WebArena-Pro: A Heterogeneous, Multimodal, Reproducible Benchmark for Web Agents

Imene Kerboua [*†1,2] **Fatemeh Pesaran Zadeh** [*†3] **Xing Han Lú** [†4,5] **Weijian Qi** [‡6] **Alexander Miller** [‡4] **Junyi Song** [‡4] **Yunjia Tian** [‡4] **Dongjin Kang** [‡3] **Seyeon Choi** [‡3] **Marzia Nouri** [4] **Ewen Gueguen** [4] **Matteo Boglioni** [5] **Fengyuan Liu** [4,5] **Zeyi Liao** [6] **Mengqi Yuan** [7] **Yue Li** [4] **Alexandre Lacoste** [8] **Alexandre Drouin** [8,5,9] **Spandana Gella** [8] **Huan Sun** [6] **Gunhee Kim** [3] **Siva Reddy** [4,5,10,8]

Abstract

Web agents powered by large language and vision-language models are increasingly applied to realistic browser work that spans heterogeneous applications, multimodal content, and stateful workflows. However, existing reproducible web-agent benchmarks cover only a small number of web applications drawn from a few software categories, and restrict modality to text and vision. Live benchmarks broaden site coverage but sacrifice reproducibility, since pages and data drift between runs. Moreover, existing benchmarks do not meaningfully evaluate whether agents can understand and use audio and video content embedded within web tasks. To address these gaps, we introduce WebArena-Pro, a benchmark comprising 300 tasks across 20 self-hosted web applications in six domain categories, spanning distinct interface conventions, workflows, and data models. Across the evaluated agents, the best performance is achieved by Gemini 3.1 Pro, which attains 37.0 % success under a 50-step budget, while open-source models' performance does not exceed 27.7% success. Among reproducible, human-curated web agent benchmarks, WebArena-Pro provides the broadest application coverage and the most comprehensive multimodal support to date. The benchmark treats audio and video as core observations alongside text and vision, with dedicated actions for extracting information from each. WebArena-Pro runs each task in isolation and supports reproducible, parallel evaluation. Tasks are authored through a dedicated annotator interface, filtered by LLM-assisted triage, and finally validated by humans before release.

1. Introduction

Web agents driven by large language and vision-language models have moved quickly from synthetic clicking tasks toward broader browser work, including drafting and triaging messages, updating records in business applications, extracting insight from dashboards, editing media, and coordinating across heterogeneous tools (Abuelsaad et al., 2024; Deng et al., 2023; Gur et al., 2023; He et al., 2024). A growing body of benchmarks has supported this progress, offering *self-hosted* web environments (Zhou et al., 2023; hattami et al., 2025), visually grounded tasks (Koh et al., 2024; Miyai et al., 2025), workplace-style scenarios (Drouin et al., 2024; Xu et al., 2025), computer-use settings beyond the browser (Xie et al., 2024), and increasingly large task pools (Aggarwal et al., 2026). As models improve and the scope of attempted tasks widens, the design of the evaluation itself becomes a central question.

Despite this progress, current self-hosted web-agent benchmarks have limitations relative to the conditions needed for realistic and diverse measurement, as summarized in Table 1. First, prior self-hosted benchmarks cover only a small number of web applications drawn from a few software categories, which limits the heterogeneity of interfaces, workflows, and data models that an agent encounters during evaluation. Second, modality coverage is limited to text and vision, even though many real browser workflows involve temporal media such as audio and video embedded in web pages. A parallel line of live benchmarks operates on real production websites (Deng et al., 2023; Xue et al., 2025; He et al., 2024), broadening site coverage beyond what self-hosted benchmarks offer. However, they sacrifice reproducibility, since pages, content, layouts, and policies drift

[*]Equal contribution [†]Lead contributors [‡]Core contributors [1]INSA Lyon, LIRIS [2]Esker [3]Seoul National University [4]McGill University [5]Mila – Quebec AI Institute [6]The Ohio State University [7]University of Hong Kong [8]ServiceNow Research [9]Université Laval [10]Canada CIFAR AI Chair. Correspondence to: Siva Reddy <siva.reddy@mila.quebec>.

Published at the Second Workshop on Agents in the Wild: Safety, Security, and Beyond (AIWILD) at ICML 2026. Copyright 2026 by the author(s).

