

Міністерство освіти і науки України Харківський національний
університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
(повна назва)

Кафедра Автоматизації проектування обчислювальної техніки
(повна назва)

АТЕСТАЦІЙНА РОБОТА

Пояснювальна записка

рівень вищої освіти другий (магістерський)
(рівень вищої освіти)

Інтелектуальний сервіс управління науково-педагогічним персоналом
кіберуніверситету

(тема)

Виконав: студент 2 курсу, групи СКСм-18-1

Кондрюков С.Е.

(прізвище, ініціали)

Спеціальність 123 – Комп'ютерна інженерія

(код і повна назва спеціальності)

Тип програми Освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Спеціалізовані комп'ютерні
системи

(повна назва освітньої програми)

Керівник доц. каф., АПВТ, Кулак Е.М.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

(підпис)

Чумаченко С.В.

(прізвище, ініціали)

2019 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
 Кафедра Автоматизації проектування обчислювальної техніки
 Рівень вищої освіти другий (магістерський)
 Спеціальність 123 – Комп'ютерна інженерія Тип
 програми Освітньо-професійна Освітня
 програма Спеціалізовані комп'ютерні системи

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« ____ » _____ 20 ____ р.

ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

студентові Кондрюкову Станіславу Едуардовичу
 (прізвище, ім'я, по батькові)

1. Тема роботи Інтелектуальний сервіс управління науково-педагогічним персоналом кіберуніверситету

затверджена наказом по університету від 04 11 2019 р. № 1624ст

2. Термін подання студентом роботи до екзаменаційної комісії 09 12 2019 р.

3. Вихідні дані до роботи мови програмування C# та Typescript, ASP.NET Core, MongoDB, Angular, _____
постанова Кабінету Міністрів України 1187

4. Перелік питань, що потрібно опрацювати в роботі _____
Проаналізувати Ліцензійні умови та постановка задачі, _____

_Проаналізувати умови додатку 12 постанови КМ України 1187 від 23.05.2018, _____

_Розробити програмний комплекс для вирахування індивідуального рейтингу _____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів)

15 слайдів

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання	02.09.2019- 07.09.2019	
2	Аналіз предметної області	08.09.2019-24.09.2019	
3	Аналіз джерел з проблемної галузі	25.09.2019-05.10.2019	
4	Дослідження постанови КабМіну	06.10.2019-19.10.2019	
5	Дослідження прикладних технологій	20.10.2019-02.11.2019	
6	Розробка програмного забезпечення	03.11.2019-16.11.2019	
7	Оформлення пояснювальної записки	17.11.2019-30.11.2019	
8	Оформлення графічного матеріалу	01.12.2019-16.12.2019	
9	Перевірка виконаного проекту керівником	17.12.2019-28.12.2019	
10			

Дата видачі завдання 02.01.2019 р.

Студент _____
(підпис)

Керівник роботи _____ доц. каф., АПВТ. Кулак Е.М.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка містить 73 сторіноки, 11 рисунків, 26 формул, 13 джерел аз переліком посилань.

РЕЙТИНГУВАННЯ, ИНТЕЛЕКТУАЛЬНА СИСТЕМА, ВНЗ, НАУКОВО-ПЕДОГОГІЧНИЙ ПЕРСОНАЛ

Метою атестаційної рботи є розробка масштабованой кіберфізичної системи online взаємодії засобів цифрового моніторингу творчої діяльності і хмарних сервісів управління кадровими ресурсами університету.

У ході виконання атестаційної роботи булирозглянуті принципи рейтингування відповідно до діючих правил ліцензування освітньої діяльності. Розглянуто принципи та особливості структури системи управління персоналом,

а також вибір технологічних підходів. Запропоновано концепцію інтелектуальної системи на основі потреб.

ABSTRACT

The explanatory note contains: 73 pages, 11 figures, 26 formulas, 13 sources according to the list of links.

PERSONNEL, INTELLECTUAL SYSTEM, METRIC SYSTEM & RATING

The purpose of the certification work is to develop a scalable cyberphysical system of online interaction of digital monitoring tools for creative activity and cloud services management of the University's human resources.

In the course of performance of the appraisal work were considered the principles of rating in accordance with the current rules for licensing educational activities. The principles and features of the structure of the personnel management system, as well as the choice of technological approaches are considered. The concept of intelligent system based on needs is offered.

ЗМІСТ

ВСТУП	5
1 КІБЕРФІЗИЧНИЙ УНІВЕРСИТЕТ	7
2 АНАЛІЗ РІНЯ НАУКОВОЇ ТА ПРОФЕСІЙНОЇ АКТИВНОСТІ	17
3 ПРОГРАМНИЙ МОДУЛЬ АНАЛІЗУ КАДРОВИХ ВИМОГ	22
3.1 Огляд компонентів розробленої системи	23

3.2 Веб прикладний програмний інтерфейс (Web API)	24
3.3 Інтерфейс користувача.....	25
3.3.1 React	25
3.3.2 Angular	27
3.3.3 Vue.....	28
3.4 База даних.....	30
3.5 Інтерфейс вводу досягнень	44
3.6 Модуль вираховування індивідуального рейтингу.....	45
3.7 Розподілення прав доступу.....	49
4. ІНФОРМАЦІЙНА БЕЗПЕКА.....	51
4.1 Аутентифікація.....	51
4.2 Авторизація за допомогою JWT-токену	54
5 ПРОЦЕДУРИ ПІДРАХУНКУ ІНДИВІДУАЛЬНОГО РЕЙТИНГУ	57
6 ІНСТРУКЦІЯ АДМІНІСТРАТОРУ	68
ВИСНОВКИ	75
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	77

ДОДАТОК А Графічна частина атестаційної роботи **Ошибка! Закладка не определена.**

ДОДАТОК Б Тези на конференцію «Радіоелектроніка та молодь у ХХІ столітті» **Ошибка! Закладка не определена.**

ВСТУП

Сьогодні більшість вищих навчальних закладів мають власні рейтингові системи оцінювання діяльності науково-педагогічних працівників. Розроблення та впровадження системи потребує виконання великого обсягу робіт, пов'язаних

зі збором та обробленням інформації стосовно науково-педагогічних працівників.

Розв'язок цієї задачі через використання паперових технологій є досить важкою проблемою, оскільки потребує великих часових затрат та збільшує вірогідність помилки під час оброблення великого обсягу первинних документів. А враховуючи затвердження Кабінетом Міністрів України ліцензійних умов щодо провадження освітньої діяльності закладів освіти, якими встановлено нові організаційні, кадрові та ліцензійні умови, обсяг робіт значно збільшиться.

Тому, цілком обґрунтовано, було прийнято рішення про необхідність розроблення програмного продукту, за допомогою якого можливо вирішити питання автоматизації процесу оброблення інформації для виведення рейтингової оцінки кожного викладача університету, що дозволить істотно зменшити розмір залучених часових та людських витрат.

Об'єкт дослідження – масштабована кіберсоціальна система як фізична інфраструктура, на якій позиціонуються хмарний аналіз відповідності науковопедогогічного персоналу вищих навчальних закладах кадровим вимогам освітнього процесу.

Предмет дослідження – кіберсоціальні моделі процесу рейтингування в науково освітніх закладах, явищ для цифрового моніторингу і хмарного метричного управління кадровим складом університету.

Мета – розробка масштабованої кіберфізичної системи online взаємодії засобів цифрового моніторингу творчої діяльності і хмарних сервісів управління кадровими ресурсами університету.

Для досягнення поставленої мети необхідно вирішити наступні задачі:

- проаналізувати кадрові вимоги;

- проаналізувати вагу окремих показників кадрових вимог;
- розробити конвертор завантаження кадрових показників у форматі doc до бази даних;
- розробити програмну систему;
- розробити критерії аналізу кадрових вимог до персоналу.

Новизна полягає у розробці інтелектуальної системи для аналізу відповідності науково-педагогічного персоналу ВНЗ кадровим вимогам освітнього процесу, тим самим зменшується потрібність у спеціально навченим персоналом та можливості автоматизації підбіру персоналу.

Практичне значення полягає у тому, що данна система може використовуватися при аналізі науково-педагогічного персоналу ВНЗ та підрахунку досягнень, тим самим спрощуючи підбір персоналу та зменшуючу кількість помилок.

1 КІБЕРФІЗИЧНИЙ УНІВЕРСИТЕТ

Кіберпростір – це віртуальний простір, який дає змогу створювати зв'язок та/або зв'язки з громадськістю за допомогою функціонування сумісних систем зв'язку та забезпечення електронного зв'язку через Інтернет та/або інші глобальні мережі передачі даних.

Кіберфізична система – це механізм, який відслідковується або контролюється комп'ютерними алгоритмами і тісно пов'язаний з Інтернетом та його користувачами. У кіберфізичних системах програмне забезпечення дуже пов'язане з фізичними об'єктами.

У наш час все більше і більше сфер нашого життя переносять свою діяльність в кіберпростір. Обговорюються розробляються різноманітні концепції

кіберфізичних систем, щоб відкрити можливості для інновацій та значних покращень у соціальному середовищі та бізнес-процесах. Концепції кіберфізичних систем розробляються або успішно впроваджуються у виробничому середовищі, охороні здоров'я, відновлюваній енергії, розумних будівлях, транспорті, сільському господарстві та інших сферах.

Тому викладачі ХНУРЕ презентували статтю, в якій було запропоновано кіберсоціальну систему Smart Cyber University (CyUni), характеризується наявністю оцифрованого простору регуляторних правил, ретельним моніторингом та активним кібер-управлінням адресованих компонентів наукових та освітніх процесів, автоматичним генеруванням регуляторних впливів, незалежних від керівників кібер-рішень щодо управління фінансовими та людськими ресурсами. Отже, кіберсоціальна система Smart Cyber University (CyUni) зможе надати:

- точний моніторинг та кібер-управління науковими та навчальними процесами
- незалежні системи прийняття рішень щодо управління фінансовими та людськими ресурсами;
- виключення паперових носіїв з науково-освітнього процесу.

Компетентність – Метрична оцінка духовної, фізичної, емоційної, інтелектуальної та професійної культури людини, що визначає її значення для потенційного застосування знань, умінь та навичок при виконанні соціальної ролі в покращенні якості життя людей та збереженні екосистеми планети.

Метрика - спосіб вимірювання відстані між процесами або явищами шляхом порівняння їх параметрів.

Згідно з доповіддю, метрикою заможного вченого на міжнародному рівні є: знання високих технологій, іноземних мов та наукових досягнень, що підтверджується публікаціями з індексами наукових баз даних. В свою чергу метрика іміджу університету влючає:

- відсутність корупції і високий рівень зарплати співробітників;
- міжнародний рівень наукових досягнень і якості випускників;
- наявність критичної маси авторитетних в світі вчених;
- виступи науковців на провідних міжнародних конференціях;
- організація та проведення власних конференцій, спонсорованих суспільством IEEE, ACM;
- членство студентів і вчених в згаданих організаціях.

Метрика відносин – оцифрований набір повідомлень про документи та регуляторні впливів (накази, статуты положення, традиції і закони), які є основою для взаємодії, оперативного та стратегічного цифрового моніторингу та кібер-управління оцифрованими наукообразовательними процесами, кадрами, інфраструктурою, фінансовими та тимчасовими ресурсами, спрямовані на створення зовнішнього іміджу та внутрішнього морально-етичного клімату з метою досягнення європейського рівня: підготовлених спеці листів, наукових досліджень і якості життя співробітників.

Відносини в вузі – найважливіша складова системи, що формує успіх університету на ринку освіти. Все інше: цілі, кадри, інфраструктура, управління, наука та освіта - безпосередньо залежать від відносин. Ротація кадрів не створює якісно нову систему - заміна несправного компонента не приводить до нового сервісу, а лише відновлює стару функціональність. Щоб створити передовий європейський університет, слід де-факто реалізувати три аксіоми законних

(кібер) відносин: Керівник сприймає працівника на основі метрики конструктивної, ефективної діяльності вченого і відображає його заслуг: вчений отримує від керівника послугу, що відповідає його метриці, яка забезпечує кар'єрний ріст, моральні та матеріальні винагороди; вчений розглядає колегу через об'єктив захоплюючих досягнень і як зразок для наслідування.

Відносини - набір соціально-технологічних зв'язків між керівництвом, персоналом та підрозділами, що формує системну структуру університету для моніторингу та управління науково-освітніми процесами, людськими ресурсами, інфраструктурою, фінансовими та тимчасовими ресурсами на основі діючих законів, статутів, постанов, положень та традицій з метою:

- виконання актуальних наукових досліджень і створення ринкової науково-технічної продукції;
- пропозиції якісних освітніх сервісів і підготовки затребуваних ринком фахівців з академічними і науковими ступенями;
- підвищення іміджу вузу і залучення зовнішніх інвестицій;
- забезпечення високого рівня життя співробітників і формування позитивного моральноетічного клімату.

Мета CyUni-системи – поліпшення якості освітніх послуг та навчальних досягнень вищої освіти шляхом створення системи метричних відносин, яка регулює правила цифрового спостереження та активного керування хмарним кібер-процесом для наукових та освітніх процесів та дозволяє знищити корупцію, залучити зовнішні інвестиції, значно покращують продуктивність та конструктивний рівень життя професорів, які створюють ринково затребувану продукцію у вигляді випускників та навчальних досягнень. Створення розумного кібер-університету пов'язано з інтеграцією технологій big data, cloud computing,

mobile services і cyber physical systems в рамках IoT-культури на основі використання сервісорієнтованих платформ програмування, пропонованих провідними компаніями IBM, Google, Microsoft, NASA, Amazon, Facebook.

Хмарні сервіси Google розподіляються на наступні підгрупи:

- обчислення, до цієї підгрупи входять сервіс розробки додатків, сервіс хмарних обчислень, сервіс управління контейнерами;
- зберігання, що включає хмарне сховище даних, хмарне сховище сервісів, хмара управління даними, хмара великих таблиць (рис. 1.1);
- великі дані. Сервіс управління запитом, хмара потоків даних, хмара процедур обробки даних, хмара обміну повідомленнями;
- сервіси включають сервіс створення інтерфейсів, сервіс автоматичного перекладу, сервіс передбачення рішень.

