

О. В. ОЛЬХОВСЬКА

кандидат фізико-математичних наук,
завідувач кафедри комп'ютерних наук та інформаційних технологій
Полтавський університет економіки і торгівлі
ORCID: 0000-0001-5366-5995

О. П. КОШОВА

кандидат фізико-математичних наук,
доцент кафедри комп'ютерних наук та інформаційних технологій
Полтавський університет економіки і торгівлі
ORCID: 0000-0003-0794-6774

Д. М. ОЛЬХОВСЬКИЙ

кандидат фізико-математичних наук,
доцент кафедри комп'ютерних наук та інформаційних технологій
Полтавський університет економіки і торгівлі
ORCID: 0000-0003-0313-6977

Д. С. СЕМИКОЗ

магістр спеціальності «Комп'ютерні науки»
Полтавський університет економіки і торгівлі
ORCID: 0009-0007-3755-695X

РОЗРОБКА WEB-ЗАСТОСУНКУ ДЛЯ ФОРМУВАННЯ РОЗКЛАДУ В ЗАКЛАДІ ВИЩОЇ ОСВІТИ

У даній роботі досліджена проблема розробки елементів програмного забезпечення для формування розкладу в вищому навчальному закладі. Були проаналізовані останні дослідження та публікації в цій галузі, які підтверджують актуальність даної проблеми. Сформульовано мету розробки елементів програмного забезпечення з використанням діаграми прецедентів. Застосунок для формування розкладу в ЗВО є важливим елементом забезпечення якості навчального процесу, оскільки дозволяє забезпечити оптимальне використання ресурсів, ефективний розподіл навчальних навантажень, зниження конфліктів та покращення якості навчання. При проектуванні застосунку необхідно враховувати потреби та очікування користувачів. Була розроблена діаграма прецедентів для викладачів, студентів та диспетчерської служби з описом їх дій при формуванні розкладу. Для цього було використано редактор діаграм для Windows і редактор векторної графіки Microsoft Visio із застосуванням сценарію виконання UML. Розглянуто основні задачі, які вирішуються при складанні розкладу занять. Сформовано рекомендовані умови для створення оптимального розкладу для студентів і викладачів. Побудовано блок-схему формування розкладу, що передбачає дотримання оптимальних обмежень, перевірку конфліктів при розподілі навчального навантаження і використанні аудиторного фонду, подальших змін за потребою. Розробка застосунку відбувається у вигляді сайту Single Page Application на мовах програмування PHP, JavaScript і їх фреймворках Laravel та Vue.js. Laravel – для розробки серверної частини додатків, а Vue.js – для розробки клієнтської частини. Описано, як встановлювати та налаштовувати ці фреймворки та як їх поєднувати для створення веб-додатків з багатофункціональними можливостями. Розробка застосунку для формування розкладу може допомогти зменшити кількість помилок, пов'язаних з ручним складанням розкладу, а також зменшити витрати на час та ресурси, які затрачаються на розклад.

Ключові слова: програмне забезпечення застосунку, формування розкладу, заклад вищої освіти.

O. V. OLHOVSKA

Candidate of Physical and Mathematical Sciences,
Head of the Department of Computer Sciences and Information Technologies
Poltava University of Economics and Trade
ORCID: 0000-0001-5366-5995

O. P. KOSHOVA

Candidate of Pedagogical Sciences,
Associate Professor at the Department of Computer Sciences
and Information Technologies
Poltava University of Economics and Trade
ORCID: 0000-0003-0794-6774

D. M. OLKHOVSKY

Candidate of Physical and Mathematical Sciences,
Associate Professor at the Department of Computer Sciences
and Information Technologies
Poltava University of Economics and Trade
ORCID: 0000-0002-0585-3307

D. S. SEMYKOZ

Master's Degree in "Computer Science"
Poltava University of Economics and Trade
ORCID: 0009-0007-3755-695X

