

**А. В. Лень, канд. іст. наук; О. Й. Карабін, канд. пед. наук, доц.;
О. В. Вовкодав, канд. техн. наук; Р. І. Іваницький, канд. техн. наук;
А. М. Ясінський**

ПОРІВНЯЛЬНИЙ АНАЛІЗ ІНСТРУМЕНТІВ UML-МОДЕЛЮВАННЯ ДЛЯ ОСВІТНІХ ЦІЛЕЙ ПІДГОТОВКИ МАЙБУТНІХ ФАХІВЦІВ У ГАЛУЗІ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

У статті наведено порівняльний аналіз сучасних інструментів для побудови UML-діаграм, що використовуються під час проєктування та моделювання інформаційних систем. Дослідження охоплює широкий спектр програмних засобів – від класичних десктопних CASE-середовищ до сучасних онлайн-сервісів та легких редакторів, що інтегруються у веб-середовище та хмарні платформи. Проаналізовано низку ключових характеристик, серед яких: тип ліцензії, підтримувана версія UML, функціональні можливості, наявність оновлень, тип інтерфейсу, підтримка командної роботи, а також придатність до використання в освітньому процесі під час підготовки майбутніх фахівців у галузі інформаційних технологій. Особливу увагу приділено розмежуванню інструментів, що орієнтовані на повноцінне моделювання інформаційних систем, і тих, які більше підходять для візуалізації готових діаграм або демонстраційних матеріалів. У процесі дослідження також проведено огляд наукових джерел та сучасних публікацій, що висвітлюють тенденції використання UML-діаграм у сфері проєктування та цифрового моделювання систем. На основі отриманих даних здійснено узагальнення практичних аспектів застосування UML-інструментарію як в освітньому середовищі, так і в професійній розробці програмного забезпечення. У результаті роботи представлено порівняльну таблицю, яка полегшує вибір середовища відповідно до потреб користувача. Зроблено висновки щодо доцільного використання кожного з розглянутих інструментів у різних контекстах – зокрема в освітніх, дослідницьких і прикладних завданнях. Також сформульовано практичні рекомендації щодо впровадження ефективних інструментів моделювання у процес підготовки молодих фахівців, а також визначено перспективи для подальших досліджень у напрямку вивчення інтеграції UML-середовищ в освітні курси та розробки діаграм для конкретних прикладів інформаційних систем.

Ключові слова: інформаційні системи, проєктування, моделювання, UML, CASE-засоби, інформаційні технології, програмне забезпечення.

Вступ

У сучасному цифровому середовищі розробка інформаційних систем (ІС) стала одним із ключових напрямів у сфері комп'ютерних наук. Інформаційні системи забезпечують збір, зберігання, обробку та поширення даних, що є визначальним чинником забезпечення ефективної роботи організацій, бізнесу, державних структур та освітніх установ. Розробка їх потребує комплексного підходу, що охоплює аналіз вимог, проєктування, реалізацію, тестування та супровід.

Одним із найважливіших етапів розробки ІС є їх проєктування, яке дозволяє сформулювати чітке уявлення про структуру, функціонування, взаємодію компонентів і поведінку в різних сценаріях. Для ефективного проєктування застосовуються спеціалізовані мови моделювання, серед яких провідне місце займає UML (Unified Modeling Language). У свою чергу UML, як стандарт у сфері комп'ютерних наук, пропонує набір графічних нотацій, що базуються на єдиній метамоделі для опису та проєктування програмних систем. Утім, широко використовується як у промислових, так і в освітніх цілях, витісняючи менш універсальні або застарілі нотації. UML підтримується великою кількістю CASE-засобів та інструментів для візуального представлення, які значною мірою полегшують побудову, аналіз і супровід моделей систем [1].

Відзначимо, що в сучасних умовах розвитку цифрових технологій та зростаючих вимог до якості програмного забезпечення, моделювання інформаційних систем набуває особливого значення. Застосування візуальних мов моделювання, зокрема UML, сприяє формуванню структурованого бачення майбутньої системи, забезпечує ефективну комунікацію між усіма учасниками проектного процесу та виступає невід'ємною складовою під час створення надійних і масштабованих ІС.

Актуальність вивчення та практичне застосування UML є вагомим в освітньому процесі закладів вищої освіти, оскільки навички моделювання відіграють важливу роль для підготовки майбутніх фахівців комп'ютерних наук. Опанування створення діаграм UML не лише розвиває системне мислення, а й забезпечує універсальним інструментарієм для візуалізації та комунікації складних ідей у командній розробці. З цією метою програмні інструменти можуть значно полегшити застосування нотації UML, особливо для спільного створення складних діаграм й симуляції реальних проектних завдань між здобувачами освіти.

