

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ПОЛІСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій обліку та фінансів  
Кафедра комп'ютерних технологій  
і моделювання систем

Кваліфікаційна робота  
на правах рукопису Янчук Назар Олександрович

УДК 004:331.108.22

## **КВАЛІФІКАЦІЙНА РОБОТА**

Інформаційна система обліку співробітників на підприємстві  
122 «Комп'ютерні науки»

Подається на здобуття освітнього ступеня бакалавр

кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Янчук.Н.О

(підпис, ініціали та прізвище здобувача вищої освіти)

Керівник роботи  
Дрейс Юрій Олександрович  
к.т.н., доцент

Житомир – 2023

**Висновок кафедри** \_\_\_\_\_  
за результатами попереднього захисту: \_\_\_\_\_

Протокол засідання кафедри \_\_\_\_\_  
№ \_\_\_ від «\_\_\_» \_\_\_\_\_ 20\_\_\_ р.

Завідувач кафедри \_\_\_\_\_  
\_\_\_\_\_  
(науковий ступінь, вчене звання) (підпис) (прізвище, ім'я, по батькові)  
«\_\_\_» \_\_\_\_\_ 20\_\_\_ р.

### **Результати захисту кваліфікаційної роботи**

Здобувач вищої освіти \_\_\_\_\_ захистив (ла)  
(прізвище, ім'я, по батькові)

кваліфікаційну роботу з оцінкою:

сума балів за 100-бальною шкалою \_\_\_\_\_  
за шкалою ECTS \_\_\_\_\_  
за національною шкалою \_\_\_\_\_

Секретар ЕК

\_\_\_\_\_  
(науковий ступінь, вчене звання) (підпис) (прізвище, ім'я, по батькові)

## АНОТАЦІЯ

Янчук Н.О. Інформаційна система обліку співробітників на підприємстві.  
– Кваліфікаційна робота на правах рукопису.

Кваліфікаційна робота на здобуття освітнього ступеня бакалавра за спеціальністю 122 – комп'ютерні науки. – Поліський національний університет, Житомир, 2023.

Дипломна робота присвячена розробці інформаційної системи кадрового обліку на підприємстві. У роботі проведений детальний аналіз предметної області кадрового обліку, включаючи функції та завдання, які повинна вирішувати система. Запропоновано архітектуру системи, що дозволяє ефективно виконувати облік персоналу, управляти кадрами .

Результатом роботи є створена інформаційна система кадрового обліку у вигляді веб-додатку, яка допомагає підприємству ефективно вести облік персоналу. Розроблена система може використовуватись для покращення роботи з персоналом та забезпечення ефективної управлінської діяльності на підприємстві.

Ключові слова: кадровий облік, інформаційна система, управління кадрами, мультиплатформеність, персонал, співробітники, веб-додаток.

## SUMMARY

Yanchuk N.O. Information system for accounting of employees at the enterprise.

– Qualification work on manuscript rights.

Qualification work for obtaining a bachelor's degree in specialty 122 - computer science. – Polis National University, Zhytomyr, 2023.

The thesis is devoted to the development of an information system of personnel accounting at the enterprise. The work provides a detailed analysis of the subject area of personnel accounting, including the functions and tasks that the system should solve. The architecture of the system is proposed, which allows efficient accounting of personnel, management of personnel.

The result of the work is the creation of a personnel accounting information system in the form of a web application, which helps the enterprise to effectively keep personnel records. The developed system can be used to improve work with personnel and ensure effective management activities at the enterprise.

Keywords: personnel accounting, information system, personnel management, multi-platform, personnel, employees, web application.

## ЗМІСТ

ВСТУП .....	6
Розділ 1 ТЕОРЕТИЧНИЙ АНАЛІЗ ІНФОРМАЦІЙНИХ СИСТЕМ КАДРОВОГО ОБЛІКУ СПІВРОБІТНИКІВ НА ПІДПРИЄМСТВІ.....	8
1.1 Аналіз інформаційних потреб і визначення предметної області дослідження .....	8
1.2 Аналіз існуючих інформаційних систем кадрового обліку.....	9
1.3 Моделювання бізнес-процесів предметної області .....	10
Висновки до першого розділу .....	11
Розділ 2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ СПІВРОБІТНИКІВ НА ПІДПРИЄМСТВІ.....	12
2.1 Інструменти реалізації інформаційної системи обліку співробітників на підприємстві.....	13
2.2 Моделювання інформаційної системи обліку співробітників на підприємстві .....	13
2.3 Проектування бази даних інформаційної системи .....	15
Висновки до другого розділу .....	19
Розділ 3 РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ .....	20
3.1 Опис програмних модулів інформаційної системи .....	20
3.2 Інструкція користувачу інформаційної системи.....	23
Висновки до третього розділу.....	24
ВИСНОВКИ .....	25
ПЕРЕЛІК ЛІТЕРАТУРНИХ ДЖЕРЕЛ .....	26

## ВСТУП

Інформаційна система обліку співробітників є невід'ємною частиною ефективного управління персоналом на будь-якому підприємстві. Ця система допомагає вирішувати багато завдань, пов'язаних з кадровою справою, включаючи підтримку бази даних про співробітників, ведення обліку робочого часу, визначення заробітної плати, формування звітів і багато іншого. Використання інформаційної системи кадрового обліку дозволяє підвищити ефективність роботи відділу кадрів, скоротити час, необхідний для вирішення завдань, зменшити ймовірність помилок та забезпечити точність інформації.

У даній роботі буде розглянута інформаційна система кадрового обліку у вигляді веб-додатку, яка дозволяє зручно та швидко виконувати необхідні завдання з будь-якого місця з доступом до Інтернету.

**Актуальність роботи** має особливе значення в сучасному бізнес-середовищі оскільки збирати, зберігати та обробляти дані про всіх співробітників на підприємстві – це великий об'єм роботи, а дана система допомагає автоматизувати багато рутинних адміністративних завдань, пов'язаних з кадрами.

**Метою** кваліфікаційної роботи є розробка (веб-орієнтованої системи) для автоматизації процесів кадрового обліку співробітників підприємства.

При виконанні даної роботи було поставлено наступні **завдання**:

- проведення детального аналізу предметної області кадрового обліку;
- аналіз існуючих інформаційних систем кадрового обліку;
- моделювання інформаційної системи обліку співробітників;
- створити інформаційну систему у вигляді веб-додатку.

**Об'єкт дослідження** – інформаційна система кадрового обліку у вигляді веб-додатку.

**Предмет дослідження** – процес створення веб-додатку для обліку співробітників на підприємстві.

**Методи дослідження**: метод аналізу, метод порівняння, метод системного аналізу, метод моделювання, метод комп'ютерного моделювання.

Аналіз літературних джерел та інформаційних ресурсів: цей метод дозволяє отримати теоретичні знання про існуючі інформаційні системи кадрового обліку.

Порівняння: цей метод дозволяє аналізувати та порівнювати існуючі інформаційні системи кадрового обліку.

Моделювання: цей метод включає розробку та тестування програмного забезпечення для інформаційної системи кадрового обліку

Комп'ютерне моделювання: цей метод використовується для розробки прототипів програмного забезпечення та тестування його роботи в різних умовах.

За темою кваліфікаційної роботи було опубліковано наукові публікації, а саме:

1.Янчук Н.О.,Удосконалення інформаційної системи обліку працівників підприємства. Студентські наукові читання – 2022 : зб. тез доповідей науково-практичної конференції «Студентські наукові читання – 2022» за результатами I туру.

2.Янчук Н.О.,Обґрунтування розробки інформаційної системи обліку працівників на підприємстві. Інформаційні технології та моделювання систем : збірник праць учасників Всеукраїнської науково-практичної конференції здобувачів вищої освіти і молодих вчених, 30 березня 2023 р. Житомир : Поліський національний університет, 2023. 140 с.

**Практичне значення** отриманих результатів полягає в: забезпеченні точності та надійності даних в системі кадрового обліку, автоматизації процесів . В результаті цього організація зможе більш ефективно облікувати своїх працівників, знизити ризики та забезпечити оптимальне використання ресурсів.

**Структура та обсяг роботи.** Дипломна робота складається зі вступу, трьох розділів основної частини, висновків, списку використаної літератури та додатків.

# **1 ТЕОРЕТИЧНИЙ АНАЛІЗ ІНФОРМАЦІЙНИХ СИСТЕМ КАДРОВОГО ОБЛІКУ СПІВРОБІТНИКІВ НА ПІДПРИЄМСТВІ**

## **1.1 Аналіз інформаційних потреб і визначення предметної області дослідження**

Облік співробітників – це безперервна робота, що полягає в збиранні та класифікації даних про кожного співробітника, а також забезпеченні їхнього зберігання, швидкого доступу до необхідної інформації та можливості її оперативного отримання з метою збору до державних чи місцевих органів управління. Для цього на підприємстві функціонує відділ кадрів, який займається формуванням та збереженням особових карток кожного співробітника, а також збереженням трудових договорів, характеристик, формуванням та збереженням трудових книжок, характеристик особового складу тощо. Кадровий облік – це ціла система заходів щодо систематизації відомостей на всіх рівнях підприємств – незалежно від розміру, обсягу бізнесу, форми власності, особливостей організаційної структури тощо. [1]

Його мета – акцентувати на найменших деталях кількісної та якісної структури персоналу компанії [1].

Основною функцією кадрової служби є формування оптимального керування об'єктами та визначення майбутніх та поточних потреб в управлінні персоналом, оскільки керівництво наразі становить 30% робочої сили організації [2].

Для вирішення поставлених завдань автоматизації обліку співробітників було прийнято рішення розробити веб-додаток. Таке рішення було прийнято з метою забезпечити зручний та швидкий доступ до системи обліку співробітників з будь-якого місця та пристрою, що має підключення до Інтернету. Розробка веб-додатку дозволить користувачам зручно взаємодіяти з системою через веб-браузер, не вимагаючи встановлення додаткового програмного забезпечення на їх пристроях.



Веб-додаток також надає можливість забезпечити багатоплатформеність системи, що означає, що вона буде доступна та працездатна на різних операційних системах, таких як Windows, macOS, Linux, а також на різних пристроях, включаючи комп'ютери, ноутбуки, планшети та смартфони. Це робить систему гнучкою та доступною для використання у різних робочих умовах та в контексті розрізних команд та розташування співробітників.

## 1.2 Аналіз існуючих інформаційних систем кадрового обліку

Зараз на ринку існує безліч систем кадрового обліку, які відрізняються функціональними можливостями, ціною та складністю використання. У даному розділі ми розглянемо найпопулярніші системи кадрового обліку та проведемо їх порівняльний аналіз. Порівняльний аналіз деяких функцій та можливостей цих програм приведено у таблиці 1 [4-7].

Таблиця 1.2 Існуючі системи кадрового обліку .

