

УДК 004.415:005.551

DOI: 10.15587/2519-4984.2023.277945

BITBUCKET ЯК РІШЕННЯ ДЛЯ ОРГАНІЗАЦІЇ КОМАНДНОЇ РОБОТИ НАД ІТ ПРОЕКТОМ

Д. М. Дмитренко, Г. М. Алексєєва, С. В. Хоменко

The article examines certain aspects of the process of professional training of students of engineering and pedagogical specialties of the computer profile, analyzes one of the most common forms of cooperation of IT specialists - team work on a project. Contradictions between the typical system of training a future specialist and the group nature of his/her future activity led to the need to develop the educational system of Ukraine, specifically in the field of training IT specialists. The skills of self-organization and self-discipline, communication skills, teamwork skills and cooperation, responsibility and adaptation, the formation of new skills and methods are becoming necessary for future computer professionals.

The basic principles of using version control systems were studied and the tools for their implementation were analyzed. For this purpose, an analysis of the most famous unique platforms Bitbucket and GitHub was carried out, their functionality was described, so that the user could decide for him/herself which system to choose; the choice of a software environment is justified due to its advantages and disadvantages, which determines the organization of effective teamwork on an IT project, where appropriate skills are required to develop individual project modules in parallel with other groups of participants. The peculiarities of using the Git and GitHub version control system during the implementation of IT projects were determined. The procedure and main steps of creating a working version of the project were described. For today's professional, teamwork is most important in training students, so understanding how to use version control system technology is a mandatory part of the professional training of future IT professionals. It has been proven, that for the organization of joint work, it is necessary to use a certain technology of interaction of a team of software project developers by means of Git version control systems

Keywords: professional training, IT specialists, teamwork, BITBUCKET, Git and GitHub

How to cite:

Dmytrenko, D., Aliksieieva, H., Khomenko, S. (2023). BITBUCKET as a solution for organizing teamwork on IT projects. ScienceRise: Pedagogical Education, 2 (53), 14–19. doi: <http://doi.org/10.15587/2519-4984.2023.277945>

© The Author(s) 2023

This is an open access article under the Creative Commons CC BY license hydrate

1. Вступ

Зміни, що відбуваються в сучасній системі освіти, входження України в Європейський освітній простір та зростання ролі діджиталізації зумовлюють процес модернізації всієї системи освіти [1, 2]. Для відповідності стрімкому розвитку інформаційних технологій професійна підготовка фахівців ІТ сфери потребує швидко реагування та адаптації до всіх змін і вимог щодо різноманітних професійних завдань [3, 4]. Аналізуючи процес підготовки студентів інженерно-педагогічних спеціальностей комп'ютерного профілю ми виявили протиріччя між типовою системою підготовки майбутнього фахівця та груповим характером його майбутньої діяльності. Особливо це стосується однієї з найпоширеніших форм співпраці ІТ фахівців як командна робота над проектом, що є досить ефективною й продуктивною роботою для розв'язання складних і громіздких завдань, де одночасно повинні працювати декілька паралельних груп [5]. Тому для організації ефективної командної роботи над ІТ проектом потрібні від-

повідні уміння здійснювати розробку окремих модулів проекту паралельно з іншими групами учасників, що сприятиме отриманню досвіду майбутніми ІТ фахівцями щодо ефективності їх подальшої участі у розробці програмного забезпечення будь-якої складності.

2. Літературний огляд

Аналіз наукової літератури свідчить про ряд особливостей. Н. Морзе [6], А. Стрюк [7] описують загальну теорію методики навчання алгоритмізації та методичні аспекти навчання системному програмуванню майбутніх інженерів-програмістів; об'єктно-орієнтований підхід до програмування розглядав науковець С. Вернигоренко [8]; проблематику вивчення окремих аспектів, які пов'язані із методикою навчання програмування здобувачів освіти різних педагогічних та технічних спеціальностей описали дослідники Ю. Біляй [9], П. Шевчук [10], О. Ящик [11]; тематику, пов'язану із версткою та web-програмуванням досліджували В. Базурін [12],

А. Ворожбит, О. Рибак [13], Н.Дегтярьова [14]; О. Овсянніков, Г. Алексеева розглядали підходи до проектування комп'ютерного середовища науково-дослідної командної роботи студентів інженерно-педагогічних спеціальностей комп'ютерного профілю [15].