DEVELOPMENT OF APPLICATION SOFTWARE ELEMENTS FOR THE FORMATION OF A SCHEDULE IN UNIVERSITIES

This work examines the problem of developing software elements for the formation of a schedule in universities. The latest studies and publications in this field were analyzed, which confirm the relevance of this problem. The goal of developing software elements using the precedent diagram is formulated. The application for creating a schedule in universities is an important element of ensuring the quality of the educational process, as it allows you to ensure optimal use of resources, effective distribution of educational loads, reduction of conflicts and improvement of the quality of education. When designing an application, it is necessary to take into account the needs and expectations of users. A precedent diagram was developed for teachers, students, and the dispatch service, describing their actions in the formation of the schedule. For this, the diagram editor for Windows and the Microsoft Visio vector graphics editor using the UML execution script were used. The main tasks that are solved when drawing up the schedule of classes are considered. Recommended conditions for creating an optimal schedule for students and teachers have been formed. A block diagram of the formation of the schedule has been built, which provides for the observance of optimal restrictions, the verification of conflicts in the distribution of the educational load and the use of the classroom fund, further changes as necessary. The development of the application takes place in the form of a Single Page Application site using the programming languages PHP, JavaScript and their Laravel and Vue.js frameworks. Laravel – for the development of the server part of applications, and Vue.js – for the development of the client part. It describes how to install and configure these frameworks and how to combine them to create rich web applications. Developing a scheduling application can help reduce the errors associated with manual scheduling and reduce the time and resources spent on scheduling.

Key words: application software, scheduling, institution of higher education.

Постановка проблеми

У більшості закладів вищої освіти (ЗВО) немає ефективної системи формування розкладу занять, що призводить до витрат часу та зусиль викладачів та адміністрації на його складання та корекцію. Недостатня автоматизація цього процесу призводить до помилок, затримок та неефективного використання ресурсів закладу. Розробка елементів програмного забезпечення для формування розкладу може вирішити ці проблеми та покращити ефективність процесу планування занять в ЗВО. Однак, розробка таких елементів може стикатися з рядом технічних та організаційних складнощів, таких як інтеграція з існуючими системами управління навчальним процесом, забезпечення безпеки даних та зручності використання користувачами різного рівня підготовки.

Аналіз останніх досліджень та публікацій

Розклад навчальних занять – важливий документ закладу вищої освіти, яким регламентується академічна робота студентів і викладачів. Вимоги, які враховуються при складанні розкладу навчальних занять, такі [1]:

- виконання робочих навчальних планів та графіків навчального процесу;
- створення належних умов праці для професорсько-викладацького складу;
- рівномірний розподіл навчального навантаження впродовж тижня (бажано уникати наявності «вікон») та дотримання усіх вимог щодо кількості пар на день, тиждень тощо;
- ефективного використання аудиторного фонду.

Одним з підходів до розробки застосунків для формування розкладу є використання алгоритмів оптимізації. Ці алгоритми дозволяють знайти оптимальне рішення для складання розкладу, враховуючи різні обмеження та вимоги, такі як кількість студентів, кількість викладачів, наявність аудиторій, розклад підгруп тощо. Ще одним підходом є використання машинного навчання та штучного інтелекту. Застосування цих технологій дозволяє автоматично аналізувати дані про студентів та викладачів, враховувати їхні переваги та обмеження та швидко складати оптимальний розклад. Окрім того, існують деякі готові рішення, які вже успішно використовуються в різних ЗВО.

Актуальність розробки програмного забезпечення застосунку для формування розкладу в навчальному закладі можна обґрунтувати з точки зору забезпечення ефективності та точності процесу складання розкладу, зменшення кількості помилок та оптимізації витрат на розклад.

Мета роботи

Розробка програмного забезпечення застосунку для формування розкладу в ЗВО, що допоможе автоматизувати процес формування розкладу.

Виклад основного матеріалу

Програмне забезпечення для формування розкладу потребує розуміння вимог користувачів, дослідження процесу формування розкладу та знання технологій розробки програмного забезпечення.

Першим етапом розробки є визначення потреб користувачів та вимог до функціональності програмного забезпечення. Наприклад, користувачі можуть бажати мати можливість вибору предметів та груп, редагування розкладу та додавання відповідних даних до розкладу. Також, вимоги можуть включати можливість попереднього перегляду розкладу на тиждень, місяць або семестр.

