

КОМП'ЮТЕРНІ НАУКИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

УДК 004.42:378.147

DOI: 10.30977/BUL.2219-5548.2023.103.0.7

ВИКОРИСТАННЯ МОВИ JAVASCRIPT ДЛЯ АВТОМАТИЗАЦІЇ ОБЛІКУ ВІДВІДУВАНОСТІ СТУДЕНТІВ У НАВЧАЛЬНІЙ СИСТЕМІ MOODLE

Лебединський А. В.

Харківський національний автомобільно-дорожній університет

Анотація. У статті розглянута проблема обліку відвідуваності в закладах вищої освіти в реаліях дистанційного навчання. У роботі проаналізовано наявні способи обліку відвідуваності студентів у сучасних умовах навчання, а також їхні недоліки та переваги. Було створено програму на мові програмування JavaScript для автоматичного обліку відвідуваності студентів у вигляді CSV-файлу з подальшим його завантаженням на вебсервер Moodle на прикладі навчального сайту Харківського національного автомобільно-дорожнього університету.

Ключові слова: JavaScript, скрипти, автоматизація, відвідуваність, програмування, Moodle.

Вступ

У сучасних умовах високий рівень автоматизації процесів дедалі більше застосовують у різноманітних сферах життя, зокрема у сфері освіти.

Реєстрація відвідуваності студентів є важливим елементом освітнього процесу, але її здійснення триває багато часу та може містити помилки через людський фактор. Нині в умовах дистанційного навчання в Україні це питання стає все більш актуальним.

Автоматизація обліку відвідуваності дозволить мінімізувати вплив людського фактора та скоротити в декілька разів час викладачів на здійснення цієї процедури під час проведення занять.

Аналіз публікацій

У багатьох країнах світу з початку 2020 року дистанційне навчання стало нормою через епідемію COVID-19 [1]. Більшість навчальних закладів перейшли на дистанційну форму навчання, використовуючи різноманітні системи управління навчанням (навчальні платформи):

- Blackboard Learn;
- Moodle;
- Canvas;
- OpenEdx;
- Google Classroom тощо.

Залежно від регіонів та або інша навчальна платформа є такою, що домінує. Наприклад, у США система керування навчальними курсами Blackboard становить приблизно 31 % від усього ринку в країні, а Moodle лише 18 % [2]. В Європі, в Латинській Америці, в Австралії і в Океанії більше 50 % ринку

становить саме навчальна платформа Moodle [3].

Питання автоматизації обліку відвідуваності студентів під час дистанційного навчання у світі розглядалася міжнародною науковою спільнотою в контексті розпізнавання облич студентів за допомогою нейронних мереж і штучного інтелекту. У [4] та [5] аналізується процес розроблення автоматизованої системи обліку відвідуваності на основі штучного інтелекту з використанням глибокого (машинного) навчання та розпізнавання образів. У [6] досліджується використання системи біометричних сенсорів для ідентифікації студентів під час заняття, а у [7] ідентифікація студентів за райдужною оболонкою ока із подальшим автоматичним збереженням інформації про відвідуваність у базах даних.

Перевагами вищезазначених методів є майже стовідсоткова ідентифікація студентів і автоматичне збереження обліку відвідуваності в базах даних. Недоліками ж усіх цих методів є складний процес впровадження та реалізації із попереднім занесенням біометричної інформації студентів для створення вибіркового аналізу, за яким вже буде здійснюватися процес ідентифікації та збереження результатів. Крім того, для створення програмно-апаратного комплексу таких систем необхідні значні витрати фінансів і часу.

Процес обліку відвідуваності студентів занять у закладах вищої освіти в Україні здійснюють різними способами, зокрема найчастіше використовують три основні:

- за допомогою таблиці Excel власноруч викладачем;

- за допомогою вбудованої системи відвідуваності «Attendance» у Moodle власноруч викладачем;
- за допомогою повідомлень самих студентів щодо своєї присутності на початку заняття.

Ці три способи мають один головний недолік, який може впливати на облік відвідуваності. Цим недоліком є наявність людського фактора. Через велику кількість факторів (вдома, поспіх, неуважність) викладач може неправильно здійснити облік відвідуваності та в деяких випадках це може призвести до зниження підсумкової оцінки студента. Якщо облік здійснюється безпосередньо студентом у Moodle, він може просто забути про це і не повідомити про свою присутність.