Хмарні сервіси Google



Рисунок 1.1 – Хмарні сервіси Google

Для висвітлення вищезазначених питань кіберсоціальна система університету розділена на 12 сервісів (рис. 1.2):

□ хмарний кібер-сервіс захищеного електронного документообігу для цифрового моніторингу та інтелектуального кіберуправління науково-освітніми процесами (створення, реалізація та утилізація документа), в форматі замкнутого циклу: «факт - вимір - оцінка - дія», який повністю виключає паперові носії шляхом використання Cloud -Mobile Service Computing, баз даних, цифрового підпису, ID-card, пошти та мобільного телефону;

□ хмарний кібер-сервіс мобільного голосування e-voting для

моніторингу громадської думки; реалізації студентських опитувань; прийняття рішень на оперативних нарадах, засіданнях вченої ради, конференціях трудового колективу; проведення виборів експертів, студентського сенату, керівного та науково-педагогічного складу при заміщенні вакантних посад;

□ хмарний кібер-сервіс управління персоналом на основі online моніторингу, вимірювання, рейтингування та накопичення цифрових метрик компетенцій для оцінювання діяльності: студентів і всіх категорій співробітників з метою вироблення прозорих регуляторних моральних і матеріальних стимулів, вибору переможців з претендентів на вакантні позиції керівників і науково-педагогічних посад;

□ хмарний кібер-сервіс управління структурним підрозділом на основі online моніторингу, вимірювання та накопичення цифрових метрик компетенцій кафедри, пов'язаних з наукообразовательним процесом для вироблення регуляторних дій, що управляють і генерують пакет документів, необхідних для життєдіяльності;

□ хмарний кібер-сервіс оцінки якості освітніх процесів і компонентів, online тестування знань і умінь, що виключає нелегітимні відносини між викладачем і студентом при здачі іспитів і заліків;

□ хмарний кібер-сервіс управління науковими процесами на основі цифрового оцінювання діяльності вчених, підрозділів, наукових результатів, проектів і пропозицій по метриках, розробленим експертами, з метою прозорого та легітимного розподілу фінансових, кадрових і часових ресурсів між підрозділами і співробітниками;

□ хмарний кібер-сервіс надання освітніх послуг у вигляді MOOC

online і onsite курсів, а також управління освітнім процесом на основі прозорого розподілу фінансових і тимчасових (кредитних) ресурсів між підрозділами і співробітниками в суворій відповідності з метричних оцінюванням вкладу кожного суб'єкта в актив і імідж університету;

□ хмарний кібер-сервіс моніторингу та управління науково-освітнім процесом студента в реальному масштабі часу, генерування і зберігання електронних документів для його супроводу в часі і просторі за допомогою створення персонального віртуального кабінету, пов'язаного з мобільним пристроєм і e-mail;

□ хмарний кібер-сервіс вимірювання і супроводу бакалаврських, магістерських та дисертаційних робіт, а також конкурсних проектів на основі інтеграції міжнародних метрик оцінювання наукової та практичної значущості результатів проведених досліджень з внутрішніми критеріями якості, розробленими експертами;

□ хмарний кібер-сервіс ліцензування та акредитації спеціальностей на основі вимірювання науково-освітньої діяльності кафедр і подальшого генерування пакета документів, необхідного для зовнішнього оцінювання якості навчальних процесів;

□ хмарний кібер-сервіс електронного 24/7 доступу та моніторингу присутності співробітників і студентів в інфраструктурних аудиторіях університету на основі використання мобільних пристроїв і ID-card, а також електронний банкінг для оплати освітніх послуг і використання корпоративних кафедральних карт для придбання товарів і послуг в межах зароблених кафедрою засобів;

□ хмарний кібер-сервіс захисту інформаційно-фізичного простору університету і санкціонування електронного доступу в усі кіберфізическі компоненти і процеси, пов'язані з життєдіяльністю вузу.

Сервіси CyUni

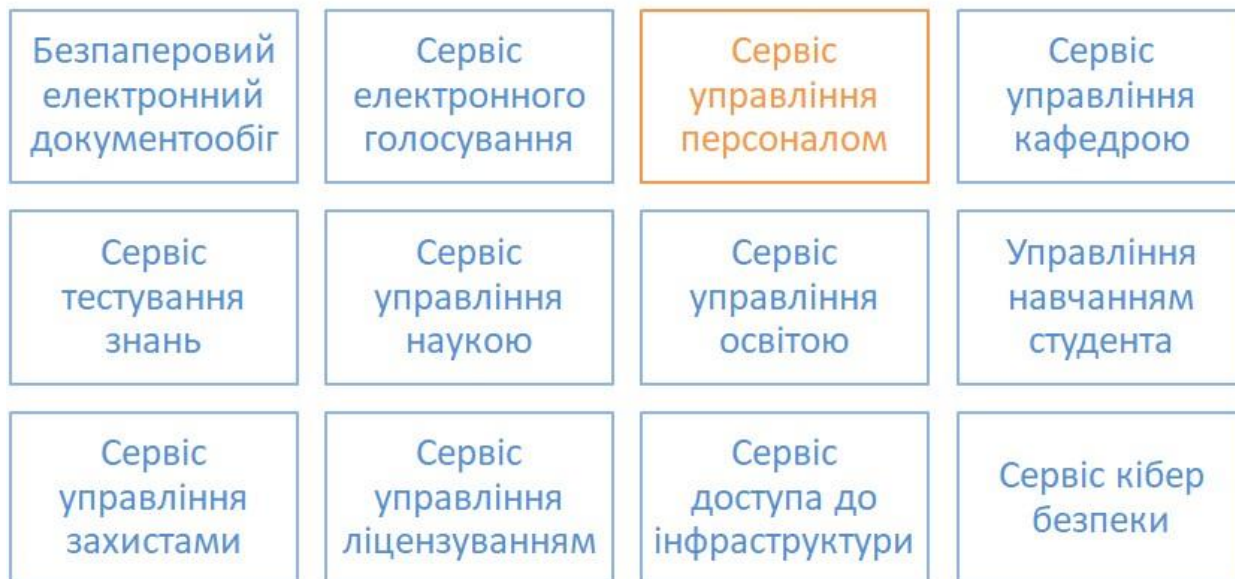


Рисунок 1.2 – Сервіси CyUni

Інтелектуальна система управління людськими ресурсами дозволяє здійснювати моніторинг, вимірювання, оцінку та накопичення цифрових метрик навичок для оцінки діяльності педагогічного колективу, забезпечувати прозорі моральні та матеріальні стимули до регулювання та вибирати переможців з кандидатів на керівні та керівні посади.

Електронний документообіг – законні інтелектуальні транзакції оцифрованих документообігів у мережі даних, призначені для здійснення безпаперових зв'язків із зовнішнім світом, безпосереднього нагляду та

безпосереднього управління науково-освітніми процесами та кафедрами університету.

За допомогою служби управління персоналом ви можете реалізувати електронний документообіг. Це означає можливість генерувати цифрові документи для створення таблиць показників та управління системою цифровими документами, зрозумілими кіберсистемою. Перехід до цифрових документів та автоматична обробка даних вказують на скорочення часу, необхідного для розрахунку оцінювання викладацького складу та людських ресурсів, що беруть участь у процесі.

Обчислювальна складова системи управління інтелектуальною власністю повинна містити метрику оцінювання рівня наукової та професійної діяльності вчених та викладачів на основі чинного освітнього законодавства в Україні.

Кожна метрика в кібер-службах університету повинна працювати зі змінними, які утворюють вимірювальний простір, який може бути як булевим, мовним, чисельним, так і нормалізованим в інтервалі $(0,1)$. Виразність інтервалу існування метричного параметра між нулем і одиницею залежить від послуг, пропонує споживачеві кіберфізичних систем: прийняття рішення - так чи ні; оцінки процесів або об'єктів для керівника - добре, погано, добре, задовільно завжди потребують коментаря; безперервний інтервал дійсних чисел від 0 до 1 точно визначає якість однотипних процесів або явищ і ранжирує їх відповідно до заданої метрикою компетенцій.

При розрахунку рейтингів не потрібно приймати жодних рішень, а використання нечітких критеріїв оцінювання не підходить для рейтингової системи. Тому найбільш очевидним та ефективним є показник, який використовує безперервний інтервал як оцінку параметра. Обраний підхід до

оцінки критеріїв дозволить отримати точні та прозорі показники, що, в свою чергу, сприяє здоровій конкуренції серед працівників університету.

Розробка данної системи дозволить перенести управління кадровим складом до сфери оцифрованих відношень, що призведе до:

- зменшення паперового документообігу для проведення рейтингових оцінювань;
- зменшення витрат людських ресурсів для складання рейтингів;
- прозорі та доступної інформації щодо досягнень викладачів.

2 АНАЛІЗ РІНЯ НАУКОВОЇ ТА ПРОФЕСІЙНОЇ АКТИВНОСТІ

Однією з функцій державного управління, тобто одним із засобів впливу держави на учасників сфери вищої освіти є процес ліцензування.

У 2018 р. Кабінет Міністрів України затвердив ліцензійні умови провадження освітньої діяльності закладів освіти, якими встановив нові організаційні, кадрові та ліцензійні умови, зокрема, для вищих навчальних закладів [4]. Цим документом затверджено єдині вимоги до провадження освітньої діяльності у сфері вищої освіти – на певному рівні вищої освіти за певною спеціальністю.

З метою забезпечення об'єктивності, прозорості процесу ліцензування та неупередженості при прийнятті рішень, прийнятою постановою затверджено єдиний перелік документів, що подаються для отримання ліцензії, єдині форми заяв про видачу ліцензії та подачі документів до органу ліцензування для всіх

суб'єктів здійснення освітньої діяльності. Загалом, вимоги щодо забезпечення провадження освітньої діяльності у сфері вищої освіти класифіковано за такими критеріями, як: кадрові, технологічні, організаційні.

Кадрові вимоги щодо провадження освітньої діяльності у сфері вищої освіти є принципово важливими в процесі ліцензування.

Проаналізувавши постанову Кабінету Міністрів України №1187 про ліцензійні умови провадження освітньої діяльності, виявлено що, відповідно до пункту 30 загальних положень постанови [4] рівень наукової та професійної активності науково-педагогічного (наукового) працівника визначається за допомогою наявності переліку документів, що підтверджують такі показники:

□ наявність за останні п'ять років наукових публікацій у періодичних виданнях, які включені до наукометричних баз, рекомендованих МОН, зокрема Scopus або Web of Science Core Collection;

□ наявність не менше п'яти наукових публікацій у наукових виданнях, включених до переліку наукових фахових видань України;

□ наявність виданого підручника чи навчального посібника або монографії;

□ наукове керівництво (консультування) здобувача, який одержав документ про присудження наукового ступеня;

□ участь у міжнародних наукових проектах, залучення до міжнародної експертизи, наявність звання «суддя міжнародної категорії»;

□ проведення навчальних занять із спеціальних дисциплін іноземною мовою в обсязі не менше 50 аудиторних годин на навчальний рік;

□ робота у складі експертних рад з питань проведення експертизи дисертацій МОН або галузевих експертних рад Національного агентства із забезпечення якості вищої освіти, або Акредитаційної комісії, або їх експертних рад, або міжгалузевої експертної ради з вищої освіти Акредитаційної комісії, або трьох експертних комісій МОН/зазначеного Агентства, або Науковометодичної ради/науково-методичних комісій (підкомісій) з вищої освіти МОН;

□ виконання функцій наукового керівника або відповідального виконавця наукової теми (проекту), або головного редактора/члена редакційної колегії наукового видання, включеного до переліку наукових фахових видань України, або іноземного рецензованого наукового видання;

□ керівництво школярем, який зайняв призове місце III-IV етапу Всеукраїнських учнівських олімпіад з базових навчальних предметів, II-III етапу Всеукраїнських конкурсів-захистів науково-дослідницьких робіт учнів - членів Національного центру «Мала академія наук України»; участь у журі олімпіад чи конкурсів «Мала академія наук України»;

□ організаційна робота у закладах освіти на посадах керівника (заступника керівника) закладу

освіти/інституту/факультету/відділення (наукової установи)/ філії/кафедри або іншого відповідального за підготовку здобувачів вищої освіти підрозділу/відділу (наукової установи)/навчальнометодичного управління (відділу)/лабораторії/іншого навчально-наукового (інноваційного) структурного підрозділу/вченого секретаря закладу освіти (факультету, інституту)/відповідального секретаря приймальної комісії та його заступника;

□ участь в атестації наукових працівників як офіційного опонента або члена постійної спеціалізованої вченої ради (не менше трьох разових спеціалізованих вчених рад);

□ наявність не менше п'яти авторських свідоцтв та/або патентів загальною кількістю два досягнення;

□ наявність виданих навчально-методичних посібників/посібників для самостійної роботи студентів та дистанційного навчання, конспектів лекцій/практикумів/методичних вказівок/рекомендацій загальною кількістю три найменування;

□ керівництво студентом, який зайняв призове місце на I етапі Всеукраїнської студентської олімпіади (Всеукраїнського конкурсу студентських наукових робіт), або робота у складі організаційного комітету/журі Всеукраїнської студентської олімпіади (Всеукраїнського конкурсу студентських наукових робіт), або керівництво постійно діючим студентським науковим гуртком/проблемною групою; керівництво студентом, який став призером або лауреатом Міжнародних мистецьких конкурсів, фестивалів та проектів, робота у складі організаційного комітету або у складі журі міжнародних мистецьких конкурсів, інших культурно-мистецьких проектів; керівництво студентом, який брав участь в Олімпійських, Паралімпійських іграх, Всесвітній та Всеукраїнській Універсіаді, чемпіонаті світу, Європи, Європейських іграх, етапах Кубка світу та Європи, чемпіонаті України; виконання обов'язків тренера, помічника тренера національної збірної команди України з видів спорту; виконання обов'язків головного секретаря, головного судді, судді міжнародних та всеукраїнських змагань; керівництво спортивною делегацією; робота у складі організаційного комітету, суддівського корпусу;

□ наявність науково-популярних та/або консультаційних (дорадчих) та/або дискусійних публікацій з наукової або професійної тематики загальною кількістю не менше п'яти публікацій;

□ участь у професійних об'єднаннях за спеціальністю;

□ досвід практичної роботи за спеціальністю не менше п'яти років;

□ наукове консультування установ, підприємств, організацій протягом не менше двох років.

Також у 30-му пункті зазначено, що під час визначення рівня наукової та професійної активності науково-педагогічного (наукового) працівника можуть зараховуватися здобутки за попередніми місцями роботи, п'ятирічний строк може продовжуватися на час перерви в роботі з об'єктивних причин (соціальна відпустка, призов/мобілізація на військову службу чи військова служба за контрактом, тривала хвороба тощо).