Для формування фундаментальних знань щодо розробки алгоритмів, програмного забезпечення та web-програмування ці теми є структурною компонентою роботи у команді, формування культури оформлення коду та, власне, контроль за версіями у процесі розробки. Для сучасного фахівця саме командна робота у підготовці здобувачів освіти є більш значимою, тому саме розуміння роботи технологій систем контролю версій є обов'язковою складовою професійної підготовки майбутніх ІТ фахівців. Деякі науковці описували використання Git у рамках дистанційного навчання програмуванню (С. Горчинський, К. Горчинська) [16]; практичний аспект використання розподілених систем контролю для організації командної роботи над проектом описує В.Гриценко і О. Подолян [17]; використання систем контролю версій на основі Git для управління результатами наукових досліджень розглянуті в роботах J. Beall [18], С. Pham, Р. Sao, Z. Kalbarczyk, R. K. Iyer, [19], А. Jacobs, [20]; використання GitHub для організації роботи з науковими публікаціям описували Р. Knoth, L. Anastasiou, G. Basile, S. Pearce, N. Pontika [21]; Van Noorden R. займався проблемою організації спільної роботи студентів з використання GitHub [22].

Проте саме питання набуття студентами компетенцій, які дозволяють забезпечити реєстрацію змін у програмному коді і синхронізувати процес написання, власне, коду програми в команді фахівців, недостатньо висвітлені у сучасній науковій літературі.

3. Мета та задачі дослідження

Мета цієї статті полягає у розкритті деяких практичних аспектів навчання роботи із систем контролю версій на прикладі програми Git у процесі професійної підготовки майбутніх ІТ фахівців.

Для досягнення мети були поставлені такі завдання:

1) проаналізувати особливості професійної підготовки студентів інженерно-педагогічних спеціальностей комп'ютерного профілю в контексті характерних навичок майбутніх ІТ фахівців, необхідних для роботи в команді;

2) описати основні засади використання систем управління версіями і здійснено аналіз інструментарію їх реалізації найвідоміших платформ Bitbucket та GitHub.

4. Матеріали і методи

Для проведення дослідження використовувалися такі методи: теоретичні аналіз та узагальнення педагогічних та методичних джерел з метою виявлення кола проблем, що потребують науково-методичного підбору необхідного програмного забезпечення відповідно поставленому технічному завданню; методи порівняльного аналізу, інтерпретації та узагальнення фактів; порівняльно-аналітичний

для аналізу навичок самоорганізації і самодисципліни, навичок комунікації, навичок роботи в команді щодо використання платформ Bitbucket та GitHub; системний аналіз щодо опису їх функціональних можливостей в процесі обґрунтування вибору програмного середовища з урахуванням переваг та недоліків, що обумовлює організацію ефективної командної роботи над ІТ проектом.

5. Результати дослідження

Існує безліч різних платформ на основі Git для спільної роботи, призначених для команд або індивідуальних користувачів [23].

Для початку потрібно з'ясувати, що таке Git. Система Git була створена в 2005 році Лінусом Торвальдсом, який раніше створив ядро операційної системи Linux. Git застосовується для керування версіями великої кількості різних проектів розробленого програмного забезпечення. Система є мультиплатформною, тобто може працювати під керуванням різних операційних систем і сумісна з безліччю інтегрованих середовищ (IDE). Git – система з розподіленою архітектурою, на відміну від деяких систем, де повна історія версій проекту доступна в якомусь одному місці, у Git кожна робоча копія проекту (або коду) сама по собі є репозиторієм. Застосування Git дозволяє забезпечити високу продуктивність процесу, його безпеку та гнучкість розподіленої системи. Bitbucket (англ. bit – «біт» та bucket – «відро») – це інструмент на основі Git, призначений для команд, хостингу систем управління версіями коду (Version Control System, VCS) та спільної роботи. Розробник Bitbucket – австралійська компанія Atlassian із Сіднея. Ця компанія є відомим розробником програмного забезпечення для керування розробкою програмного коду. Найбільш відомі продукти: Jira (система відстеження помилок коду), Confluence (система спільної роботи) [24].