Тому виникає потреба в створенні методу для здійснення процесу безпомилкового обліку відвідуваності студента із мінімальною кількістю витраченого на це часу викладача за допомогою автоматизації дій.

Мета та постановка завдання

Метою є розроблення автоматизованого методу обліку відвідуваності студентами занять під час дистанційного навчання в закладах вищої освіти, що використовують навчальну платформу Moodle.

Для досягнення поставленої мети спочатку необхідно проаналізувати основні наявні на сьогодні способи обліку відвідуваності студентів у реаліях дистанційного навчання, визначити їхні переваги й недоліки та відповідно до специфіки проведення занять на навчальній платформі Moodle створити про-

граму, яка усуне більшість недоліків наявних способів.

Аналіз функцій блоку відвідуваності Moodle

На навчальній платформі Moodle облік відвідуваності реалізований блоком «Attendance» (рис. 1), який здійснює велику кількість функцій: додавання заліково-екзаменаційних сесій, розподіл сесій на групи студентів, створення статусів відвідуваності студентів тощо.

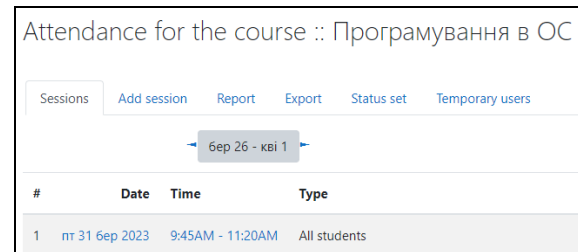


Рис. 1. Блок «Attendance» у Moodle

Функція, яку ми будемо використовувати, називається «Upload attendance by CSV» (рис. 2). Вона дозволяє завантажити файл із розширенням CSV, що містить інформацію про відвідуваність студентів певного заняття, а також їхні статуси. Стандартний перелік статусів відвідуваності є таким: П – присутній, О – спізнився, У – відсутній через поважну причину, Н – відсутній. Залежно від наповнення цього CSV-файлу автоматично буде фіксуватися відвідуваність для кожного студента.

Вміст CSV-файлу для правильного оброблення системою Moodle наведено на рис. 3.

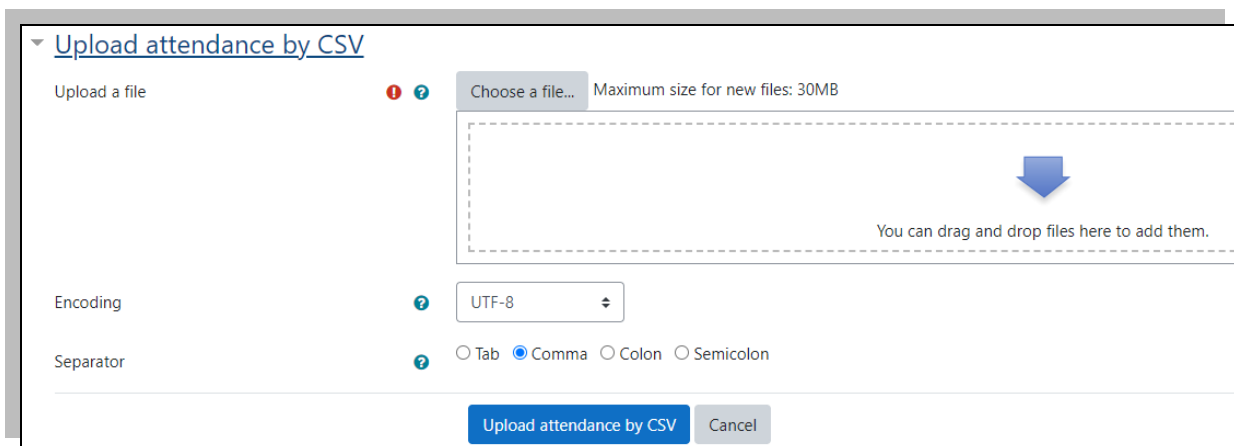


Рис. 2. Функція «Upload attendance by CSV» у Moodle

```

1 Emails, Status
2 stud_email@stud.ua, П
3 stud_email@stud.ua, П
4 stud_email@stud.ua, П
5 stud_email@stud.ua, П
6 stud_email@stud.ua, П
7 stud_email@stud.ua, П

