

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ

к.т.н, доц. Михелєв І.Л.

" ____ " _____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "магістр"

на тему:

Дослідження архітектури корпоративних застосувань та розробка інформаційної
системи позашкільного закладу освіти.

Виконав: студент групи 6144м

(підпис) *Беркунський О.Є.*

Керівник роботи:

(посада, науковий ступінь вчене звання)

К.Т.Н, ДОЦ
Михелєв І.Л.
(підпис)

м. Миколаїв – 2023 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами

Кафедра інформаційних управляючих систем та технологій

Спеціальність 126 "Інформаційні системи та технології"

Освітня програма "Інформаційні системи та технології"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми

к.т.н, доц. Чернова Лб.С.

" ____ " ____ 2023 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ на здобуття ступеня вищої освіти "магістр"

Студенту Беркунському Олександровичу Євгеновичу

(прізвище, ім'я, по батькові)

1. Тема роботи: Дослідження архітектури корпоративних застосувань та розробка інформаційної системи позашкільного закладу освіти.

Керівник роботи: Михелєв Ігор Леонідович

Затверджені наказом ректора № 1098-уч. від "18" травня 2023 року.

2. Термін подання роботи: 15 грудня 2023 р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем, матеріали наукового стажування.

4. Перелік питань, що належать до розробки (найменування розділів): аналіз предметної області і постановка задачі, розробка проекту інформаційної системи, реалізація проекту інформаційної системи, розділ з охорони праці, висновки.

5. Перелік презентаційних матеріалів _____

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1			
2			
3			
4			

7. Дата видачі завдання 05 вересня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Термін виконання етапів проекту	Примітка
1	Вивчення і опис предметної області		
2	Розробка концепції системи		
3	Розділ з розробки проектних рішень по системі (та її частинах)		
4	Реалізація проекту та розробка робочої документації		
5	Вирішення питань охорони праці		
6	Виконання графічних документів		
7	Оформлення пояснювальної записки		
8	Подання дипломної роботи (ДР) на кафедральний захист		
9	Подання ДР рецензенту		
10	Подання ДР на державний захист		

Студент

(підпис)

Беркунський Олександр Євгенович

(ПІБ)

Керівник роботи

(підпис)

Михелєв Ігор Леонідович

(ПІБ)

АНОТАЦІЯ

Дана дипломна робота має спрямованість на дослідження архітектури корпоративних застосувань та набуття навичок у розробці проекту інформаційної системи, на прикладі розробки інформаційної системи обліку учнів позашкільних навчальних закладів.

Робота виконана на ____ сторінках, містить __ розділи, __ рисунків, ____ додатків і список використаних джерел з ____ найменувань.

ABSTRACT

This diploma thesis is focused on researching the architecture of corporate applications and acquiring skills in the development of an information system project, using the example of designing an information system for tracking students' records in out-of-school educational institutions.

The work is presented on ____ pages, contains __ sections, __ pictures, __ applications and ____ references to the sources.

ЗМІСТ

ВСТУП	
РОЗДІЛ 1. ФОРМУВАННЯ ВИМОГ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ УЧНІВ ПОЗАШКІЛЬНИХ НАВЧАЛЬНИХ ЗАКЛАДІВ, РОЗРОБКА ЇЇ КОНЦЕПЦІЇ І ПОСТАНОВКА ЗАДАЧІ	
1.1 Опис предметної області і аналіз інформаційних процесів позашкільного навчального закладу	
1.2 Формування вимог до інформаційної системи	
РОЗДІЛ 2. РОЗРОБКА ТЕХНІЧНОГО ПРОЕКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ УЧНІВ ПОЗАШКІЛЬНИХ НАВЧАЛЬНИХ ЗАКЛАДІВ	
2.1 Загальносистемні рішення	
2.2 Рішення з інформаційного забезпечення	
2.3 Рішення з програмного забезпечення	
2.4 Рішення з технічного забезпечення	
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ УЧНІВ ПОЗАШКІЛЬНИХ НАВЧАЛЬНИХ ЗАКЛАДІВ	
РОЗДІЛ 4. ОХОРОНА ПРАЦІ ТА ТЕХНІКА БЕЗПЕКИ	
ВИСНОВОК	
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	
ДОДАТОК А. ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ	
ДОДАТОК Б. ПРОГРАМА І МЕТОДИКА ВИПРОБУВАНЬ	
ДОДАТОК В. ПРИКЛАДИ ЕКРАННИХ ФОРМ	
ДОДАТОК Г. КОД ПРОГРАМИ ТА КОД СТВОРЕННЯ ТАБЛИЦЬ БАЗИ ДАНИХ.....	