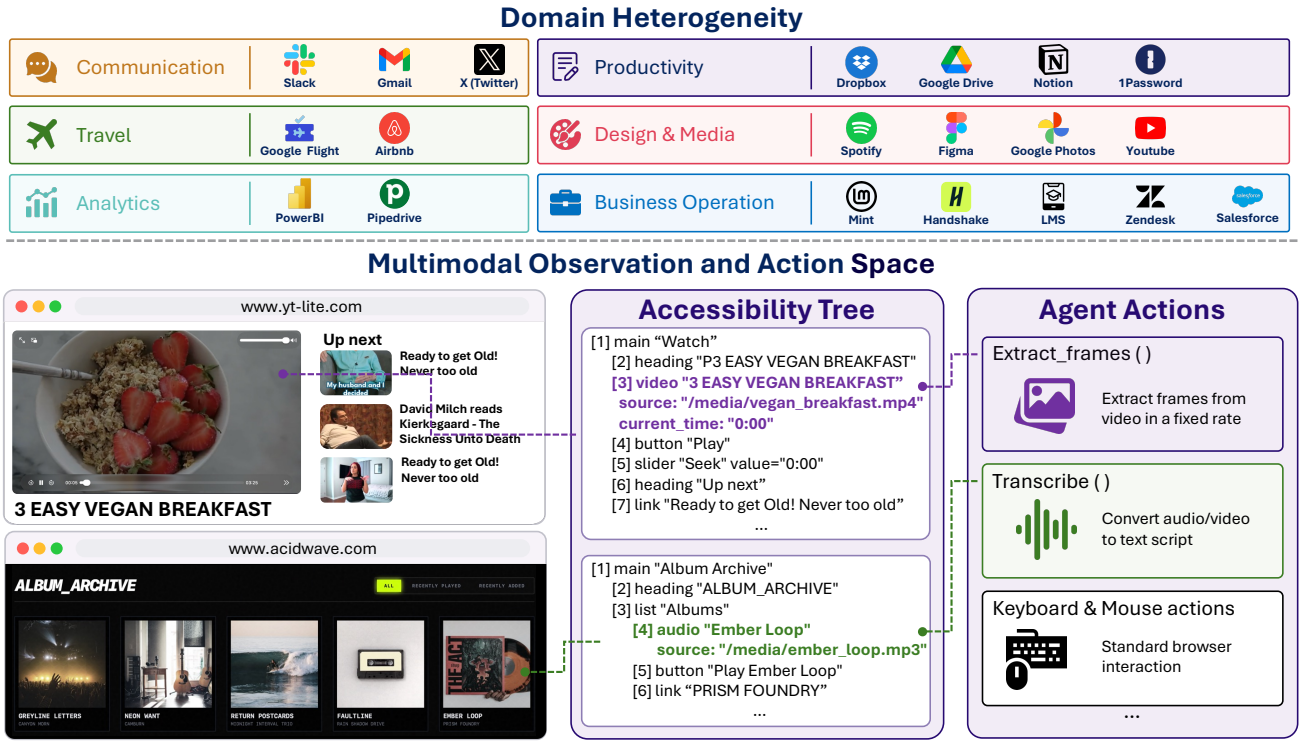

Figure 1. **Overview of WebArena-Pro. (A) Domain heterogeneity.** WebArena-Pro spans 20 self-hosted web applications across six domain categories (productivity, communication, analytics, business operations, design and media, and travel), each modeled on a widely used service such as Notion, Slack, Spotify, Figma, or Airbnb. The full set of applications and their reference services are listed in Table 2. **(B) Multimodal observation and action space.** WebArena-Pro extends the agent's observation space beyond text and vision to include audio and video. Embedded media sources are exposed to the agent through the accessibility tree, and agents invoke `transcribe()` and `extract_frames()` actions to convert temporal media into a form usable during browser workflows.

between runs, making consistent evaluation across rollouts difficult.

To this end, we introduce *WebArena-Pro*, a benchmark for web agents that addresses these limitations. First, addressing the application coverage gap, WebArena-Pro spans 20 self-hosted web applications across six domain categories (productivity, communication, analytics, business operations, design and media, and travel), with applications modeled on widely used services such as Notion, Slack, Spotify, Figma, and Airbnb (Table 2 lists all 20). Second, addressing the modality gap, WebArena-Pro evaluates audio and video as core observations alongside text and vision, exposing media sources through the accessibility tree and providing `transcribe()` and `extract_frames()` actions so that agents can incorporate temporal media directly into browser workflows. Third, addressing the reproducibility limitation of live benchmarks, WebArena-Pro runs each task in isolation against per-task seeded container state, supporting reproducible parallel evaluation that avoids drift between runs.

WebArena-Pro contains 300 tasks categorized by site coverage (225 single-site, 75 cross-site) and modality (221 text/vision, 79 audio/video), as detailed in Table 3. Unlike many prior benchmarks that generate many task instances from a small set of templates, each task in WebArena-Pro has its own dedicated template, which yields a more diverse evaluation (Table 1). Tasks are authored through a dedicated annotator interface and pass through a two-stage filter, in which LLM-assisted triage flags candidates that appear infeasible or trivial and human reviewers make the final acceptance decision. This pipeline keeps task quality grounded in human judgment while using LLM assistance to scale the authoring process. Our contributions are as follows:

- **Heterogeneous web coverage.** We construct a benchmark of 20 self-hosted web applications across six domain categories with distinct interface conventions, workflows, and data models.
- **Audio and video as core web observations.** We extend the web-agent action and observation space to temporal media, exposing audio and video sources through the accessibility tree and providing `transcribe()` and `extract_frames()` actions so that agents can use embedded media directly within browser workflows.

- **Reproducible, parallel evaluation.** We run each task in isolation against per-task seeded container state, supporting reproducible parallel rollouts and avoiding drift between runs.
- **Human-validated benchmark construction.** We release a task creation pipeline that combines an annotator interface, LLM-assisted infeasibility and triviality triage, and final human validation.

2. Related Work

Self-Hosted Web Benchmarks. WebArena is the closest predecessor to WebArena-Pro. It introduced a self-hosted benchmark for realistic web navigation over six web applications, with agents completing natural-language tasks by operating a browser (Zhou et al., 2023). VisualWebArena (Koh et al., 2024) extended the setting to visually grounded tasks where success depends on understanding screenshots rather than only DOM text, and WebChoreArena (Miyai et al., 2025) stressed long-horizon, tedious browsing patterns with self-hosted websites as well. While these benchmarks provide valuable foundations for realistic and reproducible web-agent evaluation, WebArena-Pro differs along three axes visible in Table 1. First, it covers 20 applications across six domain categories, the broadest application coverage among self-hosted web benchmarks. Second, it enforces per-task isolation by explicitly resetting application state between tasks. In contrast, these benchmarks retain state across tasks, so backend changes an agent makes while solving one task persist into the next and can contaminate later evaluations. Third, it authors each task as its own dedicated template. In contrast, these benchmarks instantiate many task instances from a smaller template set (e.g., 241 templates for 812 instances in WebArena), which may lead to a less diverse evaluation.

Enterprise Benchmarks. A parallel line of work pushes toward workplace realism. WorkArena evaluates agents on common knowledge-work tasks in ServiceNow (Drouin et al., 2024), and WorkArena++ extends this direction with compositional planning tasks and automatically generated ground-truth trajectories (Boisvert et al., 2024). TheAgentCompany simulates a small company across a handful of internal services and introduces non-binary partial scoring (Xu et al., 2025). These benchmarks emphasize enterprise workflows, long-horizon planning, and task composition. WebArena-Pro is complementary; rather than centering on one enterprise platform, it spans productivity, communication, analytics, business operations, design and media, and travel applications, while still retaining stateful authenticated workflows.

Live and Computer-Use Benchmarks. Other benchmarks broaden the evaluation setting in a different direction. Mind2Web studies generalization to real websites using collected human demonstrations (Deng et al., 2023), and OnlineMind2Web (Xue et al., 2025) considers a live variant where the target websites may change over time. Live benchmarks improve ecological validity but make repeatability difficult, since pages, data, policies, and layouts can drift between evaluations. WebArena-Pro instead uses self-hosted applications and seeded data, which makes failures easier to reproduce and enables backend-state evaluation. Beyond the browser, computer-use benchmarks such as OSWorld (Xie et al., 2024) broaden the target from browsers to full desktop environments, and Gym-Anything (Aggarwal et al., 2026) pursues a different scaling axis by using LLMs to configure environments and auto-generate evaluators. WebArena-Pro instead focuses specifically on web software, where browser instrumentation and application-level reset can be made reliable enough for controlled experiments, and keeps task authoring human-validated.