Для підтвердження кваліфікації науково-педагогічного працівника закладів вищої освіти, в яких здійснюється підготовка фахівців за мистецькими, замість наукових публікацій можуть зараховуватися такі здобутки: літературні твори, переклади літературних творів, твори живопису, декоративного мистецтва, архітектури, архітектурні проекти, скульптурні, графічні, фотографічні твори, твори дизайну, музичні твори, аудіо- та відео твори, передачі (програми) організації мовлення, медіатвори, сценічні постановки, концертні програми (сольні та ансамблеві) кінотвори, анімаційні твори, аранжування творів, рекламні твори.

Таким чином, на основі викладеного матеріалу, ми можемо стверджувати, що процес ліцензування у сфері вищої освіти слід розглядати як уніфікований порядок підготовки документів.

3 ПРОГРАМНИЙ МОДУЛЬ АНАЛІЗУ КАДРОВИХ ВИМОГ

Враховуючи показники ліцензування кадрового складу учбового закладу було сформульовано вимоги до системи рейтингування. Інтелектуальна система управління кадровим складом учбового підрозділу повинна включати:

- бази даних, які забезпечать зберігання інформації про індивідуальні досягнення викладача, а також рейтингові показники відповідно до зазначених пунктів;

- зручний інтерфейс користувача, що надасть можливість вводу досягнень, перегляд рейтингових таблиць, а також ефективне фільтрування згідно з заданими параметрами, перегляд детальної інформації про окремого викладача та його позицію у рейтинговій таблиці взагалі або за певним показником;

- процедури вирахування індивідуального рейтингу за параметрами кадрових вимог;

- розподілення прав доступу.

Розроблена концепція інтелектуальної системи керування кадровим складом повинна бути націлена та покривати такі потреби:

- зменшення навантаження на відділ управління якістю завдяки роздільного збору інформації на кафедрах;

- зменшення потоку паперових документів при наданні результатів своєї діяльності викладачами та структурними підрозділами вузу;

- автоматизація контролю оцінювання діяльності викладачів і структурних підрозділів вищого навчального закладу;

- можливість оперативного реагування при внесенні помилкових відомостей.

Переваги виконаної розробки:

- низькі вимоги до апаратної і програмної складових;
- підтримка розподілу прав доступу до введення, редагування і перегляду результатів оцінки діяльності викладачів і підрозділів;
- орієнтація на проведення оперативного та інтелектуального аналізу зібраних даних;
- гнучкі можливості масштабування системи; - низька собівартість використання.

3.1 Огляд компонентів розробленої системи

Враховуючи вище вказані вимоги до інтелектуальної системи управління кадровим складом учбового підрозділу був розроблений програмний комплекс, який складається з наступних компонентів:

- бази даних для зручного зберігання необхідної для роботи комплексу інформації;
- веб інтерфейс який забезпечує зручну та більш гнучку взаємодію між інтелектуальною системою та користувачами;
- веб прикладного програмного інтерфейсу (Web Application Program Interface – англ.) який забезпечує можливість коректної взаємодії між веб інтерфейсом користувача, базою даних та надає необхідний функціонал для роботи програмного комплексу.

3.2 Веб прикладний програмний інтерфейс (Web API)

У якості основи для реалізації Web API було обрано ASP.Net Core який є частиною платформи .Net. ASP.NET Core - це крос-платформна, високопродуктивна система з відкритим кодом для створення сучасних додатків, підключених до Інтернету на основі хмар. Мільйони розробників використовували і продовжують використовувати ASP.NET 4.x для створення веб-додатків. ASP.NET Core – це модифікація ASP.NET 4.x з архітектурними змінами, що формують більш раціональну і більш модульну платформу.

ASP.NET Core надає такі переваги:

- єдине рішення для створення призначеного для користувача вебінтерфейсу і веб-API;
- розроблено для тестованості;
- Razor Pages робить створення кодів сценаріїв для сторінок простіше і ефективніше;
- Blazor дозволяє використовувати в браузері мову C # разом з JavaScript. спільне використання серверної і клієнтської логік додатків, написаних за допомогою .NET;
- можливість розробки і запуску в ОС Windows, macOS і Linux.
- відкритий вихідний код і орієнтація на співтовариство;
- інтеграція сучасних клієнтських платформ і робочих процесів розробки;
- підтримка розміщення служб віддаленого виклику процедур (RPC) за допомогою gRPC;
- хмарна система конфігурації на основі середовища;

- вбудоване введення залежностей;
- спрощений високопродуктивний модульний конвеєр HTTP-запитів;
- управління паралельними версіями;
- інструментарій, що спрощує процес сучасної веб-розробки.

3.3 Інтерфейс користувача

У данному підрозділі буде огляд найбільш популярних фреймворків серед котрих був обран Angular як найбільш відповідний до поставлених завдань.

Основні три фреймворки у які будуть описані це React, Angular, Vue.

3.3.1 React

Випущений в 2013 році на Facebook. В основному використовується на сайтах з високим трафіком. Він був розроблений, коли реклама в Facebook почала набирати трафік і зіткнулася з проблемами в кодуванні і обслуговуванні, які відображали певні проблеми. Остання версія 16.9 була випущена 8 серпня 2019 року. Використовується Facebook, Uber.

React сам по собі це не фреймворк, а бібліотека, далі під React буде матися на увазі ціла екосистема, яка знаходиться в розробці додатків з React. Необхідно самостійно підбирати собі компоненти системи, щоб вони працювали разом.

Зараз React є найпопулярнішим вибором для рішень завдань. Але якщо в основі архітектури проекту лежить React, то це зумовлює:

- необхідність пошуку і впровадження додаткових бібліотек для реалізації кожної з задач;
- налаштування функціональної частини програми під конкретну бібліотеку;

- складність залучення до проекту інших розробників, через різницю в стеці бібліотек кожної програми.

Через високу складність оточення і необхідних знань важко вибрати React для задач.

React пропонує просте, легке створення компонентів - функціональні компоненти без збереження стану так само прості, як і раніше. Крім того, елегантний API React пропонує використовувати композицію з використанням компонентів. React дуже популярний, про що свідчить його велике співтовариство. Популярність React допомогла залучити стартапи. Наявність широкого спектру відкритих, розроблених співтовариством розширень для React дає безліч варіантів для побудови повних рішень.

Але в той же час шаблони компонентів написані з використанням JSX, а не нативного HTML, тому слід очікувати деякого періоду коригування. Оскільки він зосереджений на створенні призначених для користувача інтерфейсів, ймовірно, буде потрібно розширити React сторонніми бібліотеками. Наприклад, якщо потрібно підтримувати клієнтську маршрутизацію в додатку, ймовірно, доведеться використовувати популярну сторонню бібліотеку React Router. На щастя, є багато відмінних варіантів для виконання практично будь-яких вимог. Але наявність більшого вибору може бути приголомшливим або сбиваючим з пантелику - особливо для початківців.

У підсумку як позитивні пункти ми маємо: □

- просте і легке створення компонентів;
- елегантний API, який підтримує композицію;
- велике співтовариство підтримки;
- популярний у стартапів;

- безліч розширень з відкритим вихідним кодом.

Як негативні пункти маємо:

- налаштування JSX;
- повні розв'язання потребують сторонніх бібліотек;
- наявність опцій може перевантажувати або збивати з пантелику.

3.3.2 Angular

Випущений в 2010 році компанією Google. Це typescript орієнтований фреймворк. Сьогодні він називається просто Angular, але до 2016 року до його назви було додано суфікс «JS». JS був виключений з назви після виходу Angular 2+. 28 травня 2019 року було випущено остання версія Angular 8.0.0. Google і Wix є одними з найбільш популярних компаній, що використовують Angular.

Angular є фреймворком, так як в ньому всі необхідні інструменти і компоненти для роботи з коробки більше ніж у React або Vue. В результаті він надає більше рекомендацій про те, як створювати повні рішення. Він орієнтований на розробку додатків, про що свідчить включення таких функцій, як маршрутизація і перевірка форми.

Angular має найбільш функціональну CLI з трьох платформ. Він використовується під час розробки програми для створення нових компонентів, установки пакетів і виконання сценаріїв міграції та установки. TypeScript, з його статичної типізацією і дивовижними покращеннями редактора, пропонує досвідченим розробникам, у яких немає великої кількості JavaScript, більш дружній перехід до розробки на стороні клієнта. І хоча для звикання до його синтаксису прив'язки може знадобитися деякий час, можливість

використовувати власний HTML і CSS для шаблонів і стилів компонентів є плюсом.

Але великий API і загальний підхід Angular до розробки додатків призводить до самої крутій кривий навчання серед цих трьох фреймворків. Angular код, з його використанням статичних типів, декораторів та конфігураційного підходу, може здатися багатослівним і в рази більше складним у порівнянні з React і Vue.

У підсумку як позитивні пункти ми маємо:

- все включено у фреймворк;
- керівництво по створенню комплексних рішень;
- популярний у підприємств;
- багатофункціональний CLI;
- TypeScript;
- підтримка нативного HTML і CSS.

Як негативні пункти маємо:

- великий API;
- крута крива навчання;
- код може бути багатослівним і складним.

3.3.3 Vue

Випущений в 2014 році і розроблений колишнім інженером Google Еваном Ю. Це набирає постійну популярність. Його остання версія; Версія 2.6.10 була випущена 20 березня 2019 року. Найпопулярнішими компаніями використовують Vue це Gitlab і Alibaba.

Vue володіє плюсами React і Angular, так як є фреймворком, в якому з коробки вже є все необхідне, як в angular, але там також можна міняти практично всі компоненти системи на кастомніє як в React.

Vue має вбудовану прив'язку даних і модель MVC (модель, вид, контролер), що значно спрощує настройку в порівнянні з Angular і React.js. Платформа підтримує легшу інтеграцію з елементами HTML. У порівнянні з Angular, Vue набагато легше вивчати, розуміти і використовувати. Як і Angular, Vue також дозволяє вам використовувати свої власні навички роботи з HTML і CSS замість того, щоб вивчати нову мову шаблонів, як у випадку з React. Також він відомий своєю добре написаної документацією.

Але Vue.js – це відносно новий мову, яка не підтримується великою спільнотою активних програмістів. Аналогічним чином, команда розробників Vue.js також невелика, що означає, що платформі буде потрібно якийсь час, щоб отримати корпоративну підтримку. Станом на 2018 рік розробка структури не пов'язана з корпоративним інтересом; тому будь-які зміни в існуючій структурі в основному залежать від відгуків членів спільноти.

У світі JavaScript прийняття певної технології завжди можна розглядати як ставку проти стабільності платформи. Більша гнучкість і безліч опцій також можуть стати проблемою при розгортанні великих проектів, так як участь різних розробників може бути ускладнене. Як і всі речі, Vue поставляється з тим, що ми можемо вважати недоліками. З одного боку, його гнучкість: в той час як деякі розробники користуються свободою, яку надає Vue, інші вважають за краще мати тільки один правильний спосіб створення нових компонентів. Нарешті, багато компаній вибирають React замість Vue через меншого числа доступних програмістів Vue.js. Розробникам React пропонується набагато більше

пропозицій про роботу і, природно, на ринку з'являється більше розробників React. Компанії можуть захотіти пом'якшити потенційні проблеми з наймом, вибравши більш популярну бібліотеку для свого інтерфейсу.

У підсумку як позитивні пункти ми маємо:

- швидка настройка;
- більш проста інтеграція;
- маленька крива навчання.

Як негативні пункти маємо:

- відсутність підтримки великих проектів;
- більша гнучкість;
- мала популярність.

3.4 База даних

Збільшення обсягу та структурної складності збережених даних, а також розширення сфери використання користувачів інформаційних систем призвели до широкого прийняття найбільш зручних та відносно легких для розуміння реляційних (табличних) та нереляційних СУБД. Багатокористувацькі версії бази даних були створені для декількох користувачів, часто досить далеко один від одного та від місця зберігання даних. Так чи інакше вони вирішують конкретні проблеми паралельних процесів, цілісності (правильності) та безпеки даних, а також прав доступу.

База даних – набір даних, організований відповідно до концепції, що описує характеристики цих даних та взаємозв'язок між його елементами; цей набір підтримує щонайменше одну програму. Загалом, база даних містить схеми, таблиці, представлення даних, що зберігають процедури та інші об'єкти. Дані в

базі даних організовані відповідно до моделі організації даних. Таким чином, крім самих даних, сучасна база даних містить їх опис і може містити інструменти для їх обробки.

Система управління базами даних (СУБД / СУБД) [5] - програмне забезпечення для зберігання та управління даними. Для вирішення різних проблем розробляється все більше СУБД (реляційні та NoSQL) та відповідні програми для роботи з ними (MySQL, PostgreSQL, MongoDB, Redis тощо). СУБД базується на моделі бази даних - це спеціальні структури для обробки даних. Усі СУБД відрізняються за способом зберігання та обробки своїх даних.

СУБД повинна надавати будь-якому користувачеві доступ до даних, включаючи тих, які практично не мають та (або) не хочуть мати уявлення про:

- фізичне зберігання даних та їх описи;
- механізми пошуку запитуваних даних;
- проблеми, які виникають, коли кілька користувачів (додатків) запитують одні й ті самі дані одночасно;
- способи захисту даних від неналежного оновлення та / або несанкціонованого доступу;
- оновлення баз даних.

Хоча існує багато рішень для баз даних, лише деякі стають все більш популярними та затребуваними. Сьогодні найбільш часто використовуються реляційні СУБД і noSQL СУБД.

Реляційна модель та реляційні бази даних можуть бути дуже потужним інструментом, але лише якщо програміст знає як з ними поводитися. Останнім часом система NoSQL набуває все більшої популярності, обіцяючи усунути старі проблеми з базами даних та додати нові функції. Усуваючи жорстку структуру

даних при збереженні реляційного стилю, ці СУБД пропонують більш вільний спосіб роботи та значно ширші можливості їх адаптації. Хоча не без нових проблем.

Наприкінці 1960-х років було обговорено можливість використання різних табличних моделей даних, тобто можливість використовувати звичайні та природні способи передачі даних. Наприклад, реляційна модель, представлена в 1970 році, передбачала математичний спосіб структурування, зберігання та використання даних. По суті, він розширив плоскі та мережеві моделі, включивши їх у реляційну модель. Основною перевагою цього було групування даних, а саме реляційна модель, яка дозволяла зберігати дані у структурованій табличній формі.

Завдяки десятиліттям розробки СУБД досягла відносно високого рівня продуктивності та надійності. Досвід розробників та адміністраторів показав, що всі ці інструменти добре виконують свої функції в додатках будь-якої складності і не втраять даних навіть у разі несправного відключення.

