УДК 004.89:004.932

ВЕБЗАСТОСУНОК ДЛЯ ПРОВЕДЕННЯ ХУДОЖНІХ КОНКУРСІВ ІЗ ЗАХИСТОМ РОБІТ ВІД НЕСАНКЦІОНОВАНОГО ВИКОРИСТАННЯ ДЛЯ НАВЧАННЯ ГЕНЕРАТИВНИХ МОДЕЛЕЙ

Гурницька В.О.[1]; Нестеров Ю.М.[2]
[1,2] Національний університет «Одеська політехніка»
[1,2] Україна, Одеса
[1] 7507063@stud.op.edu.ua; [2] nesterov@op.edu.ua

Анотація. *У роботі проаналізовано вебзастосунки для проведення художніх конкурсів та визначено їхні основні недоліки. Запропоновано розроблення вебзастосунку з інтегрованою функцією отруєння зображень на сервері для скорочення часу подання конкурсних заявок і захисту робіт від використання для навчання генеративних моделей.*
Ключові слова: *художні конкурси; розроблення вебзастосунку; отруєння даних.*

Вступна частина. У професійній діяльності цифрових художників важливе місце займає участь у мистецьких конкурсах, які для 36% митців становлять суттєву частину доходу. Водночас творчий процес супроводжується значним адміністративним навантаженням: щотижня понад 10% робочих годин художники присвячують пошуку конкурсних можливостей, оформленню заявок, підготовці портфоліо та просуванню власних робіт. Унаслідок цього зменшується кількість часу, який вони можуть приділяти безпосередньо творчій діяльності [1, 2].

Ситуацію ускладнив стрімкий розвиток генеративного штучного інтелекту (ШІ). Для формування великих навчальних наборів даних розробники генеративних моделей застосовують вебскрейпінг (англ. web scraping) – автоматизований збір інформації із відкритих інтернет-ресурсів за допомогою спеціалізованих програм. Поширення такої практики викликає занепокоєння серед художників: понад 61% сприймають штучний інтелект як загрозу своїй професійній діяльності [3]. При цьому навіть прямі заборони на збирання авторського контенту не гарантують його захисту. Показовими є випадки соціальної мережі Cara, правила якої забороняють збирати та використовувати роботи художників для навчання генеративного ШІ: попри встановлені ліцензійні обмеження, платформа неодноразово потерпала від атак вебскрейперів [4].

Для протидії використанню своїх робіт у навчанні генеративних моделей художники почали застосовувати інструменти алгоритмічного отруєння зображень: вони змінюють графічні файли таким чином, щоб перешкоджати їх правильному розпізнаванню штучним інтелектом. Отруєння виконується локально на пристроях користувачів, тому тривалість обробки залежить від доступних обчислювальних ресурсів.

Оскільки участь у художніх конкурсах часто передбачає подання серії робіт, необхідність окремої обробки кожного зображення суттєво збільшує загальний час підготовки конкурсної заявки. Це обґрунтовує доцільність розробки спеціалізованого вебзастосунку для проведення художніх конкурсів, який прискорить процес подання заявок завдяки виконанню отруєння зображень на серверній частині.

Метою роботи є скорочення часових витрат на підготовку заявок для участі у художніх конкурсах шляхом розроблення спеціалізованого вебзастосунку з функцією автоматизованого алгоритмічного отруєння графічних файлів, спрямованою на унеможливлення їх коректного аналізу генеративними моделями ШІ.

Основна частина. Серед сучасних засобів для алгоритмічного отруєння зображень найбільшої прихильності серед художників набули інструменти Glaze та Nightshade. Хоча обидва передбачають внесення змін до зображень, вони відрізняються своїм призначенням і принципами дії.

Glaze призначений для запобігання відтворенню індивідуального художнього стилю автора генеративними моделями. Його робота ґрунтується на маскуванні стилістичних особливостей зображення через внесення малопомітних змін до значень пікселів. У результаті нейромережа сприйматиме отруєний малюнок як виконаний в іншому стилі. Наприклад, акварельна ілюстрація може інтерпретуватися моделлю як вугільний ескіз [5].