Функції та можливості	Міні-кадри	Кадри Плюс	Чиж-Кадри	Бухгалтерія 8
Віддалений доступ	–	+	–	+
Можливість входу з будь-якого пристрою	–	–	–	–
Ведення особистої картки користувача	+	+	+	+
Захист інформації	+	+	+	+
Мобільність	–	–	–	–
Багатоплатформеність	–	+	–	+
Пошук працівників за параметрами	–	+	+	+
Збір статистичних даних	–	+	+	–

### 1.3 Моделювання бізнес-процесів предметної області

Методологія IDEF3 є одним зі стандартів сімейства IDEF і досить широко використовується при декомпозиції моделей IDEF0 для моделювання процесів нижчого рівня, оскільки з його допомогою можна змоделювати технологічні процеси, що відбуваються на підприємстві, тобто описати можливі сценарії реалізації процесів, в рамках яких відбувається послідовна зміна властивостей об'єкта. Дана методологія дозволяє показувати можливі розгалуження в процесі. Наприклад, коли результат однієї дії може ініціювати запуск декількох дій або навпаки, щоб почати якусь дію, необхідно завершити кілька попередніх дій [8].

Декомпозиція процесу "Організувати кадровий облік, управління персоналом". Разом із заявкою надходить документація на нового співробітника. Вся документація проходить перевірку, після чого приймається рішення про прийом нового співробітника. У разі позитивного рішення відбувається прийом нового співробітника, що передбачає реєстрацію його в базі, розподіл навантаження та встановлення графіка роботи. Останньою дією складається звіт про виконану роботу.

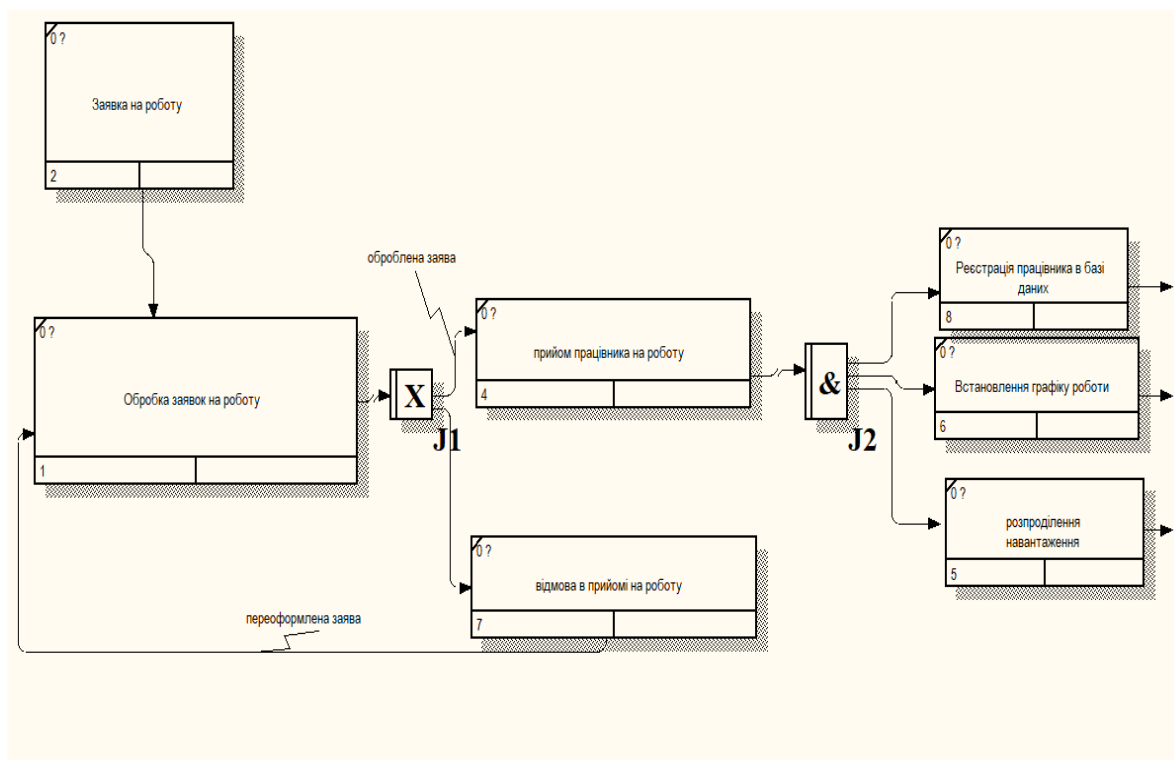


Рисунок 1.3 – Декомпозиція процесу "Організувати кадровий облік, управління персоналом"

## **ВИСНОВКИ ДО ПЕРШОГО РОЗДІЛУ**

У першому розділі роботи було проведено детальний аналіз предметної області, що стосується обліку співробітників. Описані різні аспекти цього процесу, включаючи збирання, класифікацію та зберігання даних про кожного співробітника. Також проведено аналіз існуючих систем кадрового обліку, порівняно їх функціональні можливості, доступність з різних пристроїв, рівень захисту інформації, мобільність та багатоплатформеність. На заключному етапі першого розділу продемонстровано бізнес-процес предметної області, який описує послідовність дій, які відбуваються при організації кадрового обліку та управлінні персоналом.

## 2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ СПІВРОБІТНИКІВ НА ПІДПРИЄМСТВІ

### 2.1 Інструменти реалізації інформаційної системи обліку співробітників на підприємстві

HTML (HyperText Markup Language) – це стандартна мова розмітки для створення веб-сторінок. Вона використовується для структурування та представлення вмісту на веб-сторінках, таких як текст, зображення, посилання, форми, таблиці і багато іншого. HTML використовує теги для визначення різних елементів сторінки та їх характеристик. За допомогою HTML можна створювати структуровані документи, які браузері можуть інтерпретувати та відображати користувачам у вигляді веб-сторінок [9].

Bootstrap – це CSS - framework. Bootstrap входить в число найпопулярніших фреймворків HTML, CSS і JS. Це безкоштовний набір інструментів для створення сайтів, який включає в себе елементи типографіки, готові веб-форми, кнопки, меню та інші веб-елементи. Основним напрямком фреймворку визнана розробка складних мобільних проектів. Важливою перевагою Bootstrap є велика спільнота, що в п'ять разів перевершує конкурентів [10].

Laravel – це високорівневий веб-фреймворк з відкритим кодом, розроблений для швидкої та ефективної розробки веб-додатків. Він побудований на мові програмування PHP і надає розширені можливості для створення потужних веб-додатків з чистим та елегантним кодом.

Laravel пропонує багато готових рішень та компонентів, які допомагають забезпечити швидку розробку, такі як маршрутизація, контролери, ORM (об'єктно-реляційна карта), шаблонізатор Blade, міграції бази даних та багато іншого. Він також підтримує сучасні практики розробки, включаючи MVC (Model-View-Controller) архітектуру, роутинг, аутентифікацію, кешування та безпеку [11].

Laravel став популярним вибором серед розробників для створення різноманітних веб-додатків, від невеликих сайтів до складних корпоративних проектів [12].

MySQL – це система управління базами даних, яка використовується для підтримки реляційних баз даних. Це програмне забезпечення з відкритим кодом, що підтримується корпорацією Oracle. Спочатку вона була заснована шведською компанією під назвою MYSQL AB, яка згодом була придбана сонячними мікросистемами і, нарешті, є корпорацією Oracle. Оскільки це система баз даних з відкритим кодом, вихідний код можна змінювати відповідно до наших потреб. Він також пропонує послуги преміум-класу, якщо комерційна ліцензія придбана у корпорації Oracle. MySQL – це масштабована, швидка і надійна система управління базами даних, яка може працювати на будь-якій платформі [13].

## **2.2 Моделювання інформаційної системи обліку співробітників на підприємстві**

Діаграма послідовності по задачі, що вирішується – відображає потік подій, що відбуваються в рамках варіанту використання. На цих діаграмах зображуються ті об'єкти, які безпосередньо беруть участь у взаємодії оскільки ключовим моментом є саме динаміка взаємодії об'єктів у часі і використовуються можливі статичні асоціації коїться з іншими об'єктами[14]. Одне – зліва направо як вертикальних ліній, кожна з яких зображує лінію життя окремого об'єкта, що у взаємодії. Другий вимір – вертикальна тимчасова вісь, спрямована зверху донизу. При цьому взаємодії об'єктів реалізуються за допомогою повідомлень, які надсилаються одними об'єктами іншим. Повідомлення зображуються у вигляді горизонтальних стрілок з ім'ям повідомлення і утворюють порядок за часом свого виникнення [15].

Дана візуальна діаграма послідовності ілюструє взаємодію між користувачем, веб-сервером, базою даних та системою обліку співробітників при додаванні даних співробітника. Користувач відкриває веб-сайт і взаємодіє з веб-додатком. Веб-сервер виконує запит до бази даних для перевірки прав доступу та отримання результатів. Після цього відбувається додавання даних співробітника до бази даних, підтвердження операції та повернення результатів користувачеві.

### Діаграма послідовності виконання процесів системи

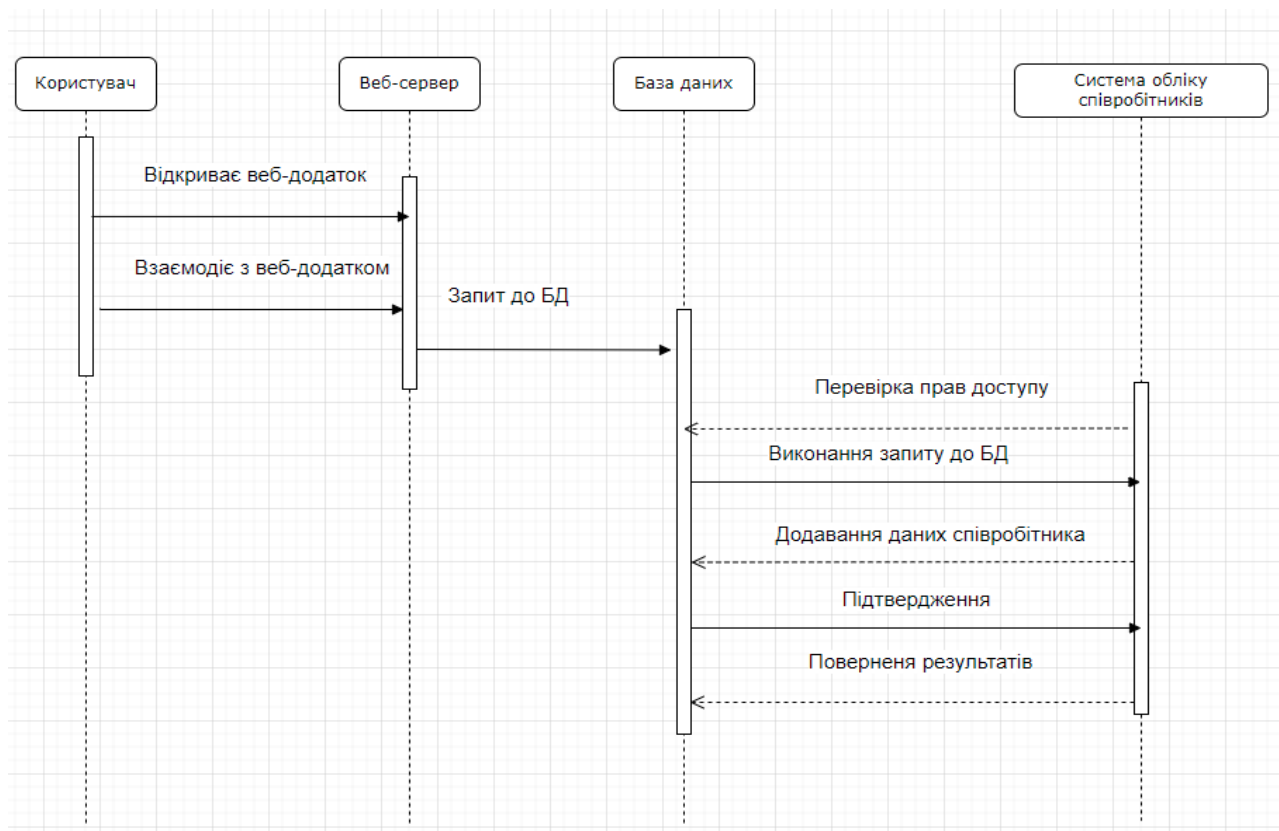


Рисунок 2.2– «Діаграма послідовності»

Діаграма діяльності з задачі, що розв'язується, застосовуються для моделювання процесу виконання операцій. При моделюванні поведінки проектованої чи аналізованої системи виникає необхідність як уявити процес зміни її станів, а й деталізувати особливості алгоритмічної і логічної реалізації виконуваних системою операцій [15].