Bitbucket не є оригінальною або навіть унікальною платформою, має «конкурентів», один з найвідоміших – GitHub. На основі порівнянь з GitHub розглянемо переваги та недоліки Bitbucket, а також її можливості.

Значимо, що обидві ці платформи мають платну та безкоштовну версію. Надалі будемо розділяти властивості та характеристики цих версій. Мабуть, найголовніша відмінність цих двох систем у тому, що Github орієнтований на загальнодоступність коду, а Bitbucket – на приватність. Безкоштовна версія Bitbucket має можливість створення приватного репозиторію для колективу розробників у кількості до п'яти осіб. Природно, що у платній версії ця кількість набагато більша. З цього робиться висновок, що Bitbucket спрямований також на закриті команди розробників [25].

Однією з переваг Bitbucket є те, що він добре інтегрується з іншими системами виробництва Atlassian: Jira (програмне забезпечення для відстеження проблем Atlassian) та Trello (інструмент управління проектами), широко відомими серед програмних та бізнес-команд. Також можлива інтеграція ще більш ніж з п'ятдесяти інструментів розробника (Microsoft Azure, Amazon, Docker Hub, Google

Analytics, Twitter, CodePlex, Google Code, SourceForge та багатьма іншими) [26].

Ще одна перевага Bitbucket у відношенні з GitHub в тому, що в Bitbucket доступні і Git, і Mercurial. Bitbucket написаний мовою програмування Python та використовує веб-фреймворк Django. Bitbucket запущений в Австралії в 2008 році і спочатку був незалежним стартапом, що пропонував хостинг виключно для проектів Mercurial. У 2010 році він був придбаний компанією Atlassian і через деякий час розробники додали підтримку репозиторіїв Git.

Семантичний пошук JQL Bitbucket сканує власне синтаксис, а не імена змінних, тому пошук проводиться швидше.

Bitbucket дозволяє створювати необмежену кількість приватних репозиторіїв. Але є обмеження на кількість користувачів не більше п'яти. Для більшої кількості потрібно придбати платний пакет. Для порівняння, на Github (на момент написання цієї статті)

усі приватні репозиторії обмежені кількістю користувачів не більше трьох. Ліміт дискового простору на один репозиторій – 2 ГБ. Github на це виділяється не більше 500 МВ для безкоштовного використання (рис. 1).

Для кожного репозиторію Bitbucket є можливість створення свого Wiki. Ця функція може містити необхідну інформацію щодо роботи з платформою. Також у Wiki можна розміщувати документацію щодо створюваного програмного забезпечення.

При роботі з Bitbucket є можливість використання хмарного сервісу або ізолизованого хмарного простору від Atlassian (Data Center). До переваг хмарного використання слід віднести отримання миттєвого доступу до новітніх можливостей завдяки автоматичним оновленням, а також відсутність серверів, сховищ або технічного обслуговування. Крім того, користувач самостійно налаштовує планувальник бекапів [26].