ВСТУП

Темою дипломної роботи є «Розробка інформаційної системи обліку учнів позашкільних навчальних закладів». Розробка зазначеної інформаційної системи є актуальною тому, що використання її позашкільними навчальними закладами зможе полегшити облік учнів, підвищити продуктивність вчителів.

Метою роботи є розробка інформаційної системи обліку учнів позашкільних навчальних закладів, яка забезпечить більш комфортні умови праці, зручну структурування груп та підвищить продуктивність вчителів. Об'єкт дослідження – робота позашкільного навчального закладу, на прикладі Миколаївського Палацу творчості учнів. Предметом дослідження є інформаційні процеси, що виконуються співробітниками навчального закладу.

У наслідок проведеного аналізу поточної роботи вчителя позашкільного навчального закладу, було сформовано вимоги до проекту інформаційної системи. Використовуючи сформовані вимоги розроблено концепцію, що дозволяє вирішити такі задачі, як: реєстрація учнів, ведення їх обліку, розподілення учнів між групами, отримання вибірок для створення звітів.

На основі концепції розробляється технічний проект, в якому приймаються та розробляються загальносистемні рішення, рішення з інформаційного, організаційного, технічного та програмного забезпечення.

Використовуючи розроблену концепцію, для реалізації інформаційної системи обрано сучасні програмні та технічні засоби. Для системи розробляється інструкція користувача, інструкція по формуванню бази даних, інструкція з початку роботи з інформаційною системою. Проведено аналіз небезпечних та шкідливих факторів в позашкільному навчальному закладі, розроблено техніку безпеки, заходи по зменшенню впливу небезпечних та шкідливих факторів на робочому місці вчителя закладу.

РОЗДІЛ 1 ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ РОЗРОБКИ КОРПОРАТИВНИХ ІНФОРМАЦІЙНИХ СИСТЕМ ТА ПОСТАНОВКА ЗАДАЧІ РОЗРОБКИ ІС ПЗО

1.1 Опис предметної області і аналіз інформаційних процесів позашкільного навчального закладу

Позашкільний навчальний заклад (заклад позашкільної освіти) - складова системи позашкільної освіти, яка надає знання, формуючи вміння та навички за інтересами, забезпечує потреби особистості у творчій самореалізації та інтелектуальний, духовний і фізичний розвиток, підготовку до активної професійної та громадської діяльності, створює умови для соціального захисту та організації змістовного дозвілля відповідно до здібностей, обдарувань та стану здоров'я вихованців, учнів і слухачів [1].

Позашкільні навчальні заклади займаються організацією навчання дітей та підлітків згідно їх інтересів та спрямованості у позашкільний час. Учні поділяються на окремі групи згідно з їх віком, класом, навичкам та ін.