Дана діаграма демонструє послідовність дій, які відбуваються в процесі взаємодії з системою обліку співробітників. Вона відображає кожен крок алгоритму та умови розгалуження. Починаючи з запуску системи, користувач вводить інформацію про співробітника, яка потім перевіряється. Якщо дані коректні, вони аналізуються, і якщо дані достатньо, зміни зберігаються в базі даних. Якщо дані не достатньо, користувач повторно вводить інформацію. Цей процес повторюється до тих пір, поки всі дані не будуть збережені в базі даних.

*Діаграма діяльності процесу введення та збереження інформації про співробітника*

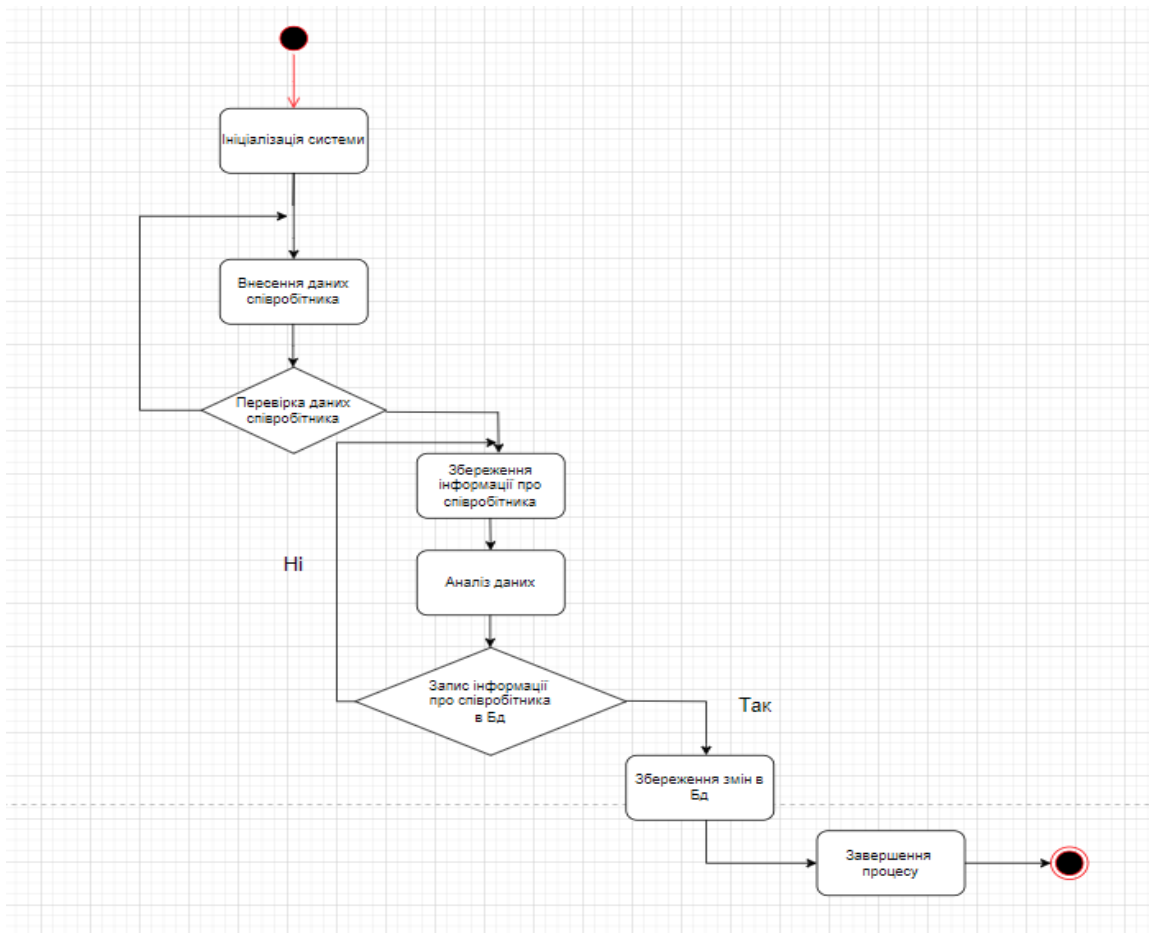


Рисунок 2.3 – «Діаграма діяльності»

### 2.3 Проектування бази даних інформаційної системи

Підприємство може складатися з виробничих структурних підрозділів (виробництв, цехів, відділень, дільниць, бригад, бюро, лабораторій тощо), а також функціональних структурних підрозділів апарату управління (управлінь, відділів, бюро, служб тощо). Таблиця 2.1 subdivisions зберігає назви цих підрозділів.

Таблиця 2.1 Структура таблиці subdivisions (підрозділи)

Ключ	Поле	Опис поля
PM	id	ID підрозділу
	name	Назва підрозділу

Посада – це визначена структурою і штатним розписом первинна структурна одиниця на підприємстві. Таблиця 2.2 positions зберігає назви посад та їх приналежність до підрозділів [16].

Таблиця 2.2 Структура таблиці positions

Ключ	Поле	Опис поля
PM	id	ID посади
	name	Назва посади
FK	subdivision_id	Підрозділ, до якої ця посада належить

В таблиці 2.3 employee, знаходиться особиста інформація про працівника, такі як його адреса, дата народження, попередні місця роботи, закінчені заклади освіти, контактні дані, фотографія тощо.

Таблиця 2.3 Структура таблиці employee.

Ключ	Поле	Опис поля
PM	id	ID співробітника
	date_of_employment	Дата найму
	date_of_dismissal	Дата звільнення
FK	position_id	ID посади
	Address	Адреса проживання співробітника
	date_of_birth	Дата народження співробітника
	job1	Попереднє місце роботи співробітника
	job1_from	Дата найму на попереднє місце роботи
	job1_to	Дата звільнення з попереднього місця роботи
	job1_position	Посада на попередньому місці роботи
	university1_name	Заклад освіти



	specialty1	Спеціальність
	university1_from	Дата початку навчання
	university1_to	Дата закінчення навчання
	university1_degree	Ступінь здобутої освіти
	married	Сімейний стан
	military	Військовий облік
	passport_seria	Серія паспорту
	passport_number	Номер паспорту
	passport_organ	Орган що видав
	passport_date	Дата видачі паспорту
	pib	Прізвище, ім'я та по- батькові співробітника

В таблиці 2.5 users зберігаються дані автентифікації користувача системи.

Таблиця 2.5 Структура таблиці users

Ключ	Поле	Опис поля
PM	id	ID посади
	name	Ім'я користувача
	email	Електронна пошта користувача системи
	password	Пароль користувача (шифрується)
	created_at	Дата і час реєстрації
	updated_at	Дата і час зміни даних

В таблиці 2.6 educational\_levels зберігаються назви рівнів освіти співробітників

Таблиця 2.6 Структура таблиці educational\_levels

Ключ	Поле	Опис поля
PM	id	ID рівня
	name	Назва рівня

В таблиці 2.7 vacations зберігається інформація про відпустки співробітників

Таблиця 2.7 Структура таблиці vacations

Ключ	Поле	Опис поля
PM	id	ID відпустки
	from	Дата початку відпустки
	to	Дата закінчення відпустки
	description	Додаткова інформація
	created_at	Дата створення запису

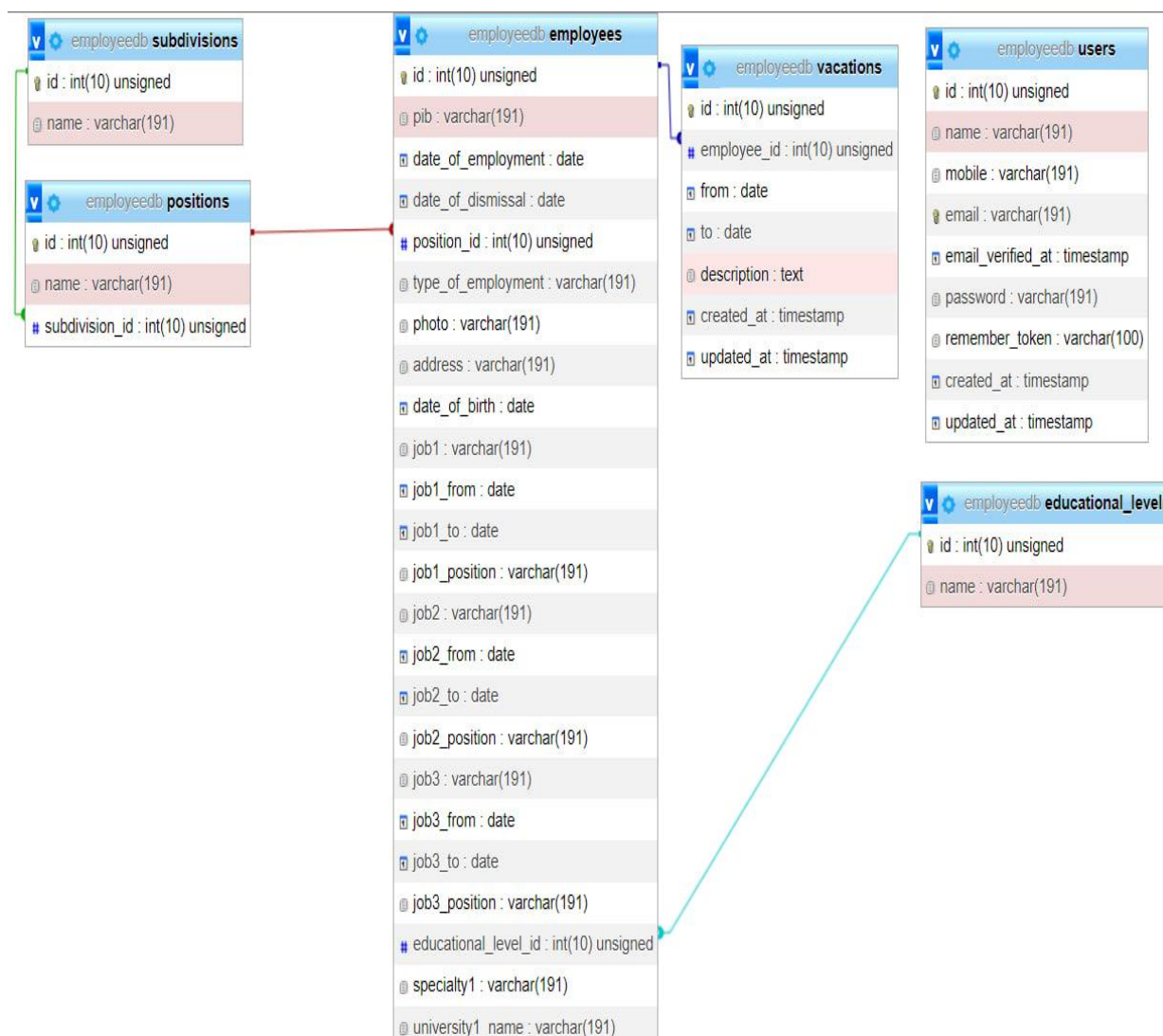


Рисунок 2.4—«Структура бази даних системи »