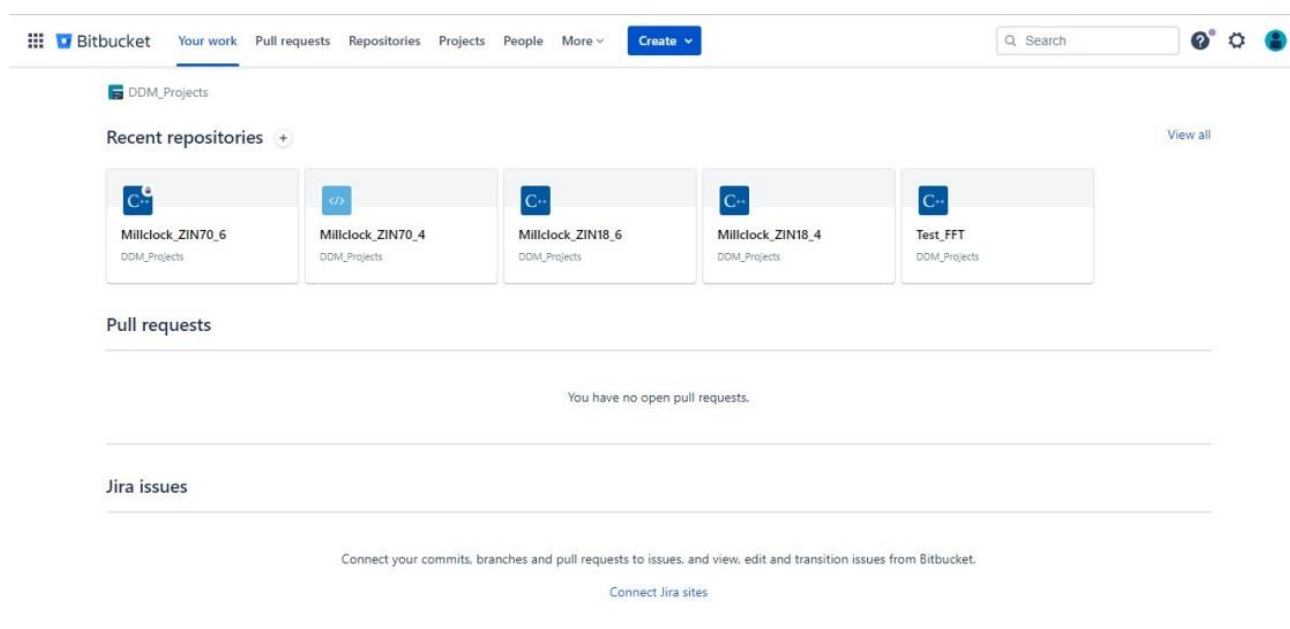


Рис. 1. Bitbucket сторінка з переліком наявних репозиторіїв користувача

Глибока інтеграція з Trello за допомогою Bitbucket Cloud здійснює безшовну інтеграцію гілок, комітів і pull-запитів у дошки Trello. Дозволи на гілку: замість того, щоб надавати доступ до кожної гілки репозиторії, Bitbucket має можливість обмежити доступ до однієї гілки [24].

Bitbucket Snippets має можливість так само створювати та керувати всіма видами багатофайлових сніпетів.

Bitbucket Pipelines – це функція Bitbucket, що забезпечує єдиний робочий простір для складання, тестування та розгортання коду з використанням інтегрованого конвеєра безперервної інтеграції та безперервної доставки (CI/CD) (рис. 2).

Онлайн-редактор коду Bitbucket – один із найкращих для редагування прямо в мережі, без попередньої перевірки на комп'ютері. Інтерфейс Bitbucket

зосереджений навколо робочої панелі, яка дає легкий доступ до інших репозиторіїв та запитів. Bitbucket надає повний інтерфейс Git, підтримуючи майже всі його функції.

Реєстрація на сервісі Bitbucket стандартна для подібних сервісів складається з двох етапів. Спочатку необхідно зайти на сторінку реєстрації офіційного сайту та ввести адресу електронної пошти та пароль. Після чого на вказану адресу прийде лист із підтвердженням реєстрації та посиланням активації облікового запису. Сервіс готовий до роботи.

Інформацію на Bitbucket можна вносити як за допомогою редактора на сайті, так і за допомогою інструментів Git, які попередньо встановлені на комп'ютері користувача. Всі операції із синхронізації стандартні, опис присутній на офіційному сайті.