Незважаючи на обмеження у створенні та управлінні даними, реляційні бази даних підтримують широкі можливості налаштування та відносно високу функціональність.

Домен називається набором атомних значень одного типу. Значення доменів полягає в наступному. Якщо значення двох атрибутів походять з одного домену, порівняння, ймовірно, має сенс. Використовуйте ці два атрибути. Якщо значення двох атрибутів походять з різних областей, їх порівняння, ймовірно, безглуздо. Співвідношення доменів D_1, D_2, \dots, D_n (не обов'язково всі різні) складається із заголовка та тіла.

Заголовок складається з такого фіксованого набору атрибутів A_1, A_2, \dots, A_n , що відповідність один на один між цими атрибутами A_i та їх визначальними доменами D_i ($i = 1, 2, \dots, n$).

Тіло складається з набору кортежів, що змінюються за часом, кожен кортеж у свою чергу складається з наборів пар атрибутно-значущих $(A_i: V_i)$, ($i = 1, 2, \dots, n$) та однієї такої пари для кожного атрибута A_i існує в заголовку. Для будь-якої пари параметр, значення атрибута $(A_i: V_i)$ V_i - це значення одного домену D_i , пов'язаного з атрибутом A_i .

Реляційна база даних - це сукупність відносин, яка містить всю інформацію, яку ви хочете зберегти в базі даних. Однак користувачі можуть сприймати базу даних як сукупність таблиць.

Наступні терміни можуть бути віднесені до будь-якої реляційної основи:

- кожна таблиця складається з однакових рядків і має унікальну назву;
- рядки мають фіксовану кількість полів (стовпців) та значень (кілька полів та повторювані групи недійсні), іншими словами, в кожній позиції тотлічаються один від одного хоча б одне значення;
- рядки таблиць неминуче дозволяють однозначно ідентифікувати кожен рядок такої таблиці;
- стовпці таблиці мають унікальну назву, і кожен запис містить однорідні значення даних;
- весь інформаційний вміст бази даних представлений у вигляді явних значень даних, і цей спосіб доставки є єдиним. Зокрема, немає спеціальних «посилань» чи покажчиків, які з'єднують одну таблицю з іншою. З'єднання між

таблицями здійснюється стовпцем таблиці, який містить унікальний специфікатор рядків з іншої таблиці;

□ для операцій над таблицею їхні рядки та стовпці можуть бути оброблені в будь-якому порядку, незалежно від їх вмісту. Цьому сприяє наявність назв таблиць та їх стовпців, а також можливість виділити будь-який рядок або набір рядків із заданими атрибутами.

NoSQL – термін, який описує набір підходів до реалізації сховищ баз даних, які суттєво відрізняються від моделей, що використовуються для доступу до даних через SQL у традиційних реляційних базах даних.

У червні 2009 року Йоганн Оскарсон провів зустріч у Сан-Франциско, на якій обговорили новітні тенденції на ринку ІТ-зберігання та обробки даних. Основним імпульсом зустрічі стали нові продукти Oporors, такі як BigTable та Dупато. Яскравий знак для зустрічі вимагав вичерпного та короткого терміну, який би ідеально вписувався в хештег Twitter. Такий термін запропонував Ерік Еванс із RackSpace – «NoSQL». Цей термін призначався лише для однієї зустрічі і не мав глибокого семантичного навантаження, але виявилось, що він поширився, як вірусна реклама по всьому світу, фактично став назвою всього напряму в ІТ-індустрії.

Термін призначався лише для однієї зустрічі і не мав великого вмісту. Однак виявилось, що вона поширилася через всесвітню павутину і стала фактичною назвою всього напряму в ІТ-індустрії.

Слід ще раз підкреслити, що термін «NoSQL» має повністю спонтанне походження і не має загальновизнаного визначення або наукової установи за ним. Ця назва швидше описує вектор розвитку ІТ далеко від реляційних баз даних. Описується як не тільки SQL, хоча є прихильники і прями визначення No SQL.

Нереляційні бази даних – це клас систем, що управляють базою даних. Основна відмінність полягає в тому, що таблиці не використовуються як структура пам'яті. Інші відмітні особливості таких систем полягають у тому, що SQL не використовується як мова запитів, операції підключення не підтримуються, а властивості ACID не гарантуються.

У найбільш загальній та класичній формулюванні об'єктно-орієнтований підхід базується на таких концепціях:

- ідентифікатор об'єкта та об'єкта. Кожна істота реального світу в об'єктно-орієнтованих мовах та системах моделюється як об'єкт. Кожен об'єкт, створюючи його, отримує унікальний ідентифікований системою ідентифікатор, який присвоюється об'єкту протягом усього його існування і не змінюється в міру зміни стану об'єкта;

- атрибути та методи. Кожен об'єкт має стан і поведінку. Статус об'єкта складається з набору значень для його атрибутів. Значенням атрибута об'єкта також може бути конкретний об'єкт або навіть декілька об'єктів. Поведінка об'єкта - це набір методів (програмний код), які застосовуються до стану об'єкта. Цей елемент відсутній у реляційній моделі. Стан та поведінка об'єкта; Взаємодія між об'єктами базується на обміні повідомленнями та впровадженні відповідних методів;

- класи. Кілька об'єктів з однаковими атрибутами та методами утворюють клас об'єктів. Об'єкт може належати лише одному класу. Примітиви Визначені класи, об'єкти екземплярів яких не мають атрибутів: цілі, рядки тощо. Клас, об'єкти якого можуть служити значеннями атрибутів об'єктів іншого класу, називається доменом цього атрибута.

Сучасні програми створюють вимоги, які не вирішуються реляційними базами даних, і це призвело до розвитку баз даних NoSQL, які мають переваги:

□ гнучка модель даних. Бази даних NoSQL відповідають різноманітним потребам сучасних додатків. Вони пропонують можливість вставляти дані без визначеної схеми. База даних NoSQL не обмежується характером даних, що зберігаються. Крім того, нові типи даних можуть бути додані за потребою під час робочого процесу. Це дозволяє легко вносити значні зміни в режимі реального часу. Зазвичай розробникам доводилося додавати код на сторінку програми, щоб забезпечити контроль якості даних, наприклад: Наприклад, із зазначенням конкретних полів, типів даних або дійсних значень. Більш складні бази даних NoSQL дозволяють застосовувати правила перевірки рівня баз даних, дозволяючи користувачам керувати даними про самі дані, зберігаючи гнучкість переваг динамічної схеми;

□ масштабованість та продуктивність. За замовчуванням бази даних NoSQL не забезпечують атомності транзакцій та узгодженості даних (але для деяких їх можна включити за потреби). Це дозволяє масштабувати базу даних до апаратного забезпечення в реальному часі, пропонуючи майже необмежений ріст із більшою пропускнуою здатністю та меншою затримкою, ніж реляційні бази даних. Хмарне сховище - чудове рішення, але масштабування полегшує обмін даними між декількома серверами. Використання, тестування та розробка локального обладнання, а потім переміщення системи до хмари, де вона працює, саме для цього створені бази даних NoSQL;

□ максимальне використання хмарних обчислень та зберігання даних. Завдяки хмарному сховищу даних ви можете ще більше розширити свою базу

даних. Це економічно вигідне рішення порівняно з вашими власними серверами та самообслуговуванням;

Крім різних переваг, таких як висока пропускна здатність та горизонтальна масштабованість, нереляційні бази даних мають і деякі недоліки.

Найважливіші наведені нижче:

- більшість нереляційних баз даних використовують програмне забезпечення з відкритим кодом, і хоча це доведено, ніхто не несе відповідальності за збій;

- багато нереляційних баз даних забезпечують властивості BASE (Basic Available, Soft State, Eventual Consistency) та жертвують стандартними властивостями ACID для підвищення продуктивності;

- через нехтування властивостями ACID рівень надійності нереляційних баз даних нижчий, ніж у реляційному випадку. Розробникам потрібно покластися на програмування для забезпечення ACID, який можна легко розгорнути у реляційній базі даних.

Відповідно до способу зберігання нереляційних баз даних прийнято поділяти на наступні типи.

- Ключове значення. При такому підході ви можете зберігати дані без певної структури. Ці дані складаються з ключа, представленого рядком та фактичних даних, що є значенням у парі ключ-значення. Дані можуть бути будь-якими примітивними елементами мови програмування (послідовність, ціле число або масив). Таким чином, це послаблює вимогу форматування даних для зберігання, тим самим робить використання фіксованої моделі даних непотрібним.

□ Зберігання документів (бази даних, орієнтовані на документи). Зберігання документів, яке також називають документами, орієнтованими на бази даних, - це в основному (комп'ютеризовані) програми, що використовуються для зберігання, отримання та оновлення даних БД. Основна структура пам'яті, що використовується в таких базах даних, - це документ. Кожне сховище документів може відрізнятися реалізацією даних, але передбачається, що дані містяться і кодуються у стандартному форматі, наприклад XML, BSON, PDF або документ Microsoft Office. Кожен документ представлений унікальним ключем, що представляє послідовність (URI - Уніфікований ідентифікатор ресурсу). API або мова запитів підтримує швидке видалення документів на основі їх вмісту. Наприклад, запит, який отримує всі документи, у яких певне поле має конкретне значення;

□ графові БД – це структуровані бази даних, які використовують діаграму як структуру даних, вершинам та ребрам яких надаються конкретні властивості, які представляють дані. Піки представлені як об'єкти (люди, компанії чи інші елементи) так само, як об'єкти представлені мовою програмування. Властивості таких об'єктів представлені у вигляді ребер, що з'єднують відповідні вершини діаграми. Послідовно переміщуючись від однієї вершини до іншої, надається необхідна інформація;

□ бази даних стовпців зберігають свої дані у форматі стовпців, на відміну від традиційного формату рядків. Цей підхід виправданий тим, що часто весь стовпець у таблиці потребує оновлення, тоді як інші дані рядків для зміни не потрібні; Н. Якщо один стовпець даних змінено, інші стовпці залишаються незмінними.

□ об'єктно-орієнтовані бази даних зазвичай називають OODBMS (Об'єктно-орієнтована система управління базами даних). Дані зберігаються як об'єкти, успадковується і підтримується, тому їх можна використовувати повторно (подібно до програмування об'єктів).

Проаналізувавши різницю між noSQL та SQL отримаємо наступні рекомендації до вживання. Ознаки проектів, для яких ідеально підійдуть SQLбази:

- є логічні вимоги до даних, які можуть бути визначені заздалегідь;
- дуже важлива цілісність даних;
- потрібна заснована на стандартах, добре зарекомендована технологія, за допомогою якої можна розраховувати на великий досвід розробників та технічну підтримку.

Властивості проектів, для яких підійде щось зі сфери NoSQL:

- вимоги до даних нечіткі, невизначені, або критерії, що розвиваються з розвитком проекту;
- проект може коригуватися з часом, при цьому важлива можливість негайного початку розробки;
- одні з основних вимог до бази даних - швидкість обробки даних і масштабованість.

Базуючись на огляд можливих у застосуванні баз даних було прийнято рішення використовувати noSQL. Використання не реляційної бази даних додає можливостей для внесення швидких змін у принципи роботи програми та легку масштабованість. Для даного проекту була обрана документоорієнтована БД MongoDB.

MongoDB - це система управління базами даних з відкритим кодом (СУБД), яка використовує орієнтовану на документи модель бази даних, яка підтримує різні форми даних. Це одна з численних нереляційних технологій баз даних, що виникла в середині 2000-х під банером NoSQL для використання у великих програмах даних та інших роботах з обробки даних, що включають дані, що не вписуються в жорстку реляційну модель. Замість використання таблиць і рядків, як у реляційних базах даних, архітектура MongoDB складається з колекцій та документів.

MongoDB був створений Дуайтом Мерріманом та Еліотом Горовицем, які зіткнулися з проблемами розробки та масштабованості з традиційними підходами до реляційних баз даних під час створення веб-додатків у DoubleClick, компанії з реклами в Інтернеті, яка зараз належить компанії Google Inc. Назва бази походить від слово «humongous» для представлення ідеї підтримки великого обсягу даних.

Мерріман та Горовіц допомогли створити 10Gen Inc. у 2007 році для комерціалізації MongoDB та відповідного програмного забезпечення. Компанія була перейменована в MongoDB Inc. у 2013 році та оприлюднила в жовтні 2017 року під тикерним символом MDB.

СУБД була випущена як програмне забезпечення з відкритим кодом у 2009 році та доступна за умовами версії 3.0 Загальної публічної ліцензії GNU Affero Фонду вільного програмного забезпечення, крім комерційних ліцензій, пропонуваніх MongoDB Inc.

На момент написання цієї наукової роботи, серед користувачів є наступні компанії: страхова компанія MetLife використовує MongoDB для додатків обслуговування клієнтів, веб-сайт Craigslist використовує його для архівування

даних, лабораторія фізики CERN використовує його для збирання та виявлення даних, а Нью-Йорк Times використовує MongoDB для підтримки програми для створення форм для подання фотографій.

Запис у MongoDB - це документ, який представляє собою структуру даних, що складається з пар полів і значень. Документи MongoDB схожі на об'єкти JavaScript Object Notation, але використовують варіант під назвою Binary JSON (BSON), який вміщує більше типів даних. Поля в документах схожі на стовпці реляційної бази даних, і значення, які вони містять, можуть бути різними типами даних, включаючи інші документи, масиви та масиви документів, згідно з посібником користувача MongoDB.

Документи, які також повинні містити первинний ключ як унікальний ідентифікатор, є базовою одиницею даних у MongoDB. Колекції містять набори документів і функціонують як еквівалент таблиць реляційних баз даних. Колекції можуть містити будь-який тип даних, але обмеження в тому, що дані в колекції не можуть бути розповсюджені в різних базах даних.

Оболонка mongo – це інтерактивний інтерфейс JavaScript для MongoDB, який дозволяє користувачам запитувати та оновлювати дані та проводити адміністративні операції. Оболонка є стандартним компонентом дистрибутивів з відкритим кодом MongoDB. Після встановлення MongoDB користувачі підключають оболонку mongo до своїх запущених екземплярів MongoDB. Формат зберігання документів та обмін даними BSON, використовуваний у MongoDB, забезпечує двійкове представлення документів, схожих на JSON. Автоматичне заточування - ще одна ключова особливість, яка дозволяє розподіляти дані в колекції MongoDB по декількох системах для горизонтальної

масштабованості по мірі збільшення обсягів даних та вимог пропускнуої здатності.

