



## ХАРАКТЕРИСТИКИ ОНЛАЙН-СЕРЕДОВИЩ ДЛЯ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**Твердохліб Ігор,**  
*доцент кафедри інформаційних технологій і програмування,  
кандидат педагогічних наук, доцент,  
Український державний університет імені Михайла Драгоманова  
i.a.tverdokhlib@npu.edu.ua*

**Оніщенко Сергій,**  
*здобувач наукового ступеня,  
Український державний університет імені Михайла Драгоманова,  
s.m.onishchenko@npu.edu.ua*

На сьогоднішній день процес розробки та створення програмних засобів стрімко розвивається і набуває значних змін завдяки швидкому розвитку технологій. До основних тенденцій, які можна виділити у створенні десктопних, мобільних і веб-додатків, можна виокремити кросплатформність, мікросервісну архітектуру, розвиток хмарних технологій, широке використання Low-Code/No-Code платформ, застосування штучного інтелекту та машинного навчання, зростання популярності технології Progressive Web Apps (PWA), використання контейнерів (Docker) та систем оркестрації (Kubernetes) для забезпечення надійності, масштабованості та гнучкості розгортання додатків, стрімкий розвиток інтернету речей (IoT). Ці інновації не лише покращують якість і функціональність програмних продуктів, але й роблять їх більш доступними для широкого кола користувачів.

Зокрема, важливим аспектом сучасної розробки є використання хмарних сервісів та платформ, які забезпечують зручні інструменти для створення, тестування та розгортання додатків. Наприклад, такі платформи, як AWS, Azure,

і Google Cloud, пропонують розробникам потужні інструменти для роботи з додатками різної складності та масштабу. Це дозволяє скоротити час на розробку та забезпечити високу якість кінцевого продукту.

В умовах сьогодення альтернативними (до інтегрованих середовищ розробки) засобами навчання програмування у закладах вищої освіти можуть бути використані онлайн-сервіси, які надають інтерактивне середовище для розробки та тестування комп'ютерних програм без необхідності встановлення будь-якого спеціального програмного забезпечення на комп'ютері користувача. Таких онлайн-сервісів існує багато, і перед тим, як обрати один із них за основний засіб навчання програмування, варто розглянути і ознайомитися з основними характеристиками, якими цей засіб має володіти, і які необхідні для якісного навчання.

1. Насамперед, це має бути повноцінне інтегроване середовище розробки з підтримкою різноманітних функцій щодо створення, збереження, налагодження та компіляції проектів різних типів та потрібної складності.
2. Підтримка сучасних найпоширеніших мов програмування.
3. Онлайн виконання коду.
4. Наявність колаборативних засобів.
5. Хмарне зберігання проектів.
6. Наявність відповідної документації щодо використання даного середовища, а також посилань на ресурси зі специфікаціями мов програмування.
7. Інтеграція з Git.
8. Підтримка мобільності.

З урахування проведеного аналізу вітчизняних досліджень та власного досвіду використання різних онлайн-середовищ розробки програмного

забезпечення можна зробити висновок про те, що перелічені характеристики мають онлайн-середовища Replit, GitHub Codespaces, GitLab Web IDE, Visual Studio Code Online, які в умовах змішаного навчання можна рекомендувати як засіб навчання програмування.

Для якісного навчання програмування обраною мовою варто окремо присвятити достатній час для ознайомлення та характеристики процесу створення програмного засобу. Зрозуміло, що потрібно детально ознайомитися із засобами та функціями самого середовища, його призначення. Слід приділити увагу структурі проекту та місця його розміщення. Оцінити та перевірити роботу текстового редактора. Також потрібно дослідити етапи тестування програмного засобу, як здійснюється виведення результатів роботи на різних виконуваних версіях робочого пристрою – персональних комп'ютерах, планшетах, смартфонах.

Важливим є розгляд процесу трансляції створеного програмного коду за розробленим алгоритмом у машинний код, що безпосередньо буде виконуватися в даній операційній системі. І тут варто приділити увагу таким поняттям як компіляція та інтерпретація програмного коду.