Multimodal Web Benchmarks. VisualWebArena showed that many web tasks require visual grounding and that text-only agents miss information that is obvious to humans (Koh et al., 2024). WebArena-Pro follows this motivation and extends multimodal support beyond static visual content to include audio and video. To enable this, we add two new actions to the action space of the agent, `transcribe()` and `extract_frames()`. The agent invokes `transcribe()` to convert an audio source to text and `extract_frames()` to sample representative frames from a video source. The resulting transcripts are merged into the accessibility-tree observation as synthetic nodes, while extracted frames are returned as image content alongside the AXTree, so the agent perceives temporal media within the same observation structure it already uses for the rest of the page (Figure 1).

3. WebArena-Pro

3.1. Web Environments

WebArena-Pro consists of 20 self-hosted web applications grouped into six domain categories, listed in Table 2. Each application runs as a Docker container with seeded data and fixed login credentials, and is reachable at a fixed local URL during evaluation. Before each task, the container state is reset to its seeded baseline, so changes made by an agent during one task do not persist into the next. Local hosting also fixes the application version and the database contents across runs, which removes the drift in pages, data, or policies.

The six domains are productivity, communication, analytics, business operations, design and media, and travel. Within each domain the applications share a broad task family, such as file management in productivity or dashboard analysis in

Table 1. Comparing WebArena-Pro to existing reproducible benchmarks for web and computer-use agents. "*Isolation*" refers to per-task environment reset enabling contamination-free parallel execution. "*Human-curated*" indicates that tasks are authored or verified by humans rather than auto-generated. A half-circle (◐) indicates partial support. [§]WorkArena Suite combines WorkArena (33 templates instantiated to 19,912 instances) and WorkArena++ (682 compositional tasks, each sampling from thousands of valid configurations). [†]OSWorld targets desktop applications rather than websites, with "# Envs" showing its 9 application domains. [‡]Gym-Anything's environments and tasks are LLM-generated at scale rather than human-curated.

		Tasks		Modalities			Eval & Execution	
Benchmark	**# Envs**	**Templates**	**Instances**	**Vision**	**Audio**	**Video**	**Isolation**	**Human-curated**
WebArena (Zhou et al., 2023)	6	241	812	✗	✗	✗	✗	✓
VisualWebArena (Koh et al., 2024)	3	233	910	✓	✗	✗	✗	✓
WebChoreArena (Miyai et al., 2025)	4	117	532	✓	✗	✗	✗	✓
WorkArena Suite[§] (Boisvert et al., 2024; Drouin et al., 2024)	1	715	19,912	✓	✗	✗	◐	✓
TheAgentCompany (Xu et al., 2025)	4	175	175	✓	✗	✗	✓	✓
OSWorld[†] (Xie et al., 2024)	9	369	369	✓	✗	✗	✓	✓
Gym-Anything[‡] (Aggarwal et al., 2026)	200	10,000+	10,000+	✓	✗	✗	✓	✗
WebArena-Pro (ours)	20	300	300	✓	✓	✓	✓	✓

analytics, but differ in interface conventions, authentication flows, and backend state schemas. NextCloud and MyDrive both store files, for example, but they expose distinct REST APIs, distinct folder and permission models, and distinct Document Object Model (DOM) structures, so an agent that has learned to operate one is not guaranteed to transfer to the other. This grouping is what allows WebArena-Pro to test transfer at the level of interface and backend conventions, rather than memorization of a single product.

Each application is modeled on a familiar reference application, listed in the third column of Table 2, which names the workflow pattern reproduced rather than implying data or feature parity.

3.2. Tasks and Action Space

A task in WebArena-Pro is specified by a natural-language intent, one or more starting sites, an authenticated browser state, and an evaluation specification that determines success and partial credit (Section 3.3). WebArena-Pro contains 300 tasks, with each task authored as its own dedicated template (Table 1).

We categorize these tasks along two independent dimensions, namely site coverage and modality, with qualitative examples in Table 3. Along site coverage, 225 tasks are **single-site** and require the agent to complete a workflow within one application, while 75 are **cross-site** and require the agent to coordinate between two or more applications, probing transfer across the interface and backend conventions of Section 3.1. Along modality, 221 tasks rely only on text and vision (**text/vision** tasks), while 79 are **audio/video** tasks that require information embedded in temporal media on a page, such as the spoken content of a podcast or the visual content of a video.

The default action space follows the BrowserGym convention (Chezelles et al., 2025), with element-targeted actions (`click`, `fill`, `select_option`, `scroll`) and navigation actions (`goto`, `go_back`). The observation space exposes each page as both a structured accessibility tree and a screenshot, enabling evaluation of text-only and vision-language agents under the same task specifications.

To support audio/video tasks, we extend this default action space by surfacing audio and video sources as nodes in the accessibility tree and introducing two actions that operate on them (Figure 1).

- `transcribe(url, start, end)`: Returns a transcript of the audio segment from `start` to `end` (in seconds) using a local Whisper model.
- `extract_frames(url, n, start, end)` (vision-language agents only): Returns n evenly spaced frames from the video segment between `start` and `end` (in seconds).

The agent invokes these actions when it judges that the audio or video content is needed for the task. The resulting transcripts and extracted frames are returned as observations in the next step, with transcripts merged into the accessibility tree as synthetic nodes and frames returned as image content alongside the accessibility tree.

3.3. Evaluation

We evaluate each web-agent trajectory with a rubric-based verifier. Each task is paired with a small set of human-authored natural-language criteria, tagged as `[action]` or `[outcome]`. Action rubrics measure whether the agent performed required operations, such as searching, filtering, sorting, or submitting a form. Outcome rubrics measure whether the resulting browser state or final answer satisfies the user's request; for question-answering tasks, these rubrics additionally include a reference answer against which the verifier judges the agent's response.

Table 2. The 20 self-hosted web applications in WebArena-Pro, grouped by domain. Each application is a self-hosted instance whose interface and workflow patterns are modeled on a widely-used reference application, while data and exact feature scope are controlled by the benchmark.