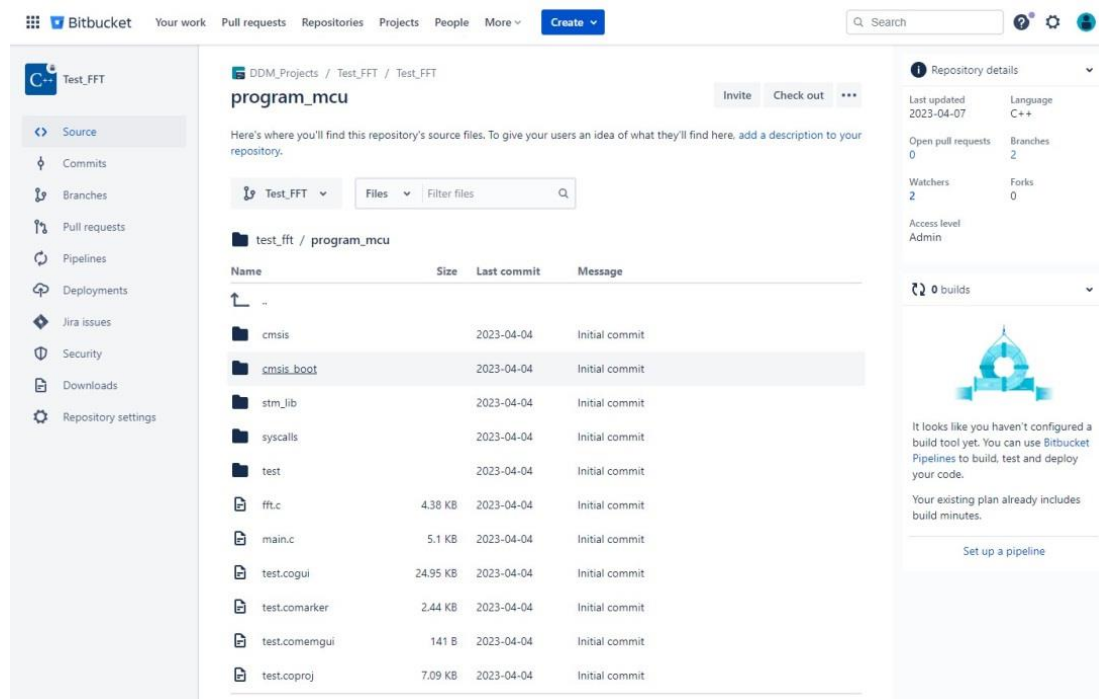


Рис. 2. Bitbucket – перелік файлів репозиторію

З недоліків сервісу Bitbucket можна відзначити відсутність локалізації, тобто є лише одна мова – англійська. Хоча, у Github такий же недолік. Для програмістів, на мою думку, це не буде проблемою. Частково інтерфейс Bitbucket перекладено деякими мовами, але до повної локалізації (на момент написання статті) ще далеко.

Крім того, на Bitbucket відсутня можливість створення репозиторіїв із відкритим вихідним кодом. Ще один недолік – на відміну від GitHub, у Bitbucket відсутнє підсвічування синтаксису.

Bitbucket – платформа, яка дозволяє користувачам керувати проектами або ділитися ними.

Таким чином, хоча Bitbucket і відрізняється меншою кількістю користувачів (проти Github), але, в порівнянні з Github, він досить зручний і вигідний в економічному та практичному плані для невеликих команд або індивідуальних розробників. Для великих команд, що працюють над кількома проектами, найкращим варіантом буде GitHub.

Наявність власного хмарного сховища та безпечний доступ до нього за протоколами HTTPS та SSH підвищує безпеку та надійність цього сервісу [4].

Всі вищесказані матеріали ми зібрали в табл. 1.