СУБД NoSQL використовує єдину головну архітектуру для узгодженості даних із вторинними базами даних, які підтримують копії первинної бази даних. Операції автоматично копіюються на ті вторинні бази даних для автоматичного резервного зберігання.

Як і інші бази даних NoSQL, MongoDB не вимагає попередньо визначених схем, і він зберігає будь-який тип даних. Це дає користувачам можливість гнучко створювати будь-яку кількість полів у документі, полегшуючи масштабування баз даних MongoDB порівняно з реляційними базами даних.

Однією з переваг використання документів є те, що ці об'єкти відображають на рідні типи даних у ряді мов програмування. Також вбудовані документи зменшують потребу в приєднанні до бази даних, що може зменшити витрати.

Основною функцією MongoDB є її горизонтальна масштабованість, що робить її корисною базою даних для компаній, що працюють із програмами великих даних. Крім того, шардинг дозволяє базі даних розподіляти дані по кластеру машин. Новіші версії MongoDB також підтримують створення зон даних на основі осколкового ключа.

MongoDB підтримує низку двигунів зберігання даних та надає плагінні інтерфейси API, що дозволяють третім сторонам розробляти власні двигуни зберігання даних для MongoDB.

СУБД також має вбудовані можливості агрегації, які дозволяють користувачам запускати MapReduce код безпосередньо на базі даних, а не запускати MapReduce

на Hadoop. MongoDB також включає власну файловою системою під назвою GridFS, схожу з розподіленою файловою системою Hadoop (HDFS), в основному для зберігання файлів, що перевищують обмеження розміру BSON в 16 Мб на документ. Ці подібності дозволяють використовувати MongoDB замість Hadoop, хоча програмне забезпечення баз даних інтегрується з Hadoop, Spark та іншими структурами обробки даних.

Хоча переваг багато, у MongoDB є деякі недоліки. За допомогою своєї стратегії автоматичного відмови користувач встановлює лише один головний вузол у кластері MongoDB. Якщо майстер виходить з ладу, ведений вузол автоматично перетвориться на нового ведучого. Цей перемикач обіцяє безперервність, але це не миттєво - це може зайняти до хвилини. Для порівняння, база даних Cassandra NoSQL підтримує декілька головних вузлів, так що якщо один майстер виходить з ладу, інший буде готовий до високодоступної інфраструктури баз даних.

Єдиний головний вузол MongoDB також обмежує швидкість запису даних у базу даних. Запис даних повинен бути записаний на головний майстер, а записування нової інформації в базу даних обмежено ємністю цього головного вузла.

MongoDB доступний у спільній та комерційній версіях через постачальника MongoDB Inc. MongoDB Community Edition - це відкритий вихід, в той час як MongoDB Enterprise Server пропонує додаткові функції безпеки, механізм зберігання в пам'яті, функції адміністрування та автентифікації та можливості моніторингу через Ops Manager.

Графічний користувацький інтерфейс (GUI) під назвою MongoDB Compass дає користувачам можливість працювати зі структурою документа, проводити запити, дані індексу тощо. MongoDB Connector для BI дозволяє користувачам підключати базу даних NoSQL до своїх інструментів бізнесрозвідки для

візуалізації даних та створення звітів за допомогою SQL-запитів. Слідом за іншими постачальниками баз даних NoSQL, MongoDB Inc. в 2016 році запустила хмарну базу даних як послугу під назвою MongoDB Atlas. Atlas працює на AWS, Microsoft Azure та Google Cloud Platform. Зовсім недавно MongoDB випустила платформу під назвою Stitch для розробки додатків у MongoDB Atlas, плануючи поширити її на локальні бази даних.

Компанія також додала підтримку багатодокументних транзакцій ACID в рамках MongoDB 4.0 у 2018 році. Дотримуючись властивостей ACID - атомності, послідовності, ізоляції та довговічності - на декілька документів розширюються типи транзакційних навантажень, з якими MongoDB може гарантовано впоратися точність та надійність.

3.5 Інтерфейс вводу досягнень

При виборі параметрів інтерфейсу введення даних було враховано кілька варіантів.

1. Введення даних через звичайний інтерфейс у вигляді форм, наприклад реєстраційні форми. Цей варіант не підходить для великої кількості інформації. Незважаючи на знайомість та чіткість використання, необхідний час заповнення збільшується зі збільшенням кількості полів, які потрібно заповнити.

2. Використання готових документів (електронних таблиць Excel або Word). Дозволяє завантажувати велику кількість даних, при цьому завантажуючи лише один файл підготовленої інформації.

Проаналізувавши вищезазначені моменти, було зроблено висновок, що найкращим варіантом буде проведення аналізу даних із вихідного файлу у міру

необхідності завантаження великих масивів даних та використання інтерфейсу у формі форм для зміни необхідних полів.

Рекомендується подавати рейтинг як підсумок всіх показників, що беруть участь у розрахунку рейтингу, при цьому таблицю сортують окремо за необхідним показником або загальним показником рейтингу, а також з можливістю глобального пошука по прізвищу викладача чи показників.

Профіль повинен містити повну інформацію про результативність роботи педагога та оцінку відповідно до балів, які відображають результативність та рекомендації щодо покращення показників у разі реалізації модуля відповідального за таку дію.

3.6 Модуль вирахування індивідуального рейтингу

Відповідно до розробленої метричної системи, необхідно розробити модуль, який обчислює бали оцінювання відповідно до цієї системи.

Враховуючи необхідність оновлення поточного рейтингу, результати рейтингу потрібно постійно перераховувати. Хоча оцінка визначених показників не передбачає складних обчислень, велика кількість даних може зажадати значної кількості обчислювальних ресурсів, що, в свою чергу, може перевантажити сервер і викликати його нестабільну роботу. Для вирішення цієї проблеми було вирішено скористатися віддаленими обчисленнями та послугами обробки даних або так званими «хмарними» послугами.

Хмарні обчислення - досить поширений термін, який поєднує в собі безліч підходів і моделей надання та управління ІТ-послугами. Тому на практиці цей термін усі розуміють по-різному. Деякі вважають хмарні обчислення

хостинговими віртуальними машинами або серверами сайтів лише на основі доступу до ресурсів мережі, інші розуміють хмарні обчислення як сервіси користувачів, такі як Dropbox та Google Drive. Тобто більшість користувачів визначають хмарні обчислення лише за атрибутом - доступом до мережі. Однак хмарні обчислення є набагато більшою одиницею. Як визначено Національним інститутом стандартів та технологій США (NIST), хмарні обчислення є моделлю для надання повсюдного та зручного доступу до мережі за запитом до спільного пулу налаштованих обчислювальних ресурсів, які можна швидко розгорнути та з мінімальним адмініструванням можна звільнитись та зателефонувати провайдеру.

Виділяють 3 типи хмарних обчислень [7] по моделі обслуговування:

- інфраструктура як послуга (IaaS – англ., Infrastructure as a Service);
- програмне забезпечення як послуга (SaaS – англ., Software as a Service);
- платформа як послуга (PaaS – англ., Platform as a Service).

Програмне забезпечення як послуга - модель, коли споживач може використовувати постачальника програмного забезпечення, який працює в хмарній інфраструктурі і доступний з різних клієнтських пристроїв або через тонкий клієнт, такий як браузер (наприклад, веб-пошта) або через інтерфейс програми. Контроль та управління базовою фізичною та віртуальною хмарною інфраструктурою, включаючи мережі, сервери, операційні системи, сховище даних або навіть окремі функції програми (за винятком обмеженої кількості налаштувань конфігурації програми), здійснюється хмарним постачальником. Програмне забезпечення як послуга надає користувачам можливість отримати доступ до конкретного програмного забезпечення. Клієнт не платить за

установку оновлень, обслуговування сервера, а оновлення сервісу відбувається автоматично. Замовник платить за оренду ПО. Якщо ж необхідність в обраній програмі відпала, то завжди можна від неї відмовитися. За контрактом користувач не тільки оплачує доступ до софту, але і витрати на захист даних.

Плюси послуги SaaS:

- клієнту не треба оновлювати, встановлювати, налаштовувати ПО;
- оплачуються тільки використовувані ресурси;
- більшість додатків можна запустити через браузер, це означає що не потрібно примусово завантажувати модулі;
- клієнт може зайти в програму з будь-якого пристрою, підключеного до інтернету.

Платформа як послуга – модель, де споживачі можуть використовувати хмарну інфраструктуру для розміщення базового програмного забезпечення для подальшого розгортання нових або існуючих додатків (нативних, спеціальних або придбаних реплікаційних програм). Ці платформи включають інструменти для створення, тестування та запуску програмного забезпечення, систем управління базами даних, папок та мов програмування, які надаються постачальником хмарних послуг. Платформа як послуга представляє собою хмарну платформу для розгортання додатків. Клієнт не зможе управляти мережевим і серверним обладнанням, операційними системами. Йому доступна тільки можливість встановити програмне забезпечення, Користувач зможе управляти базами даних, оновлювати софт. Це рішення підходить для розробників ПЗ, бізнес-аналітиків, компаній, які вирішили запустити новий сервіс в тестовому режимі. Плюси послуги PaaS:

- не потрібно для запуску сервісу купувати додаткове обладнання або нести інші витрати, пов'язані з технікою;
- підтримка динамічного масштабування ресурсів;
- скорочення часу роботи над софтом за рахунок залучення декількох співробітників;
- платформа постійно оновлюється провайдером.

Інфраструктура як послуга надає можливість використовувати хмарну інфраструктуру для незалежного управління обробкою, зберіганням, мережевими та іншими основними обчислювальними ресурсами. Наприклад, споживач може встановлювати та запускати будь-яке програмне забезпечення, включаючи операційні системи, платформи та прикладне програмне забезпечення. Споживач може контролювати операційні системи, віртуальні системи зберігання даних та встановлені програми, а також мати обмежений контроль над низкою доступних мережеслужб (наприклад, брандмауер, DNS). Контроль та управління базовою фізичною та віртуальною хмарною інфраструктурою, включаючи мережі, сервери, типи операційних систем та системи зберігання даних, здійснює хмарний провайдер. Інфраструктура як послуга - послуга, що надає користувачеві самостійно управляти ресурсами для обчислення, обробки, збереження інформації. Провайдер зобов'язаний забезпечувати доступність віртуальних серверів (private cloud), а все інше лягає на плечі користувача. Фактично клієнт отримує готову для настройки інфраструктуру. Послуга IaaS має наступні плюси:

- гнучке, динамічне масштабування, що дозволяє в будь-який момент додати або відмовитися від частини ресурсів;

- повне управління всіма системами, можливість установки різного ПО, вибору ОС;
- доступ до хмарної інфраструктури можна отримати з будь-якої точки земної кулі.

Таким чином, модель хмарних обчислень Інфраструктура як послуга уникає складних та ресурсомістких операцій, які, у свою чергу дозволяють уникнути відмови сервера і призводять до більш стабільних операцій.

3.7 Розподілення прав доступу

Права доступу дозволяють учасникам отримати доступ до певних функцій веб-сайту курсу або проекту, залежно від їх ролі та функцій, призначених їм власником або адміністратором сайту.

Ролі - це сукупність прав доступу. Деякі ролі забезпечують учаснику легкий доступ до читання матеріалів, а інші дозволяють змінити сайт та інструменти, такі як: Додавання учасників, зміна матеріалу або дозволу на інші ролі. Усі ролі НЕ рівноцінні.

Ролі можуть запропонувати різні способи роботи з різними інструментами. Однак у більшості випадків творець сайту проекту чи інструктор має всі права доступу на сайт курсу та може додавати або видаляти матеріали з сайту.

Параметри для кожного інструменту різні. Адміністратор сайту визначає, які ролі встановлюються та які дозволи надаються. Ролі та відповідні дозволи на різних системах будуть різні.

Для системи управління кадровим складом університету було розроблено ієрархічну схему розподілення прав доступу, яка включає в себе наступні ролі.

- адміністратор;
- викладач;
- представник адміністрації університету або овнер.

Адміністратор має можливість додавання, редагування та видалення користувачів системи, також у цієї ролі є повний доступ до всіх можливостей системи, зокрема: редагувати рейтингові коефіцієнти, переглядати рейтингові списки та детальну інформацію по кожному з користувачів. Адміністратор не може лише додати власний рейтинговий файл щоб відобразитися у системі.

Представник адміністрації університету має можливість редагувати рейтингові коефіцієнти, переглядати рейтинговий список та детальну інформацію по кожному з користувачів, також він має можливість додати власний файл для того щоб відобразитися у рейтинговому списку.

Викладач має можливість лише додавати та переглядати власний рейтинговий файл.

Таким чином найбільш повні повноваження має адміністратор системи, однак він не може бути учасником рейтингування і не може додати свій файл до системи. У той же час представник адміністрації університету має цю можливість, однак він не може редагувати інформацію о користувачах.

4. ІНФОРМАЦІЙНА БЕЗПЕКА

Для кожної системи яка надає можливість доступу до конфіденційної інформації повинно приділятися велика кількість уваги інформаційній безпеці та захисту від несанкціонованих дій. Це означає що кожний запит до API повинен бути авторизован, а користувач має бути аутентифікован, щоб користувач не мав можливості виконати дії які не є у рамках його ролі. У данному розділі розглядаються деякі з існуючі методів інформаційної безпеки та ті дії які були зроблені для забезпечення інформаційної безпеки розробленої системи.

4.1 Аутентифікація

Аутентифікація - це процес перевірки автентичності чого-небудь. Прикладом аутентифікації може бути порівняння пароля, введеного користувачем, з паролем, який збережений в базі даних сервера. Подібна перевірка може бути як односторонньою, так і взаємної - все залежить від способу захисту і політики безпеки сервісу.

Авторизація - це перевірка прав користувача на доступ до певних ресурсів.

Якщо коротко, ідентифікація - процес визначення, що за людина перед нами; аутентифікація - процес підтвердження, що ця людина саме той, за кого себе видає; авторизація - процес прийняття рішення про те, що саме цієї аутентифіцироваться персони дозволяється робити.

SQL ін'єкція або SQL-ін'єкції (англ. SQL injection) [6] - один з поширених способів злому сайтів і програм, які працюють з базами даних, заснований на

впровадженні в запит довільного SQL-коду. Впровадження SQL, в залежності від типу використовуваної СУБД і умов впровадження, може дати можливість атакуючому виконати довільний запит до бази даних (наприклад, прочитати вміст будь-яких таблиць, видалити, змінити або додати дані), отримати можливість читання і / або запису локальних файлів і виконання довільних команд на атакується сервері.