Другим етапом розробки програмного забезпечення є дослідження процесу формування розкладу в ЗВО. Розклад формується на основі даних про доступність приміщень, доступність викладачів та їх освітніх компонентів, а також доступність студентів та їх груп.

Третім етапом є вибір технологій розробки програмного забезпечення та розробка його елементів. Наприклад, для розробки веб-застосунку можуть використовуватись технології HTML, CSS, JavaScript та PHP.

Перший етап. Діаграма прецедентів демонструє можливі дії користувачів, які будуть взаємодіяти з застосунком для формування розкладу. Вона показує, що застосунок для формування розкладу може бути корисним для різних груп користувачів, таких як викладачі, студенти, адміністратори, які відповідають за розклад університету. Застосунок може допомогти їм швидко та ефективно створювати розклад, враховуючи всі необхідні обмеження та вимоги.

Окрім того, розробка застосунку для формування розкладу може допомогти зменшити кількість помилок, пов'язаних з ручним складанням розкладу, а також зменшити витрати на час та ресурси, які затрачаються на розклад.

Застосунок повинен бути здатний забезпечувати різні функції для різних груп користувачів.

Для викладачів:

- відображення розкладу за допомогою фільтрів;
- перегляд розкладу занять та іспитів;
- коригування розкладу із-за відповідних причин та автоматичне надсилання запиту про зміну диспетчерській службі.

Для студентів:

- відображення розкладу за допомогою фільтрів;
- перегляд розкладу;
- надсилання запиту викладачу про перенесення заняття або некоректність розкладу.

Для працівників диспетчерської служби:

- встановлення обмежень та вимог щодо розкладу;
- створення розкладу занять та іспитів;
- відслідковування процесу формування розкладу;
- відображення розкладу за допомогою фільтрів;
- перегляд розкладу;
- опрацювання запитів про зміну розкладу;
- редагування розкладу;
- перевірка наявності конфліктів у розкладі та їх вирішення.

Процес створення діаграми прецедентів для формування розкладу можна описати наступним чином:

1. Визначення основних акторів системи – визначається, які основні користувачі будуть взаємодіяти з системою формування розкладу.

2. Визначення основних прецедентів – визначається, які дії може виконувати актор у системі.

3. Визначення залежностей між прецедентами – визначається, які прецеденти взаємодіють між собою і яка є послідовність виконання цих прецедентів.

4. Створення діаграми прецедентів – на основі отриманої інформації складається діаграма прецедентів для системи формування розкладу.

5. Перевірка та вдосконалення діаграми – після створення діаграми прецедентів необхідно перевірити, чи відображає вона усі дії, які може виконувати актор у системі, та чи відображено правильну послідовність виконання цих дій. У разі необхідності діаграма може бути вдосконалена.

З огляду на це було створено діаграми для кожної групи користувачів. Для цього використано редактор діаграм для Windows і редактор векторної графіки Microsoft Visio із застосуванням сценарію виконання UML [2].

Викладачі можуть переглядати розклад і коригувати його за потреби, але зміни відбудуться лише після опрацювання відповідного запиту працівником диспетчерської служби (рис. 1) [3].