Таблиця 1

Аналіз інструментарію найвідоміших платформ Bitbucket та GitHub

Параметр	Bitbucket	GitHub
1	2	3
Підтримувані системи контролю версій (VCS)	Підтримуються як Mercurial, так і Git	Підтримується лише Git
Публічний репозиторій	Необмежена кількість безкоштовних сховищ. Майже будь-яка кількість користувачів може отримати доступ до публічних сховищ.	Необмежена кількість безкоштовних сховищ.
Приватний репозиторій	Приватний репозиторій на 5 користувачів в безоплатній версії.	Приватний репозиторій на 3 користувача в безоплатній версії.
Навігація	Bitbucket не має функції навігації по коду.	Функція навігації по коду присутня
Інтерфейс користувача	Впорядкований і зрозумілий інтерфейс	Інтерфейс простий. Але, на відміну від Bitbucket, в деяких моментах заплутаний.
Wiki	Дозволяє додавати wiki до кожного зі своїх сховищ. Є вибір доступності або приватності wiki	Немає можливості додати wiki до безкоштовних приватного репозиторіїв, лише – до загальнодоступних.
Підтримка та спільнота	Бізнесові та корпоративні команди використовують, в основному, для приватних репозиторіїв.	Є найбільшим хостом для публічного коду, тому має велику спільноту
Аналіз проекту	Візуалізація аналізу за допомогою діаграм.	Наразі не пропонує цю функцію. Але користувачі мають можливість переглядати історію комітів для файлів

Продовження табл. 1

1	2	3
Пряма інтеграція	Jira, Bamboo, Crucible, JenkinsExternal.	Zendesk, Azure, Cloudbees, Google Cloud, Amazon, Code Climate, Heroku
Гнучкість	Пропонує більшу гнучкість за замовчуванням, оскільки може імпортувати сховища з Git, CodePlex, Google Code, SourceForge і SVN.	Може імпортувати репозиторії з Git, SVN, HG і TFS
Додаткові функції	Інтеграція Jira, відстеження проблем, Bitbucket Snippets, Bitbucket Pipelines. Зовнішня аутентифікація через Twitter, Facebook і Google	Двофакторна аутентифікація, підсвічування синтаксису
Ціна	Приблизно у 10 разів дешевший (дані на травень 2022 р.), але пропонує відносно однакову функціональність	Користуватися Github відносно дорожче, ніж Bitbucket.

Отже, спільна робота над проектом не можлива без використання хмарних репозиторіїв, найбільш відомі із них були нами описані. Важливою перевагою використання цих сервісів є можливість створити спільне робоче середовище для студентів та викладачів, що дозволяє не тільки навчити студентів працювати із системою керування версіями, а й створювати спеціалізовані завдання, виконувати їх перевірку, спільно реалізувати навчальні проекти.

Дослідження не повністю враховує особливості організації індивідуально-групового навчання під час командної реалізації ІТ проектів у нинішніх швидкоплинних умовах. За цих обставин постійно настає проблема пошуку технологічних рішень і розробки методичних прийомів щодо організації командної роботи і навчання, механізмів і закономірностей міжособистісної взаємодії майбутніх ІТ фахівців. Використання елементів керування версіями досить добре впроваджуються в наукових дисциплінах, хоча їх використання все ще далеко до універсального.

6. Висновки

1) Проаналізовано особливості професійної підготовки студентів інженерно-педагогічних спеціальностей комп'ютерного профілю в контексті характерних навичок майбутніх ІТ фахівців, необхідних для роботи в команді. При цьому, системи управління версіями розглядатимуться водночас як засоби навчання, як одна з цілей навчання, як професійний інструмент для розроблення програмного забезпечення. Це дозволяє організувати більш ефективну взаємодію викладачів і студентів щодо виконання завдань із використанням сучасних професійних інструментів ІТ-фахівців і хмарних сервісів; значно

спрощує виконання будь-яких колективних проектів (з будь-якою кількістю учасників) з програмування або вебтехнологій; отримати хостинг для розміщення результатів роботи (програмного коду).

2) Описано основні засади використання систем управління версіями, здійснено аналіз інструментарію реалізації двох платформ Bitbucket та GitHub. Таким чином, хоча Bitbucket і відрізняється меншою кількістю користувачів (проти Github), але, в порівнянні з Github, він досить зручний і вигідний в економічному та практичному плані для невеликих команд або індивідуальних розробників. Для великих команд, що працюють над кількома проектами, найкращим варіантом буде GitHub. Ці дві системи однаково потужні, різниця в малому, просто Bitbucket новіша, зі своїми відмінностями. Користувача треба підвести до рішення не недоліками та перевагами, а саме функціональністю, для кого наявність або відсутність якогось функціоналу може бути як перевагою, так і недоліком.