Domain	Site	Reference application	Description
Productivity	MyDrive	Dropbox	Personal cloud file storage and sharing
	NextCloud	Google Drive	Collaborative file storage with calendars, contacts, and chat
	Notion	Notion	Notes and knowledge-management workspace
	Password-Manager	1Password	Password and credential vault
Communication	Mattermost	Slack	Team messaging with channels and direct messages
	Thunderbird	Gmail	Email client with folders and contacts
	X-Lite	X (Twitter)	Microblogging and social posting
Analytics	Metabase	Power BI	Business-intelligence dashboards and SQL queries
	PipeCRM	Pipedrive	Sales pipeline and contact management
Business ops.	ERP-Next	Salesforce	Customer and supplier chain management
	Firefly	Mint	Personal finance and budget management
	JobPortal	Handshake	Job postings and application tracking
	Minerva-Lite	University LMS	Courses, assignments, and grades
	Peppermint	Zendesk	Customer-support ticketing
Design & media	AcidWave	Spotify	Music streaming with lyrics and playlists
	Immich	Google Photos	Photo library with albums and search
	Penpot	Figma	Web-based design and prototyping
	YT-Lite	YouTube	Video hosting and playback
Travel	Flight-Search	Google Flights	Flight search and itinerary planning
	TravelMap	Airbnb	Travel and lodging booking

The verifier observes the trajectory step by step. At each step, it receives the agent's reasoning, the action string, and pre- and post-action screenshots. When the action targets a UI element, its bounding box is highlighted in the pre-action screenshot. The judge may call a structured `grade_rubric` tool to update any rubric with a new verdict and rationale. The latest rubric state, a verdict (pending, pass, or fail) paired with a rationale history, persists in context as the judge's working memory. Ungraded rubrics remain pending until sufficient evidence appears; earlier verdicts can be revised as new evidence emerges. On the final step, the judge is prompted once more to resolve all remaining pending rubrics. A trajectory succeeds if and only if every rubric passes (logical AND).

3.4. Benchmark Construction

Mock Data and Personas. Each application is shipped with seeded data so that tasks operate on a realistic, non-empty state. Seed contents are application-appropriate and include documents, files, messages, calendar events, dashboards, listings, support tickets, and similar records, as well as the relations between them, such as channel memberships, file shares, and assignments. Each application is also seeded with one or more user personas, represented by credentials and pre-populated account state, so that tasks begin from an authenticated browser session.

Task Creation Pipeline. Tasks are authored by an annotator team composed of the authors of this paper. Through a dedicated annotator interface, an annotator writes the natural-language intent, selects the relevant sites, and drafts the evaluation rubric. Each candidate then undergoes an automated triage step that flags trivial tasks by running against an LLM-based agent and discarding those solved correctly on the first attempt. Tasks that pass triage are reviewed by a second annotator on the live application, who confirms feasibility, finalizes the rubric items used for scoring, and confirms the difficulty label. Human validation is the gating step before a task enters the released benchmark, and we do not release tasks that fail any of these checks.

4. Experiments

4.1. Experimental Setup

Environment Setup. We evaluate agents by running each task in a BrowserGym-compatible browser environment (Chezelles et al., 2025). At the beginning of an episode, the benchmark starts a fresh website container, waits for

Table 3. Qualitative examples of WebArena-Pro tasks, organized by site coverage (single-site or cross-site) and modality (text/vision or audio/video).

Coverage	Modality	Sites	Example intent
Single-site	Text/Vision	NextCloud	Set the displayed currency to USD. Find the fastest one-way economy flight from New York (JFK) to Los Angeles (LAX) on 2026-06-15. If multiple flights are tied for the shortest total travel time, choose the first tied flight shown after sorting by Fastest.
	Audio/Video	YT-Lite	In the video "Tips for winter biking", does the creator record himself from a first person point of view?
Cross-site	Text/Vision	Notion, Pepper-mint	Create a Notion page summarizing the VPN drops ticket and naming its assigned owner.
	Audio/Video	Mattermost, Acidwave	Send Mike a direct message recommending a rock song suitable for someone going through a breakup.

Table 4. Qualitative rubric example from the NextCloud task configuration. The judge verifies whether the agent performed actions 1 and 2, and verifies outcomes 3 and 4 at each step.

#	Type	Criterion
1	Action	The agent viewed the Project Alpha Talk history.
2	Action	The agent typed "Client Meeting Update" as the name for a new file.
3	Outcome	The content of the "Client Meeting Update" file states that the client meeting moved to next Thursday.
4	Outcome	A file named "Client Meeting Update" exists in the Nextcloud file directory.

Table 5. Agreement between each judge model and human evaluation across six models. We sample 50 trajectories from each agent model, which results in 300 annotated trajectories.

	Judge Agreement Rate (%)		
Model	**Qwen3.6-27B**	**Gemini-3.1 Pro**	**Kimi K2.6**
Gemini-3.1 Pro	98.0	92.0	86.0
Gemini-3.1 Flash Lite	96.0	94.0	86.0
Claude Opus 4.7	90.0	80.0	84.0
Gemma4-31B	92.0	82.0	87.5
Qwen-3.5-9B	88.0	80.0	76.0
Qwen-3.6-27B	78.0	80.0	78.0
Overall	**90.3**	84.7	82.3

readiness, performs the required UI login, and navigates the browser to the task start URL. The agent receives the natural-language intent, with the list of accessible websites for the task, and then acts in the browser until it calls `done()` or `send_msg_to_user()`, or reaches the maximum number of steps. Each task is limited to 50 browser actions. We report success rates using the benchmark's evaluator output.

Modeling. We evaluate representative browser agents built from current language and vision-language models. Each baseline receives the same task intent, browser observation, and high-level action space. All agents operate on the accessibility tree and screenshots augmented with Set-of-Mark (Yang et al., 2023). Agents use our WebArena-Pro custom action set, which is derived from the bid action set from BrowserGym, and augmented with `done()`, `transcribe()`, and `extract_frames()` actions, which respectively allow finishing the current task, transcribing audio segments, and extracting frames from video segments (Section 3.2). We select seven models as agent backbones: four open-source models, including Qwen3.5-9B, Qwen3.6-27B (Qwen Team, 2026), Gemma4-24B-A4B-IT, Gemma4-31B-IT (Farabet & Lacombe, 2026). And three closed-source models, including Gemini-3.1-flash, Gemini-3.1-pro-preview (Pichai et al., 2025), and Claude-Opus-4.7 (Anthropic). For all models, we enable extended thinking and set the temperature to 1.0, following the recommendations of the respective model developers. We also cap the maximum number of tokens per step at 40k (Chezelles et al., 2025; Drouin et al., 2024).