Аналіз останніх досліджень і публікацій

Питання використання мов моделювання та CASE-інструментів у процесі проектування ІС і підготовки майбутніх фахівців у галузі інформаційних технологій розглядається в працях як зарубіжних, так і українських дослідників, які зробили вагомий внесок у розвиток цієї сфери знань. Серед закордонних науковців UML-моделювання вивчали: Ю. Люй, К. А. Александру, Е. С. С. Фрейре, Г. К. Олівейра, М. Е. С. Гомес, Дж. Ромео, М. Ральянті, Ч. Надь, М. Ланца, А. Альмасабє, С. Луді, А. О. Ельфакі, Х. Коч, А. М. Ердоган, Ю. Баржаклі, С. Пекер [2 – 6].

Зокрема, у дослідженнях Ю. Люй, К. А. Александру представлено методологію та результати систематичного огляду інструментів для побудови UML-діаграм, а також запропоновано набір рекомендацій щодо розробки інструментів для навчальних дисциплін у галузі інформаційних технологій [2]. Науковці Е. С. С. Фрейре, Г. К. Олівейра та М. Е. С. Гомес провели аналіз тринадцяти інструментів з відкритим кодом, що дозволяють створювати п'ять UML-діаграм, які найчастіше використовуються фахівцями у сфері розробки програмного забезпечення. Метою дослідження було визначити, наскільки такі інструменти дотримуються рекомендованих практик моделювання, що сприяють створенню якісного програмного продукту [3].

Поряд з тим науковці Дж. Ромео, М. Ральянті, Ч. Надь і М. Ланца досліджували еволюцію та поточний стан використання UML у проєктах з відкритим вихідним кодом, проаналізувавши близько 13 тисяч репозиторіїв на GitHub. Автори виявили, що хоча UML тривалий час залишався малопоширеним серед розробників, останнім часом спостерігається певне відродження інтересу до нього – завдяки поширенню текстових форматів моделювання. Дослідники також наголошують на нових викликах та можливостях для подальшого розвитку UML у контексті відкритих програмних проєктів [4]. У роботах А. Альмасабє, С. Луді й А. О. Ельфакі проведено систематичний аналіз засобів моделювання, зосереджений на підтримці UML та інтерактивних методів навчання в курсах, пов'язаних із розробкою програмного забезпечення та проектуванням ІС. Науковці застосували метод метаєтнографії для синтезу якісних результатів і виявили низку популярних та ефективних інструментів, визначивши їхні сильні сторони, обмеження та доцільність використання в освітньому процесі [5]. Варто звернути увагу на праці Х. Коча, А. М. Ердогана, Ю. Баржаклі й С. Пекера, які вивчали застосування UML-діаграм у проектуванні програмного забезпечення. У межах свого огляду науковці проаналізували 128 найрелевантніших наукових праць, опублікованих у період 2000–2019 років. Дослідження показали, що UML-діаграми найчастіше використовуються для моделювання та проектування систем, а діаграми класів є найбільш поширеним типом, оскільки вони допомагають візуалізувати

структуру програмних застосунків і визначати їхні функціональні вимоги [6].

Серед українських науковців, які займаються проблематикою UML, можна виділити праці В. Авраменка, А. Авраменка, О. Коваленка, Л. Доброської, О. Зінов'євої, С. Постової, М. М'ястковської, О. Гудзоватої, В. Костирка, І. Артишук та інших [1, 7 – 11]. Так у роботах В. Авраменка, А. Авраменка [7], О. Коваленка, Л. Доброської [8] викладено основи теорії проєктування ІС. Дослідники приділили особливу увагу методам аналізу та проєктування з використанням мови UML, розкриваючи ключові питання життєвого циклу програмного забезпечення, архітектури ІС, а також сучасних технологій створення програмного забезпечення, включно з функціями та компонентами CASE-засобів. Науковиця О. Зінов'єва відзначила переваги застосування CASE-технології для проєктування програмного забезпечення, окреслила характеристики найбільш популярних CASE-засобів для проєктування ІС [1]. Дослідниця С. Постова описала онлайн-сервіси, які використовуються для проєктування ІС, підкреслюючи їхню доступність для швидкого набуття навичок у побудові діаграм [9]. У свою чергу М. М'ястковська розглянула низку CASE-засобів для побудови UML-діаграм, класифікуючи їх за типами ліцензій – платні та безкоштовні, що є важливим аспектом для вибору інструментів у сучасному контексті [10]. У нашому баченні, цікавий погляд на сучасні засоби проєктування інформаційних систем та можливості мови UML запропоновано в дослідженнях О. Гудзоватої, В. Костирка, І. Артишук. Науковці виокремили переваги UML як ефективного інструмента моделювання прикладних завдань і виступає чинником, що сприяє спрощенню процесу кодування та прискорення реалізації ІС [11].