## ВИСНОВКИ ДО ДРУГОГО РОЗДІЛУ

У другому розділі були описані інструменти, за допомогою яких була створена інформаційна система обліку співробітників. Було використано сучасні технології та інструменти розробки, зокрема мови програмування, фреймворки та засоби веб-розробки. Далі було продемонстровано процес моделювання інформаційної системи за допомогою нотації UML для цього була використана діаграма послідовності, яка відображає послідовність дій між користувачем та системою. Далі, була розроблена діаграма діяльності, яка ілюструє послідовність дій під час введення та збереження інформації про співробітника. Дана діаграма показує процес взаємодії між користувачем, веб-сервером, базою даних та системою обліку співробітників. Після цього була розроблена структура бази даних інформаційної системи. Були визначені таблиці та їхні взаємозв'язки, що описують дані про співробітників. Ця структура бази даних забезпечує зручне та ефективно зберігання інформації, необхідної для обліку співробітників.

## 3 РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ СПІВРОБІТНИКІВ НА ПІДПРИЄМСТВІ

### 3.1 Проектування інтерфейсу інформаційної системи

Для авторизації на сайті була розроблена зручна форма в якій користувачеві необхідно лише ввести свій пароль та пошту.

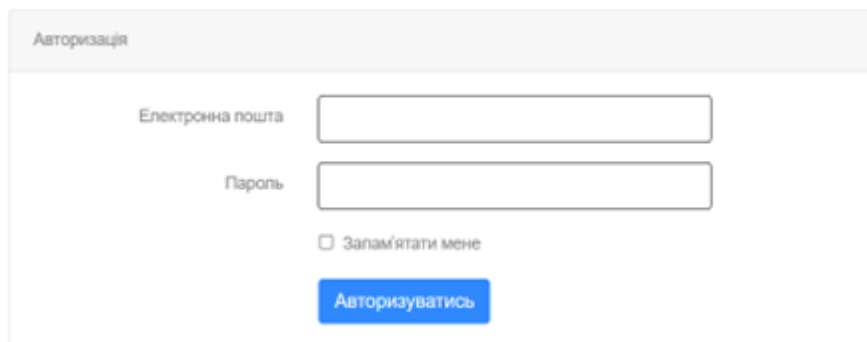
The image shows a web form titled "Авторизація" (Authorization). It contains two input fields: "Електронна пошта" (Email) and "Пароль" (Password). Below the password field is a checkbox labeled "Запам'ятати мене" (Remember me). At the bottom of the form is a blue button labeled "Авторизуватись" (Log in).

Рисунок 3.1—«Форма авторизації»

Після авторизації в системі адміністратор перейде на головну сторінку сайту.

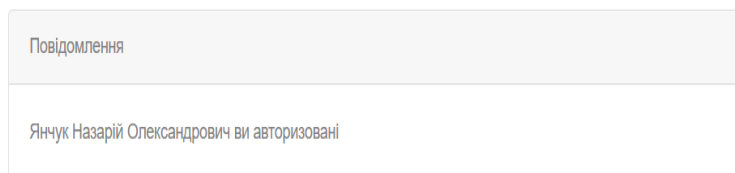
The image shows a notification message box with a light gray header labeled "Повідомлення" (Notification). The main content of the message reads: "Янчук Назарій Олександрович ви авторизовані" (Yanchuk Nazariy Oleksandrovich you are authorized).

Рисунок 3.2—«Головна сторінка»

Форма посади відділи дає змогу адміністратору створювати нові посади та відділи.

#	Посада \ Підрозділ	Кількість співробітників
1	Директор \ Адміністративний	1
2	Заступник директора \ Адміністративний	1
3	Головний бухгалтер \ Бухгалтерський	2
4	Системний Адміністратор \ IT-відділ	0
5	Програміст \ IT-відділ	0
6	Технічна підтримка \ Адміністративний	0
7	Архітектор систем \ IT-відділ	2
8	Спеціаліст по БД \ IT-відділ	1

Рисунок 3.3—«Форма Посади/Відділи»

Форма співробітників дає можливість адміністратору створити нового користувача та при необхідності знайти інформацію про нього в таблиці.

Ім'я	Посада	Вік	Стать	Одружений	Стаж	Статус працівника	Пільгова категорія	Стоїть на військовому обліку
<a href="#">Григорян Арсен Петрович</a>	Архітектор систем	45	Чоловік	Так	15 років	Працює	Особи, які постраждали від торгівлі людьми	Так
<a href="#">Марченко Сергій володимирович</a>	Спеціаліст по БД	36	Чоловік	Так	3 років	Працює	Немає	Так
<a href="#">Михайло Володимир Олександрович</a>	Заступник директора	22	Чоловік	Ні	2 років	Працює(у відпустці)	Немає	Так
<a href="#">Ольга Василівна Годзь</a>	Головний бухгалтер	29	Чоловік	Так	1 років	Працює	Жінки, яким присвоєно почесне звання України "Мати-героїня"	Ні
<a href="#">Петрівський Олег Олександрович</a>	Архітектор систем	28	Чоловік	Так	6 років	Працює(у відпустці)	Учасники бойових дій	Так
<a href="#">Янчук Назарій Олександрович</a>	Директор	32	Чоловік	Ні	1 років	Працює(у відпустці)	Немає	Так
<a href="#">Янчук Сергій Олександрович</a>	Програміст	36	Чоловік	Так	1 років	Не працює	Малозабезпечені сім'ї	Так
<a href="#">Ясенюк Тамара Петрівна</a>	Головний бухгалтер	22	Жінка	Так	23 років	Працює	Жінки, яким присвоєно почесне звання України "Мати-героїня"	Ні

Рисунок 3.4—«Форма Співробітники»

Головна сторінка співробітника містить інформацію про співробітника і дає змогу адміністратору редагувати інформацію та звільнити користувача.

**Петриківський Олег  
Олександрович**  
Архітектор систем, IT-відділ  
Працює

Повне ім'я: Петриківський Олег Олександрович  
Email: example@gmail.com  
Телефон: (097) 234-5678  
Мобільний: (098) 765-4321  
Адреса: Житомир  
Шлюб: Так  
Військовий облік: Так  
Категорія пільги: Учасники бойових дій

Паспортні дані:  
Номер: 1233123  
Серія: 1231313  
Орган, що видав: 2345  
Дата видачі: 2016-05-02

Освіта: Чернівецький університет, Інформаційні технології, 2012 - 2016, Молодший спеціаліст

Попереднє місце роботи: "Атлас", Архітектор систем, 2014.01 - 2015.05

Рисунок 3.5—«Сторінка співробітника»

Форма відпустки дає змогу адміністратору відправити у відпустку співробітника.

Додати відпустку

Show 10 entries Search:

Співробітник	Посада	Підрозділ	Срок	Кількість днів	Дії
<a href="#">Бульба Олександр Григорович</a>	Менеджер проєктів	Відділ IT	2023-06-13 - 2023-06-23	10 днів	<a href="#">Скасувати відпустку</a>
<a href="#">Янух Назарій Олександрович</a>	Директор	Адміністративний	2023-06-08 - 2023-06-14	6 днів	<a href="#">Скасувати відпустку</a>

Showing 1 to 2 of 2 entries Previous 1 Next

Рисунок 3.6—«Форма відпустки»

Форма статистика дає змогу адміністратору переглядати статистику по віковому рівню та за рівнем освіченості.

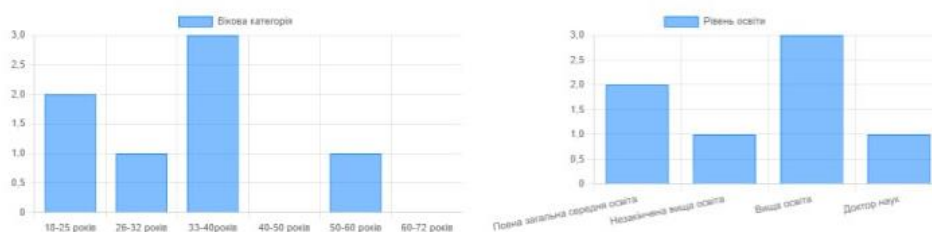


Рисунок 3.7—«Форма статистика»

### 3.2 Інструкція користувачу інформаційної системи

1. Для початку роботи з системою необхідно авторизуватись. Щоб увійти у систему потрібно ввести електронну пошту а також пароль.

2. Після введення даних та вдалої авторизації, на екрані бачимо повідомлення про успішний вхід в систему.

3. Для того, щоб переглянути список посад та відділів потрібно натиснути на кнопку «Посади/Відділи» в панель-меню. Буде здійснено перехід на сторінку з таблицею посад та відділів. В цій таблиці, також, можна буде побачити кількість співробітників закріплених за посадами. Сторінка «Посади/Відділи».

4. Щоб додати новий відділ або нову посаду необхідно натиснути на кнопку «Новий відділ» або «Нова посада» відповідно, заповнити необхідні дані та натиснути кнопку «Створити».

5. Щоб переглянути список співробітників, необхідно натиснути на кнопку «Співробітники» . Буде здійснено перехід на сторінку з таблицею співробітників. В цій таблиці можемо бачити ПІБ співробітника, його посаду та відділ, тип зайнятості, статус (працює чи звільнений) і дату прийому на роботу.

6. Для того, щоб створити нового співробітника натискаємо кнопку «Створити співробітника». Буде здійснено перехід на сторінку з формою для заповнення даних про нового співробітника. Дані діляться на два типа: особисті (дата народження, адреса, освіта, попередні місця та інші) і корпоративні (ПІБ, посада та підрозділ, дата прийняття на роботу, дата звільнення, статус працівника та інші). Після заповнення форми натискаємо на кнопку «Створити».

7. Для того, щоб переглянути інформацію про конкретного працівника системи натискаємо на його посилання на профіль в таблиці співробітників. Здійснюється перехід на сторінку з профілем співробітника. Ця форма дає змогу адміністратору редагувати інформацію про працівника та при необхідності його звільнити.