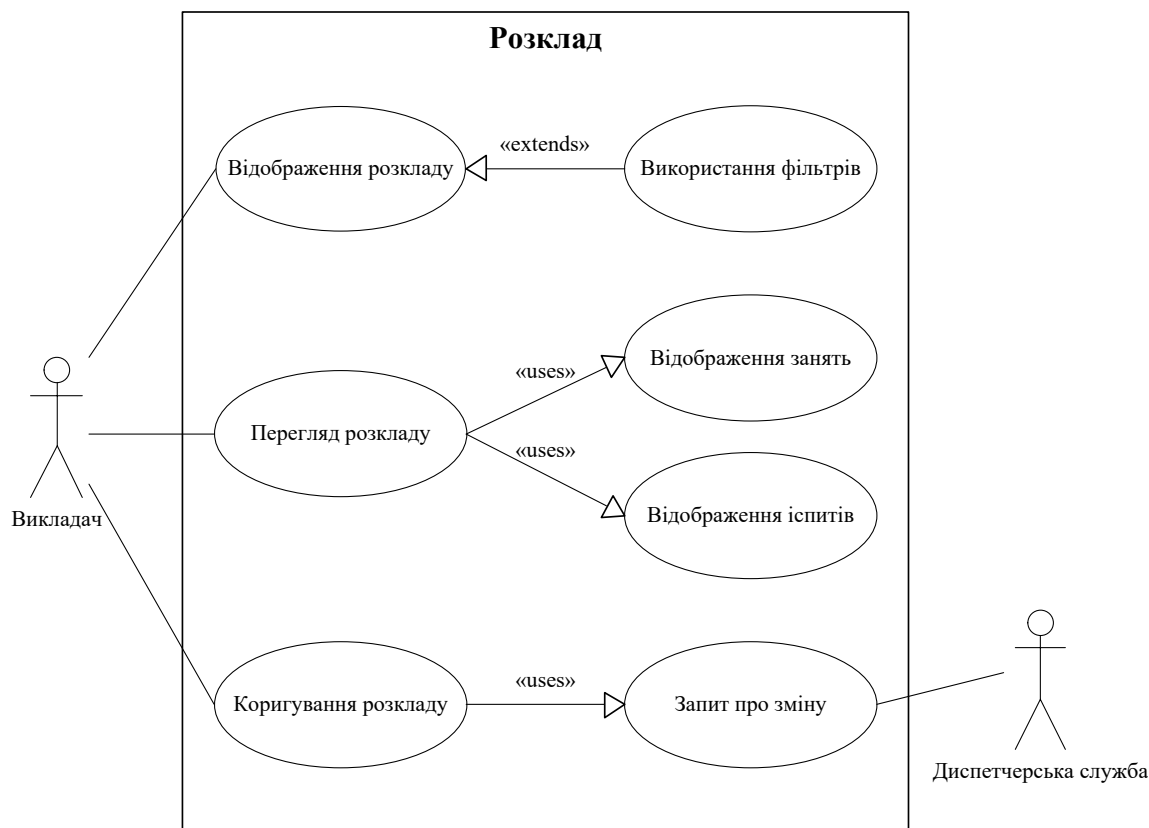


Рис. 1. Діаграма прецедентів для викладачів

Диспетчерська служба має повний доступ до формування та керування розкладом і приймає запити про подальші зміни в ньому. Важливим пунктом є перевірка конфліктів і їх вирішення, оскільки вони обов'язково будуть з'являтися (рис. 2) [3].

Студенти лише можуть переглядати розклад і при необхідності попросити викладача про перенесення занять. Якщо ж на погляд студентів розклад для групи сформовано некоректно для відвідування чи навчання, то також можна звернутися до викладача (рис. 3) [3].

Другий етап. Основні задачі, які вирішуються при складанні розкладу занять:

- виконання робочих навчальних планів та графіків навчального процесу;
- створення оптимального режиму роботи студентів протягом семестру;
- створення оптимальних умов праці для викладацького складу;
- рівномірний розподіл навчального навантаження впродовж тижня;
- ефективне використання аудиторного фонду.

Розклад навчальних занять повинен складатися на базі облікових одиниць навчального часу здобувачів освіти. Кредит ЄКТС – одиниця вимірювання обсягу навчального навантаження здобувача вищої освіти, необхідного для досягнення визначених (очікуваних) результатів навчання.

Навчальний день – складова частина навчального часу здобувачів вищої освіти тривалістю не більше 9 академічних годин. Таким чином, розклад занять здобувачів освіти не повинен перевищувати 4-х навчальних занять на день з урахуванням перерв між ними. Максимальне тижневе аудиторне навантаження за розкладом занять не повинно перевищувати 36 годин [4].

Студентоцентрирована організація навчального процесу передбачає забезпечення максимально прийнятних умов для навчання здобувачів вищої освіти, у тому числі і за рахунок оптимального розкладу навчальних занять. Рекомендовані умови для створення оптимального розкладу [4]:

- від 2-х до 4-х занять (пар) на день для академічної групи (підгрупи);
- рівномірний розподіл занять за днями тижня;
- щільне групування занять (мінімізація «вікон» між заняттями).