Всі методи аутентифікації [12] можна розділити на односторонню (перевірка здійснюється тільки однією стороною) і взаємну (в перевірці даних беруть участь обидві сторони). Також виділяють однофакторний і криптографічний спосіб. Найпопулярнішим прикладом застосування однофакторних систем є паролі. Залежно від рівня організації і ступеня конфіденційності даних, вони можуть бути багаторазовими (менш захищений варіант) і одноразовими. Всі способи аутентифікації можна розташувати по зростанню їх складності:

- базова аутентифікація. При застосуванні цього виду аутентифікації логін користувача і його пароль входять до складу веб-запиту. Будь-який перехоплювач пакета інформації легко впізнає засекречені дані;

- дайджест-аутентифікація. Вид аутентифікації, який має на увазі передачу призначених для користувача паролів в хешировань стані. На перший погляд може здатися, що рівень захисту в даному випадку трохи відрізняється від базової перевірки. На ділі це не так: до кожного паролю додається довільна рядок, що складається з символів (хеш), яка генерується окремо на кожен новий веб-запит. Постійне оновлення хешу не дає зловмиснику можливості розшифрувати пакет даних - кожне нове підключення утворює інше значення пароля. На основі даного методу аутентифікації працює більшість інтернетбраузерів (Mozilla, Google Chrome, Opera);

□ HTTPS. Цей протокол дає можливість шифрування не тільки логіна і пароля користувача, але і всіх інших даних, що передаються між інтернетклієнтом і сервером. У протоколу є один істотний недолік - він значно уповільнює швидкість з'єднання;

□ аутентифікація з пред'явленням цифрового сертифікату. Такий спосіб має на увазі використання протоколів із запитом і відповідним відповіддю на нього. Сторінка аутентифікації направляє до користувача певний набір символів («адреса»). Відповіддю є запит сервера, який підписаний за допомогою персонального ключа. Аутентифікація по відкритому ключу застосовується в якості захисного механізму в протоколах SSL, Kerberos та RADIUS;

□ аутентифікація з використанням Cookies. Невеликий масив даних, який відправляється інтернет-сервером і зберігається на ПК користувача. Браузер при кожній спробі підключення до даного ресурсу посилає Cookies як одну із складових частин HTTP-запиту. Дана технологія, крім аутентифікації, використовується для: збереження індивідуальних налаштувань і переваг; стеження за станом сеансу; збору статистичних даних про користувачів (частота відвідувань, унікальні відвідування і т.д.). Як засіб аутентифікації куки використовуються для систем безпеки чатів, форумів і різних інтернет-ігор. Cookies володіють низьким ступенем захисту - якщо сесія погано фільтрується, то викрасти їх не складає труднощів. Тому застосовується додаткова прив'язка по IP-адресою, з якого користувач увійшов в систему;

□ децентралізована аутентифікація Виділяють кілька основних протоколів, що працюють за принципом децентралізованої аутентифікації: OpenID, OpenAuth та OAuth.

Базуюсь на необхідності високого рівня безпеки та раніше оголошених методів механізму аутентифікації було обрано використання одразу декількох механізмів. Були обрані механізм аутентифікації за цифровим сертифікатом, HTTPS та за допомогою JWT-токену. Для реалізації данного комплексу був згенерован сертифікат SSL, а також увімкнута передача даних через протокол HTTPS.

4.2 Авторизація за допомогою JWT-токену

Для авторизації користувачів було обрано JSON Web Token. Це сучасний стандарт, який можна використовувати для надання авторизації.

Токени - це інструмент авторизації для кожного запиту від клієнта до сервера. Токени (і сигнатура токена також) генеруються на сервері на основі секретного ключа (який зберігаються на сервері) та корисного навантаження (далі payload). Потім токен зберігається на клієнті і використовується за необхідності для авторизації запитів.

JSON Web Token (JWT) - це JSON об'єкт, який визначений у відкритому стандарті RFC 7519. Він вважається одним з безпечних способів передачі інформації між двома учасниками. Для його створення необхідно визначити заголовок (header) із загальною інформацією по токені, корисні дані (payload), такі як ід користувача, його роль і т.д., і підписи (signature). Ця інформація може бути перевірена, бо вона має цифровий підпис.

Стандартна схема створення та використання токену (рис 4.1) має наступні кроки:

- користувач входить до додатку за допомогою логіну та паролю;
- сервер перевіряє достовірність введених користувачем даних та у разі успіху створює та повертає JWT токен користувачеві;
- клієнт зберігає токен та використовує access токен для подальшої авторизації запитів;
- перед кожним запитом клієнт попередньо перевіряє час життя access токена та якщо він минув використовує refresh токен для оновлення access та refresh токенів, далі використовує для запиту новий access токен.



Рисунок 4.1 Схема створення та використання JWT-токену

JWT використовують криптографічні алгоритми для підпису вмісту заголовку та інформаційної частини.

Відповідно до JWA (JSON Web Algorithms), специфікація розрізняє групи алгоритмів шифрування з використанням симетричних та асиметричних алгоритмів. Симетричні алгоритми включають HS256, HS384, HS512, кожен з яких використовує відповідні алгоритми хешування HMAC SHA-256, HMAC SHA-384 або HMAC SHA-512. Код автентифікації повідомлень на основі хеша (HMAC) - це механізм перевірки цілісності інформації за допомогою секретного рядка для передачі переданих даних у поєднанні з хеш-функцією.

Асиметричними алгоритмами є RS256, RS384, RS512 з RSASSA-PKCS1v1_5 - асиметричним криптографічним алгоритмом. При цьому використовується пара що містить як приватні так і відкриті ключі.

Завдяки невеликому розміру JWT можливо передавати у URL, у POST параметрі чи всередині HTTP заголовка. До того ж, компактність збільшує швидкість передачі. Самодостатність означає, що токен містить усю необхідну інформацію про користувача, що дозволяє не робити запити до бази даних більш ніж один раз. JWT складаються з трьох частин які будуть розглянуті далі:

- заголовок. Містить дві частини: алгоритм хешування і тип токена (JWT), який використовується (наприклад, SHA256, HMAC або RSA). Заголовок кодується за допомогою Base64Url та є першою частиною JWT токену;
- корисне навантаження (payload). Містить інформацію про

користувача додатку та додаткові метадані. Є зарезервовані, публічні та приватні поля. Корисне навантаження також кодується Base64Url та є другою частиною JWT токєну;

□ підпис. Для створення підпису беруться закодовані корисне навантаження і заголовок, а також секретне слово після чого підписуються за допомогою вказаного в заголовку алгоритму. Підпис використовується аби впевнитись, що відправник JWT – той, за кого він себе видає, і повідомлення не було змінєне на шляху до серверу;

5 ПРОЦЕДУРИ ПІДРАХУНКУ ІНДИВІДУАЛЬНОГО РЕЙТИНГУ

Методики оцінювання науково-педагогічних працівників мають на меті створити оцифрований взаємозв'язок, який дозволить зробити прозорий розподіл моральних та матеріальних стимулів між учасниками метричного оцінювання до утворєнними рейтингами процесів чи явищ. Такий підхід створить «досягнення – оцінення – стимулювання – досягнення», що дасть змогу швидкому зростанню висококваліфікованих кадрів, що, в свою чергу, забезпечить кращу освіту та покращить репутацію університету за рахунок особистої діяльності кожного працівника.

Будь-яка діяльність науково-педагогічних працівників повинна враховуватися при складанні рейтингу, але всі сфєри діяльності мають різну вагу, залежно від важливості досягнення. Крім того, всі пункти досягнєнь повинні бути розділені на активні та пасивні.

Зокрема, активні пункти потребують постійної підтримки

результативності. Сюди можна віднести, наприклад, наявність наукового видання в журналі, яке входить до наукових метрик, наявність наукових публікацій (авторські свідоцтва, патенти) у наукових публікаціях, що входять до переліку наукових видань України тощо.

Пасивні досягнень усувають потребу в підтримці продуктивності. Єдине, що враховує, це наявність досягнутого результату. Пасивні досягнення включають захист дисертації, члена вченої ради, заступника декана тощо.

Усі компоненти рейтингу повинні бути розраховані окремо за відповідними формулами [13]. По-перше, ми повинні отримати кількісні дані вимірювань для кожного елемента оцінювання. Далі розглянемо процес отримання кожної метрики.

Формула для розрахунку кількісного показника рейтингового пункту щодо наявності наукової публікації у періодичному виданні, яке включено до наукометричних баз (Scopus, Web of Science Core Collection та інші) за останні 5 років має вигляд:

$$S_1 = \sum_{j=1}^m \sum_{i=1}^n P_{i,j}, m = \overline{1,2,3}, n = \overline{1,2,3},$$

де $P_{i,j}$ - публікація у періодичному виданні, що включено до науко

метричних баз, $\sum_{i=1}^n P_{i,j}$ - загальна кількість публікацій в певному періодичному

$m \quad n$

виданні, таким чином $\sum_{j=1}^m \sum_{i=1}^n P_{i,j}$ - сума усіх публікацій у періодичних виданнях,

що входять до науко метричних баз.

Оцінка показника наявності принаймні п'яти наукових публікацій у наукових виданнях включених до переліку наукових фахових видань України:

$$S_2 \square 1, n_{\text{при}} \square \sum_{i \square 1}^n P_i \square 5, n \square 1, 2, 3,$$

або

$$S_2 \square 0, n_{\text{при}} \square \sum_{i \square 1}^n P_i \square 5, n \square 1, 2, 3,$$

де P_i - наукова публікація у наукових виданнях, включених до переліку наукових фахових видань України;

Рейтинг наявності виданого підручника, навчального посібника, рекомендованого Міністерством освіти і науки, іншим центральним органом виконавчої влади чи вченою радою навчального закладу, та існування монографії (для співавторів з фіксованим власним внеском) відповідає формулі:

$$S_3 \square \sum_{i \square 1}^n \sum_{j \square 1}^m P_{\text{під}i} \square \sum_{y \square 1}^r P_{\text{нав}j} \square \sum_{y \square 1}^r P_{\text{му}}, m \square 1, 2, 3, n \square 1, 2, 3, r \square 1, 2, 3,$$

де - кількісний показник виданих підручників;

m

$\square P_{\text{нав}j}$ - кількісний показник виданих навчальних посібників;

лп1

r

лп1 $P_{му}$ - кількісний показник наявних монографій.

улп1

Розрахування рейтингового показника пункту про наукове керівництво (консультування) здобувача, який одержав документ про присудження наукового ступеня відбувається за допомогою формули:

$$S_4 = \frac{\sum_{i=1}^n N_{i,n}}{n}, \quad i=1,2,3,$$

де N_i - показник наукового керівництва або консультування певного здобувача,

що отримав документ про присудження наукового ступеня.

Формула для розрахунку рейтингу щодо залучення до міжнародної експертизи або участі у міжнародному науковому проекті, наявність звання «суддя міжнародної категорії».

$$S_5 = U_{мнп} + U_{ме} + U_{см},$$

де $U_{мнп}$ - наявність участі у міжнародному науковому проекті ; $U_{ме}$ -

показник, що характеризує залучення до міжнародної експертизи;

$U_{смк}$ - наявність звання «суддя міжнародної категорії».

Рейтинг пункту проведення навчальних занять іноземною мовою (крім мовних навчальних дисциплін) в обсязі не менше 50 аудиторних годин на навчальний рік вираховується за формулою:

$$S_6 \geq 1, \text{ при } \sum_{i=1}^n H_i \geq 50, n \geq 1, 2, 3,$$

або

$$S_6 \geq 0, \text{ при } \sum_{i=1}^n H_i \geq 50, n \geq 1, 2, 3,$$

n

де $\sum_{i=1}^n H_i$ - кількість аудиторних годин навчальних занять проведених

іноземною мовою за навчальний рік.

Рейтинг пункту відповідальності за наявність роботи в спеціалізованих комітетах з вивчення дисертацій МОН або галузевих технічних комітетів Національного агентства з забезпечення якості вищої освіти або комісії з акредитації, спеціалізованих комітетів, міжгалузевого комітету з вищої освіти акредитаційних комісій, трьох спеціалізованих комітетів науково-методичну раду або науково-методичні комісії з вищої освіти Міністерства освіти і науки, робочі групи з розробки стандартів вищої освіти України можливо отримати через формулу:

n —

$$S_7 = \sum_{i=1}^n N_{i,n} \quad i=1,2,3 ,$$

яка зазначає, що кількісний показник 7-го пункту дорівнює сумі наявності кожного з зазначених у пункті досягнень.

Вирахування рейтингу пункту щодо виконання функцій наукового керівника госпдоговорної або держбюджетної теми, відповідального виконавця наукової теми (проекту), головного редактора або члена рекадційної колегії наукового видення включеного до переліку наукових фахових видань України іноземного рецензованого наукового видання відбувається по формулі:

$$S_8 = \sum_{i=1}^n N_{i,n} \quad i=1,2,3 ,$$

яка зазначає, що кількісний показник 8-го пункту дорівнює сумі наявності кожного з зазначених у пункті досягнень (виконання функцій наукового керівника держбюджетної або госпдоговорної теми і т.д.).

Рейтингова оцінка пункту, що відповідає за керівництво школярем, який зайняв призове місце III-IV етапу Всеукраїнських учнівських олімпіад з базових навчальних предметів, II-III етапу Всеукраїнських конкурсів, захистів науково-дослідницьких робіт учнів - членів Національного центру «Мала академія наук України»; участь у журі конкурсів чи олімпіад «Мала академія наук України»:

$$S_9 = \sum_{i=1}^n N_{i,n} \quad i=1,2,3 ,$$

n

де $\sum_{i=1}^n N_i$ - сума наявності студентів, що зайняли призове місце у одному з n зазначених у пункті конкурсів під керівництвом педагога;

Організаційна робота у закладах освіти на посадах керівника (заступника керівника): інституту, закладу освіти, факультету, відділення (наукової установи), кафедри, філії або іншого відповідального за підготовку здобувачів вищої освіти: відділу (наукової установи), підрозділу, навчально-методичного управління (відділу), іншого навчальнонаукового (інноваційного) структурного підрозділу, лабораторії, вченого секретаря закладу освіти (факультету, інституту), відповідального секретаря приймальної комісії.

Таким чином $S_{10} \geq 1$, якщо виконується не менш ніж один з зазначених вище критеріїв та $S_{10} \geq 0$ у протилежному випадку.