```

Рис. 3. CSV-файл із обліком відвідуваності після заняття

Як спосіб ідентифікації студента можна вибрати такі дані: електронна пошта студента, ID студента в Moodle та Username студента. У роботі було використано спосіб ідентифікації студента за e-mail. Такий спосіб не є безпечним щодо можливого зчитування інформації зі скрипту третіми особами, але його застосовують через неможливість використання ID студента та Username, оскільки відсутня реалізація цього функціонала на сервері Moodle.

Використання систем управління навчанням, зокрема Google Classroom, Moodle та інших, передбачає аутентифікацію студентів у ній способом входу до особистих акаунтів з особистим логіном і паролем. Тому питання саме ідентифікації студента під час входу на онлайн-заняття в статті не розглядалося.

Передумови до роботи

Оскільки велика кількість занять проводиться саме на платформі для онлайн-конференцій BigBlueButton (рис. 4), то передумовою до роботи цього способу є присутність студентів під час онлайн-заняття та написання до загального чату будь-якого повідомлення під час перевірки викладачем відвідуваності (рис. 5). Прізвища студентів були приховані.

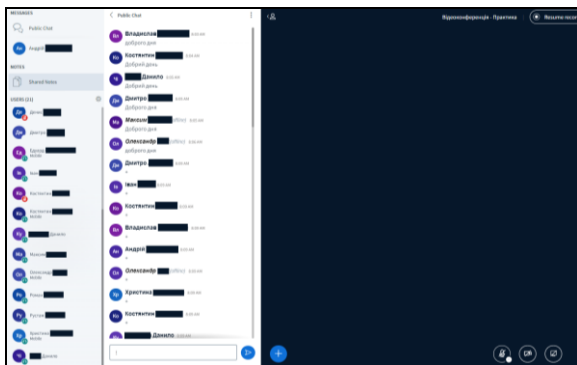


Рис. 4. BigBlueButton онлайн-конференції

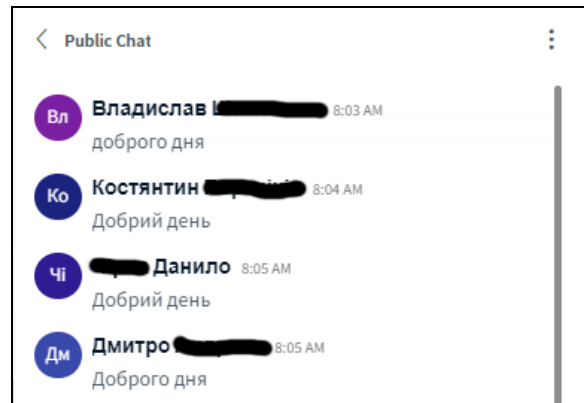


Рис. 5. Загальний чат BigBlueButton

Зміст запропонованого способу

За допомогою залучення скрипту на мові JavaScript у консоль вебсторінки пропонується автоматично генерувати до CSV-файлу залежно від кількості присутніх на занятті студентів, що написали будь-яке повідомлення в онлайн-чат. Скрипт також можна налаштувати на будь-який певний символ (наприклад, символ «+»), під час залучення якого до загального чату система реєструє присутність студента на онлайн-конференції. У цьому випадку всі інші повідомлення, які не будуть містити в собі цей символ, автоматично ігноруватимуться.

Алгоритм роботи скрипту наведено на рис. 6. Текстовий опис алгоритму:

1) визначаємо змінні: змінна «scrollDiv» посилається на контейнер із загальним чатом, де знаходяться текстові повідомлення студентів із їхнім прізвищем та ім'ям. Цей контейнер має клас "ReactVirtualized_Grid.ReactVirtualized_List" в html-документі. Додатково визначаємо масиви для зберігання даних про користувачів;

2) створення словника: змінна «dict» здійснює функцію створення об'єкта із множиною пар ключ-значення, де як ключ використовується набір прізвищ та імен студентів, як значення використовується їхня поштова адреса;