8. Для того щоб створити відпустку треба натиснути на кнопку «Додати відпустку» після цього з'явиться вікно зі списком співробітників в якому

адміністратор вибирає співробітника, після чого в наступному вікні треба вибрати дату початку відпустки і дату кінця і натиснути кнопку «створити».

9. Щоб переглянути статистику в панель-меню необхідно натиснути на кнопку «Статистика» після чого користувач перейде на форму з статистичними даними.

10. Для виходу із системи необхідно натиснути на кнопку «Вихід» в панель-меню. Після чого користувач повернеться на сторінку авторизації.

### **ВИСНОВКИ ДО ТРЕТЬОГО РОЗДІЛУ**

У даному розділі було продемонстровано інтерфейс інформаційної системи обліку працівників на підприємстві. Інтерфейс системи візуально представлено у вигляді вікон та кнопок, які надають користувачам можливість взаємодії з програмним продуктом. Для зручності користувачів, інтерфейс системи був розроблений таким чином, щоб бути простим та зрозумілим для використання. Описано також інструкцію для користування інформаційною системою користувачам. Ця інструкція містить детальну інформацію про функціональні можливості системи, послідовність дій для виконання певних операцій. Завдяки інструкції, користувачі можуть ефективно використовувати систему, а також швидко вирішувати проблеми, які виникають під час її використання.



## ВИСНОВКИ

В результаті проведеної роботи було успішно розроблено інформаційну систему обліку співробітників у вигляді веб-додатку. Ця система надає широкі можливості для ефективного обліку персоналу на підприємстві. Вона дозволяє здійснювати облік працівників, та аналізувати дані про працівників.

Загальною метою створеної інформаційної системи є спрощення та автоматизація процесів кадрового обліку на підприємстві. Це дозволяє підвищити продуктивність, зменшити помилки та забезпечити точність обліку персоналу.

Розроблена інформаційна система кадрового обліку є потужним інструментом для підтримки управлінських процесів обліку персоналу на підприємстві. Її впровадження дозволить зменшити час та зусилля, які раніше витрачалися на ручний облік та обробку даних про працівників.

Важливо зазначити, що створена система має потенціал для подальшого розвитку та розширення функціоналу. З використанням фреймворку Laravel і стандартів програмування, майбутні програмісти, що підтримуватимуть систему, зможуть легко внести необхідні зміни та додаткові функціональні можливості, що відповідають змінюючимся потребам підприємства.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Облік персоналу й кадрове діловодство Url: <https://zkg.ua/oblik-personaluj-kadrove-dilovodstvo/>.
2. Розробка програмних засобів для кадрових відділів Url: <http://elibrary.asu.ru/xmlui/bitstream/handle/asu/5055/vkr.pdf?sequence=1&isAllowed=y>.
4. Програма кадрового обліку «Чиж-Кадри». URL : <http://kadri.chizh.ua/>.
5. Програми обліку співробітників організації. URL: <https://norvegus.ru/uk/pogrammy-dlya-ucheta-sotrudnikov-organizacii-kakaya-programma-luchshe-podhodit/>.
6. Інтернет магазин програмного та апаратного забезпечення. URL: <https://www.softkey.ua/ua/catalog/business/programma-kadrovogo-ucheta-kadry-plyus/>.
7. Автоматизація бухгалтерського обліку. URL : <https://dinn.com.ua/services/1c-accounting>
8. Методологія Idef3 Url: [https://stud.com.ua/87186/ekonomika/metodologiya\\_idef3](https://stud.com.ua/87186/ekonomika/metodologiya_idef3).
9. Що таке HTML та як за допомогою нього увійти в IT Url: <https://mc.today/uk/shho-take-html-ta-yak-za-dopomogoyu-nogo-uvijti-do-it/>.
10. Bootstrap вивчення та як його використовувати Url: <https://druzy.com.ua/bootstrap-sho-ce-z-chogo-pochati-vivchennia-i-iak-vikoristovyvati/>.
11. Підручник для початківців Laravel Url: <https://uk.myservername.com/php-laravel-tutorial>.
11. Що таке база даних MySQL Url: <https://uk.education-wiki.com/2686691-what-is-mysql-database>.
12. Розробка інформаційної системи процесу управління персоналом Url: <https://www.evkoval.org/kursovye-raboty/razrabotka-informatsionnoj-sistemyi-protssesa-upravleniya-personalom#Заклучение>.
13. Про затвердження інструкції роботи з кадрами Url: <https://zakon.rada.gov.ua/rada/show/v0083299-01#Text>.

14. ВебDatabaseApplicationwith PHP and MySQL», 2nd EditionByDavidLane, Hugh E. Williams. © O'Reilly, May 2004. ISBN: 0-596-00543-1.
15. Петюх В. М. Управління персоналом: [навч.-метод. посібник для самоств. вивч. дисц. ] / В.М. Петюх. – К.: КНЕУ, 2000. – 124 с.
16. Маслак О.О. Стратегічні аспекти управління персоналом / О. Маслак // Науковий вісник НЛТУ України. – 2010. № 20/6.- С.123-129.
17. MySQL. Довідник. MySQL АВ. — М: «Вільямс», 2006 — 521 с.
18. Пількевич І. А., Молодецька К. В., Сугоняк І. І., Лобанчикова Н. М. Основи побудови автоматизованих систем управління : навч. посібник. Житомир: ЖДУ ім. І. Франка, 2014. 226 с.
19. Линенко А. В. Сутність й аналіз персоналу підприємства за показниками його ефективності / А. В. Линенко, О. Л. Шевченко // Держава та регіони. Серія : Економіка та підприємництво. — 2008. — № 4. — С. 111— 116.
20. Галузинський, Г. П., Гордієнко І. В. Сучасні технологічні засоби обробки інформації : навч. посіб. К. : КНЕУ, 1998. 224 с.
21. Пушкар, О. І., Гіковатий В. М., Євсєєв О. С., Потрашкова Л. В. Системи підтримки прийняття рішень : навч. посібник; ред. О. І. Пушкар. Харків : Інжек, 2006. 304 с
22. Дерлоу Д. Ключові управлінські рішення : Технологія прийняття рішень/ Дес Дерлоу,; Пер. з англ. Р.А.Семків, Р.Л.Ткачук. -К.: Всеуито: Наук. думка, 2001. 242 с.
- 23.Bootstrap – бібліотека готових рішень Url: <https://getbootstrap.com/docs/5.3/customize/color/>

# ДОДАТКИ

## ДОДАТОК А

## Моделювання бізнес-процесів предметної області

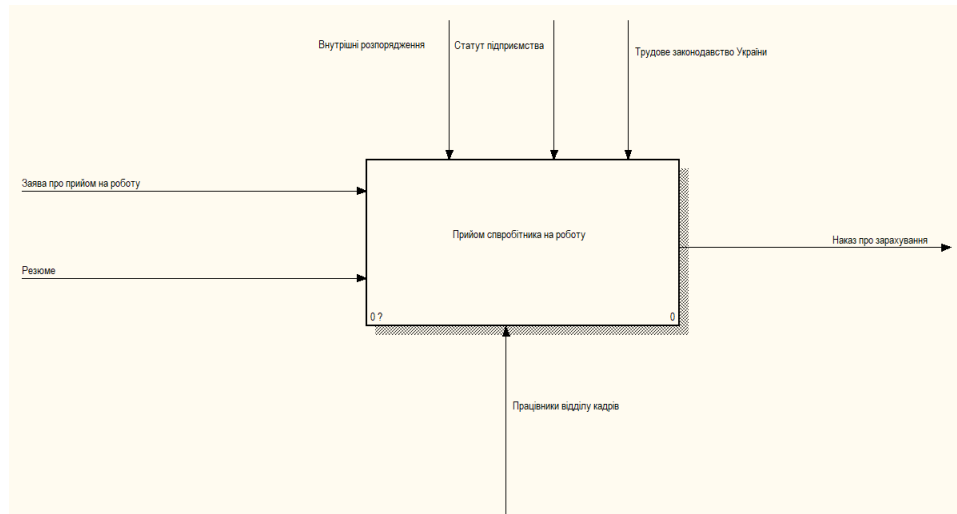


Рисунок 1 – Контекстна діаграма  
«Прийом співробітника на роботу»

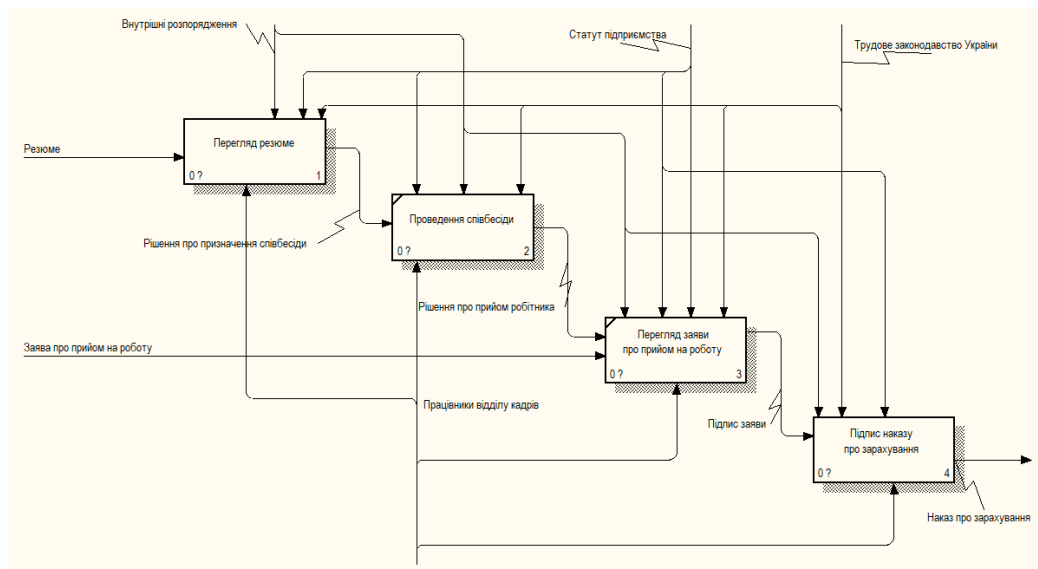


Рисунок 2 – Декомпозиція процесу  
«Прийом співробітника на роботу»