Приблизна організаційна структура окремого закладу позашкільної освіти:

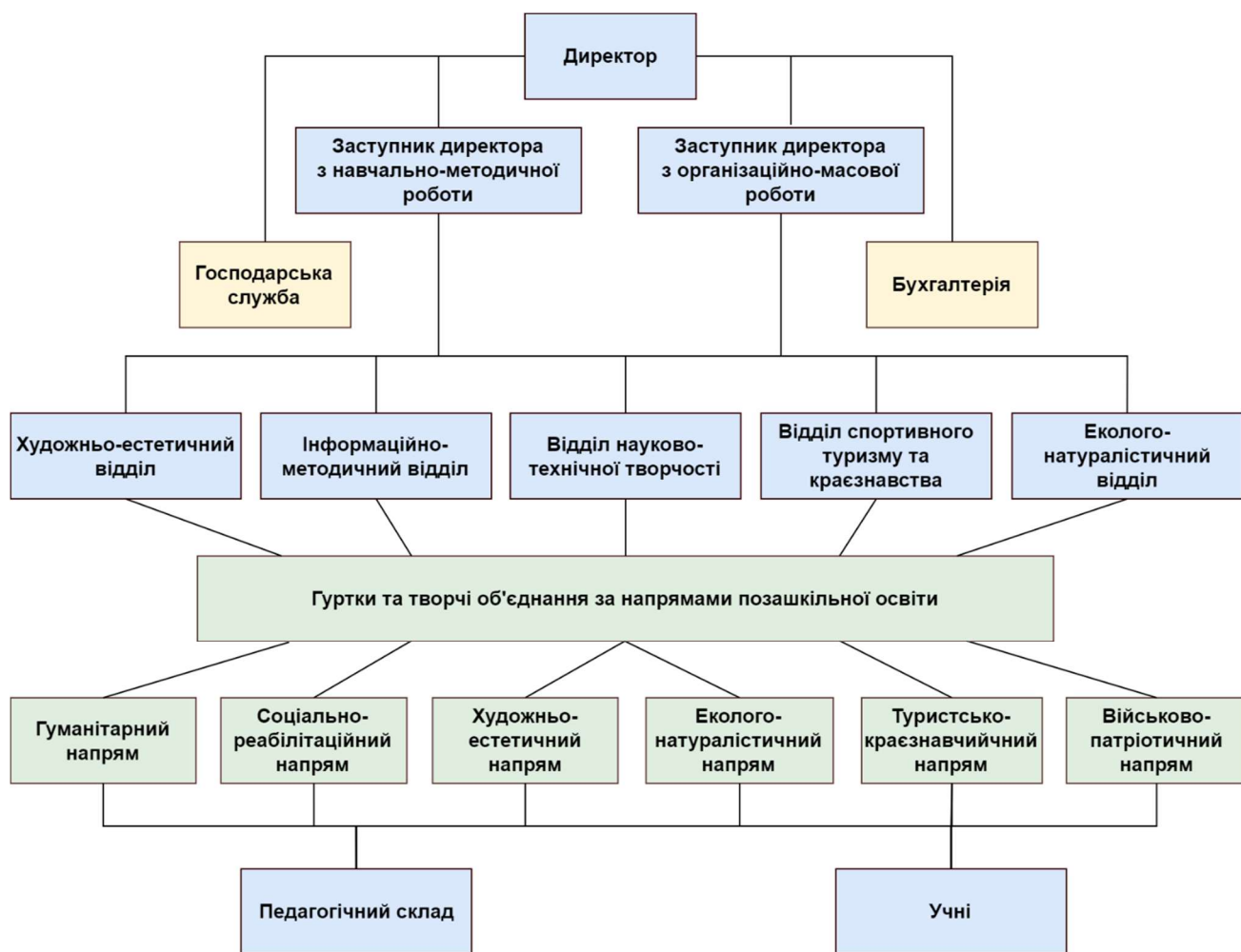


Рисунок 1.1 – Організаційна структура позашкільного навчального закладу

Директор керує закладом позашкільної освіти, стежить за виконанням обов'язків іншими працівниками закладу, виконує представницьку функцію та посередницьку функцію між Міністерством Освіти та педагогами.

Господарська служба виконує роботи з забезпечення умов праці, поточного ремонту та обліку наявної техніки.

Бухгалтерія виконує бухгалтерський облік, виплату заробітної плати та займається плануванням бюджету навчального закладу.

Заступники директора виконують комунікацію між директором та педагогічним складом, стежать за виконанням обов'язків.

