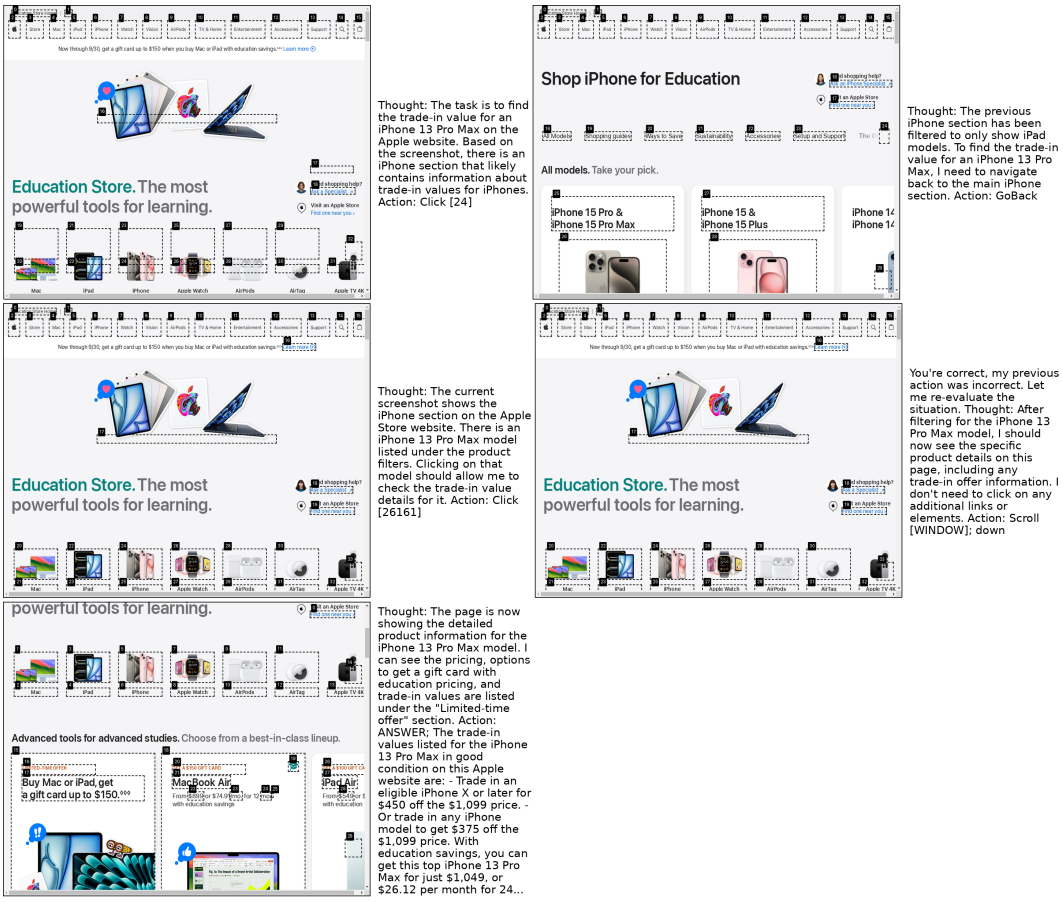


Figure 26: Extra full trajectories of fail trajectory 3, with error type **Visual Hallucination**, executed by model LLaVa-7B SFT. The task is 'Find out the trade-in value for an iPhone 13 Pro Max in good condition on the Apple website'.

Figure 27: Extra full trajectories of fail trajectory 4, with error type **Timeouts**, executed by model Claude 3.5 Sonnet. The task is 'Go to the Plus section of Cambridge Dictionary, find Image quizzes and do an easy quiz about Animals and tell me your final score'.

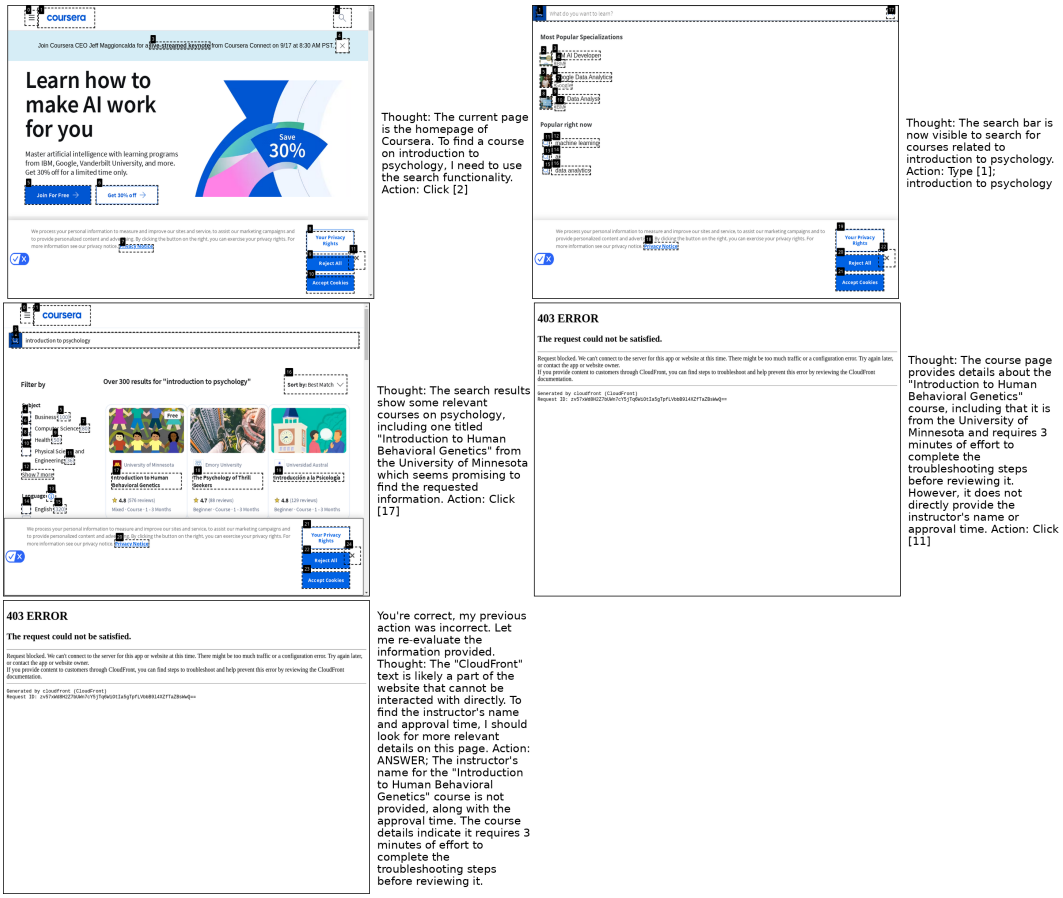


Figure 28: Extra full trajectories of fail trajectory 5, with error type **Technical issues**, executed by model LLaVa-7B PAE. The task is 'Identify a course on Coursera that provides an introduction to Psychology, list the instructor's name, the institution offering it, and how many hours it will approximately take to complete'.