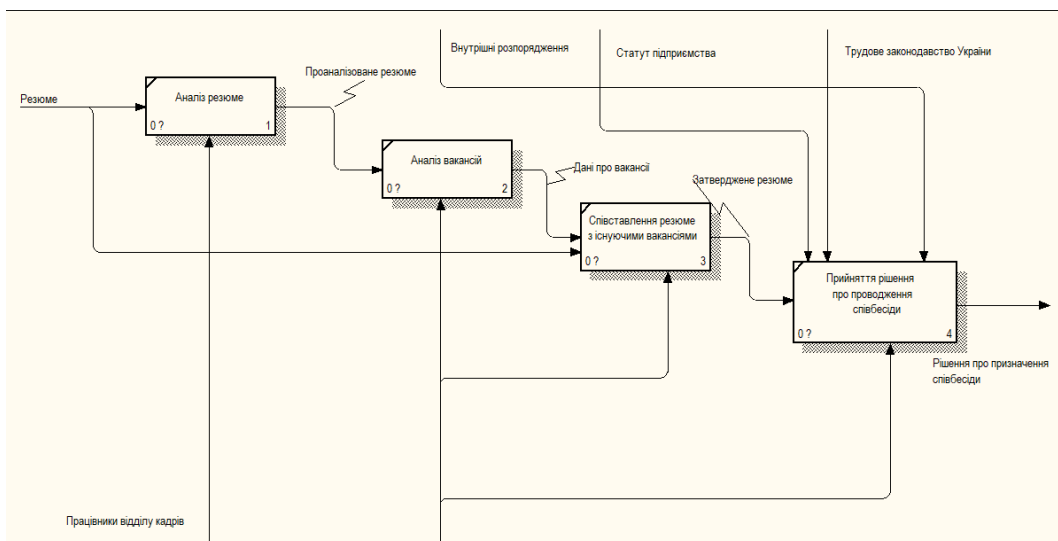


Рисунок 3 – Декомпозиція процесу «Перегляд резюме»

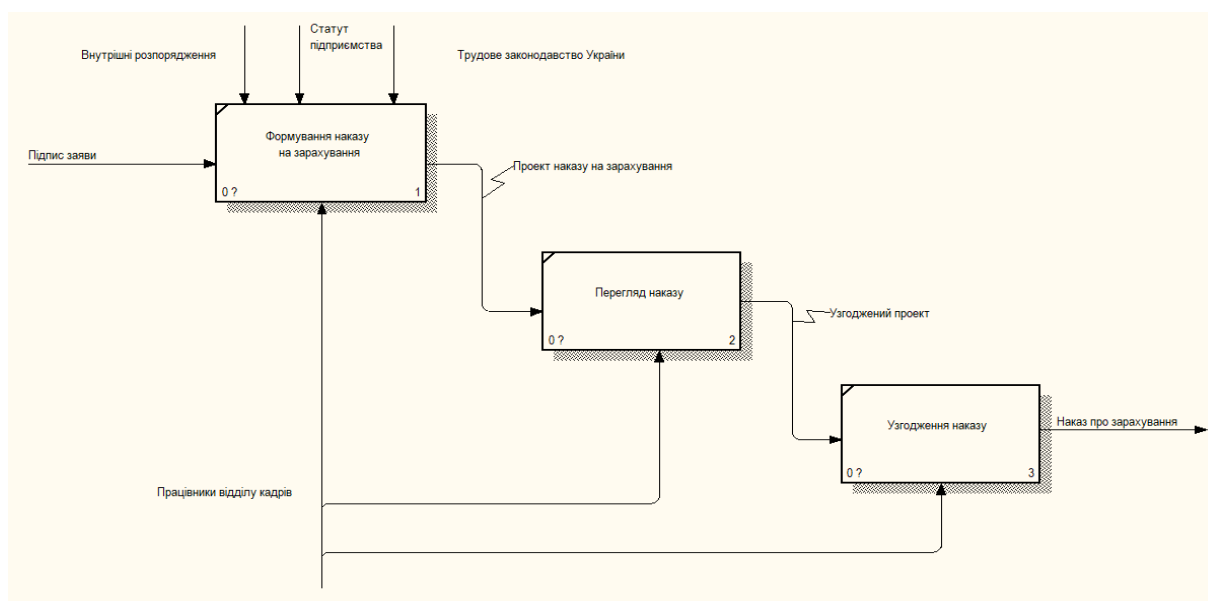


Рисунок 4 – Декомпозиція процесу «Підпис наказу про зарахування»

## ДОДАТОК Б

## Форма створення співробітника

## Створення співробітника

Особисті дані працівника		Корпоративні дані працівника	
Провідник ім'я по-батькові		Дата приєднання на роботу	
<input type="text"/>		<input type="text" value="дд.мм.ттт"/>	
Дата народження		Дата звільнення <i>*не обов'язково для заповнення*</i>	
<input type="text" value="дд.мм.ттт"/>		<input type="text" value="дд.мм.ттт"/>	
Місце проживання		Посада	
<input type="text"/>		<input type="text" value="Директор, Адміністративний"/>	
Фотографія співробітника		Тип зайнятості	
<input type="button" value="Вибрати файл"/> файл не вибран		<input type="text" value="Повна зайнятість"/>	
Паспорт: серія		Пільгова категорія	
<input type="text"/>		<input type="text" value="Немає"/>	
Паспорт: номер		Стать	
<input type="text"/>		<input type="text" value="Чоловік"/>	
Паспорт: орган що видає		Стоїть на військовому обліку	
<input type="text"/>		<input type="text" value="Так"/>	
Паспорт: дата видачі		Одружений	
<input type="text" value="дд.мм.ттт"/>		<input type="text" value="Так"/>	
Повторіть місце роботи <i>*не обов'язково для заповнення*</i>			
Місце роботи 1	Місце роботи 2	Місце роботи 3	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Посада	Посада	Посада	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Дата прийому на роботу	Дата прийому на роботу	Дата прийому на роботу	
<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	
Дата звільнення	Дата звільнення	Дата звільнення	
<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	
Причина звільнення	Причина звільнення	Причина звільнення	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Освіта <i>*не обов'язково для заповнення*</i>			
Назва університету	Назва університету	Назва університету	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Спеціальність	Спеціальність	Спеціальність	
<input type="text"/>	<input type="text"/>	<input type="text"/>	
Початок навчання	Початок навчання	Початок навчання	
<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	
Закінчення навчання	Закінчення навчання	Закінчення навчання	
<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	<input type="text" value="дд.мм.ттт"/>	
Ступінь освіти	Ступінь освіти	Ступінь освіти	
<input type="text" value="Молодий спеціаліст"/>	<input type="text" value="Молодий спеціаліст"/>	<input type="text" value="Молодий спеціаліст"/>	
Рівень освіченості	Рівень освіченості		
<input type="text" value="Повна загальна середня освіта"/>	<input type="text" value="Повна загальна середня освіта"/>		
<input type="button" value="Створити"/>			

## ДОДАТОК Д

Код контролера EmployeeController відповідає за створення редагування та виведення інформації про співробітника

```
<?php
namespace App\Http\Controllers\Employee;

use App\Models\EducationalLevel;
use App\Models\Employee;
use App\Models\EmployeeData;
use App\Models\Mail;
use App\Models\Position;
use App\Models\Subdivision;
use App\User;
use Illuminate\Http\Request;
use App\Http\Controllers\Controller;
use Illuminate\Support\Facades\Storage;
use Illuminate\Support\Facades\File;

/*
 * Контролер відповідає за створення, редагування та виведення інформації про співробітника
 */

class EmployeeController extends Controller
{
    public $position;
    public $educational_level;
    public $employeeData;
    public $employee;
    public $user;
    public $mail;
    public $subdivision;

    public function __construct(Position $position, EducationalLevel $educational_level, EmployeeData
$employeeData,
    Employee $employee, User $user, Mail $mail, Subdivision $subdivision)
```



```
{

    $this->position = $position;
    $this->educational_level = $educational_level;
    $this->employeeData = $employeeData;
    $this->employee = $employee;
    $this->user = $user;
    $this->mail = $mail;
    $this->subdivision = $subdivision;
}

public function index()
{
    $data = [
        'employees' => $this->employee->withTrashed()->get() // SELECT * FROM employees
    ];

    return view('employee.index', $data);
}

public function create()
{
    $data = [
        'positions' => $this->position->all(), //SELECT * FROM positions
        'educational_levels' => $this->educational_level->all() // SELECT * FROM educational_levels
    ];
    return view('employee.create', $data);
}

public function store(Request $request)
{
    /*
    *
    * Збереження співробітника ділиться на 2 фази
    * 1 - запис особистої інформації співробітника в таблицю employee_data і дістаємо ІД запису
    * 2 - запис загальної інформації співробітника в таблицю employees
    */
}
```

```
*

```

```
*/

```

```
$employee = new Employee();

```

```
$employeeData = new EmployeeData();

```

```
/*

```

```
*

```

```
* Обробляємо фото

```

```
*/

```

```
$dir = $this->employee->count()+1;

```

```
$photo = $request->file('photo');

```

```
$extension = $photo->getClientOriginalExtension();

```

```
Storage::disk('employees')->put($dir.'/'.$photo->getFilename().'.'.$extension, File::get($photo));

```

```
$employeeData->photo = $dir.'/'.$photo->getFilename().'.'.$extension;

```

```
$employeeData->address = $request->input('address');

```

```
$employeeData->date_of_birth = $request->input('date_of_birth');

```

```
/*

```

```
* Перша робота

```

```
* */

```

```
$employeeData->job1 = $request->input('job1');

```

```
$employeeData->job1_from = $request->input('job1_from');

```

```
$employeeData->job1_to = $request->input('job1_to');

```

```
$employeeData->job1_position = $request->input('job1_position');

```

```
$employeeData->reason1 = $request->input('reason1');

```

```
/*

```

```
* Друга робота

```

```
* */

```

```
$employeeData->job2 = $request->input('job2');

```

```
$employeeData->job2_from = $request->input('job2_from');

```

```
$employeeData->job2_to = $request->input('job2_to');

```

```
$employeeData->job2_position = $request->input('job2_position');
$employeeData->reason1 = $request->input('reason2');

/*
 * Третя робота
 * */
$employeeData->job3 = $request->input('job3');
$employeeData->job3_from = $request->input('job3_from');
$employeeData->job3_to = $request->input('job3_to');
$employeeData->job3_position = $request->input('job3_position');
$employeeData->reason1 = $request->input('reason3');

$employeeData->educational_level_id = $request->input('educational_level_id');

/*
 * Університет 1
 * */
$employeeData->university1_name = $request->input('university1_name');
$employeeData->specialty1 = $request->input('specialty1');
$employeeData->university1_from = $request->input('university1_from');
$employeeData->university1_to = $request->input('university1_to');
$employeeData->university1_degree = $request->input('university1_degree');

/*
 * Університет 2
 * */
$employeeData->university2_name = $request->input('university2_name');
$employeeData->specialty2 = $request->input('specialty2');
$employeeData->university2_from = $request->input('university2_from');
$employeeData->university2_to = $request->input('university2_to');
$employeeData->university2_degree = $request->input('university2_degree');

/*
 * Університет 3
 * */
$employeeData->university3_name = $request->input('university3_name');
```

```

$employeeData->specialty3 = $request->input('specialty3');
$employeeData->university3_from = $request->input('university3_from');
$employeeData->university3_to = $request->input('university3_to');
$employeeData->university3_degree = $request->input('university3_degree');

$employeeData->pilgova_category = $request->input('pilgova_category');
$employeeData->military = $request->input('military');
$employeeData->married = $request->input('married');
$employeeData->passport_seria = $request->input('passport_seria');
$employeeData->passport_number = $request->input('passport_number');
$employeeData->passport_organ = $request->input('passport_organ');
$employeeData->passport_date = $request->input('passport_date');

$employeeData->save(); //Зберігаємо запис в таблицю employee_data

$employeeDataId = $employeeData->id; //Витягуємо id

$employee->pib = $request->input('pib');
$employee->date_of_employment = $request->input('date_of_employment');
$employee->date_of_dismissal = $request->input('date_of_dismissal');
$employee->position_id = $request->input('position_id');
$employee->type_of_employment = $request->input('type_of_employment');
$employee->employee_data_id = $employeeDataId;

$employee->save();

$employeeSavedId = $employee->id;

$message = "Співробітника ".$employee->pib." було успішно створено. ID ".$employeeSavedId;
return redirect()->back()->withErrors(['message' => $message]);

}

public function show($id)