Проведений аналіз засвідчує актуальність та важливість використання UML як універсального засобу моделювання у процесі проєктування ІС, що підтверджується як вітчизняними, так і зарубіжними дослідженнями, зосередженими на ефективності, доступності й освітньому потенціалі UML інструменту.

Мета і завдання статті

Метою статті є визначення можливостей ефективного застосування UML-діаграм у процесі проєктування ІС, з акцентом на їхню роль у підготовці майбутніх фахівців галузі інформаційних технологій. Особливу увагу приділено порівняльному аналізу програмних засобів UML-моделювання та особливостей їх застосування в освітньому процесі.

Завдання дослідження полягають у тому, щоб проаналізувати потенціал застосування UML-діаграм у процесі проєктування ІС; з'ясувати їхню роль у професійній підготовці майбутніх фахівців; дослідити наявні CASE-засоби та інструменти моделювання з урахуванням їхніх функціональних можливостей, зручності використання в освітньому процесі, підтримки командної роботи та відповідності навчальним цілям; зіставити різні типи інструментів – десктопні, онлайн та текстові редактори – з метою визначення їх переваг і недоліків для різних сценаріїв застосування; сформулювати практичні рекомендації щодо вибору та впровадження UML-середовищ у систему підготовки ІТ-фахівців; а також окреслити перспективні напрями подальших досліджень щодо інтеграції UML-інструментів в освітню практику.

Основна частина

Основна частина статті присвячена аналізу сучасних інструментів моделювання, що можуть бути використані в освітньому процесі закладів вищої освіти під час вивчення проєктування інформаційних систем. Проте, перш ніж перейти до безпосереднього огляду CASE-засобів, доцільно розглянути види UML-діаграм, які становлять основу візуального моделювання програмних систем.

Слід обумовити, що уніфікована мова моделювання (UML) є міжнародним стандартом, який розробляється та підтримується консорціумом Object Management Group (OMG). Утім,

зазначимо, що в різних джерелах наводиться різна кількість UML-діаграм, що зумовлено особливостями інтерпретації стандарту та його версій. Актуальна версія стандарту UML 2.5.1 була опублікована у грудні 2017 року. Відповідно до цієї версії, UML включає 14 типів діаграм, які поділяються на три основні групи: діаграми структури, діаграми поведінки та діаграми взаємодії. У таблиці 1 наведено перелік усіх UML-діаграм із класифікацією за групами, а також їхні назви українською та англійською мовами [12].

Таблиця 1

Класифікація UML-діаграм

№	Український варіант		Англійський варіант	
1.	Діаграма класів	Структурні діаграми	Structure Diagrams	Class Diagram
2.	Діаграма компонентів			Component Diagram
3.	Діаграма об'єктів			Object Diagram
4.	Діаграма композитної структури			Composite Structure Diagram
5.	Діаграма розгортання			Deployment Diagram
6.	Діаграма пакетів			Package Diagram
7.	Діаграма профілю			Profile Diagram
8.	Діаграма діяльності	Поведінкові і діаграми	Behavior Diagrams	Activity Diagram
9.	Діаграма варіантів використання			Use Case Diagram
10.	Діаграма станів			State Machine Diagram
11.	Діаграма послідовності	Діаграми взаємодії	Interaction Diagrams	Sequence Diagram
12.	Діаграма комунікації			Communication Diagram
13.	Діаграма огляду взаємодії			Interaction Overview Diagram
14.	Діаграма синхронізації			Timing Diagram

UML забезпечує зручний інструментарій для візуального опису архітектури, поведінки та взаємодії компонентів ІС за допомогою різних типів діаграм. UML-діаграми є потужним засобом візуалізації, який дозволяє представити загальний і водночас структурований вигляд функціонування програмного забезпечення або інформаційної системи. Вони базуються на наборі стандартизованих елементів, що забезпечують єдиний підхід до моделювання різних аспектів системи – від її структури до поведінки. А також, дозволяє формалізувати вимоги, структурувати логіку системи та зменшити ризики під час реалізації проєкту. Означені діаграми допомагають як розробникам, так і замовникам краще зрозуміти ключові процеси, взаємозв'язки та функції майбутнього продукту. Застосування UML-діаграм значно спрощує процес проєктування, сприяє виявленню потенційних проблем на ранніх етапах і забезпечує супровід документації. Саме завдяки цим властивостям UML активно використовується не лише в промислових проєктах, а й в освітньому процесі навчальних дисциплін із моделювання та проєктування ІС [7].