Конфлікт інтересів

Автори декларують, що не мають конфлікту інтересів стосовно даного дослідження, в тому числі фінансового, особистісного характеру, авторства чи іншого характеру, що міг би вплинути на дослідження та його результати, представлені в даній статті.

Фінансування

Дослідження проводилось без фінансової підтримки.

Доступність даних.

Рукопис не має пов'язаних даних.

Література

1. Glazunova, O., Korolchuk, V., Parhomenko, O., Voloshyna, T., Morze, N., Smyrnova-Trybulska, E. (2020). Methodology for using Cloud-oriented Environment for Flipped Learning of the Future IT Specialists. Proceedings of the 1st Symposium on Advances in Educational Technology. Kyiv: SciTePress. doi: <https://doi.org/10.5220/0010925100003364>
2. Diachenko, A., Vusyk, H., Bielova, Y., Shurdenko, M., Titenko, O. (2021). Educational role in COVID-19 terms of ethnodesign graphic function in higher education practical activities. International Journal of Health Sciences, 5 (3), 584–593. doi: <https://doi.org/10.53730/ijhs.v5n3.2540>
3. Алексеева, Г. М. (2014). Використання інформаційно-комунікаційних технологій в процесі професійної підготовки студентів педагогічних вузів. Актуальні питання природничо-математичної освіти. Суми, Україна: ВВП «Мрія», 184–191.
4. Несторенко, Т. П. (2020). «Економіка суперзірок»: можливості та загрози для сфери освіти. Український журнал прикладної економіки, 5 (7), 8–15. doi: <https://doi.org/10.36887/2415-8453-2020-2-1>