Гуртки та творчі об'єднання за напрямками позашкільної освіти існують відповідно до спрямування навчального закладу, створюють відповідні

навчальні програми, організовують навчальний процес, повідомляють про набір до гуртків, беруть участь в конкурсах, мають педагогічний склад.

Проведений аналіз роботи бухгалтерії та господарської служби підтвердив наявність простих та якісних налагоджених рішень, що дозволяють ефективно налагодити фінансову та господарську діяльність в позашкільному навчальному закладі.

За результатом аналізу роботи гуртків та творчих об'єднань було виявлено, що набір учнів проводиться педагогами використовуючи застарілі системи та в паперовому вигляді, вибірки для звітів виконуються в ручному вигляді, не використовуючи будь-які засоби автоматизації. Це підтверджує зниження ефективності педагогів, зменшення часу на підготовку навчального процесу, а відповідно і зменшення якості позашкільної освіти. Є необхідність вирішення проблеми продуктивності та ефективності роботи педагога.

Можливі проблеми при здійсненні роботи педагога:

- 1) Неправильне або некоректне заповнення заявок до гуртка батьками учнів;
- 2) Зберігання некоректної або неповної інформації про групи;
- 3) Несвоєчасна перевірка заявок до гуртка чи творчого об'єднання;
- 4) Некоректні вибірки даних для звітів.

Шляхами вирішення зазначених проблем можуть бути:

- 1) Використання форм з автоматичною валідацією даних;
- 2) Автоматичне сповіщення вчителя про отримання нових заявок до групи;
- 3) Генерація вчителем відповідних вибірок учнів для звітів.

За результатами проведеного аналізу сформовано вимоги до інформаційної системи.

1.2 Формування вимог до інформаційної системи

На основі проблем, з якими стикаються у навчальному закладі, сформовано такі вимоги за функціоналом:

- 1) Структуроване зберігання даних;
- 2) Приймання даних відбувається після їх перевірки;
- 3) Автоматичне сповіщення вчителя про створені заяви;
- 4) Автоматичне сповіщення учнів про результат розгляду заяви;
- 5) Шаблонне виведення даних для звітів.

Для виконання функціональних вимог необхідні наступні програмні засоби:

- 1) Операційна системи;
- 2) Мови програмування для розробки веб-додатку;
- 3) Реляційна СУБД;
- 4) Веб-браузер для користування інформаційною системою.

Для розробки необхідні такі програмні засоби:

- 1) IDE – інтегроване середовище розробки;
- 2) Система контролю версій;
- 3) Фреймворк генерування документації програмного коду;
- 4) Графічний редактор для створення діаграм.

1.3 Дослідження технологій та засобів розробки корпоративних інформаційних систем

На основі аналізу існуючих програмних компонентів, що планується інтегрувати в корпоративну інформаційну систему, було визначено, що мова програмування Java, де-факто є найпоширенішим вибором для створення

корпоративних інформаційних систем. Зокрема, слід відзначити багату історію існування цієї мови, велику кількість існуючих бібліотек та фреймворків, велику спільноту. Але зворотним боком такої популярності є те, що у екосистемі мови Java було створено декілька різних підходів до створення саме корпоративних (великих, комплексних) застосувань.

Як вже зазначалося, у світі розробки корпоративних застосувань на платформі Java існує багато засобів та технологій. Ось кілька ключових компонентів:

- **JakartaEE (раніше Java EE – Java Enterprise Edition)** – набір специфікацій, які дозволяють створювати корпоративні застосування. Java EE містить різноманітні технології, такі як сервлети, EJB (Enterprise JavaBeans), JPA (Java Persistence API), JMS (Java Message Service) і багато інших.
- **Spring Framework (Spring)** – один з найбільш популярних фреймворків для розробки Java-додатків. Він надає широкий спектр інструментів і модулів для створення різноманітних корпоративних застосувань, включаючи управління об'єктами, інтеграцію з базами даних, аспекти програмування тощо.
- **Hibernate** – фреймворк для роботи з базами даних в Java. Hibernate дозволяє зручно працювати з об'єктно-реляційним відображенням (ORM) і спрощує роботу з реляційними базами даних.
- **Apache Maven** – інструмент для управління залежностями, будівництвом та конфігурацією проектів на Java. Maven спрощує процес розробки, допомагаючи в організації проекту та його бібліотек.
- **JUnit** – фреймворк для тестування коду на Java, він дозволяє створювати тести для перевірки правильності роботи коду, що сприяє підтримці та розвитку проекту.