Візуальні засоби структурного та об'єктно-орієнтованого аналізу відіграють важливу роль у процесі проєктування інформаційних систем. Проте ефективне застосування їх було б утрудненим без відповідної інструментальної підтримки – адже ручна побудова моделей потребує багато часу, схильна до помилок і ускладнює внесення змін. Саме тому виникла потреба в спеціалізованих програмних засобах – CASE-засобах, які автоматизують процес аналізу, проєктування та документування програмного забезпечення [1]. Розглянемо низку CASE-засобів, які дозволяють будувати UML-діаграми та мають потенціал для використання під час навчання у закладах освіти.

У рамках нашого дослідження зазначимо, що на сучасному ринку CASE-засобів та

інструментів для моделювання представлено широке розмаїття програмних рішень. Їх можна класифікувати за різними критеріями, зокрема за способом розгортання та доступу. За цим принципом виділяють три основні категорії: десктопні застосунки, що вимагають локальної інсталяції; онлайн (хмарні) сервіси, доступ до яких здійснюється через веб-браузер та які функціонують на віддалених серверах; а також гібридні рішення, що поєднують обидва підходи.

Попри зовнішню схожість, принципово важливо розрізняти інструменти, призначені виключно для малювання UML-діаграм (тобто графічні редактори з набором фігур), та повноцінні середовища моделювання (CASE-засоби), які працюють із логічно цілісною моделлю системи, підтримують її узгодженість і забезпечують інжиніринг коду.

Enterprise Architect від компанії Sparx Systems здобув широке визнання серед фахівців та науковців завдяки своїй багатофункціональності, що постійно оновлюється та вдосконалюється. Серед ключових переваг *Enterprise Architect* варто виділити його вичерпну підтримку всіх видів діаграм у нотації UML, що дозволяє охопити всі аспекти системного проєктування. Інструмент також надає потужні можливості інженерії коду, підтримуючи широкий спектр сучасних мов програмування (зокрема C++, Java, C#, Python, PHP, ActionScript та ін.). Включає функції прямої та зворотної інженерії, а також забезпечення синхронізації програмного коду та елементів моделей, що є критично важливим для підтримки узгодженості між проєктом та його реалізацією. Крім того, *Enterprise Architect* ефективно працює з моделюванням баз даних, що сприяє повторному використанню перевірених архітектурних рішень. Постійне оновлення та наявність вичерпної документації додатково підкреслюють його статус як комплексного, орієнтованого на підприємства рішення для повного життєвого циклу розробки програмного забезпечення, що робить його одним з найбільш універсальних і потужних інструментів [1].

Enterprise Architect надає повнофункціональну пробну версію для ознайомлення з її широкими можливостями. Проте, для повноцінного та довготривалого використання в освітньому процесі для здобувачів освіти й освітніх установ передбачена спрощена процедура отримання спеціальних освітніх ліцензій.

IBM Rational Rose заслужено вважається класичним стандартом у сфері UML-проєктування, що підтримує повний цикл розробки – від бізнес-аналізу до зворотного проєктування програмного коду. Існує декілька версій продукту, орієнтованих на різні аспекти проєктування: бази даних, бізнес-моделювання, системи реального часу тощо [13].

Borland Together ControlCenter – це інтегроване середовище, яке поєднувало в собі підтримку UML, патернів проєктування, гнучких методологій та функції автоматизованого розгортання систем. Особливість цього середовища орієнтація на командну розробку складних корпоративних систем [1].

Варто зазначити, що *IBM Rational Rose* та *Borland Together ControlCenter* є одними з найвідоміших та історично впливових інструментів для UML-моделювання. Ці десктопні програмні комплекси були розроблені в час активного впровадження UML у промислове проєктування програмного забезпечення і свого часу вважались одними з найпотужніших інструментів на ринку. Сьогодні ці продукти не оновлюються і вважаються застарілими, однак їх ще можна використовувати для базового моделювання або в навчальних цілях.

Dia Diagram Editor є вільним кросплатформним редактором діаграм, який може бути легко встановлений та використаний як самостійний застосунок. Серед беззаперечних переваг для українських користувачів є повна підтримка української мови в інтерфейсі. Цей інструмент відзначається своєю доступністю, легкістю та універсальністю, дозволяючи створювати широкий спектр візуалізацій: від блок-схем алгоритмів програм та статичних структур UML до діаграм баз даних. Хоча темпи його оновлень можуть бути менш інтенсивними порівняно з комерційними CASE-засобами, стабільність *Dia*, його простота у використанні та відсутність платної ліцензії роблять його надзвичайно ефективним вибором

для швидкого створення загальних діаграм та для освітніх цілей, де не потрібен складний функціонал професійних систем моделювання. Він ідеально підходить для початкового вивчення принципів діаграмування завдяки своїй невибагливості до ресурсів та інтуїтивно зрозумілому інтерфейсу [10].