5. Овсянніков, О., Алексеева, Г. (2019). Комп'ютерне середовище науково-дослідної роботи студентів інженерно-педагогічних спеціальностей комп'ютерного профілю як об'єкт проектування. Молодь і Ринок, 9 (176). doi: <https://doi.org/10.24919/2308-4634.2019.182241>
6. Алексеева, Г. М. (2014). Формування готовності майбутніх соціальних педагогів до застосування комп'ютерних технологій у професійній діяльності. Бердянськ: БДПУ, 269.
7. Стрюк, А. М. (2011). Системне програмування у підготовці фахівців з програмної інженерії. Збірник наукових праць Уманського державного педагогічного університету, 3, 260–271.
8. Вернигоренко, С. А. (2007). Вивчення основ об'єктно орієнтованого програмування у класах фізико-математичного профілю. Науковий часопис НПУ імені М. П. Драгоманова. Серія 2. Комп'ютерно-орієнтовані системи навчання, 5 (12), 63–74.
9. Біляй, Ю. П. (2018). Реактивне програмування. Науковий часопис НПУ імені М.П.Драгоманова. Серія 2. Комп'ютерно-орієнтовані системи навчання, 20 (27), 124–130.
10. Шевчук, П. Г. (2010). Програмно-технологічні умови використання мови С# для навчання програмування в загальноосвітніх навчальних закладах. Науковий часопис НПУ імені М.П. Драгоманова. Серія 2. Комп'ютерно-орієнтовані системи навчання, 10 (17), 80–84.
11. Ящик, О. Б. (2016). Машина Тьюрінга як універсальний виконавець алгоритмів та її застосування в процесі поглибленого вивчення алгоритмізації і основ програмування старшокласниками. Інформаційні технології і засоби навчання, 52 (2), 10–17.
12. Базурін, В. М. (2014). Особливості навчання веб-програмування мовою JavaScript студентів-математиків. Вісник Житомирського державного університету, 1 (73), 79–83.
13. Ворожбит, А. В., Рибак О. С. (2018). Огляд курсу за вибором «Основи верстки та веб-програмування». Фізико-математична освіта, 1 (15), 20–27.
14. Дегтярьова, Н. В. (2017). Методичні особливості навчання студентів розмітки блоків при вивченні таблиць каскадних стилів. Фізико-математична освіта, 1 (11), 32–36.
15. Kravchenko, N. V., Alyeksyeyeva, H. M., Gorbatyuk, L. V. (2018). Curriculum Optimization by the Criteria of Maximizing Professional Value and the Connection Coefficient of Educational Elements, Using Software Tools. ICTERI 2018: 14th International conference on ICT in education, research, and industrial applications). Kyiv: CEUR Workshop Proceedings, 1, 365–378.
16. Горчинський, С., Горчинська, К. (2022). Використання github при вивченні дисципліни «основи веб-технологій». Professional Pedagogics, 1 (24), 195–202. doi: <https://doi.org/10.32835/2707-3092.2022.24.195-202>
17. Гриценко, В. Г., Подолян, О. М. (2014). Використання системи управління версіями GIT для організації командної роботи над IT проектом. Інформаційні технології і засоби навчання, 39 (1), 250–263.
18. Beall, J. (2016). Ban predators from the scientific record. Nature, 534 (7607), 326. doi: <https://doi.org/10.1038/534326a>
19. Pham, C., Phuong Cao, Kalbarczyk, Z., Iyer, R. K. (2012). Toward a high availability cloud: Techniques and challenges. IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN 2012). doi: <https://doi.org/10.1109/dsnw.2012.6264687>
20. Jacobs, A. (2019). Network Device Configuration Management with Oxidized and Bitbucket (No. SAND2019-5380C). Sandia National Lab.(SNL-CA). Livermore.
21. Knoth, P., Anastasiou, L., Basile, G., Pearce, S., Pontika, N. (2017). Machine accessibility of Open Access scientific publications from publisher systems via ResourceSync. OAI10.
22. Van Noorden, R. (2014). Online collaboration: Scientists and the social network. Nature, 512 (7513), 126–129. doi: <https://doi.org/10.1038/512126a>
23. Іваніньська, І. І., Абдурайімов, Л. Н. (2013). Застосування веб-сервісу GitHub при розробці програмних проектів студентами в процесі навчання. Proceedings of the third international scientific and practical conference FOSS Lviv.
24. Abyzova, L., Babenko, O., Nestorenko, T., Reshetova, I., Semeniuk, M., Shevchenko, O. (2017). Educational management in Ukraine: the place of displaced universities. Sustainable Development Goals: The 2030 Agenda & Does environmental diplomacy reflect new challenges regarding climate change? Workshop 8.11.2017. Bratislava. Available at: <https://cutt.ly/SYcVUXq>
25. Lavrik, V., Cortez, L., Aliksieieva H., Garcia, G. T., Juarez, P. G., Poblano, J. (2014). Development of the CAD system for designing non-standard constructions from elastomers. Development, 3 (3).
26. Mosiuk, O. (2021). Особливості навчання майбутніх учителів інформатики роботі із системою контролю версій GIT. Науковий вісник Ужгородського університету. Серія: «Педагогіка. Соціальна робота», 2 (49), 107–110.

Received date 07.02.2023

Accepted date 23.03.2023

Published date 31.03.2023

Дмитро Миколайович Дмитренко, аспірант, Бердянський державний педагогічний університет, вул. Шмідта, 4, м. Бердянськ, Україна, 71100

Ганна Миколаївна Алексеева*, Кандидат педагогічних наук, доцент, кафедра комп'ютерних технологій в управлінні та навчанні й інформатиким Бердянський державний педагогічний університет, вул. Шмідта, 4, м. Бердянськ, Україна, 71100

Світлана Валеріївна Хоменко, кандидат педагогічних наук, доцент, кафедра професійної педагогіки, Бердянський державний педагогічний університет, вул. Шмідта, 4, м. Бердянськ, Україна, 71100

**Corresponding author: Hanna Aliksieieva, e-mail: alekseeva@ukr.net*