- **Jenkins** – інструмент для продуктивності, який використовується для автоматизації процесу збирання, тестування та розгортання програмного забезпечення.

Перелічені інструменти та технології є лише частиною багат шарового стеку розробки корпоративних застосунків на платформі Java. Комбінування цих інструментів дозволяє розробникам створювати потужні, масштабовані та надійні програмні продукти для підприємств. Розглянемо їхні переваги та недоліки більш докладно. Слід відмітити також, що принципово вибір розробника поділяється між першими двома: Jakarta EE та Spring. Технології Hibernate (реалізація JPA), Maven (універсальний механізм збирання Java проєктів), JUnit (фреймворк для розробки та супроводження модульних тестів) та Jenkins (сервер безперервних збирань – Continuous Integration) мають можливості інтеграції з будь-яким з них.

Таким чином, для визначення того, яку платформу слід обрати, проведемо аналіз особливостей двох платформ: Jakarta EE та Spring. Визначимо їхню схожість та відзначимо розбіжності.

1.4 Порівняльний аналіз платформ Jakarta EE та Spring Архітектура

Jakarta EE – стандарт платформи для побудови корпоративних застосунків, який включає у себе набір специфікацій для створення різноманітних компонентів, таких як Servlets, JSP, EJB (Enterprise JavaBeans), JPA (Jakarta Persistence API), JMS (Jakarta Message Service) та інших. Jakarta EE забезпечує комплексний підхід до розробки, але імплементація може бути менш гнучкою. Завдяки таким особливостям, Jakarta EE (раніше Java EE) лежить в основі таких відомих рішень, як Privat24, UniCredit, Deutsche Bank, інформаційних систем компаній Lufthansa, AIRBUS, сайту міжнародної студентської олімпіади ICPC та інших.

Spring – фреймворк, який пропонує гнучкий підхід до розробки, заснований на принципах інверсії управління (IoC) та введення залежностей (DI). Spring надає модульну структуру, що дозволяє вибирати та використовувати тільки ті частини фреймворку, які потрібні для конкретного проекту. На теперішній час Spring є потужним і популярним фреймворком для розробки програмного забезпечення мовою Java, від простих консольних програм до складних корпоративних застосунків. Spring використовується компаніями різного рівня, від стартапів до відомих компаній, таких як AliExpress, Netflix, Spotify та Amazon

Проаналізуємо схожість та відмінності архітектур. Структура найпростішого Jakarta EE застосування показана на рисунку 1.2.