Натомість Nightshade спрямований на захист змісту (семантики) зображення шляхом порушення зв'язку між його візуальним вмістом і текстовим описом. Унаслідок цього під час навчання нейромережа може засвоювати хибні відповідності: наприклад, зображення собаки асоціюватиметься не з відповідним об'єктом, а зі шкіряною сумкою. Якщо значна кількість отруєних зображень потрапить до навчальної вибірки, це погіршить здатність генеративної моделі правильно відтворювати об'єкти за текстовими запитами [6].

Аби визначити вимоги до розроблюваного вебзастосунку було проаналізовано аналогічні програмні рішення. Call For Entry (CaFÉ) – спеціалізований вебзастосунок для мистецьких організацій, що дозволяє проводити необмежену кількість етапів конкурсного відбору з різним складом журі. Проте платформа підтримує лише зображення у форматі JPEG розміром до 5 МБ, що може погіршувати їхню якість.

Zealous поєднує проведення мистецьких конкурсів із формуванням публічного портфоліо художника. Блоковий конструктор дозволяє організаторам гнучко налаштовувати конкурсні анкети, проте доступ до окремих функцій залежить від тарифного плану.

ArtStation – популярна платформа для цифрових художників, конкурси на якій передбачають ведення «щоденника розробки» з публікацією проміжних етапів створення роботи. Однак функція організації конкурсів недоступна звичайним користувачам.

Соціальна мережа Cara орієнтована на художників та використовує автоматизовану фільтрацію ШІ-контенту. Платформа дозволяє завантажувати зображення без жорстких обмежень щодо розміру, проте не має спеціалізованих інструментів для проведення конкурсів. Зведену порівняльну характеристику розглянутих рішень наведено в табл. 1.

Таблиця 1 – Порівняльна характеристика програм-аналогів

Характеристика	Застосунок			
	CaFÉ	*Zealous*	*ArtStation*	*Cara*
Створення конкурсу	Платна підписка	Безкоштовно або преміум	Лише партнери платформи	Безкоштовно через допис
Формат створення анкети	Стандартний шаблон	Блоковий конструктор	Сторінка на платформі	Допис із медіа
Багатоетапне оцінювання	Необмежена кількість	Залежить від тарифу	Задається платформою	Відсутнє
Формати зображень	JPEG	JPEG, PNG, PDF	JPEG, PNG, GIF, MP4	JPEG, PNG, GIF
Обмеження розміру файлу	До 5 МБ	До 50 МБ	Залежить від типу файлу	Без обмежень
Фільтрація ШІ-робіт	Відсутня	Відсутня	Перевірка етапів малювання	Перевірка алгоритмом
Блокування збереження робіт	Відсутнє	Відсутнє	Відсутнє	Відсутнє
Механізм отруєння зображень	Відсутній	Відсутній	Відсутній	Відсутній

Проведений аналіз показав, що жодна з розглянутих платформ не поєднує повноцінні інструменти для організації художніх конкурсів із вбудованою функцією алгоритмічного отруєння зображень, що підтверджує доцільність розроблення вебзастосунку.

На рис. 1 наведено основні функціональні можливості розроблюваної системи. Всього було визначено чотири актори: гість, учасник, організатор та журі. Гість є неавторизованим відвідувачем вебзастосунку; учасник, організатор та журі є зареєстрованими користувачами, функціональні можливості яких залежать від обраної під час реєстрації бажаної діяльності.