Серед сучасних безкоштовних CASE-засобів, які можуть слугувати альтернативою комерційним рішенням, особливе місце посідає *Modelio* – потужне десктопне середовище з відкритим вихідним кодом. Підтримує всі основні стандарти моделювання, включаючи UML 2.x, SysML та BPMN. *Modelio* також надає інструменти для моделювання архітектури підприємств із підтримкою таких фреймворків, як ArchiMate та MODAF, що дозволяє працювати з проєктами на різних рівнях абстракції [3]. Модульна структура та підтримка Java API забезпечують гнучке налаштування під потреби конкретного проєкту. Завдяки активній спільноті, відкритій базовій ліцензії та широкому функціоналу, *Modelio* є доступним і ефективним інструментом для навчання та професійного використання без обмежень комерційних ліцензій [11].

Наступним кросплатформним програмним забезпеченням, що заслуговує на увагу, є *StarUML*. Інструмент вирізняється підтримкою стандартів моделювання, зокрема UML 2.x, а також можливістю генерації коду для таких мов програмування, як Java, C#, C++ і Python, що реалізується через плагіни (деякі з відкритим вихідним кодом).

Важливою перевагою *StarUML* є зберігання моделей у форматі JSON, що спрощує автоматизацію роботи з ними за допомогою інтерфейсу командного рядка (CLI). Окрім того, застосунок легко розширювати за допомогою сторонніх модулів, а моделі можна експортувати у вигляді HTML-документації для зручної спільної роботи. Проте варто зауважити, що *StarUML* є платним програмним продуктом, не має безкоштовної повнофункціональної версії та не підтримує вебдоступу, що може обмежувати його застосування в освітньому середовищі [14].

Серед безкоштовних CASE-засобів варто згадати *ArgoUML* – одне з перших середовищ UML-моделювання з відкритим вихідним кодом. Завдяки реалізації на Java, воно є кросплатформним і працює на будь-якій ОС. *ArgoUML* розповсюджується за відкритою ліцензією Eclipse Public License 1.0, що забезпечує його повну безкоштовність [15].

Однак інструмент орієнтований на застарілу специфікацію UML 1.4 та підтримує лише базовий набір діаграм. Відсутність підтримки UML 2.x, обмежений функціонал і відсутність активного розвитку обмежують можливості *ArgoUML* у контексті сучасного навчання й професійного моделювання.

Microsoft Visio, хоча й не є повноцінним CASE-середовищем, пропонує шаблони UML-діаграм і дозволяє створювати базові візуалізації архітектури програмних систем. Перевагою тут є знайоме середовище та інтеграція з іншими продуктами Microsoft [1]. Водночас *Visio* є платним програмним забезпеченням і не завжди входить до стандартного пакета Microsoft Office, що може обмежувати його доступність у закладах вищої освіти.

UMLet є відкритим інструментом для створення UML-діаграм, розробленим на базі Java, що робить його кросплатформним. Особливістю цього середовища є мінімалістичний інтерфейс користувача, який дозволяє змінювати та налаштовувати фігури за допомогою простих текстових кодів форматування, що спрощує швидке створення діаграм без надлишку значків чи діалогових вікон. *UMLet* підтримує експорт діаграм у різні формати зображень та документів (EPS, JPG, SVG, PDF), а також дозволяє користувачам створювати власні елементи та шаблони. Варто зазначити, що це середовище також має онлайн-версію під назвою (*UMLetino*), яка дозволяє створювати діаграми безпосередньо у веб-браузері, розширюючи доступність інструменту [16].

Унікальним інструментом, що кардинально відрізняється від традиційних графічних редакторів є *PlantUML*. Середовище дозволяє створювати широкий спектр діаграм за допомогою спеціальної простої текстової мови розмітки. Цей генератор діаграм функціонує

як гібридне рішення, доступне як для локального використання на комп'ютері, так і через веб-браузер. Завдяки такому підходу, PlantUML ідеально підходить для швидкого прототипування, ефективного управління версіями діаграм як коду, а також для їхньої легкої інтеграції в автоматизовані процеси розробки [3].

PlantUML розроблений мовою програмування Java, що забезпечує його кросплатформність. Він функціонує як гібридне рішення: його можна використовувати локально на персональному комп'ютері, або ж за допомогою численних онлайн-сервісів, що генерують діаграми безпосередньо у веб-браузері. Застосунок є відкритим програмним забезпеченням і повністю безкоштовна для використання [4].

Нарівні з PlantUML, значну популярність здобуває *Mermaid* – ще один інструмент, що втілює концепцію «діаграми як код». Подібно до PlantUML, Mermaid дозволяє створювати візуальні діаграми, описуючи їх за допомогою простої та інтуїтивної текстової мови розмітки, що є альтернативою традиційному графічному редагуванню.