Rubric-based Judge. To select a judge model for the rubric-based verifier described in Section 3.3, we compare three candidates, Qwen3.6-27B, Kimi-K2.6, and Gemini-3.1-Pro. Specifically, we measure how well each candidate's rubric verdicts correlate with human-annotated verdicts on a fixed set of 300 trajectories produced by six agent models in Table 5. As reported in Table 5, Qwen3.6-27B achieves the highest correlation with human evaluation, so we use it as the rubric verifier in all subsequent experiments unless otherwise specified.

Metrics. We report task Success Rate (SR) as our primary metric, defined as the fraction of tasks for which every rubric passes under the rubric-based verifier of Section 3.3. To capture partial progress on unsuccessful tasks, we also report a partial-completion score that measures the fraction of rubric items passed per task, in Appendix A.

Table 6. Success rates of agents on WebArena-Pro at different step budgets (30 steps and 50 steps), broken down by domain category. Domain columns abbreviate Productivity (Prod.), Communication (Comm.), Analytics (Anlt.), Business Operations (BusOps), Design & Media (Design), and Travel.

	Domains SR (%)						
Model	**Prod.**	**Comm.**	**Anlt.**	**BusOps**	**Design**	**Travel**	**Overall SR (%)**
30 steps							
Gemini-3.1 Pro	**30.4**	**26.7**	**37.5**	**22.0**	**32.0**	**33.3**	**29.3**
Gemini-3.1 Flash Lite	14.3	13.3	**37.5**	19.0	27.8	22.2	21.4
Claude Opus 4.7	16.1	20.0	25.0	15.3	29.9	22.2	22.0
Qwen3.6-27B	8.9	15.6	12.5	8.5	17.5	18.5	13.7
Qwen3.5-9B	3.6	0.0	6.2	5.1	11.3	3.7	6.0
Gemma4-31B	21.4	20.0	**37.5**	**22.0**	20.6	29.6	22.7
Gemma4-24B-A4B	17.9	24.4	25.0	**22.0**	18.8	14.8	20.1
50 steps							
Gemini-3.1 Pro	**37.5**	33.3	43.8	**27.1**	**43.3**	**37.0**	**37.0**
Gemini-3.1 Flash Lite	19.6	26.7	43.8	25.9	34.0	25.9	28.4
Claude Opus 4.7	17.9	22.2	37.5	20.3	40.2	29.6	28.3
Qwen3.6-27B	10.7	15.6	12.5	8.5	20.6	18.5	15.0
Qwen3.5-9B	5.4	4.4	6.2	6.8	17.5	3.7	9.3
Gemma4-31B	26.8	28.9	**56.2**	25.4	23.7	29.6	27.7
Gemma4-24B-A4B	23.2	**35.6**	43.8	25.4	21.9	14.8	25.4

4.2. Main Results

Table 6 reports the success rate of each agent at two step budgets, 30 and 50. Gemini-3.1 Pro leads at 37.0% with a 50-step budget, with Gemini Flash Lite and Claude Opus 4.7 close behind. Even the best agent leaves nearly two thirds of the benchmark unsolved. Several other agents cluster between 25 and 28% at the larger budget, and the smaller Qwen models lag well behind at 15.0% and 9.3%. Doubling the budget from 30 to 50 steps yields gains of 5 to 8 points for most agents and smaller gains for the weakest Qwen models. This shows that WebArena-Pro is a challenging and unsaturated benchmark even for state-of-the-art agents, with additional steps narrowing the gap only modestly.

Performance varies across domains, which shows the heterogeneity of the benchmark (Section 3.1). Analytics yields the highest success rates for every agent, with Gemma4-31B reaching 56.2% and both Gemini variants at 43.8% at 50 steps, reflecting the relative simplicity of dashboard queries. Productivity is the hardest domain for closed-source agents Gemini Flash Lite 19.6%, Claude Opus 17.9% despite their competitive overall scores. Design and Media exposes a different gap, where the open-source Gemma family drops to 23.7% and 21.9% while Gemini-3.1 Pro and Claude Opus 4.7 stay above 40%, suggesting that closed source models show bigger gaps in environements that require multimodal and vision capability compared to open source models.

4.3. Site Coverage and Modality

In Table 7, we report the success rate of each agent decomposed along two task axes, site coverage (single-site and cross-site) and modality (text/vision and audio/video). Cross-site tasks are uniformly harder than single-site tasks, but the magnitude of the drop varies across agents. Claude Opus 4.7 shows the largest gap, falling from 33.8% on single-site tasks to 12.0% on cross-site tasks, a drop of nearly 22%. This indicates that coordinating actions across multiple applications remains a substantial challenge for current agents.

Regarding the modality, audio and video tasks yield higher success rates than text and vision tasks for nearly every agent. We attribute this to the structured access provided by `transcribe()` and `extract_frames()` actions, which lifts the audio/video perception into the language and image channels that these models already handle well. The Gemma family is the only exception, where audio and video tasks yield comparable or lower success than text and vision tasks, indicating that the multimodal action extension does not benefit much in this model family.

5. Discussion

Limitations. The rubric-based verifier relies on a judge model whose agreement with human annotators is high but imperfect, so individual task verdicts carry some residual