Рейтингова оцінка за участь в атестації наукових працівників як члена постійної спеціалізованої вченої ради (не менше трьох разових спеціалізованих вчених рад) або офіційного опонента вираховується за формулою.

$$S_{11} \geq 1, n \text{ при } \sum_{i=1}^n N_i \geq 3, n \geq 1, 2, 3,$$

або

$$S_{11} \geq 0, n \text{ при } \sum_{i=1}^n N_i \geq 3, n \geq 1, 2, 3,$$

$$\sum_{i=1}^n$$

де $\sum_{i=1}^n N_{i,n} \square 1,2$ - кількість раз у яких педагог приймав участь у

спеціалізованих вчених радах.

Вирахування рейтингу пункту щодо наявності не менше п'яти авторських свідоцтв та/або патентів загальною кількістю два досягнення відбувається по формулі:

$$S_{12} \square 1, n \text{ при } \sum_{i=1}^n N_{i,n} \square 5, n \square 1,2,3,$$

або

$$S_{12} \square 0, n \text{ при } \sum_{i=1}^n N_{i,n} \square 5, n \square 1,2,3,$$

$$\sum_{i=1}^n$$

де $\sum_{i=1}^n N_{i,n} \square 1,2$ кількість наявних виданих свідоцтв, патентів і т.д.

$\sum_{i=1}^n$

Рейтинг щодо наявності виданих навчально-методичних посібників/посібників для дистанційного навчання та самостійної роботи студентів, конспектів, практикумів, лекцій, рекомендацій та методичних вказівок загальною кількістю три найменування отримується за формулою:

$$S_{13} \square 1, n \text{ при } \sum_{i=1}^n N_{i,n} \square 1, n \square 1,2,3,$$

або

$$S_{13} = \sum_{i=1}^n N_{i,n} \cdot \sum_{n=1,2,3} N_{i,n}$$

де $\sum_{i=1}^n N_{i,n}$ кількість наявних виданих навчально-методичних

посібників/посібників для дистанційного навчання та самостійної роботи студентів, конспектів лекцій тощо.

Рейтингова оцінка за керівництво студентом, який зайняв призове місце на I етапі Всеукраїнської студентської олімпіади (Всеукраїнського конкурсу студентських наукових робіт), або робота у складі організаційного комітету/журі Всеукраїнської студентської олімпіади (Всеукраїнського конкурсу студентських наукових робіт), або керівництво постійно діючим студентським науковим гуртком/проблемною групою; керівництво студентом, який став призером або лауреатом Міжнародних мистецьких конкурсів, фестивалів та проектів, робота у складі організаційного комітету або у складі журі міжнародних мистецьких конкурсів, інших культурномистецьких проектів; керівництво студентом, який брав участь в

Олімпійських, Паралімпійських іграх, Всесвітній та Всеукраїнській Універсіаді, чемпіонаті світу, Європи, Європейських іграх, етапах Кубка світу та Європи, чемпіонаті України; виконання обов'язків тренера, помічника тренера національної збірної команди України з видів спорту; виконання обов'язків головного секретаря, головного судді, судді міжнародних та всеукраїнських

змагань; керівництво спортивною делегацією; робота у складі організаційного комітету, суддівського корпусу вираховується за формулою:

$$S_{14} = \frac{1}{n} \sum_{i=1}^n N_{i,n} \quad 1,2,3,$$

яка зазначає, що кількісний показник 14-го пункту дорівнює сумі наявності кожного з зазначених у пункті досягнень (за керівництво студентом, який зайняв призове місце на I етапі Всеукраїнської студентської олімпіади або робота у складі організаційного комітету/журі Всеукраїнської студентської олімпіади тощо).

Вирахування рейтингу пункту щодо наявності науково-популярних та/або консультаційних (дорадчих) та/або дискусійних публікацій з наукової або професійної тематики загальною кількістю не менше п'яти публікацій відбувається по формулі:

$$S_{15} = \frac{1}{n} \sum_{i=1}^n N_i \quad 1,2,3,$$

або

$$S_{15} = \frac{0}{n} \sum_{i=1}^n N_i \quad 1,2,3,$$

де $N_{i,n}$ кількість наявних виданих науково-популярних та/або

i=1

консультаційних (дорадчих) та/або дискусійних публікацій і т.д.

Формула для розрахунку рейтингу щодо участі у професійних об'єднаннях за спеціальністю має вигляд:

$$S_{16} = \frac{\sum_{i=1}^n N_{i,n} \cdot 1,2,3,}{n}$$

де - кількість раз, коли науко-педагогічний працівник приймав участь у професійних об'єднаннях за спеціальністю;

Вирахування рейтингу пункту щодо досвід практичної роботи за спеціальністю не менше п'яти років по формулі:

$$S_{17} = \frac{\sum_{i=1}^n N_{i,n} \cdot 5,1,2,3,}{n}$$

або

$$S_{17} = \frac{\sum_{i=1}^n N_{i,n} \cdot 5,1,2,3,}{n}$$

де $\sum_{i=1}^n N_{i,n} \cdot 1,2$ кількість років досвіду практичної роботи за спеціальністю.

Рейтингова оцінка за наукове консультування установ, підприємств, організацій протягом не менше двох років вираховується за формулою:

$$S_{18} = \frac{1}{n} \sum_{i=1}^n N_{i,2,n} \cdot 1,2,3,$$

або

$$S_{18} = \frac{0}{n} \sum_{i=1}^n N_{i,2,n} \cdot 1,2,3,$$

де $\sum_{i=1}^n N_{i,n} \cdot 1,2$ кількість років наукове консультування установ,

підприємств, організацій протягом.

Таким чином, враховуючи усі зазначені вище процедури вираховування рейтингових значень по кожному пункту збірний рейтинг можливо отримати наступною формулою:

$$S = \sum_{i=1}^n S_{i,n} \cdot 1,2,3..1,$$

де S_i - відповідний кількісний показник i -го пункту рейтингу.

6 ІНСТРУКЦІЯ АДМІНІСТРАТОРУ

У данному розділі будуть наведені приклади використання розробленого програмного комплексу а також детальна інструкція доступного функціоналу.

Представлений програмний комплекс має три ролі: адміністратор, власник (особа яка є відповідальною у кожному окремо взятому підрозділі навчального закладу) та викладач.

Для початку розглянемо можливості адміністратора системи. Першою сторінкою яку бачить користувач системи є сторінка входу до системи.

Скріншот данної сторінки наведений у рис. 6.1.

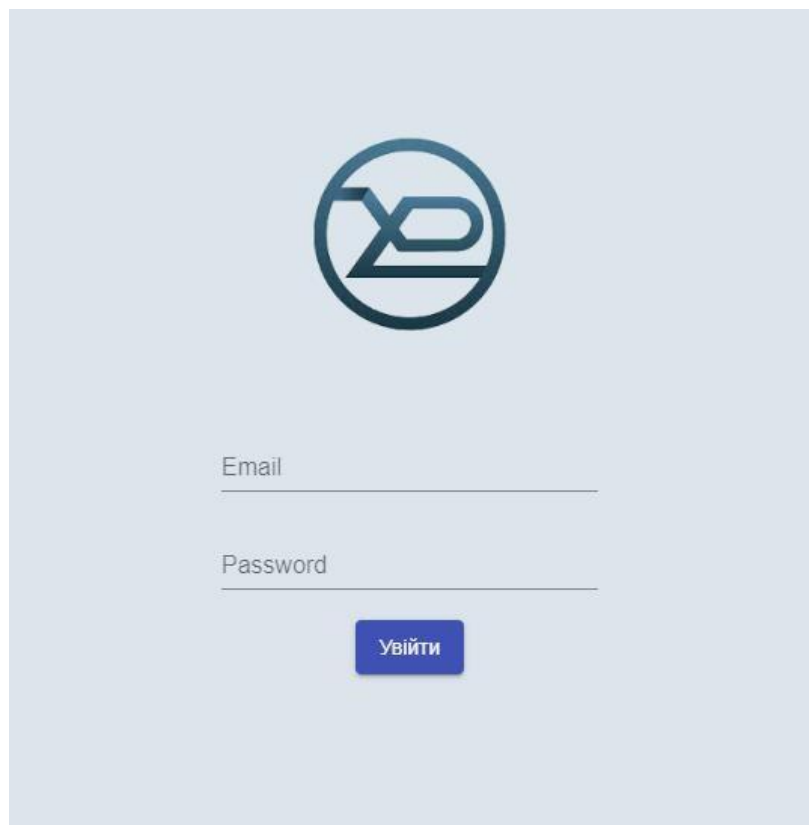


Рисунок 6.1 – Сторінка входу

Нові користувачі не мають можливості зареєструватися у програмі. Для того щоб користувач мав змогу увійти до веб інтерфейсу йому необхідно мати данні для

входу, такі як email та пароль, які йому (користувачу) повинен надати адміністратор системи. У свою чергу додавати нових користувачів має змогу лише адміністратор системи на сторінці «Працівники» (див. рис. 6.2) у розділі «Додати працівника» (рис. 6.3). Адміністратор системи додається розробником програмного забезпечення або персоною яка відповідає за перше розгортання програмного комплексу на серверах.

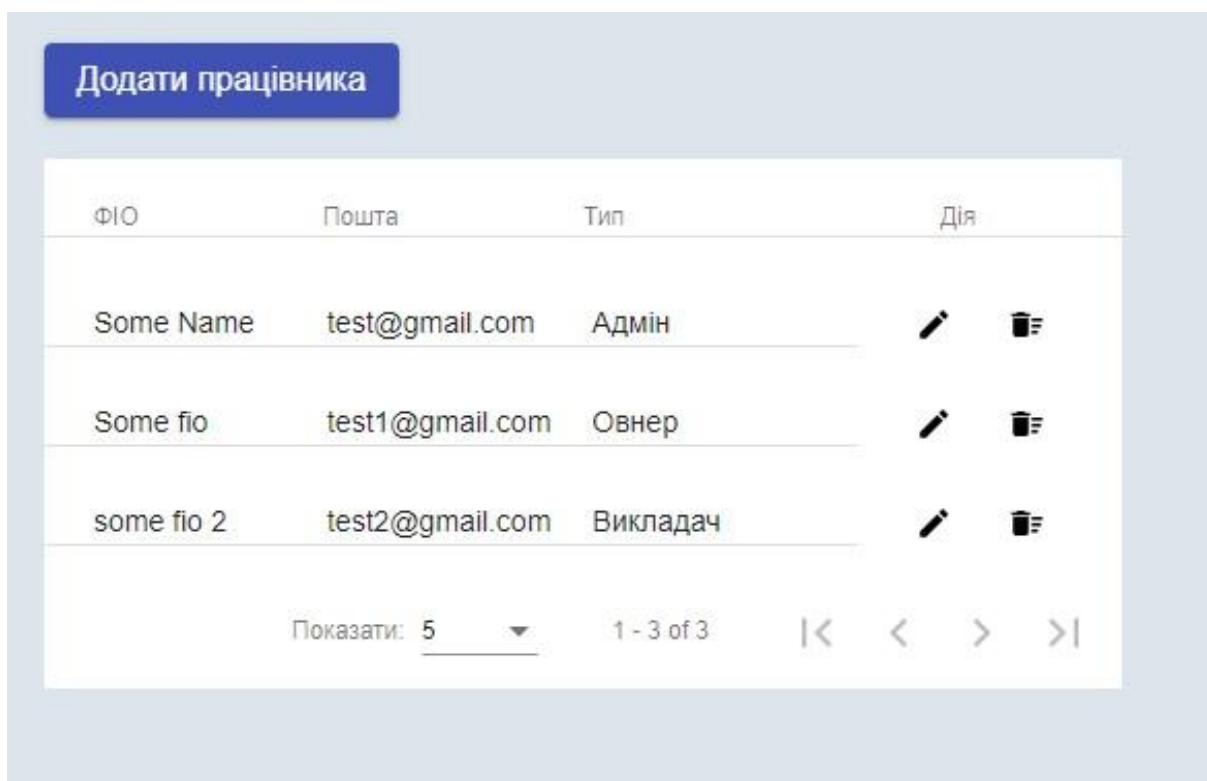


Рисунок 6.2 - Сторінка «Працівники»

На сторінці «Працівники» адміністратор має можливість переглянути список користувачів та такі дання як ФІО (колонка «ФІО»), електронна пошта(колонка «Пошта» та правову групу користувача (колонка «Тип»). Також адміністратор має можливість редагувати інформацію користувача чи видалити його із системи.

Функціонал редагування схожий з функціоналом додавання користувача, тому ми детально зупинимось лише на функціоналі додавання співробітника.

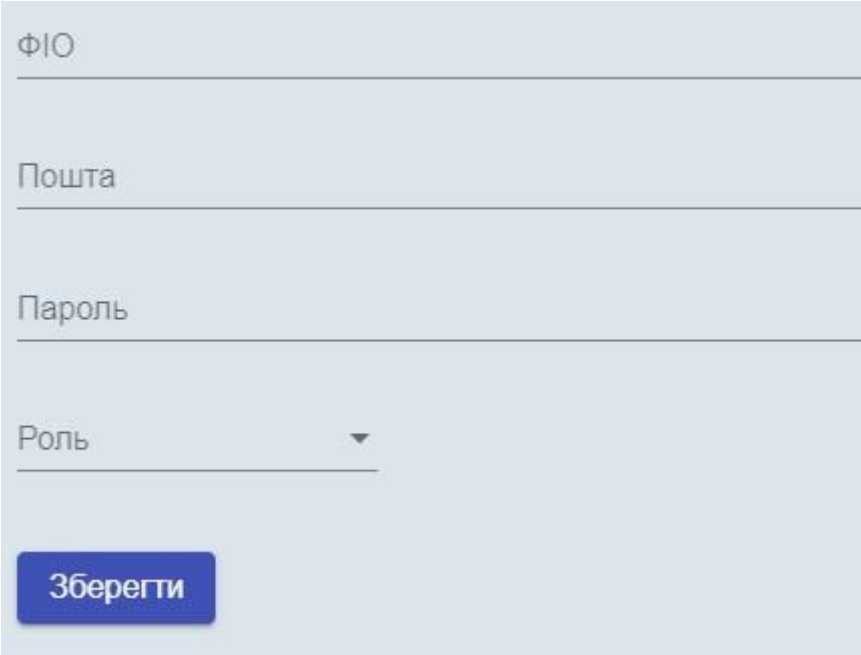


Рисунок 6.3 - Сторінка «Додати працівника»

Для того щоб додати нового користувача системи Ви повинні виконати наступні кроки:

1. Заповнити поле «ФІО» українською мовою.
2. Ввести електронну пошту працівника у просторстві доменних імен pure.ua.
3. Задати пароль користувача.
4. Задати роль користувача. Роль може бути «Адмін», «Овнер», «Викладач»
5. Натиснути «Зберегти».