3) додавання обробників подій: на цьому етапі здійснюється створення та додавання обробників подій «scroll» та «dblclick» (рис. 7), перший з яких здійснює функцію збереження всіх повідомлень усіх студентів під час скролінгу вікна чату, а другий – функцію генерації CSV-файлу із потрібним форматкуванням необхідних даних: поштової адреси студентів і їхніх статусів;

4) вилучення повідомлень присутніх студентів із вебсторінки. Способом прохो-

дження вздовж кожного з елементів загального чату «scrollDiv» вилучається текст з кожного такого елемента. Вилучені дані зберігаються в масиві «users»;

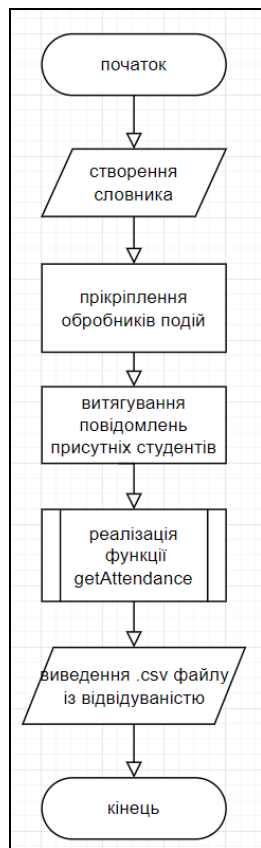


Рис. 6. Алгоритм роботи скрипту attendance.js

5) реалізація функції «getAttendance». Функція «getAttendance» складається з декількох етапів:

5.1) видаляються імена та прізвища, що повторюються (тобто ті студенти, які декілька разів щось написали до загального чату);

5.2) створюється масив «attendanceStatus» із символом «П», який відповідає за статус відвідування «присутній» для кожного студента;

5.3) імена та прізвища присутніх студентів перетворюються на адреси електронної пошти, використовуючи дані зі словника «dict»;

5.4) створюється масив «arr» з парами адрес електронної пошти та статусом відвідування «П»;

5.5) здійснюється підготовка наповнення файлу CSV способом об'єднання цих двох масивів за допомогою ком та символу нового рядка.

5.6) створюється об'єкт «Blob» із вмістом відвідуваності (рис. 8);

5.7) створюється посилання для заванта-

ження форматovanого CSV-файлу;

б) вилучення CSV-файлу із відвідуваністю: подвійним натисканням у будь-якому місці поля загального чату на web-сторінці згенерований CSV-файл буде завантажений до будь-якого місця в операційній системі користувача.

```

scrollDiv.addEventListener('dblclick', function(e) {
  getAttendance();
});
  
```

Рис. 7. Підключення обробника подій подвійним натисканням миші для здійснення функції «getAttendance»

```

let csvContent = arr.map(e => e.join(",")).join("\n");
const blob = new Blob([csvContent], { type: 'text/csv;charset=utf-8' });
const objUrl = URL.createObjectURL(blob);
const link = document.createElement('a');
link.setAttribute('href', objUrl);
link.setAttribute('download', 'File.csv');
link.textContent = 'Click to Download';
document.querySelector('body').append(link);
link.click();
  
```

Рис. 8. Код створення об'єкта «Blob» із вмістом CSV-файлу

Результат роботи програми

Результатом роботи програми attendance.js є завантажений і відформатований згідно з правилами CSV-файл, який містить електронну поштову адресу кожного із тих студентів, які під час проведення онлайн-заняття написали будь-яке повідомлення до загального чату BigBlueButton, і статус відвідуваності.

Цей файл можна завантажувати до системи Moodle блоку «Attendance», де автоматично буде зазначена відвідуваність кожного студента на цьому занятті. Якщо процес імпортування CSV-файлу до Moodle буде з помилками, у системі з'явиться відповідне повідомлення (рис. 9). Залежно від налаштувань блоку відвідуваності всі інші студенти, які не були присутні на занятті, не будуть відмічені або матимуть статус відвідуваності «Н», тобто відсутній.