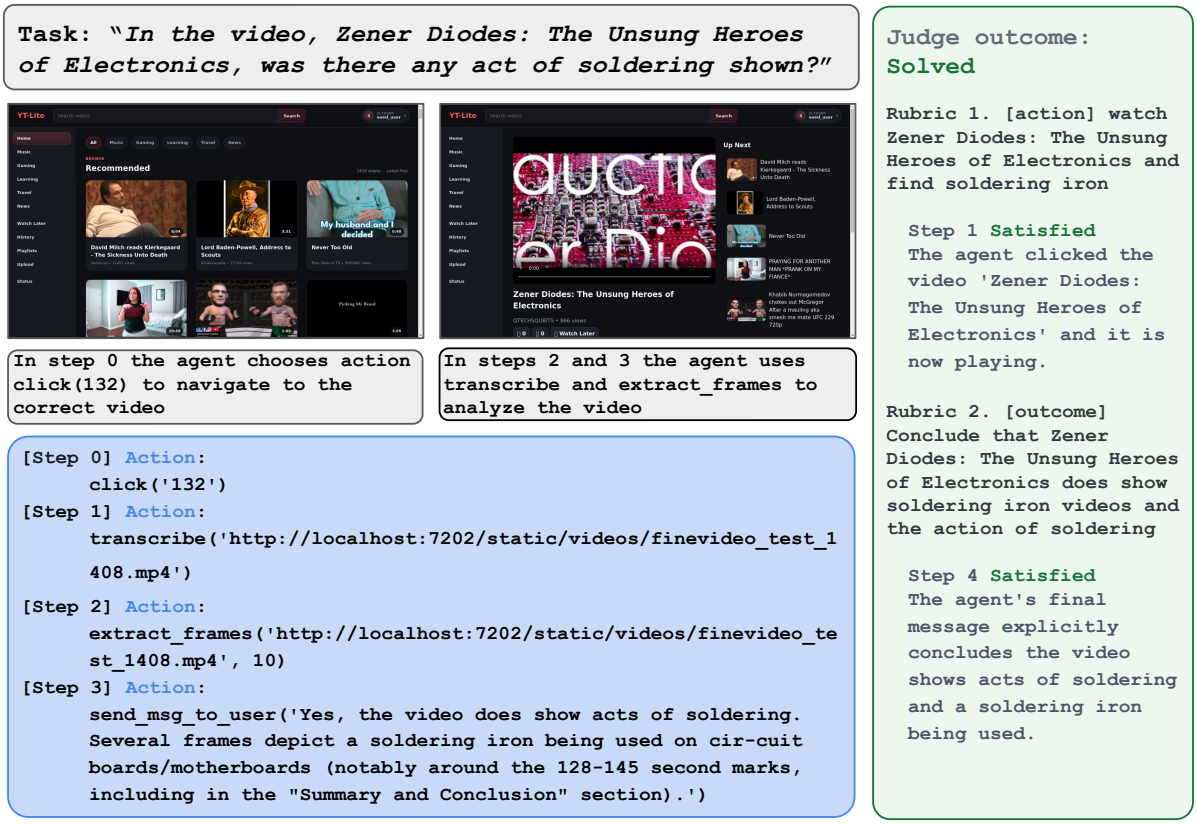

Figure 2. **Qualitative example of a Qwen3.6-27B agent trajectory in WebArena-Pro.** Given a single-site audio/video task on YT-Lite, namely "*In the video, Zener Diodes: The Unsung Heroes of Electronics, was there any act of soldering shown?*", the agent first clicks the matching video thumbnail. On the video page, it invokes `transcribe()` to read the spoken content and `extract_frames()` to sample 10 representative frames. The transcript does not mention soldering, but the extracted frames at $t = 128.6$ s and $t = 144.7$ s show a soldering iron, and the agent answers accordingly. The rubric verifier confirms a passing trajectory.

Table 7. Success rate (SR) of each agent on WebArena-Pro at a 50-step budget, decomposed along the two task axes of Section 3.2. The site coverage axis splits the 300 tasks into 225 single-site and 75 cross-site tasks, and the modality axis splits them into 221 text/vision and 79 audio/video tasks. Best values are bold and second-best are underlined.

	Site Coverage		Modality	
Model	**Single-site (%)**	**Cross-site (%)**	**Text/Vision (%)**	**Audio/Video (%)**
Gemini-3.1 Pro	**40.4**	**26.7**	**32.1**	**50.6**
Gemini-3.1 Flash Lite	29.9	<u>24.0</u>	25.5	36.7
Claude-Opus-4.7	<u>33.8</u>	12.0	21.7	<u>46.8</u>
Qwen3.6-27B	18.2	5.3	12.2	22.8
Qwen3.5-9B	12.0	1.3	5.9	19.0
Gemma4-31B	28.9	<u>24.0</u>	<u>29.4</u>	22.8
Gemma4-24B-A4B	28.1	17.3	25.5	25.3

error, and per-domain success rates for the smaller domains have non-trivial sampling variance. Our action-space extension also covers only prerecorded audio segments and discrete video frames, leaving richer temporal interaction modalities such as voice dialogue and live video streams to future work.

Future work. The same self-hosted infrastructure that supports parallel evaluation can also serve as a training environment, where the rubric-based verifier provides a per-task reward signal for reinforcement learning or rejection sampling. The controlled environment also supports robustness studies that perturb the underlying web pages to test resilience to UI drift and prompt injection. Finally, expanding the action space to richer temporal modalities such as live voice and streaming video would let the benchmark track agents that move beyond static page interaction.

6. Conclusion

We presented WebArena-Pro, a reproducible benchmark of 300 human-validated tasks across 20 self-hosted web applications spanning six domain categories. It treats audio and video as core observations alongside text and vision, and runs each task in container isolation for reproducible parallel evaluation. Across seven language and vision-language model backbones, the strongest agent reaches only 37.0%

success at a 50-step budget, with cross-site tasks remaining substantially harder than single-site ones. These results indicate considerable headroom for progress on heterogeneous, multimodal web agents.

Impact Statement

This work aims to improve the evaluation of web agents by providing a more realistic, diverse, and reproducible benchmark. It may support progress toward more capable assistants for digital workflows, accessibility, and productivity. At the same time, advances in web agents could also increase risks such as unwanted automation, spam, phishing, or misuse of online services. By grounding evaluation in controlled self-hosted environments, this benchmark can help researchers study both capability and safety before deployment on real-world websites.

References

Abuelsaad, T., Akkil, D., Dey, P., Jagmohan, A., Vempaty, A., and Kokku, R. Agent-E: From Autonomous Web Navigation to Foundational Design Principles in Agentic Systems, July 2024.

Aggarwal, P., Neubig, G., and Welleck, S. Gym-anything: Turn any software into an agent environment, 2026. URL https://arxiv.org/abs/2604.06126.

Anthropic. Introducing Claude Opus 4.7. URL https://www.anthropic.com/news/claude-opus-4-7.

Boisvert, L., Thakkar, M., Gasse, M., Caccia, M., Chezelles, T. L. S. D., Cappart, Q., Chapados, N., Lacoste, A., and Drouin, A. WorkArena++: Towards Compositional Planning and Reasoning-based Common Knowledge Work Tasks, July 2024.

Chezelles, T. L. S. D., Gasse, M., Drouin, A., Caccia, M., Boisvert, L., Thakkar, M., Marty, T., Assouel, R., Shayegan, S. O., Jang, L. K., Lù, X. H., Yoran, O., Kong, D., Xu, F. F., Reddy, S., Cappart, Q., Neubig, G., Salakhutdinov, R., Chapados, N., and Lacoste, A. The BrowserGym Ecosystem for Web Agent Research, February 2025.

Deng, X., Gu, Y., Zheng, B., Chen, S., Stevens, S., Wang, B., Sun, H., and Su, Y. Mind2Web: Towards a Generalist Agent for the Web, June 2023.