```

```

{
    $data = [
        'employee' => $this->employee->select()->where('id', $id)->withTrashed()->first()
    ];
    return view('employee.show', $data);
}

public function edit($id)
{
    $data =
    [
        'employee' => $this->employee->select()->where('id', $id)->withTrashed()->first(),
        'positions' => $this->position->all(),
        'educational_levels' => $this->educational_level->all()
    ];

    return view('employee.edit', $data);
}

public function update(Request $request)
{
    /*
    *
    * Збереження співробітника ділиться на 2 фази
    * 1 - запис особистої інформації співробітника в таблицю employee_data і дістаємо ІД запису
    * 2 - запис загальної інформації співробітника в таблицю employees
    *
    */

    $employee = $this->employee->where('id', $request->input('id'))->withTrashed()->first();
    $employeeData = $this->employeeData->where('id', $employee->employee_data->id)->first();

    /*
    *
    * Обробляємо фото
    */
}

```

```

if($request->file('photo') != null)
{
    $dir = $this->employee->where('id', $request->input('id'))->first()->id;
    $photo = $request->file('photo');
    $extension = $photo->getClientOriginalExtension();
    Storage::disk('employees')->put($dir.'/'.$photo->getFilename().'.'.$extension, File::get($photo));
    $employeeData->photo = $dir.'/'.$photo->getFilename().'.'.$extension;
}

/*    $request->photo->store('employees', 'public');
    $employeeData->photo = $request->photo->hashName();*/

$employeeData->address = $request->input('address');
$employeeData->date_of_birth = $request->input('date_of_birth');

/*
 * Перша робота
 * */
$employeeData->job1 = $request->input('job1');
$employeeData->job1_from = $request->input('job1_from');
$employeeData->job1_to = $request->input('job1_to');
$employeeData->job1_position = $request->input('job1_position');

/*
 * Друга робота
 * */
$employeeData->job2 = $request->input('job2');
$employeeData->job2_from = $request->input('job2_from');
$employeeData->job2_to = $request->input('job2_to');
$employeeData->job2_position = $request->input('job2_position');

/*
 * Третя робота
 * */
$employeeData->job3 = $request->input('job3');

```

```

$employeeData->job3_from = $request->input('job3_from');
$employeeData->job3_to = $request->input('job3_to');
$employeeData->job3_position = $request->input('job3_position');

$employeeData->educational_level_id = $request->input('educational_level_id');

/*
 * Університет 1
 */
$employeeData->university1_name = $request->input('university1_name');
$employeeData->specialty1 = $request->input('specialty1');
$employeeData->university1_from = $request->input('university1_from');
$employeeData->university1_to = $request->input('university1_to');
$employeeData->university1_degree = $request->input('university1_degree');

/*
 * Університет 2
 */
$employeeData->university2_name = $request->input('university2_name');
$employeeData->specialty2 = $request->input('specialty2');
$employeeData->university2_from = $request->input('university2_from');
$employeeData->university2_to = $request->input('university2_to');
$employeeData->university2_degree = $request->input('university2_degree');

/*
 * Університет 3
 */
$employeeData->university3_name = $request->input('university3_name');
$employeeData->specialty3 = $request->input('specialty3');
$employeeData->university3_from = $request->input('university3_from');
$employeeData->university3_to = $request->input('university3_to');
$employeeData->university3_degree = $request->input('university3_degree');
$employeeData->married = $request->input('married');
$employeeData->passport_number = $request->input('passport_number');
$employeeData->passport_seria = $request->input('passport_seria');
$employeeData->passport_date = $request->input('passport_date');

```

```
$employeeData->passport_organ = $request->input('passport_organ');  
$employeeData->pilgova_category = $request->input('pilgova_category');  
$employeeData->military = $request->input('military');  
$employeeData->sex = $request->input('sex');
```

```
$employeeData->save(); //Зберігаємо запис в таблицю employee_data
```

```
$employeeDataId = $employeeData->id; //Витягуємо id
```

```
$employee->pib = $request->input('pib');  
$employee->date_of_employment = $request->input('date_of_employment');  
$employee->date_of_dismissal = $request->input('date_of_dismissal');  
$employee->position_id = $request->input('position_id');  
$employee->type_of_employment = $request->input('type_of_employment');  
$employee->employee_data_id = $employeeDataId;
```



## ДОДАТОК Г

Код контролера Vacation Controller відповідає за створення відпусток для співробітників

```
<?php

namespace App\Http\Controllers\Vacation;

use App\Models\Employee;
use App\Models\Vacation;
use Illuminate\Http\Request;
use App\Http\Controllers\Controller;

class VacationController extends Controller
{
    public $vacation;
    public $employee;
    public function __construct(Vacation $vacation, Employee $employee)
    {
        $this->vacation = $vacation;
        $this->employee = $employee;
    }
    public function index()
    {
        $data = [
            'vacations' => $this->vacation->all()
        ];

        return view('vacations.index', $data);
    }
    public function create()
    {
        $data =
        [
            'employees' => $this->employee->select('id', 'pib')->get()
        ];
    }
}
```

```
    return view('vacations.create', $data);
}

public function store(Request $request)
{
    $data = $request->all();
    unset($data['_token']);
    Vacation::create($data);
    return redirect()->route('vacations.index');
}

public function delete($id)
{
    $vacation = $this->vacation->where('id', $id)->first();
    $message = "Відпустка працівника ".$vacation->employee->pib." з ".$vacation->from." по
".$vacation->to." була скасована";
    $vacation->delete();
    return redirect()->route('vacations.index')->withErrors(['message' => $message]);
}
```

## ДОДАТОК Д

## Код контролера StatisticsController(відповідає за створення статистики)

```

<?php

namespace App\Http\Controllers\Statistics;

use App\Models\EmployeeData;
use Illuminate\Http\Request;
use App\Http\Controllers\Controller;
use DateTime;
use Illuminate\Support\Facades\DB;
/*
 * Відповідає за статистику
 */
class StatisticsController extends Controller
{
    function calculateAge($birthdate)
    {
        $today = new DateTime();
        $diff = $today->diff(new DateTime($birthdate));
        return $diff->y;
    }
    public function index()
    {
        // Графік по віковій категорії
        $labels = ['18-25 років', '26-32 років', '33-40 років', '40-50 років', '50-60 років', '60-72 років'];

        $data = [];

        foreach ($labels as $label) {
            $range = explode('-', $label);
            $minAge = intval($range[0]);
            $maxAge = intval($range[1]);

            $employeesCount = EmployeeData::whereRaw("TIMESTAMPDIFF(YEAR, date_of_birth, CURDATE())

```

```
>= $minAge")
    ->whereRaw("TIMESTAMPDIFF(YEAR, date_of_birth, CURDATE()) <= $maxAge")
    ->count();
    $data[] = $employeesCount;
}
$labels2 = []; // Замініть цей масив на ваші рівні освіти
$data2 = [];
$educationLevels = DB::table('educational_levels')->pluck('name', 'id');
foreach ($educationLevels as $id => $name) {
    $employeesCount = EmployeeData::where('educational_level_id', $id)->count();
    $labels2[] = $name;
    $data2[] = $employeesCount;
}
return view('chart', compact('labels2', 'data2', 'labels', 'data'));
return view('chart', compact('labels', 'data', 'labels2', 'data2'));
}
```

## Код employee/create.blade.php(Форма для створення співробітника)

```

@extends('layouts.app')

@section('content')

<div class = "container">
  @if($errors->any())
    <div class="alert alert-success alert-dismissible fade show" role="alert">
      <h4 class="alert-heading">Чудово!</h4>
      <p>{{ $errors->first() }}</p>
      <button type="button" class="close" data-dismiss="alert" aria-label="Close">
        <span aria-hidden="true">&times;</span>
      </button>
    </div>
  @endif
  <h2>Створення співробітника</h2>
  <form method = "post" action = "{{ route('employee.store') }}" enctype="multipart/form-data">
    @csrf
    <div class = "row">
      <div class = "col-md-6">

<!--Карточка Дані про працівника-->
      <div class="card">
        <div class="card-header">
          Особисті дані працівника
        </div>
        <div class="card-body">

          <!-- pib -->
          <div class="form-group">
            <label for="exampleFormControlInput1">Прізвище ім'я по-батькові</label>
            <input type="text" class="form-control" style = "border: 1px solid;"
placeholder="Прізвище ім'я по-батькові" name = "pib">
          </div>
          <!-- pib END-->

          <!--date_of_birth-->
          <div class="form-group">
            <label for="exampleFormControlSelect1">Дата народження</label>
            <input type = "date" class = "form-control" style = "border: 1px solid;" name =
"date_of_birth">
          </div>
          <!--date_of_birth END-->

          <!--address-->
          <div class="form-group">

```

```

        <label for="exampleFormControllInput1">Місце проживання</label>
        <input type="text" class="form-control" style = "border: 1px solid;"
placeholder="Місце проживання" name = "address">
    </div>
<!--address END-->

<!--photo-->
    <div class="form-group">
        <label for="exampleFormControllInput1">Фотографія співробітника</label>
        <input type="file" class="form-control" style = "border: 1px solid;" name = "photo"
required>
    </div>
<!--photo END-->

    <div class = "form-group">
        <label for="exampleFormControllInput1">Паспорт: серія</label>
        <input type="text" class="form-control" style = "border: 1px solid;"
placeholder="Паспорт серія" name = "passport_seria" required>
    </div>

    <div class = "form-group">
        <label for="exampleFormControllInput1">Паспорт: номер</label>
        <input type="text" class="form-control" style = "border: 1px solid;"
placeholder="Паспорт номер" name = "passport_number" required>
    </div>

    <div class = "form-group">
        <label for="exampleFormControllInput1">Паспорт: орган що видав</label>
        <input type="text" class="form-control" style = "border: 1px solid;"
placeholder="Паспорт орган що видав" name = "passport_organ" required>
    </div>

    <div class = "form-group">
        <label for="exampleFormControllInput1">Паспорт: дата видачі</label>
        <input type="date" class="form-control" style = "border: 1px solid;" name =
"passport_date" required>
    </div>

</div>
</div>

<!--Карточка Дані про працівника-->
</div>

<div class = "col-md-6">
    <div class = "card">
        <div class = "card-header">
            Корпоративні дані працівника
        </div>
        <div class = "card-body">

```

```

<!--date_of_employment-->
<div class="form-group">
  <label for="exampleFormControlInput1">Дата прийняття на роботу</label>
  <input type="date" class="form-control" style = "border: 1px solid;" name =
"date_of_employment" required>
</div>
<!--date_of_employment END-->