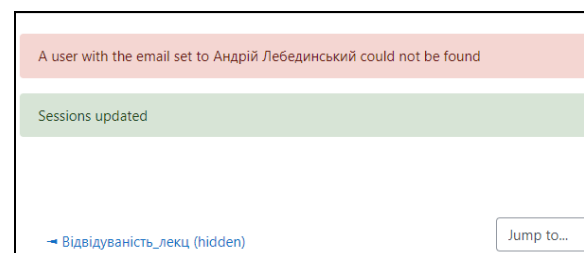


Рис. 9. Діалогове вікно з помилками після імпорту відвідуваності студентів з CSV-файлу

Можливі напрями наступних етапів розроблення

Одним з пріоритетних напрямів наступних етапів розроблення програми є її автоматизація і зменшення кількості часу на здійснення процесу обліку відвідуваності викладачем. Використання розширень «Chrome Extensions» дозволить створити розширення на основі вже наявної програми, яке автоматично буде завантажуватися й активуватися тільки тоді, коли викладач приєднується до онлайн-конференції BigBlueButton. Таким чином, викладачу непотрібно буде на кожному занятті додавати скрипт до консолі та скролити вікно загального чату.

Ще одним напрямом для подальшого розроблення є додавання додаткових статусів відвідувань, зокрема статусів «спізнився» та «відсутній через поважну причину». Наприклад, реалізація першого статусу можлива за умови використання часового проміжку початку та завершення заняття. Крім прізвища та імені, у загальному чаті подано час написання повідомлення. Залежно від цього можна модифікувати функцію «getAttendance» так, щоб в разі, якщо студент вчасно не написав повідомлення до чату, він би отримав статус «спізнився» (замість «присутній»).

Висновки

У статті було розглянуто процес створення скрипту на мові програмування JavaScript, який дозволяє автоматизувати облік відвідуваності студентів під час дистанційного навчання на платформі онлайн-навчання Moodle. Способом моніторингу загального чату онлайн-конференції BigBlueButton створюється форматований CSV-файл, який містить інформацію про всіх студентів, які написали будь-яке повідомлення до чату, та їхній статус відвідуваності «П», тобто «присутній». Після завершення заняття подвійним натисканням лівої клавіші миші по елементу загального чату на веб-сторінці цей CSV-файл завантажується на комп'ютер викладача. Отриманий результат, тобто CSV-файл, може бути легко імпортований до системи Moodle, де автоматично позначено статус відвідуваності студентів для подальшого використання в навчальних процесах. Програма дозволяє викладачам фіксувати присутність студентів й автоматизувати процес оцінювання.

У разі виникнення помилок під час імпорту CSV-файлу до системи Moodle користувач буде інформований про це за допомогою діалогового вікна з помилками. Такий підхід

дозволяє забезпечити точність та надійність обліку відвідуваності студентів під час онлайн-занять.

Загалом запропонований спосіб визначає потенціал для покращення процесів онлайн-навчання та оптимізації роботи викладачів у віртуальному навчальному середовищі.

Література

1. COVID-19 and Distance Learning: Effects on Georgia State University School of Public Health Students / E. Armstrong-Mensah та ін. *Frontiers in Public Health*. 2020. Т. 8. URL: <https://doi.org/10.3389/fpubh.2020.576227> (дата звернення: 11.07.2023).
2. Contributors to Wikimedia projects. Moodle - Wikipedia. *Wikipedia, the free encyclopedia*. URL: <https://en.wikipedia.org/wiki/Moodle> (дата звернення: 11.07.2023).
3. BENADLA D., HADJI M. EFL Students Affective Attitudes towards Distance E-Learning Based on Moodle Platform during the Covid-19the Pandemic: perspectives from Dr. Moulay-Tahar University of Saida, Algeria. *Arab World English Journal*. 2021. № 1. С. 55–67. URL: <https://doi.org/10.24093/awej/covid.4> (дата звернення: 11.07.2023).
4. Automated Attendance System in the Classroom Using Artificial Intelligence and Internet of Things Technology / D. D. Nguyen та ін. 2021. *8th NAFOSTED Conference on Information and Computer Science (NICS)*, м. Hanoi, Vietnam, 21–22 груд. 2021 р. 2021. URL: <https://doi.org/10.1109/nics54270.2021.9700991> (дата звернення: 11.07.2023).
5. Deep Learning Based Automated Attendance System / J. Joshan Athanesious та ін. *Procedia Computer Science*. 2019. Т. 165. С. 307–313. URL: <https://doi.org/10.1016/j.procs.2020.01.045> (дата звернення: 11.07.2023).
6. Rahman S., Rahman M., Rahman M. M. Automated Student Attendance System using Fingerprint Recognition. *Edelweiss Applied Science and Technology*. 2018. С. 90–94. URL: <https://doi.org/10.33805/2576.8484.120> (дата звернення: 11.07.2023).
7. Design and implementation of iris recognition-based attendance management system / A. Khatun та ін. 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), м. Dhaka, 21–23 трав. 2015 р. 2015. URL: <https://doi.org/10.1109/iceeict.2015.7307458> (дата звернення: 11.07.2023).