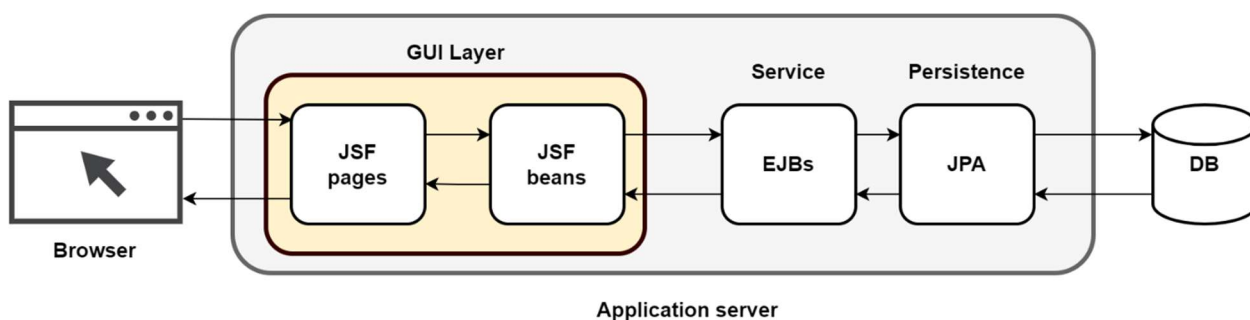


Рисунок 1.2 – Структура застосування на платформі Jakarta EE

Легко побачити, що структура достатньо зрозуміла. Запит обробляється JSF сторінкою (використовується технологія Jakarta Server Faces), а потім біном – спеціальним об’єктом, що призначений для обслуговування саме цієї сторінки. Отримує запити та перенаправляє їх до сервісів застосування, а отримавши від них результати, передає їх у зручній формі на сторінку. Управління далі йде на бізнес-логіку в EJB (із використанням технології Enterprise Java Beans), яка працює з базою даних через JPA (специфікація роботи з шаром постійності – Jakarta Persistence API). Ці зв’язки є простими і зрозумілими. Крім того, технологія Jakarta гарантує, що якщо навантаження на це застосування збільшиться на кілька порядків, його схема не зміниться взагалі ніяк. Просто застосування треба буде встановити на великий кластер з

кількох потужних серверів – вузлів (Nodes), і все застосування працюватиме в кластерному оточенні без жодних правок. Якщо потужності все одно не вистачатиме, то достатньо додати додаткові сервери в у кластер.

Саме через це рішення на стеку Jakarta EE рекомендовано використовувати в тих випадках, коли питання масштабування для застосування найкритичніше.

Ми розглянули базову структуру простої програми Jakarta EE. Тепер звернемо увагу на потужніші можливості цієї платформи.