Drouin, A., Gasse, M., Caccia, M., Laradji, I. H., Del Verme, M., Marty, T., Boisvert, L., Thakkar, M., Cappart, Q., Vazquez, D., Chapados, N., and Lacoste, A. WorkArena: How Capable Are Web Agents at Solving Common Knowledge Work Tasks?, March 2024.

Farabet, C. and Lacombe, O. Gemma 4: Byte for byte, the most capable open models, April 2026. URL https://blog.google/innovation-and-ai/technology/developers-tools/gemma-4/.

Gur, I., Furuta, H., Huang, A., Safdari, M., Matsuo, Y., Eck, D., and Faust, A. A Real-World WebAgent with Planning, Long Context Understanding, and Program Synthesis, October 2023.

hattami, A. E., Thakkar, M., Chapados, N., and Pal, C. Webarena verified: Reliable evaluation for web agents. In *Workshop on Scaling Environments for Agents*, 2025. URL https://openreview.net/forum?id=94tlGxmqkN.

He, H., Yao, W., Ma, K., Yu, W., Dai, Y., Zhang, H., Lan, Z., and Yu, D. WebVoyager: Building an End-to-End Web Agent with Large Multimodal Models, February 2024.

Koh, J. Y., Lo, R., Jang, L., Duvvur, V., Lim, M. C., Huang, P.-Y., Neubig, G., Zhou, S., Salakhutdinov, R., and Fried, D. VisualWebArena: Evaluating Multimodal Agents on Realistic Visual Web Tasks. https://arxiv.org/abs/2401.13649v2, January 2024.

Miyai, A., Zhao, Z., Egashira, K., Sato, A., Sunada, T., Onohara, S., Yamanishi, H., Toyooka, M., Nishina, K., Maeda, R., Aizawa, K., and Yamasaki, T. Webchorearena: Evaluating web browsing agents on realistic tedious web tasks, 2025. URL https://arxiv.org/abs/2506.01952.

Pichai, S., Hassabis, D., and Kavukcuoglu, K. A new era of intelligence with Gemini 3, November 2025. URL https://blog.google/products-and-platforms/products/gemini/gemini-3/.

Qwen Team. Qwen3.6-27B: Flagship-level coding in a 27B dense model, April 2026. URL https://qwen.ai/blog?id=qwen3.6-27b.

Xie, T., Zhang, D., Chen, J., Li, X., Zhao, S., Cao, R., Hua, T. J., Cheng, Z., Shin, D., Lei, F., Liu, Y., Xu, Y., Zhou, S., Savarese, S., Xiong, C., Zhong, V., and Yu, T. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 52040–52094. Curran Associates, Inc., 2024. doi: 10.52202/079017-1650. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/5d413e48f84dc61244b6be550f1cd8f5-Paper-Datasets_and_Benchmarks_Track.pdf.

Xu, F. F., Song, Y., Li, B., Tang, Y., Jain, K., Bao, M., Wang, Z. Z., Zhou, X., Guo, Z., Cao, M., Yang, M., Lu, H. Y., Martin, A., Su, Z., Maben, L., Mehta, R., Chi, W., Jang, L., Xie, Y., Zhou, S., and Neubig, G. Theagentcompany: Benchmarking llm agents on consequential real world tasks, 2025. URL https://arxiv.org/abs/2412.14161.

Xue, T., Qi, W., Shi, T., Song, C. H., Gou, B., Song, D., Sun, H., and Su, Y. An illusion of progress? assessing the current state of web agents. *arXiv preprint arXiv:2504.01382*, 2025.

Yang, J., Zhang, H., Li, F., Zou, X., Li, C., and Gao, J. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

Zhou, S., Xu, F. F., Zhu, H., Zhou, X., Lo, R., Sridhar, A., Cheng, X., Ou, T., Bisk, Y., Fried, D., Alon, U., and Neubig, G. WebArena: A Realistic Web Environment for Building Autonomous Agents, October 2023.

A. Partial Completion Score

Table 8 shows the partial completion score computed based on the verdict of the judge on the 821 rubrics of the benchmarks.

Table 8. Partial completion scores of agents on WebArena-Pro with a 50-step budget, broken down by domain category. Domain columns abbreviate Productivity (Prod.), Communication (Comm.), Analytics (Anlt.), Business Operations (BusOps), Design & Media (Design), and Travel.

	Domains ↑ (%)						
Model	**Prod.**	**Comm.**	**Anlt.**	**BusOps**	**Design**	**Travel**	**Overall ↑ (%)**
50 steps, Partial correctness score							
Gemini-3.1-flash	45.9	57.9	55.0	43.0	54.7	45.6	50.6
Gemini-3.1-pro-preview	60.0	50.2	63.7	40.6	64.4	50.2	55.4
Claude-Opus-4.7	38.2	41.0	48.3	29.4	55.8	47.1	43.9
Qwen3.5-9B	17.7	17.4	19.2	15.2	33.6	13.3	22.0
Qwen3.6-27B	27.2	32.3	30.6	20.2	37.8	37.4	31.1
Gemma4-24B-A4B-IT	49.6	59.0	54.2	41.7	47.7	36.8	48.1
Gemma4-31B-IT	48.6	55.6	62.0	44.5	45.0	46.4	48.2

B. Failure Modes Analysis

When analyzing the 7 models we found that per-model patterns of failure modes differ. Claude and Qwen3.5-9B most often fail by exhausting the action budget or looping without recovery. Gemini 3.1 Pro has the lowest failure rate, but its failures still often come from omitted final artifacts and wrong target entities. Both Gemma models show many premature or empty finalizations, especially Gemma 4 31B. Qwen3.6-27B improves over Qwen3.5-9B but still frequently misses required end-state actions and selects wrong entities or values.

The two dominant failure modes are incomplete workflows and long-horizon stagnation. In half of the failures, the agent reached some intermediate state but did not complete a required artifact or action such as sending, saving, submitting, sharing, downloading, booking, or updating. In the other half, the agent hit the step limit or entered a repeated navigation/search loop.

The next tier of failures reflects constraint tracking and evidence quality. An exact-constraint error means the agent used a nearby but rubric-invalid condition, such as filtering a Metabase category as ‘Gadget‘ when the required exact value was ‘Electronics‘, or identifying a flight without applying the required cheapest/date/currency constraint. We detail the failure patterns with examples in Table 9. Figure 3 shows the distribution of the trajectory length by model for solved and unsolved tasks.