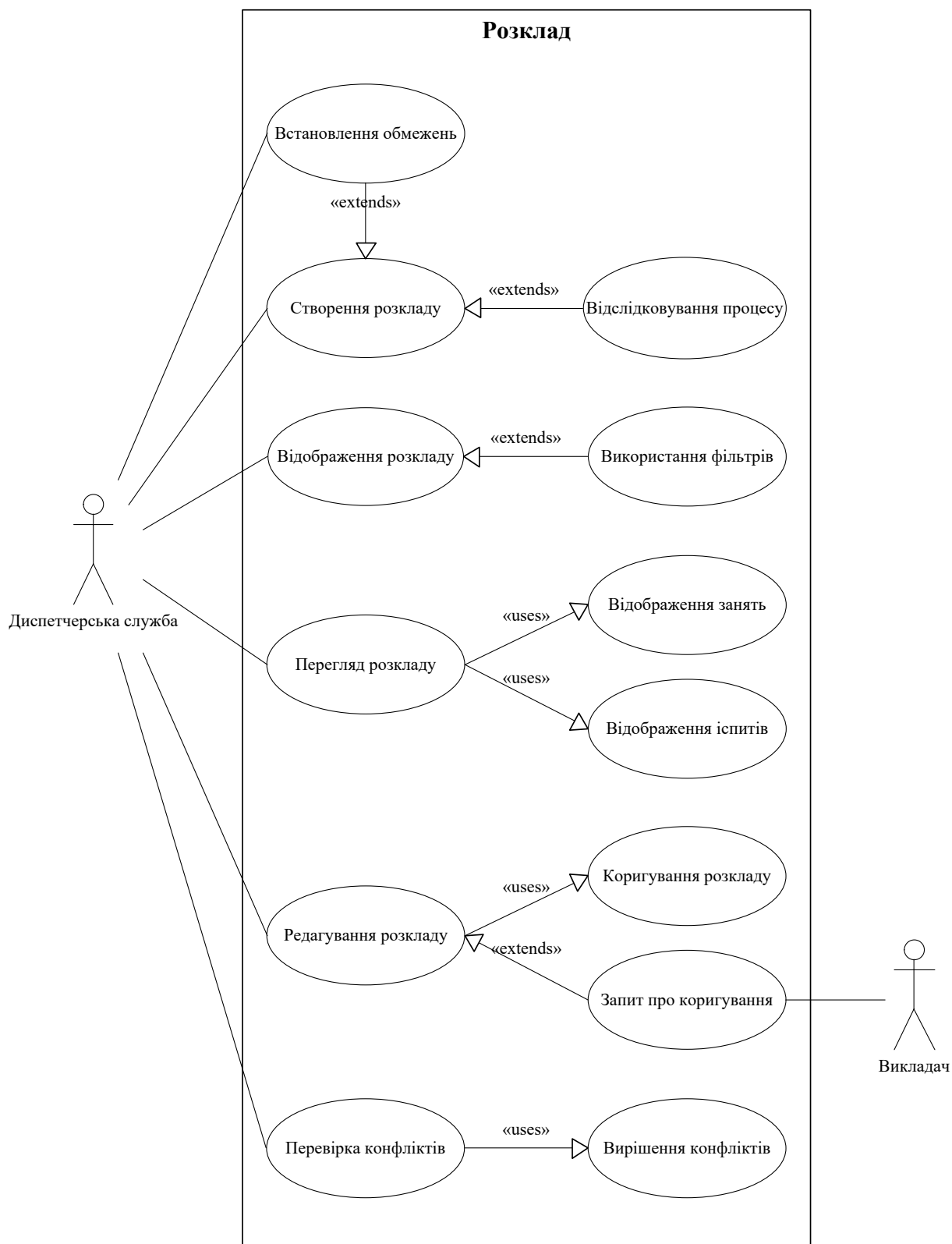


Рис. 2. Діаграма прецедентів для диспетчерської служби

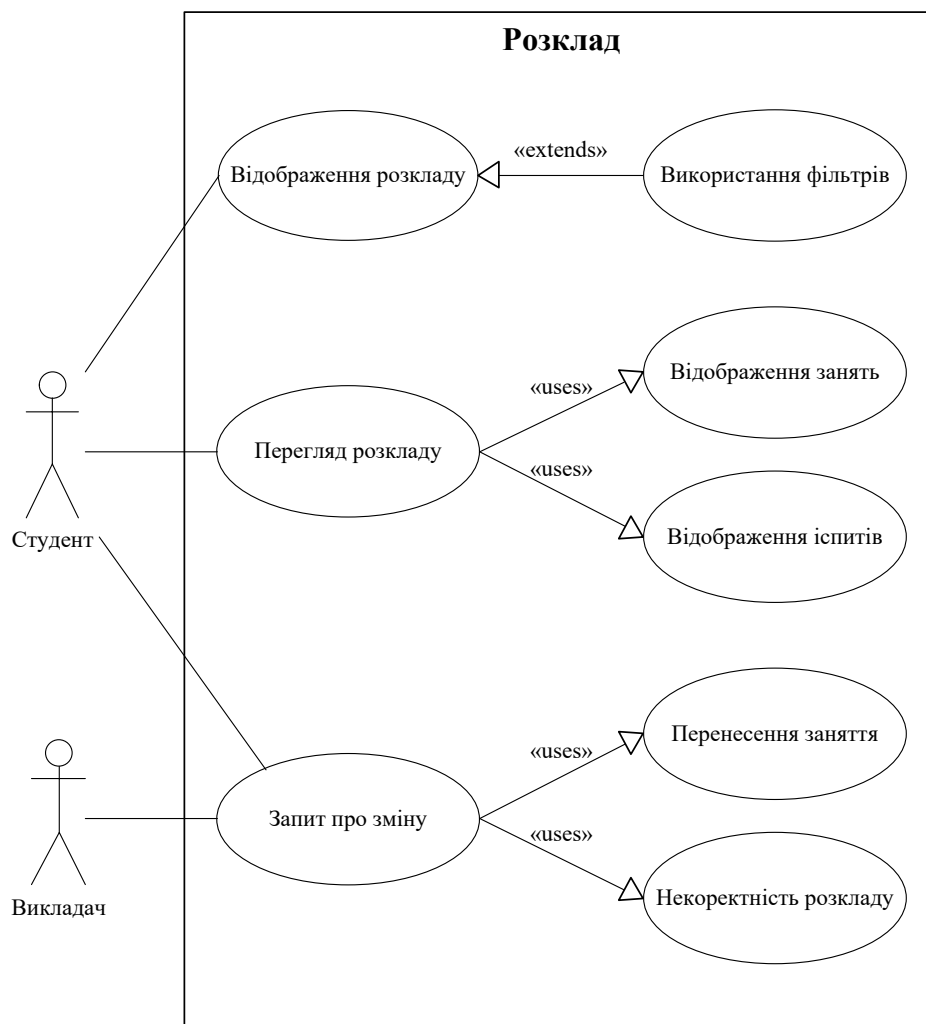


Рис. 3. Діаграма прецедентів для студентів

На основі цього було створено блок-схему формування розкладу, що передбачає дотримання оптимальних обмежень, перевірку конфліктів при розподілі навчального навантаження і використанні аудиторного фонду, подальших змін розкладу за потребою (рис. 4).

Третій етап. Розробка програми у вигляді сайту Single Page Application (SPA) є досить актуальною в наші дні, особливо з урахуванням зростання популярності веб-додатків та зменшення обсягу роботи, що потрібен для створення настільних програм.

Основна перевага SPA полягає в тому, що він забезпечує більш плавну та швидку взаємодію з користувачем, тому що сторінки не перезавантажуються під час навігації. Це дозволяє зменшити витрати на обробку та передачу даних, що забезпечує покращення продуктивності та швидкості веб-додатка.

Розробка сайту відбувається на двох мовах програмування, а саме PHP і JavaScript. Були використані Laravel та Vue.js – це два потужних інструменти для розробки сучасних веб-додатків. Laravel – це фреймворк для розробки серверної частини додатків, а Vue.js – це фреймворк для розробки клієнтської частини.