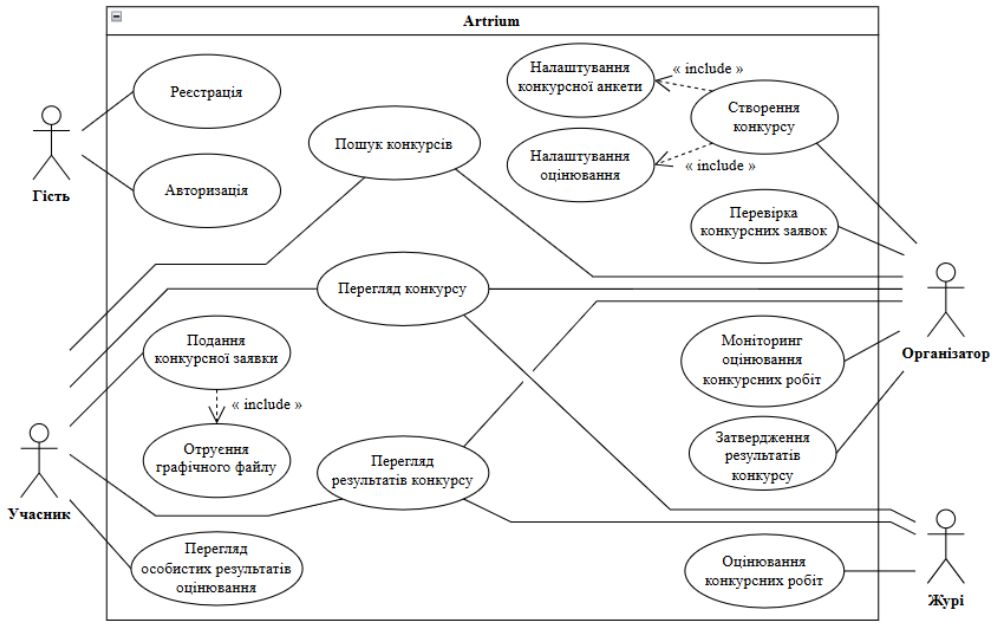


Рисунок 1 – Діаграма прецедентів розроблюваного вебзастосунку

Для забезпечення високої продуктивності вебзастосунку обрано архітектурний стиль Web-Queue-Worker, який дозволяє відокремити ресурсомістке отруєння зображень від обробки користувацьких запитів. На рис. 2 наведено архітектурну схему системи.

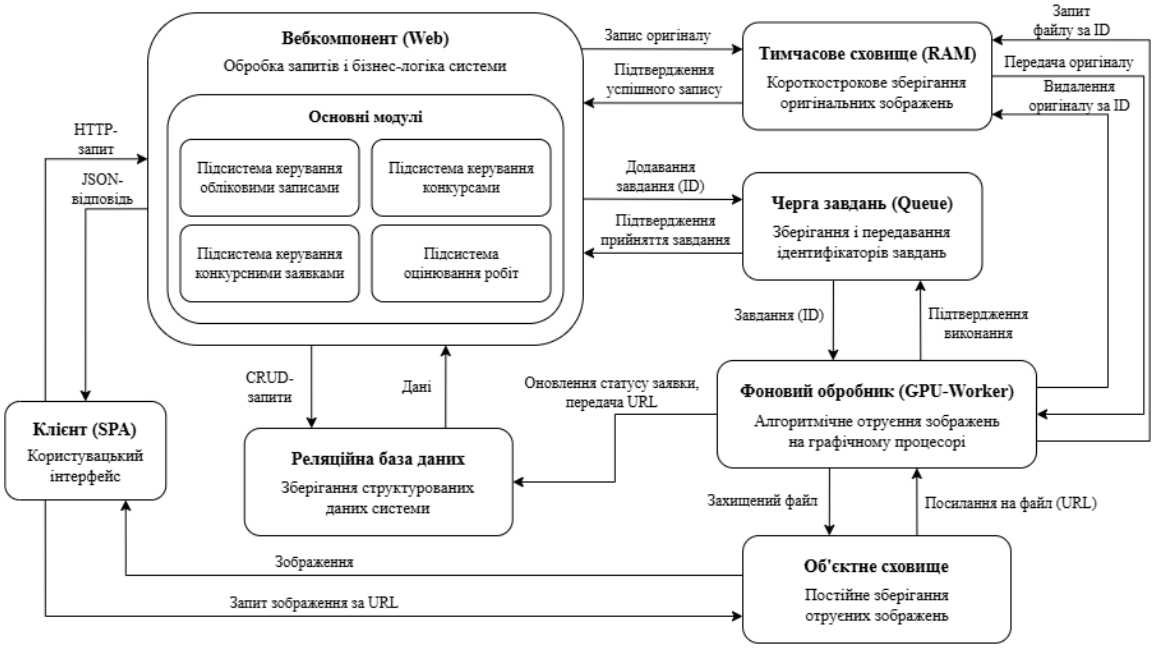


Рисунок 2 – Архітектурна схема розроблюваного вебзастосунку

Клієнт передає оригінал зображення вебкомпоненту, який зберігає його в тимчасовому сховищі оперативної пам’яті сервера та реєструє завдання в черзі. Фоновий GPU-обробник асинхронно виконує отруєння файлу, зберігає захищену копію в об’єктному сховищі та видаляє оригінал із пам’яті, після чого URL-посилання записується в реляційну базу даних.