Ключова відмінність Mermaid полягає у його JavaScript-орієнтованій реалізації та синтаксисі, що наближений до Markdown, що робить його зручним для інтеграції у вебсередовища, технічну документацію та інструменти розробника [4].

Хоча Mermaid підтримує обмежену кількість UML-діаграм і не дотримується повною мірою офіційного стандарту UML, він забезпечує базовий функціонал для побудови діаграм ключових типів. Завдяки своїй легкості, простоті використання та високій інтеграції Mermaid є ефективним вибором для швидкої візуалізації та спільної роботи безпосередньо в текстових документах.

Наступне середовище – *Visual Paradigm*, яке є одним із найбільш потужних, комплексних та багатофункціональних комерційних CASE-засобів для моделювання та архітектурного проєктування. Це середовище позиціонується як гібридне рішення, пропонуючи як повноцінну десктопну версію, так і гнучку онлайн-платформу, що забезпечує можливості для спільної роботи та доступності з будь-якого веббраузера.

Visual Paradigm надає вичерпну підтримку всіх типів діаграм UML відповідно до актуальних стандартів, що дозволяє детально описувати структуру й поведінку системи та має потужні можливості автоматичної генерації вихідного коду безпосередньо з UML-моделей. Підтримує широкий спектр популярних мов програмування, що дозволяє швидко створювати каркаси застосунків і зменшувати кількість ручних помилок. Застосунок здатний аналізувати наявний вихідний код й автоматично генерувати з нього UML-моделі та діаграми.

Однією з найцінніших функцій є можливість підтримувати двосторонню синхронізацію між програмним кодом і моделлю. Зміни, внесені в код, можуть бути автоматично відображені в моделі, і навпаки, забезпечуючи постійну узгодженість між проєктом та його реалізацією. Наявні потужні інструменти для генерації налаштованих звітів у різних форматах (HTML, PDF, DOC, Excel), що є важливим для документування та комунікації [17].

Visual Paradigm є комерційним програмним забезпеченням, що пропонується за різними типами ліцензій та підписками, що відрізняються набором функцій. Однак, розробник надає спеціальні освітні ліцензії зі спрощеною процедурою отримання або значними знижками для використання у закладах вищої освіти.

Visual Paradigm Online – це графічна та браузерна версія, що робить її надзвичайно доступною та зручною для швидкого створення діаграм і спільної роботи. Вона є повноцінним хмарним середовищем для моделювання, а не просто інструментом для малювання, і дозволяє створювати UML-діаграми, де елементи є частиною логічної моделі системи. Ще однією перевагою є підтримка текстових мов опису діаграм – PlantUML та Mermaid, що дає змогу швидко генерувати схеми з коду, автоматизувати документування та інтегрувати моделі у середовища розробки [17].

Таким чином, онлайн-версія виступає як централізоване хмарне сховище та платформа для спільної роботи над моделлю, а десктопна версія є «робочою станцією» з повним функціоналом для глибокої інженерії та взаємодії з кодом.

Наступним онлайн-ресурсом для створення діаграм є *Draw.io*, що надає широкий набір фігур, включно з елементами UML, але основний фокус зроблений на графічне відображення та візуальну організацію інформації. Це середовище повністю безплатне, підтримує українську мову та легко інтегрується з хмарними сховищами. Підтримує спільне редагування діаграм у реальному часі, що ідеально підходить для командної роботи та мозкових штурмів під час навчання [10].

Lucidchart – це один із провідних комерційних онлайн-сервісів для створення діаграм, відомий своїм інтуїтивно зрозумілим інтерфейсом та широкими можливостями для візуалізації інформації. Як і *draw.io*, *Lucidchart* є насамперед потужним інструментом для візуального представлення UML-діаграм. Окрім усіх типів UML, сервіс пропонує велику бібліотеку шаблонів для блок-схем, мережесхем, ER-діаграм, організаційних структур, ментальних карт, макетів інтерфейсів та багато іншого. Онлайн-сервіс має глибоку інтеграцію з екосистемами Google Workspace та Microsoft 365, а також підтримує спільну роботу в реальному часі, що дозволяє кільком користувачам одночасно працювати над однією діаграмою. Він надає безкоштовний тарифний план, який дозволяє створювати до трьох документів із максимально 60 об'єктами в кожному – цього зазвичай достатньо для навчальних потреб або ознайомлення з функціоналом [18].

Таким чином, розглянуті інструменти відображають широку палітру підходів до створення UML-діаграм – від простих графічних редакторів до повноцінних CASE-середовищ, здатних забезпечити повний цикл розробки програмного забезпечення. Для зручності сформовано зведену інформацію щодо порівняння можливостей інструментів з користувацькими характеристиками кожного з них та наведено в таблиці 2.