Якщо Ви виконали усі кроки коректно то вас буде переведено на сторінку «Працівники» та у списку буде новий користувач якого Ви щойно додали до системи. Для редагування користувача Ви повинні відредагувати необхідні поля, які у Вас будуть вже заповнені поточними даними користувача, та натиснути «Зберегти».

Наступна сторінка яка є доступною для адміністратора та володаря системи – сторінка «Рейтингові списки» (рис. 6.4). На данній сторінці Ви має згоду переглянути рейтингові списки інших користувачів які вже завантажили свою рейтингову інформацію. Найбільш важливою інформацією данного списку є рейтинговий бал та перелік існуючих балів яки Ви маєте можливість бачити у списку.

ФІО	Бал	Існуючі бали	Дія
Some fio	3.10	{1,2,3,4,10,11,16,}	
some fio 2	3.10	{1,2,3,4,10,11,16,}	

Показати: 5 1 - 2 of 2

Рисунок 6.4 – Сторінка «Рейтингові списки»

За бажанням Ви маєте можливість переглянути повну інформацію за кожним користувачем, вам лише потрібно клікнути на зображення

«Переглянути». Далі ви побачите сторінку «Перегляд файлу» частина інформації з якої предоставлена на рисунку 5.5.

Ім'я	Шкіль Олександр Сергійович>		
Академічний статус	"канд.техн.наук"		
Освіта	Старший науковий співробітник		
Кафедра	Автоматизації проектування обчислювальної техніки>		
Остання зміна файлу	11.11.2019 15:22		

Найменування підпункту Пункту 5 Приміток до Додатку 12	Наявність, реквізити матеріалів	Рік	Найменування дисципліни
1			
Scopus	1. Shkil A. Design Automation of Testable Finite State Machines / M.Miroschnyk, Y. Pakhomov, E. German, A. Shkil, E. Kulak, D. Kucherenko // Proceedings of the Internat. Sympos. EWDTS'2017, September 29 - October 2, Novi Sad, Serbia, 2017. – P.203–208. 2. Shkil A. Expert evaluation model of the computer system diagnostic features / G. Krivoulya, A. Shkil, D. Kucherenko, A. Lipchansky, Ye. Sheremet // EWDTS'2014: Proc. of international conf., 26-29 september, 2014. – Kiev, Ukraine: KhNURE, 2014. – P. 286-289.	2017 2014	1.Логічне моделювання 1. Методи підвищення якості програмного забезпечення
Web of Science Core Collection	1. A. S. Shkil'. Analysis of Production Rules in Expert Systems of Diagnosis / G.F. Krivoulya, A.S. Shkil', D.Ye. Kucherenko / AUTOMATIC CONTROL AND COMPUTER SCIENCES. – 2013. – Vol.47. – No. 6. P.331-341.	2014	1.Логічне моделювання
	—	—	—
2			
наявність наукових публікацій у наукових виданнях, включених до переліку наукових фахових видань України	1. Шкіль А.С. Автоматизация поиска ошибок проектирования в HDL-моделях конечных автоматов / А.С. Шкіль, Г.П. Фастовец, А.С. Серокурова // АСУ и приборы автоматизи.– 2014.– Вып. 168.– С. 43-52. 2. Шкіль, А.С. Кубитные структуры данных вычислительных устройств / В.И. Хаханов, Baahdadi Ammar Awni Abbas, Е.И. Литвинова, А.С. Шкіль // Электронное моделирование.	2014 2015	1.Логічне моделювання 1.Логічне моделювання

Рисунок 6.5 – Верхня частина сторінки «Перегляд файлу»

Наступна сторінка до якої має доступ як адміністратор так і володар системи, це сторінка «Коефіцієнти» (рис. 6.6). На данній сторінці предоставлена можливість змінювати ті чи інші коефіцієнти для кожного конкретного пункту рейтингування. Для того щоб змінити коефіцієнт Вам потрібно лише вибрати необхідний коефіцієнт, натиснути «+» або «мінус» для його інкрементування чи декрементування, або ввести значення у поле вводу. Після того як всі необхідні коефіцієнти були встановлені, Вам потрібно лише натиснути кнопку «Зберегти». Після виконання вище вказаних дій системою автоматично буде перерахован рейтингвий бал для кожного з користувачів системи які вже добавили свої

рейтингові данні до профілю. Як додати свої рейтингові данні до профілю буде розглянуто нижче у цьому розділі.

The image shows a web interface for setting coefficients. It consists of 18 individual input fields arranged in a 6x3 grid. Each field is labeled 'Коефіцієнт' followed by a number from 1 to 18. Each field contains a numerical value and has minus and plus signs on either side for adjustment. The values are: Коефіцієнт 1: 0.55; Коефіцієнт 2: 0.15; Коефіцієнт 3: 0.1; Коефіцієнт 4: 0.1; Коефіцієнт 5: 0.1; Коефіцієнт 6: 0.1; Коефіцієнт 7: 0.1; Коефіцієнт 8: 0.1; Коефіцієнт 9: 0.2; Коефіцієнт 10: 0.1; Коефіцієнт 11: 0.1; Коефіцієнт 12: 0.1; Коефіцієнт 13: 0.1; Коефіцієнт 14: 0.1; Коефіцієнт 15: 0.1; Коефіцієнт 16: 0.1; Коефіцієнт 17: 0.1; Коефіцієнт 18: 0.1. At the bottom center of the grid is a blue button labeled 'Зберегти' (Save).

Коефіцієнт 1	Коефіцієнт 2	Коефіцієнт 3
- 0.55 +	- 0.15 +	- 0.1 +
Коефіцієнт 4	Коефіцієнт 5	Коефіцієнт 6
- 0.1 +	- 0.1 +	- 0.1 +
Коефіцієнт 7	Коефіцієнт 8	Коефіцієнт 9
- 0.1 +	- 0.1 +	- 0.2 +
Коефіцієнт 10	Коефіцієнт 11	Коефіцієнт 12
- 0.1 +	- 0.1 +	- 0.1 +
Коефіцієнт 13	Коефіцієнт 14	Коефіцієнт 15
- 0.1 +	- 0.1 +	- 0.1 +
Коефіцієнт 16	Коефіцієнт 17	Коефіцієнт 18
- 0.1 +	- 0.1 +	- 0.1 +

[Зберегти](#)

Рисунок 6.6 – Сторінка «Коефіцієнти»

Наступним пунктом який ми розглянемо буде сторінка «Додати файл» (рис. 6.7) до якої є доступ у володаря системи та вчителя. Щоб користувач появився у рейтинговому списку йому необхідно додати свій файл до системи. Для цього Вам як користувачу необхідно натиснути на кнопку «Choose File» та вибрати необхідний файл.

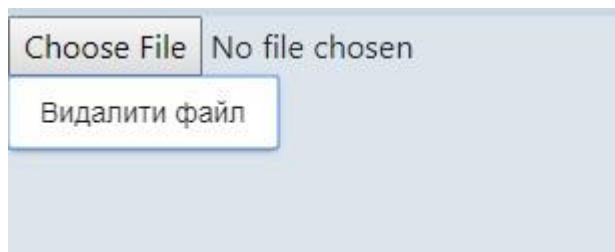


Рисунок 6.7 – Сторінка «Додати файл» коли користувач ще не додав файл

Після того як ви додали файл до системи у вас є можливість його видалити, переглянути та редагувати. Частина сторінки після додавання файлу зображена на рисунку 6.8. Щоб відредагувати файл вам необхідно натиснути на поле яке Ви бажаєте редагувати, внести необхідні зміни та натиснути кнопку «Зберегти зміни».

 A screenshot of a web interface showing a file management system. At the top left, there are buttons for 'Видалити файл' (Delete file) and 'Зберегти зміни' (Save changes). Below these is a metadata panel for a selected file, displaying:

- Ім'я: Шкіль Олександр Сергійович
- Академічний статус: канд.техн.наук
- Освіта: Старший науковий співробітник
- Кафедра: Автоматизації проектування обчислювальної техніки
- Остання зміна файлу: 01.12.2019 23:57

 Below the metadata panel is a table with the following columns: 'Найменування підпункту Пункту 5 Приміток до Додатку 12', 'Наявність, реквізити матеріалів', 'Рік', and 'Найменування дисципліни'. The table contains three rows of data:

Найменування підпункту Пункту 5 Приміток до Додатку 12	Наявність, реквізити матеріалів	Рік	Найменування дисципліни
1 Scopus	1. Shkil A. Design Automation of Testable Finite State Machines / M.Miroschnyk, Y. Pakhomov, E. German, A. Shkil, E. Kulak, D. Kucherenko // Proceedings of the Internat. Sympos. EWDTS'2017, September 29 - October 2, Novi Sad, Serbia, 2017. – P.203–208.	2017	1.Логічне моделювання
	2. Shkil A. Expert evaluation model of the computer system diagnostic features / G. Krivoulya, A. Shkil, D. Kucherenko, A. Lipchansky, Ye. Sheremet // EWDTS'2014: Proc. of international conf., 26-29 september, 2014. – Kiev, Ukraine: KhNURE, 2014. – P. 286-289.	2014	1. Методи підвищення якості програмного забезпечення
Web of Science Core Collection	1. A. S. Shkil, Analysis of Production Rules in Expert Systems of Diagnosis / G.F. Krivoulya, A.S. Shkil, D.Ye. Kucherenko / AUTOMATIC CONTROL AND COMPUTER SCIENCES. – 2013. – Vol.47. – No. 6. P.331-341.	2014	1.Логічне моделювання

Рисунок 6.8 – Сторінка «Додати файл» після додавання файлу до системи

ВИСНОВКИ

Сьогоднішній світ досяг нової ери, в якій інформація відіграє важливу роль, а також економіки, яка базується на ній. Сучасна еволюція інформаційного суспільства безпосередньо пов'язана з потребою збирати, обробляти та передавати великі обсяги інформації та перетворювати інформацію в загально цінний товар.

На мою думку, це стосується, зокрема, освіти та науки. Розробка та впровадження інтелектуальних систем скорочує персонал і час, дозволяє уникнути помилок і неточностей через повну або часткову відсутність людських факторів.

У цьому дослідженні були враховані умови провадження освітньої діяльності навчальних закладів. Були встановлені необхідні прикладні модулі, необхідні для розробки повної та функціональної інтелектуальної системи управління навчальним персоналом, та розглядалися технології для найбільш продуктивної та відповідної реалізації визначених модулів. Були обрані методи захисту від зловмисників.

Визначені компоненти інтелектуальної системи охоплюють усі потреби, виявлені під час дослідження предметної області.

На основі проаналізованої постанови Кабінету Міністрів України 1187 були розроблені процедури розрахунку рейтингу для працівників навчального закладу.

Результатом цієї роботи є розробка концепції інтелектуальної системи управління персоналом навчального закладу, яка повинна призвести до зниження витрат на списки оцінювання викладачів та до підвищення якості освіти пропонованої вищим навчальним закладом.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Постанова Кабінету Міністрів України 1187 про затвердження ліцензійних умов провадження освітньої діяльності від 30 грудня 2015 р.
[Електронний ресурс] / Режим доступу : [www/ URL: http://zakon2.rada.gov.ua/laws/show/1187-2015-%D0%BF/ed20151230](http://zakon2.rada.gov.ua/laws/show/1187-2015-%D0%BF/ed20151230) – 06.06.2018 – Загл. с екрана.
2. Постанова Кабінету Міністрів України 1187 про затвердження ліцензійних умов провадження освітньої діяльності від 23 травня 2018 р.
[Електронний ресурс] / Режим доступу : [www/ URL: http://zakon2.rada.gov.ua/laws/show/1187-2015-%D0%BF/ed20180523](http://zakon2.rada.gov.ua/laws/show/1187-2015-%D0%BF/ed20180523) ed20151230 – 06.06.2018 – Загл. с екрана.
3. Антифеев Дм.Д. Современные средства построения корпоративных систем поддержки принятия управленческих решений [Текст] Дм.Д Антифеев. – М. :Терн, 2001. – 94 с.
4. Рогач І. Ф. Інформаційні системи у фінансово-кредитних установах [Текст] : навч. посібник. / І. Ф. Рогач, М. А. Сендзюк, В. А. Антонюк— 2-ге вид. – М. :Терн, 2001. – 172 с.
5. Савчук Т.О. Організація баз даних і знань [Текст] / Т.О. Савчук. – В. : ВДТУ, 2000. – 63 с.
6. Системи обробки економічної інформації. [Електронний ресурс] / Режим доступу : [www / URL: http://www.zsu.zp.ua/lab/mathdep/mme/IV/soei/index.htm](http://www.zsu.zp.ua/lab/mathdep/mme/IV/soei/index.htm) – 06.06.2018 – Загл. с екрана.

7. Бондар Є. С. Хмарні обчислення та їх застосування [Текст] / Є. С. Бондар, М. М. Глибовець, С. С. Гороховський // Вісник КНУ ім. Т. Шевченка. – 2011. – К.: КНУ, № 1. – С. 74–82.
8. Кормич Б. А. Інформаційна безпека: організаційно-правові основи: навч. посібник [Текст] / Б. А. Кормич. – К.: Кондор, 2004. – 384 с.
9. Белов Е. Б. Основы информационной безопасности: учеб. пособие [Текст] / Е. Б. Белов, В. П. Лось, Р. В. Мещеряков, А. А. Шелупанов. – М.: Горячая линия-Телеком, 2006. – 544 с.
10. Беляева А. П. Интегративно-модульная педагогическая система профессионального образования. [Текст] / А. П. Беляева. – СПб. : Радом, 1997. – 226 с.
11. Запровадження інтелектуальних інформаційних технологій для систем підтримки прийняття рішень [Текст] / А. О. Морозов, В. Л. Косолапов, Л. В. Ромашкина, Р. П. Муренко – М. : НТІ, 2003. – 41 с.
12. Універсальна модель процесу функціонування інформаційного об'єкта в інтелектуальній системі управління / О. В. Барабаш, С. В. Ленков, Я. І. Лепіх та ін. [Текст] // Сенсор. електроніка і мікросистем. технології. – 2010. – № 3. – С. 87–90.
13. Кондрюков С.Е. Интеллектуальная система анализа соответствия научно-педагогического персонала вузов кадровым требованиям образовательного процесса / С.Е. Кондрюков // Радиоелектроніка та молодь у ХХІ столітті. – 2019. – № 5. – С. 9–10.