Інтерфейс користувача розроблено за допомогою бібліотеки React. Серверний вебкомпонент реалізовано на базі фреймворку Java Spring Boot із використанням СУБД PostgreSQL, брокера повідомлень Redis та об’єктного сховища MinIO. Фоновий обробник реалізовано мовою Python, а для отруєння робіт інтегровано алгоритми AdvPoison та Mist-V2.

Висновки. Апробацію розробленого вебзастосунку здійснено шляхом порівняння тривалості виконання повного сценарію подання конкурсної заявки, що охоплює алгоритмічне отруєння зображень та супутні дії користувача, за трьох умов обробки графічних файлів: локально на центральному процесорі (CPU), локально на графічному процесорі (GPU) та на серверному фоновому обробнику. Результати експериментального дослідження підтвердили ефективність запропонованого рішення, забезпечивши скорочення загальних часових витрат користувача на подання конкурсної заявки у 2,76 рази порівняно з локальною обробкою на GPU та у 26,28 рази порівняно з локальною обробкою на CPU.

Використання інструментів ШІ. Під час написання тез доповіді було використано Gemini 3.1 Pro. Конкретна мета використання: стилістичне покращення тексту та перевірка на наявність помилок. Інструмент ШІ застосовувався під час написання розділів «Вступна частина», «Основна частина» та «Висновки».

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Throsby, D., & Petetskaya, K. (2017). *Making art work: An economic study of professional artists in Australia.* Strawberry Hills: Australia Council for the Arts. https://doi.org/10.13140/RG.2.2.18064.99847.

[2] Throsby, D., & Petetskaya, K. (2024). *Artists as workers: An economic study of professional artists in Australia.* Sydney: Creative Australia. https://creative.gov.au/research/artists-workers-economic-study-professional-artists-australia.

[3] Lovato, J., Zimmerman, J. W., Smith, I., Dodds, P., & Karson, J. L. (2024). Foregrounding artist opinions: A survey study on transparency, ownership, and fairness in AI generative art. *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 7(1), 905–916. https://doi.org/10.1609/aies.v7i1.31691.

[4] Salkowitz, R. (2026, August 23). Artists built a site to escape AI. Scrapers are coming for it anyway. *Forbes*. https://www.forbes.com/sites/robsalkowitz/2026/08/23/artists-built-a-site-to-escape-ai-scrapers-are-coming-for-it-anyway/.

[5] Shan, S., Cryan, J., Wenger, E., Zheng, H., Hanocka, R., & Zhao, B. Y. (2023). Glaze: Protecting artists from style mimicry by text-to-image models. In *32nd USENIX Security Symposium (USENIX Security 23)* (pp. 2187-2204). USENIX Association. https://doi.org/10.48550/arXiv.2302.04222.

[6] Shan, S., Ding, W., Passananti, J., Wu, S., Zheng, H., & Zhao, B. Y. (2024). Nightshade: Prompt-specific poisoning attacks on text-to-image generative models. In *2024 IEEE Symposium on Security and Privacy (SP)* (pp. 807–825). IEEE. https://doi.org/10.1109/SP54263.2024.00207.

Hurnytska V.O., Nesterov Y.M.

Web Application for Organizing Art Competitions with Protection of Artworks against Unauthorized Use for Training Generative Models

Abstract. *This paper analyzes web applications for organizing art competitions and identifies their main shortcomings. The development of a web application with an integrated server-side image poisoning feature is proposed to reduce the time required to submit competition entries and protect artworks from being used to train generative models.*

Keywords: *art competitions; web application development; data poisoning.*