Таблиця 2

Порівняльна характеристика програмних середовищ для UML-моделювання

Назва середовища	Тип версії	Покликання для доступу	Версія підтримки UML	Тип ліцензії
Enterprise Architect	desktop	https://sparxsystems.com/	UML 2.x	комерційна (знижка для освіти)
IBM Rational Rose	desktop	https://www.ibm.com/support/pages/ibm-rational-rose-enterprise-7004-ifix001	UML 2.0	комерційна
Borland Together ControlCenter	desktop	https://borland-together-controlcenter-v-6-2.software.informer.com/download/	UML 2.0	комерційна
Microsoft Visio	desktop	https://www.microsoft.com/uk-ua/microsoft-365/visio/flowchart-software	UML 2.0 (частково)	комерційна
Dia Diagram Editor	desktop	http://dia-installer.de/	UML 1.x (часткова)	відкрита
Modelio	desktop	https://www.modelio.org/	UML 2.x	відкрита (базова)
StarUML	desktop	https://staruml.io/	UML 2.x	комерційна (пробний період, знижка для освіти)
ArgoUML	desktop	argouml-tigris-org.github.io	UML 1.4	відкрита
UMLet	desktop	https://umlet.com/	UML 2.x (обмежено)	відкрита
Umletino	online	https://www.umletino.com/umletino.html	UML 2.x (обмежено)	відкрита
PlantUML	desktop	https://plantuml.com/	UML 2.x	відкрита
Mermaid	online	https://www.mermaidchart.com/	UML (обмежена)	відкрита

Продовження таблиці 2				
Visual Paradigm	desktop	https://www.visual-paradigm.com/	UML 2.x	комерційна (пробний період, знижка для освіти)
Visual Paradigm Online	online	https://online.visual-paradigm.com/	UML 2.x	відкрита
Draw.io	online	app.diagrams.net	UML 2.x	відкрита
Lucidchart	online	https://lucid.app/	UML 2.x	комерційна (три документи безплатно)

Відзначимо, що вибір інструмента значною мірою залежить від освітньої або професійної мети, вимог до функціональності, доступності та типу ліцензування. Завдяки великій кількості як комерційних, так і безкоштовних рішень, користувачі мають змогу обрати оптимальний варіант для моделювання, що найкраще відповідає їхнім потребам та досвіду.

Висновки

У процесі дослідження було здійснено порівняльний аналіз інструментів UML-моделювання і виявлено значну різноманітність програмних засобів, що доступні для проєктування та документування інформаційних систем. Детально розглянуто широкий спектр рішень – від класичних десктопних CASE-середовищ до гнучких онлайн-редакторів і гібридних платформ. Ключовою відмінністю, що була зазначено, є розмежування між засобами, призначеними для глибокого семантичного моделювання та інструментами, що зосереджені переважно на візуалізації й швидкому діаграмуванні. За результатами аналізу підкреслено критичну важливість свідомого й обґрунтованого вибору інструментарію відповідно до конкретних освітніх потреб, етапу проєктування та компетентностей майбутніх фахівців.

Із огляду на вимоги до освітнього процесу закладів вищої освіти, доступності й зручності використання, рекомендовано використовувати середовище Modelio як базовий засіб моделювання UML-діаграм у вищій освіті. Зазначене середовище надає вичерпні можливості для практики з усіма типами UML-діаграм, розуміння архітектури систем та освоєння складних концепцій моделювання без обмежень комерційних ліцензій, що є ідеальним для використання в освітньому процесі майбутніх фахівців комп'ютерних наук.

Для підтримки спільної роботи здобувачів освіти й створення візуалізацій у командних проєктах доцільно застосовувати онлайн-середовища Visual Paradigm Online, Draw.io та Lucidchart, які забезпечують створення діаграм через веб-браузери, підтримку шаблонів і можливість інтеграції з хмарними сховищами.

Також з'ясовано, що доцільно впроваджувати текстові інструменти моделювання, зокрема PlantUML та Mermaid, які дозволяють поглиблювати розуміння структури діаграм, генерувати візуалізації на основі коду та експериментувати з автоматизацією побудови схем. Їх інтеграція з сучасними платформами та підтримка генерації за допомогою штучного інтелекту дозволяє створювати перспективу для інноваційного навчання та розвитку аналітичного мислення у майбутніх фахівців.

Перспективним напрямом подальших досліджень вважаємо розробку методичних підходів до використання UML-середовищ у процесі створення прикладних моделей інформаційних систем. Зокрема, це передбачає практичне конструювання різних типів UML-діаграм на основі реальних кейсів, аналіз їхньої ефективності для опису архітектури, функціонування та взаємодії компонентів систем. Також доцільним є вивчення відмінностей у візуальному та текстовому представленні моделей, що може сприяти кращому розумінню складних концепцій та розвитку навичок аналітичного мислення у майбутніх фахівців комп'ютерних наук.