```

```

<!--date_of_dismissal-->
<div class="form-group">
  <label>
    Дата звільнення <span style = "color:red;">*не обов'язково для
заповнення</span>
  </label>
  <input type="date" class="form-control" style = "border: 1px solid;" name =
"date_of_dismissal">
</div>
<!--date_of_dismissal END-->

```

```

<!--position_id-->
<div class="form-group">
  <label>Посада</label>
  <select class="form-control" style = "border: 1px solid;" name = "position_id">
    @foreach($positions as $position)
      <option value = "{{ $position->id }}">{{ $position->name }}, {{ $position-
>subdivision->name }}</option>
    @endforeach
  </select>
</div>
<!--position_id END-->

```

```

<!--type_of_employment-->
<div class = "form-group">
  <label>Тип зайнятості</label>
  <select class="form-control" style = "border: 1px solid;" name =
"type_of_employment">
    <option value = "Повна зайнятість">Повна зайнятість</option>
  </select>
</div>
<!--type_of_employment END-->

```

```

<div class = "form-group">
  <label>Пільгова категорія</label>
  <select class="form-control" name = "pilgova_category">
    <option value = "Немає">Немає</option>
    <option value = "Малозабезпечені сім'ї">Малозабезпечені сім'ї</option>
    <option value = "Громадяни, які постраждали внаслідок Чорнобильської
катастрофи">Громадяни, які постраждали внаслідок Чорнобильської катастрофи</option>
    <option value = "Особи, які постраждали від торгівлі людьми">Особи, які
постраждали від торгівлі людьми</option>

```

```

    <option value = "Багатодітні сім'ї">Багатодітні сім'ї</option>
    <option value = "Жінки, яким присвоєно почесне звання України "Мати-героїня"">Жінки, яким присвоєно почесне звання України "Мати-героїня"</option>
    <option value = "Учасники бойових дій">Учасники бойових дій</option>
  </select>
</div>

```

```

<div class = "form-group">
  <label>Стать</label>
  <select class="form-control" name = "sex">
    <option value = "Чоловік">Чоловік</option>
    <option value = "Жінка">Жінка</option>
  </select>
</div>

```

```

<div class = "form-group">
  <label>Стоїть на військовому обліку</label>
  <select class="form-control" name = "military">
    <option value = "Так">Так</option>
    <option value = "Ні">Ні</option>
  </select>
</div>

```

```

<div class = "form-group">
  <label>Одружений</label>
  <select class="form-control" name = "married">
    <option value = "Так">Так</option>
    <option value = "Ні">Ні</option>
  </select>
</div>

```

```

  </div>
</div>
</div>

```

```

<div>

```

```

  </div>
</div>

```

```

<div class="card" style = "margin-top: 30px;">
  <div class="card-header">
    Попередні місяці роботи <span style = "color:red;">*не обов'язково для
заповнення</span>
  </div>
  <div class="card-body">
    <div class = "row">

```



```

<div class = "col-md-4">

    <!--job1-->
    <div class="form-group">
        <label for="exampleFormControllInput1">Місце роботи 1</label>
        <input type="text" class="form-control" style = "border: 1px solid;"
placeholder="Місце роботи 1" name="job1">
    </div>
    <!--job1 END-->

    <!--job1_position-->
    <div class = "form-group">
        <label for="exampleFormControllInput1">Посада</label>
        <input type="text" class="form-control" style = "border: 1px solid;"
placeholder="Посада" name = "job1_position">
    </div>
    <!--job1_position END-->

    <!--job1_from-->
    <div class="form-group">
        <label>Дата прийому на роботу</label>
        <input type = "date" class = "form-control" style = "border: 1px solid;" name =
"job1_from">
    </div>
    <!--job1_from END-->

    <!--job1_to-->
    <div class="form-group">
        <label for="exampleFormControlSelect1">Дата звільнення</label>
        <input type = "date" class = "form-control" name = "job1_to">
    </div>
    <!--job1_to END-->

    <div class = "form-group">
        <label>Причина звільнення</label>
        <input type = "text" name = "reason1" class="form-control">
    </div>

</div>

<div class = "col-md-4">

    <!--job2-->
    <div class="form-group">
        <label for="exampleFormControllInput1">Місце роботи 2</label>
        <input type="text" class="form-control" placeholder="Місце роботи 2" name =
"job2">
    </div>
    <!--job2 END-->

```

```

<!--job2_position-->
<div class = "form-group">
  <label for="exampleFormControlInput1">Посада</label>
  <input type="text" class="form-control" placeholder="Посада" name =
"job2_position">
</div>
<!--job2_position END-->

<!--job2_from-->
<div class="form-group">
  <label for="exampleFormControlSelect1">Дата прийому на роботу</label>
  <input type = "date" class = "form-control" name = "job2_from">
</div>
<!--job2_from END-->

<!--job2_to-->
<div class="form-group">
  <label for="exampleFormControlSelect1">Дата звільнення</label>
  <input type = "date" class = "form-control" name="job2_to">
</div>
<!--job2_to END-->

<div class = "form-group">
  <label>Причина звільнення</label>
  <input type = "text" name = "reason2" class="form-control">
</div>

</div>

<div class = "col-md-4">

  <!--job3-->
  <div class="form-group">
    <label>Місце роботи 3</label>
    <input type="text" class="form-control" placeholder="Місце роботи 3" name="job3">
  </div>
  <!--job3 END-->

  <!--job3_position-->
  <div class = "form-group">
    <label for="exampleFormControlInput1">Посада</label>
    <input type="text" class="form-control" placeholder="Посада" name =
"job3_position">
  </div>
  <!--job3_position END-->

```

```

<!--job3_from-->
<div class="form-group">
  <label for="exampleFormControlSelect1">Дата прийому на роботу</label>
  <input type = "date" class = "form-control" name = "job3_from">
</div>
<!--job3_from END-->

```

```

<!--job3_to-->
<div class="form-group">
  <label for="exampleFormControlSelect1">Дата звільнення</label>
  <input type = "date" class = "form-control" name = "job3_to">
</div>
<!--job3_to END-->

```

```

<div class = "form-group">
  <label>Причина звільнення</label>
  <input type = "text" name = "reason3" class="form-control">
</div>
</div>

```

```

</div>

```

```

</div>

```

```

</div>

```

```

<div class = "card" style = "margin-top:30px;">
  <div class = "card-header">
    Освіта <span style = "color:red;">*не обов'язково для заповнення</span>
  </div>
  <div class = "card-body">
    <div class = "row">
      <!--university1-->
      <div class = "col-md-4">

```

```

      <!--university1_name-->
      <div class = "form-group">
        <label>Назва університету</label>
        <input class = "form-control" type="text" name = "university1_name">
      </div>
      <!--university1_name END-->

```

```

      <!--specialty1-->
      <div class = "form-group">
        <label>Спеціальність</label>
        <input class = "form-control" type = "text" name = "specialty1">
      </div>

```

```
<!--specialty1 END-->

<!--university1_from-->
<div class = "form-group">
  <label>Початок навчання</label>
  <input class = "form-control" type = "date" name = "university1_from">
</div>
<!--university1_from END-->

<!--university1_to-->
<div class = "form-group">
  <label>Закінчення навчання</label>
  <input class = "form-control" type = "date" name = "university1_to">
</div>
<!--university1_to END-->

<!--university1_degree-->
<div class = "form-group">
  <label>Ступінь освіти</label>
  <select class="form-control" name = "university1_degree">
    <option value = "Молодший спеціаліст">Молодший спеціаліст</option>
    <option value = "Бакалавр">Бакалавр</option>
    <option value = "Магістр">Магістр</option>
    <option value = "Доктор наук">Доктор наук</option>
  </select>
</div>
<!--university1_degree END-->

</div>
<!--university1-->

<!--university2-->
<div class = "col-md-4">

  <!--university2_name-->
  <div class = "form-group">
    <label>Назва університету</label>
    <input class = "form-control" type="text" name = "university2_name">
  </div>
  <!--university2_name END-->

  <!--specialty2-->
  <div class = "form-group">
    <label>Спеціальність</label>
    <input class = "form-control" type = "text" name = "specialty2">
  </div>
  <!--specialty2 END-->
```

```

<!--university2_from-->
<div class = "form-group">
  <label>Початок навчання</label>
  <input class = "form-control" type = "date" name = "university2_from">
</div>
<!--university2_from END-->

<!--university2_to-->
<div class = "form-group">
  <label>Закінчення навчання</label>
  <input class = "form-control" type = "date" name = "university2_to">
</div>
<!--university2_to END-->

<!--university2_degree-->
<div class = "form-group">
  <label>Ступінь освіти</label>
  <select class="form-control" name = "university2_degree">
    <option value = "Молодший спеціаліст">Молодший спеціаліст</option>
    <option value = "Бакалавр">Бакалавр</option>
    <option value = "Магістр">Магістр</option>
    <option value = "Доктор наук">Доктор наук</option>
  </select>
</div>
<!--university2_degree END-->

</div>
<!--university2 END-->

<!--university3-->
<div class = "col-md-4">

  <!--university3_name-->
  <div class = "form-group">
    <label>Назва університету</label>
    <input class = "form-control" type="text" name = "university3_name">
  </div>
  <!--university3_name END-->

  <!--specialty3-->
  <div class = "form-group">
    <label>Спеціальність</label>
    <input class = "form-control" type = "text" name = "specialty3">
  </div>
  <!--specialty3 END-->

  <!--university3_from-->
  <div class = "form-group">

```

```

    <label>Початок навчання</label>
    <input class = "form-control" type = "date" name = "university3_from">
</div>
<!--university3_from END-->

```

```

<!--university3_to-->
<div class = "form-group">
    <label>Закінчення навчання</label>
    <input class = "form-control" type = "date" name = "university3_to">
</div>
<!--university3_to END-->

```

```

<!--university3_degree-->
<div class = "form-group">
    <label>Ступінь освіти</label>
    <select class="form-control" name = "university3_degree">
        <option value = "Молодший спеціаліст">Молодший спеціаліст</option>
        <option value = "Бакалавр">Бакалавр</option>
        <option value = "Магістр">Магістр</option>
        <option value = "Доктор наук">Доктор наук</option>
    </select>
</div>
<!--university3_degree END-->

```

```

</div>
<!--university3 END-->
</div>

```

```

<!--educational_level-->
<div class = "form-group">
    <label>Рівень освіченості</label>
    <select class = "form-control" name = "educational_level_id">
        @foreach($educational_levels as $educational_level)
            <option value = "{{ $educational_level->id }}">{{ $educational_level->name}}</option>
        @endforeach
    </select>
</div>
<!--educational_level END-->

```

```

</div>

```

```

<!--submit button-->
<div class = "row">
    <div class = "col-md-4">
        <div class = "form-group">
            <button class="btn btn-primary" type = "submit">Створити</button>
        </div>
    </div>
</div>
<!--submit button END-->

```

```
</div>
```

```
    </form>
```

```
</div>
```

```
<style>
```

```
  input.form-control{  
    border: 1px solid;  
  }
```

```
  select.form-control{  
    border: 1px solid;  
  }
```

```
</style>
```

```
@endsection
```