References

1. COVID-19 and Distance Learning: Effects on Georgia State University School of Public Health Students / E. Armstrong-Mensah et al. *Frontiers in Public Health*. 2020. Т. 8. URL:

- <https://doi.org/10.3389/fpubh.2020.576227> (date of access: 11.07.2023).
2. Contributors to Wikimedia projects. Moodle - Wikipedia. *Wikipedia, the free encyclopedia*. URL: <https://en.wikipedia.org/wiki/Moodle> (date of access: 11.07.2023).
 3. BENADLA D., HADJI M. EFL Students Affective Attitudes towards Distance E-Learning Based on Moodle Platform during the Covid-19the Pandemic: perspectives from Dr. Moulay-Tahar University of Saida, Algeria. *Arab World English Journal*. 2021. № 1. С. 55–67. URL: <https://doi.org/10.24093/awej/covid.4> (date of access: 11.07.2023).
 4. Automated Attendance System in the Classroom Using Artificial Intelligence and Internet of Things Technology / D. D. Nguyen et al. *2021 8th NAFOSTED Conference on Information and Computer Science (NICS)*, Hanoi, Vietnam, 21–22 December 2021. 2021. URL: <https://doi.org/10.1109/nics54270.2021.9700991> (date of access: 11.07.2023).
 5. Deep Learning Based Automated Attendance System / J. Joshan Athanesious et al. *Procedia Computer Science*. 2019. Т. 165. С. 307–313. URL: <https://doi.org/10.1016/j.procs.2020.01.045> (date of access: 11.07.2023).
 6. Rahman S., Rahman M., Rahman M. M. Automated Student Attendance System using Fingerprint Recognition. *Edelweiss Applied Science and Technology*. 2018. С. 90–94. URL: <https://doi.org/10.33805/2576.8484.120> (date of access: 11.07.2023).
 7. Design and implementation of iris recognition-based attendance management system / A. Khatun et al. *2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, Dhaka, 21–23 May 2015. 2015. URL: <https://doi.org/10.1109/iceeict.2015.7307458> (date of access: 11.07.2023).

Лебединський Андрій Володимирович, док. філ., доц. кафедри комп'ютерних систем, тел. +38 093 660-57-04, lebedynskiy@khadi.kharkov.ua

Using JavaScript for Automating Attendance Tracking in the Moodle Learning System

Abstract. Problem: The article addresses the issue of automating attendance tracking for students in the context of distance learning in higher education institutions. **Methodology:** The paper analyzes existing methods of student attendance tracking in virtual learning environments, identifying their advantages and drawbacks. Additionally, it proposes a JavaScript-based solution for the automatic attendance tracking, generating a CSV file that is subsequently uploaded to the Moodle web server. **Goal:** The primary goal of the article is the development and exploration of a software tool capable of automating the process of tracking student attendance in the online educational system Moodle. **Originality:** The paper introduces an original approach to solving the attendance tracking issue using the JavaScript programming language, providing an innovative character to the proposed methodology. **Results:** The developed JavaScript script enables the automatic creation of a CSV file with student attendance data, which is then uploaded to the Moodle web server. The results demonstrate the efficiency of this method. **Practical value:** The work holds significant practical value for educators and administrators in educational institutions, simplifying and automating the student attendance tracking process in the virtual learning environment based on the Moodle platform.

Key words: JavaScript, scripts, automation, attendance, programming, Moodle.

Lebedynskiy Andrii¹, PhD, Assoc. Prof., Computer Systems Department, tel. +38 093-660-57-04, lebedynskiy@khadi.kharkov.ua

¹Kharkiv National Automobile and Highway University, 25, Yaroslava Mudrogo str., Kharkiv, 61002, Ukraine.