Розробка сайту з використанням Laravel та Vue.js дозволяє забезпечити ефективну та зручну розробку додатків з високою продуктивністю та інтерактивністю. Laravel дозволяє забезпечити швидку та зручну розробку серверної частини додатка, включаючи роботу з базою даних, аутентифікацію та авторизацію користувачів, маршрутизацію та інші функції.

Vue.js забезпечує зручну та швидку розробку клієнтської частини додатка, зокрема, забезпечує роботу зі станом додатка, динамічним оновленням сторінок та інші функції.

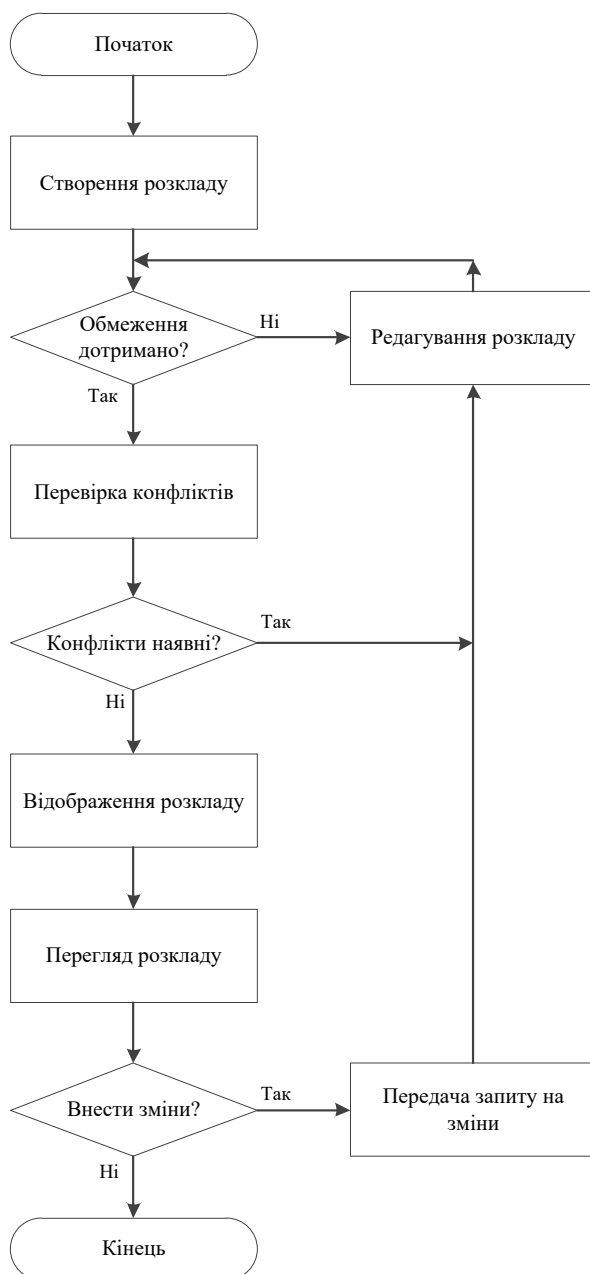


Рис. 4. Блок-схема формування розкладу

На етапі реалізації серверної частини створюються моделі бази даних, міграції, контролери, сервіси, провайдери та інші компоненти. Клієнтська частина – створюються компоненти, директиви, фільтри, сторінки та інші елементи, що забезпечують взаємодію з користувачем. Після створення серверної та клієнтської частин додатку проводиться їх інтеграція, що передбачає підключення до бази даних, взаємодію між сервером та клієнтом, обробку запитів.

Встановлюється Laravel за допомогою Composer, виконавши команду в терміналі:

```
composer create-project --prefer-dist laravel/laravel Rozklad
```

Після успішного встановлення було скопійовано «.env.example» файл в «.env» і виконано команду, щоб згенерувати ключ за шифрування: *php artisan key:generate*. Налаштування з'єднання з базою даних в файлі «.env» відбувається через заповнення наступних полів: DB_DATABASE, DB_USERNAME та DB_PASSWORD. Потім створюються таблиці в базі даних: *php artisan migrate*. Для запуску сервера виконується команда: *php artisan serve*. Щоб встановити глобальний пакет vue-cli, слід виконати команду в терміналі: *npm install -g vue-cli*. Створюється новий проект: *vue init webpack rozklad*. Встановлюються залежності проекту і його запуск, виконавши команди: *npm install, npm run dev*.