СПИСОК ЛІТЕРАТУРИ

1. Зінов'єва О. Г. Використання CASE-засобів для проєктування інформаційних систем. *Українські студії в європейському контексті : збірник наукових праць*. 2023. № 7. С. 220–227.
2. Lu Y., Alexandru C. A. Systematic review of UML diagramming software tools for higher education software engineering courses, in Proc. *United Kingdom Ireland Comput. Educ. Res. (UKICER) Conf.*, Sep, 2023. P. 1–7.
3. Freire E. S. S., Oliveira G. C., Sousa Gomes M. E. Analysis of open-source CASE tools for supporting software modeling process with UML. In *Proceedings of the XVII brazilian symposium on software quality*. Oct, 2018. P. 51–60.
4. UML is Back. Or is it? Investigating the Past, Present, and Future of UML in Open Source Software / J. Romeo et al. *ICSE 2025 47th International Conference on Software Engineering*, 2025. URL: <https://www.inf.usi.ch/phd/raglianti/publications/Romeo2025a.pdf> (date of access: 02.05.2025).
5. Almasabe A., Ludi S., Elfaki A. O. Scrutinizing UML Teaching and Learning Modeling Tools. *International Journal of Computer Science & Network Security*. 2024. № 24 (12). P. 179–187.
6. UML diagrams in software engineering research: a systematic literature review / H. Koc et al. *7th International Management Information Systems Conference*, 9–11 December 2020. Proceedings 2021. № 74 (1), 13. P. 1 – 8.
7. Авраменко В. С., Авраменко А. С. Проєктування інформаційних систем : навчальний посібник. Черкаси : Черкаський національний університет ім. Б. Хмельницького, 2017. 434 с.
8. Проєктування інформаційних систем: Загальні питання теорії проєктування ІС (конспект лекцій) : навчальний посібник для студентів спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського; уклад. : О. С. Коваленко, Л. М. Добровська. Київ : КПІ ім. Ігоря Сікорського, 2020. 192 с.
9. Постова С. А. Онлайн сервіси проєктування інформаційних систем. *Digital transformation and technologies for sustainable development all branches of modern education, science and practice*. 2023. Т. 4. С. 64–67.
10. Мясковська М. О. Проєктування інформаційних систем. *Наукові праці Кам'янець-Подільського національного університету імені Івана Огієнка: збірник за підсумками звітної наукової конференції викладачів, докторантів і аспірантів*. Кам'янець-Подільський: Кам'янець-Подільський національний університет імені Івана Огієнка, 2023. Вип. 22. С. 665–668.
11. Гудзовата О. О., Костирко В. І., Артишук І. В. Використання уніфікованої мови візуального моделювання UML (Unified Modeling Language) як інструменту підтримки проєктування інформаційних систем. *Підприємництво і торгівля*, 2019. Вип. 24. С. 108–114.
12. Object Management Group. OMG Unified Modeling Language (OMG UML), Version 2.5.1. December 2017. URL: <https://www.omg.org/spec/UML/2.5.1/> (date of access: 05.05.2025).
13. Forward and reverse engineering using uml with rational rose and object-oriented programming language / B. Lakshmi et al. *Journal of Theoretical and Applied Information Technology*. 2025. № 103 (3). P. 1038–1044.
14. StarUML. URL: <https://staruml.io/> ((date of access: 10.05.2025).
15. AgroUML. URL: <https://argouml-tigris-org.github.io/tigris/argouml/> (date of access: 12.05.2025).
16. UMLet. URL: <https://umlet.com/> (date of access: 15.05.2025).
17. Луценко Г. В. Програмне середовище Visual Paradigm у навчанні технологій проєктної діяльності студентів-інженерів. *Інформаційні технології і засоби навчання*. 2021. Т. 83. № 3. С. 208–225.
18. Lucidchart. URL: <https://www.lucidchart.com/> (date of access: 15.05.2025).

Стаття надійшла до редакції 13.06.2025.

Стаття пройшла рецензування 02.09.2025.

Лень Андрій Володимирович – канд. іст. наук, асистент кафедри інформатики та методики її навчання.

Карабін Оксана Йосифівна – канд. пед. наук, доцент кафедри інформатики та методики її навчання.

Вовкодав Олександр Валерійович – канд. техн. наук, викладач кафедри інформатики та методики її навчання.

Іваницький Роман Іванович – канд. техн. наук, асистент кафедри інформатики та методики її навчання.

Ясінський Андрій Михайлович – здобувач другого (магістерського) рівня вищої освіти освітньо-професійної програми «Комп'ютерні науки».

Тернопільський національний педагогічний університет імені Володимира Гнатюка.