Overall, the failure are more about brittle long-horizon execution: agents lose task constraints, fail to recover from unproductive navigation, skip verification, or stop before the UI state actually satisfies the rubric.

C. Websites

WebArena-Pro includes 20 containerized website environments. Of these, 9 are built by adapting existing open-source applications with benchmark-specific seeded data and reset logic, while 11 are custom websites developed with AI assistance and human review. We provide details about the sources and Docker image sizes in Table 10. Additionally, in WebArena-Pro, there is no data is seeded after the container creation. For most websites, the data is seeded when building the image; the container can then be built fast. Exceptions are made for ERP-Next, which initializes the containers with data when it starts.

Media datasets. Some WebArena-Pro environments required media-heavy seed datasets that are stored outside Git and baked into Docker images at build time. For Acidwave, we created a music catalog with CC-licensed MP3 assets and accompanying metadata, including title, artist, album, genre, duration, play counts, license, attribution, and source URL; the current seed references 50 unique audio files. For `yt-lite`, we used samples from FineVideo: videos were decoded locally, paired with generated thumbnails, and transcoded to 720p HEVC/H.265 for compression before being packaged

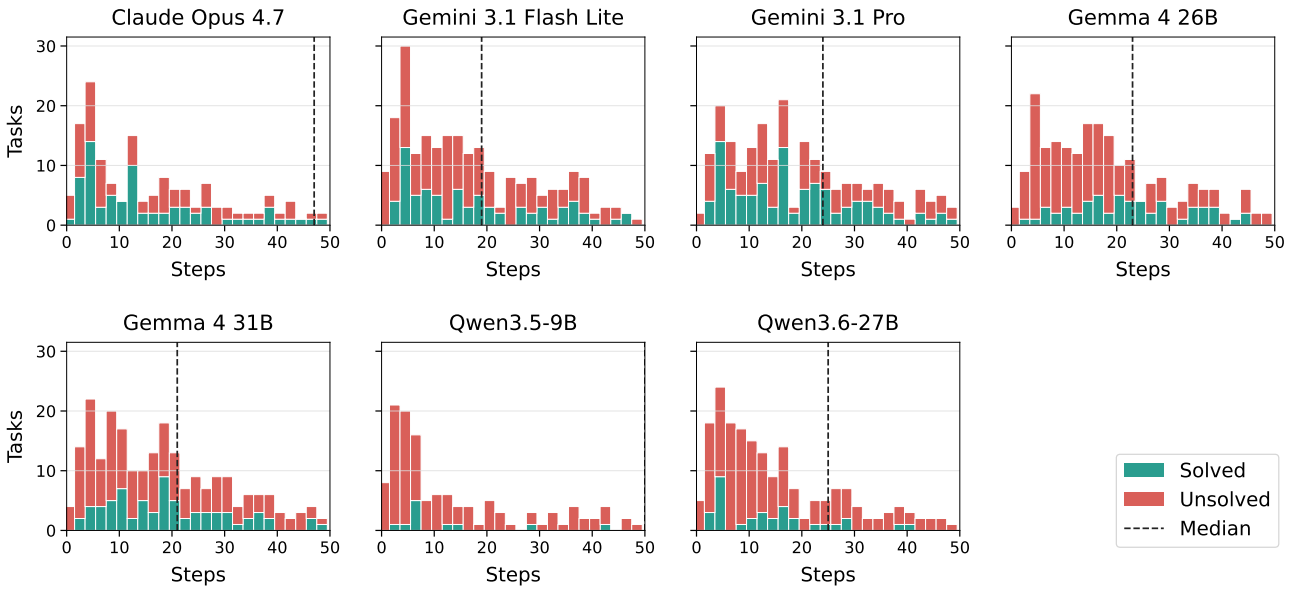

Figure 3. Trajectory length distribution per model for solved and unsolved tasks.

in the yt-lite image. For Immich, we sampled photos from Open Images V7, assigned them across the admin and agent accounts, manually curated them to remove low-quality, inappropriate, duplicate, or privacy-sensitive images, and then baked the uploaded photos, thumbnails, database state, and ML cache into the final image.

Table 9. Representative failure modes, the websites where they most often appeared, and one example of each pattern.

Failure pattern	**Main websites**	**Example**
Wrong answer/entity	Thunderbird, Acidwave, Mattermost, Notion, YT-Lite	On Acidwave, one agent answered `00:00` for a folk-song duration task whose expected answer was `33:22`.
Filter/sort/exact-constraint error	Flight-search, Firefly, Metabase, Mattermost, Notion	On Metabase, an agent built a product-rating table but filtered `Gadget` instead of the exact category `Electronics`.
Skipped verification	YT-Lite, Acidwave, Immich, TravelMap, Notion	On YT-Lite, an agent answered from the wrong Warhammer video instead of opening or watching the requested Warhammer III video.
Navigation/search failure	Flight-search, Notion, Mattermost, YT-Lite, TravelMap	In a Nextcloud and Mattermost task, the agent failed to open the budget file and never navigated to Mattermost to message Jane.
Premature or missing final response	YT-Lite, TravelMap, Peppermint, Acidwave, Flight-search	On Peppermint, an agent reported a database-comment task as infeasible without opening the target discussion.

Table 10. Docker image sizes and upstream sources for WebArena-Pro website environments.

Website	Construction basis / upstream repository	Docker image size
`acidwave`	AI-assisted custom	1.04 GB
`erp-next`	Open-source ERPNext/Frappe	2.84 GB
`firefly`	Open-source Firefly III	1.78 GB
`flight-search`	AI-assisted edited from airline-route-data	270 MB
`immich`	Open-source Immich	16.6 GB
`jobportal`	AI-assisted edited from jobportal	275 MB
`mattermost`	Open-source Mattermost	2.4 GB
`metabase`	Open-source Metabase	2.72 GB
`minerva-lite`	AI-assisted custom	209 MB
`mydrive`	AI-assisted custom	2.77 GB
`nextcloud`	Open-source Nextcloud	2.1 GB
`notion`	AI-assisted custom	1.74 GB
`password-manager`	AI-assisted custom	244 MB
`penpot`	Open-source Penpot	2.0 GB
`peppermint`	Open-source Peppermint	1.46 GB
`pipedrive`	AI-assisted custom	400 MB
`thunderbird`	Roundcube + Dovecot webmail stack	494 MB
`travelmap`	AI-assisted custom	278 MB
`x-lite`	AI-assisted custom	210 MB
`yt-lite`	AI-assisted custom	60.8 GB