Після цього вже виконуються додаткові налаштування і розпочинається процес розробки застосунку для формування розкладу в ЗВО.

Висновки

Таким чином, реалізовано web-застосунок для формування розкладу в ЗВО, що допоможе автоматизувати процес формування розкладу. Визначено основні групи користувачів і дії, які вони можуть виконувати у системі. Так викладачі і студенти мають схожий набір дій з розкладом, а працівники диспетчерської служби мають повний доступ до формування та керування розкладом. Для кожної групи користувачів створена відповідна діаграма прецедентів. Розглянуто процес формування розкладу в ЗВО і побудована відповідна блок-схема формування розкладу. Розробка застосунку відбувається у вигляді сайту Single Page Application (SPA) на мовах програмування PHP і JavaScript з використанням фреймворків Laravel та Vue.js. Планується подальша реалізація застосунку, його тестування і внесення додаткових налаштувань для покращення роботи з розкладом.

Список використаної літератури

1. Голуб Б.Л., Ветрова Д.В., Пронішина К.О. Програмна система формування розкладу занять у закладі вищої освіти [Електронний ресурс]. *Математичні машини і системи*. 2019. № 4. С. 100-109. Режим доступу: <https://doi.org/10.34121/1028-9763-2019-4-100-109>
2. Створення схеми сценарію виконання UML. [Електронний ресурс]. Режим доступу: <https://support.microsoft.com/uk-ua/office/створення-схеми-сценарію-виконання-uml-92cc948d-fc74-466c-9457-e82d62ee1298>
3. Як будувати UML-діаграми. Розбираємо три найпопулярніші варіанти. Режим доступу: <https://dou.ua/forums/topic/40575/>
4. Положення про розклад навчальних занять у Харківському національному економічному університеті імені Семена Кузнеця. *Офіційний сайт Харківського національного економічного університету імені Семена Кузнеця*. Режим доступу: <https://www.hneu.edu.ua/wp-content/uploads/2018/11/Polozhennya-pro-rozklad-navchalnykh-zanyat-u-HNEU-2018.pdf>

References

1. Holub B.L., Vetrova D.V., Pronishyna K.O. (2019). Prohramna systema formuvanniarozkladuzaniatuzakladi vyshchoi osvity [Prohramna systema formuvannia rozkladu zaniat u zakladi vyshchoi osvity]. *Matematychni mashyny i systemy [Mathematical Machines and Systems]*, no. 4, P.100–109. Access mode: <https://doi.org/10.34121/1028-9763-2019-4-100-109>
2. Stvorennia skhemy stsenariiu vykonannia UML [Create a UML use case diagram [Electronic resource]]. Access mode: <https://support.microsoft.com/en-us/office/create-a-uml-use-case-diagram-92cc948d-fc74-466c-9457-e82d62ee1298>
3. Iak buduvaty UML-diahramy. Rozbyraiemo try naipopuliarnishi varianty. [How to build UML diagrams. We analyze the three most popular options [Electronic resource]]. Access mode: <https://dou.ua/forums/topic/40575/>
4. Polozhennia pro rozklad navchalnykh zaniat u Kharkivskomu natsionalnomu ekonomichnomu universyteti imeni Semena Kuznetsia [Regulations on the schedule of educational classes at Semyon Kuznets Kharkiv National University of Economics]. *Ofitsiynyi sait Kharkivskoho natsionalnoho ekonomichnoho universytetu imeni Semena Kuznetsia*. [The official website of Semyon Kuznets Kharkiv National University of Economics]. Access mode: <https://www.hneu.edu.ua/wp-content/uploads/2018/11/Polozhennya-pro-rozklad-navchalnykh-zanyat-u-HNEU-2018.pdf>