Для пояснення поняття інтерпретації досить вдалим прикладом можуть бути процеси виконання програмних кодів програмами-браузерами, які за javascript-кодом, розміщеному у html-документі (який є звичайним текстовим файлом), або у окремому js-файлі, на який є посилання у тому ж html-документі, відтворюють сучасні досить складні динамічні веб-сторінки. Іншим вдалим прикладом інтерпретації може бути серверний скрипт – програмний код, створений на мові Python. В обох випадках текстові програми подаються на вхід інтерпретаторам, які перетворюють на машинний код кожен інструкцію окремо, здійснюючи її лексичний та синтаксичний аналіз, а також за необхідністю її

рекурсивний розбір, і одразу її виконують. Це є одним із способів реалізації кросплатформності програмного забезпечення. І до речі, це суттєво сприяло створенню та використанню програм-браузерів, різних онлайн-інтерактивних середовищ для розробки, створення та тестування комп'ютерних програм.

При розгляді поняття компіляції слід приділити увагу процесам, за допомогою яких здійснюється повна або часткова компіляція тексту програми. При повній компіляції тексту програмного коду створюється виконуваний файл для виконання на певній платформі. Програму, створену за такою технологією, ще називають десктопною версією програмного засобу. Тобто це програма, призначена для роботи на персональному комп'ютері і потребує відповідного процесу інсталяції. При частковій компіляції текст програмного коду компілюється у так званий байт-код, який уже на певній платформі буде виконуватися спеціально для неї розробленою віртуальною машиною.

Розуміння окреслених процесів створення програмних засобів є важливим при навчанні програмування із використанням онлайн-інтерактивних середовищ розробки. Так, для створення програмних додатків різного призначення, різних версій типу веб-додатку або десктопного додатку, добираються відповідні мови програмування, що в свою чергу потребує відповідного середовища розробки. При навчанні програмування певною мовою варто розуміти, що відбувається на стороні сервера при надсиланні йому запиту на виконання набраної програми, яким способом отримується результат виконання програми у відповідній частині екрану, які повідомлення можуть бути отримані у випадку невиконання програми і як це слід виправити.

Користуючись онлайн-сервісами, які надають інтерактивне середовище для розробки та тестування комп'ютерних програм, студенти отримують знання з алгоритмізації та програмування, знання набору сучасних мов програмування,

практичні навички роботи з сучасними інструментами та технологіями, опрацьовують засоби віддаленої роботи над проектом, отримують досвід командної роботи. Все це підготує студентів до реальних умов праці в ІТ-секторі.

### *Література*

1. Ковтанюк М.С. Переваги використання онлайн-середовища розробки Replit під час вивчення Програмування. Комп'ютерні технології: інновації, проблеми, рішення : тези доп. IV Всеукр. наук.-техн. конф., 18–20 листопада 2021. Житомир : Житомирська політехніка, 2021. С. 97–98. URL: <https://dspace.udpu.edu.ua/bitstream/123456789/14755/2/%D0%9A%D0%BE%D0%B2%D1%82%D0%B0%D0%BD%D1%8E%D0%BA%20%D0%96%D0%B8%D1%82%D0%BE%D0%BC%D0%B8%D1%80.pdf>

2. Шевченко Г.В. Переваги використання онлайн-середовища розробки «Replit» для вивчення мови програмування «Python» у закладах освіти під час дистанційного навчання. «Вересень»,1(96), 2023, с. 141-152. URL: <https://doi.org/10.54662/veresen.1.2023.11>

3. Величко В.Є., Ананьєв М.С., Іванюк С.В., Шеремет М.М. Електронне навчання у процесі вивчення програмування: Інформатика та методика її навчання. Збірник наукових праць фізико-математичного факультету ДДПУ, 2023, 13: 54-61. URL: <https://doi.org/10.31865/2413-26672415-3079132023295330>

4. Idea to software, fast. URL: <https://replit.com/>

5. Start coding instantly with Codespaces. URL: <https://github.com/features/codespaces>

6. GitLab Web IDE. URL: [https://docs.gitlab.com/ee/user/project/web\\_ide/](https://docs.gitlab.com/ee/user/project/web_ide/)

7. Visual Studio Code Online. URL: <https://vscode.dev/>