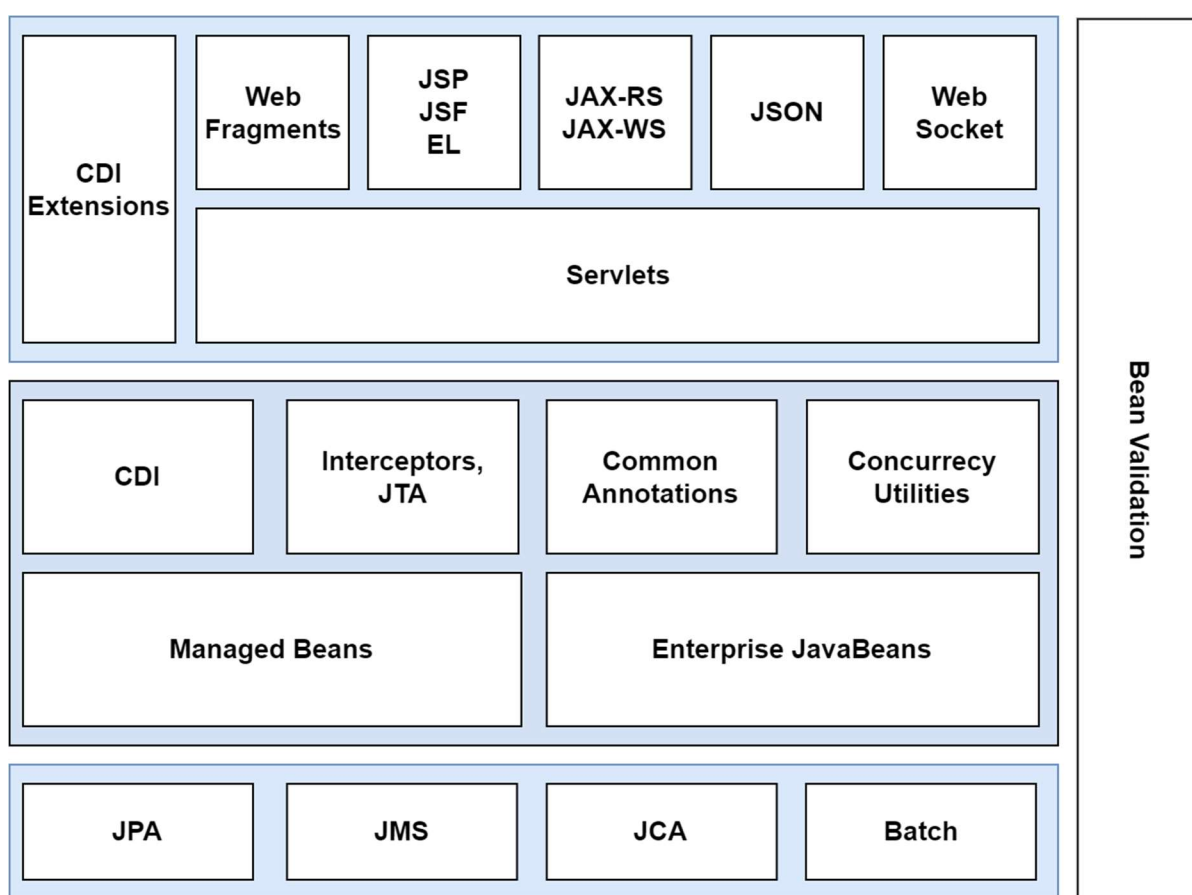


Рисунок 1.3 – Компоненти платформи Jakarta EE

З точки зору документації Jakarta EE є чимось, з чим можна оперувати без особливих складнощів – для цього не треба вивчати велику кількість джерел інформації. Разом з тим весь необхідний функціонал доступний і вже вбудований в будь-якому сервері застосувань (Enterprise Application Server). Таких серверів існує велика кількість: GlassFish, Payara, JBoss, WildFly, TomEE

Важливо, що вся ця функціональність без проблем працює в сукупності та в розподіленому (кластерному) середовищі.

Просте застосування із використанням стеку Spring не дуже сильно відрізняється від Jakarta EE за своєю архітектурою.



Рисунок 1.4 – Архітектура простого застосування Spring

При детальному співставленні можна зрозуміти, що цей рисунок відрізняється від архітектури з рисунку 1.2 лише більшою деталізацією. Так що середня програма на платформі Spring практично складається з таких саме модулів, які зв'язані подібним чином, як і у випадку платформи Jakarta EE.

Єдина, але дуже суттєва відмінність, про яку треба завжди пам'ятати та брати до уваги при виборі платформи, це те, що застосування на платформі Jakarta EE може працювати в загальному випадку тільки в рамках корпоративного сервера застосувань (Enterprise Application Server), при цьому варто згадати, що найпоширеніший сервер для Java застосувань – Apache Tomcat ним не є, а застосування на платформі Spring може працювати на будь-чому. Наприклад, на тому ж Tomcat, або на GlassFish / Payara і навіть взагалі без сервера (оскільки запустить його в собі самостійно). Така особливість робить платформу Spring Framework ідеальною для реалізації елементів

мікросервісної архітектури, але за рахунок відмови від підтримки всіх можливостей корпоративних серверів застосувань. У загальному випадку кластерна програма – це проєкт, який хоч і можливо, але дуже складно збудувати із використанням Spring. Тому питання масштабування для Spring застосувань повинні вирішуватися окремо. Найчастіше – із значно більшими витратами.

Розглянемо весь стек бібліотек Spring Framework. На жаль, як вже було зазначено, зараз бібліотек надто багато, тому всі їх виносити на одне зображення було б дуже складно. На сайтах, що містять документацію по Spring, зокрема на офіційному сайті [2] досі демонструється зображення приблизно п'ятирічної давнини:

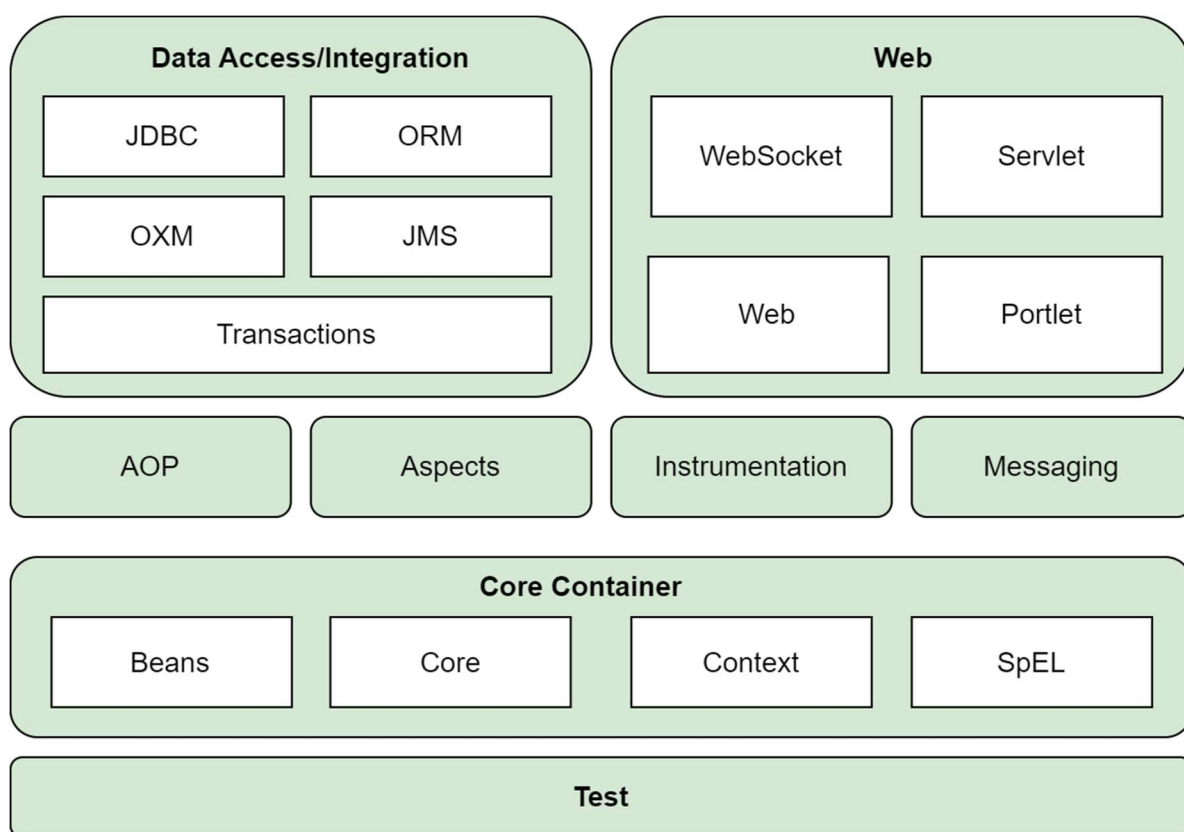


Рисунок 1.5 – Архітектура та складові Spring Framework

При найближчому детальному розгляді справді можна помітити дивовижну схожість між представленими зображеннями. Справа в тому, що у розробників та архітекторів обох платформ, що існують на сьогоднішній день, практично

ідентичне уявлення про те, як слід розробляти великі корпоративні застосування.

Обидва стеки дотримуються ідеї, що об'єкти, які являють собою дані предметної області, повинні бути лише значеннями (Value Object), у той час як бізнес-логіка повинна бути винесена окремо від них. Для таких задач відповідно застосовують сервіси Spring та EJB Jakarta EE. І це, як не дивно, суперечить основам об'єктно-орієнтованого програмування, коли об'єкти мають містити дані, що характеризують їхній стан та описувати поведінку - методи. Це скоріше, процедурний код із його власними недоліками. Такий підхід також обмежує використання різних шаблонів проєктування, що здається дивним з огляду на прогрес у галузі інформатики за останні роки. Проте, слід також відзначити, що декілька загальновідомих шаблонів проєктування також можуть бути реалізовані на цих платформах із використанням принципу "Введення залежностей" (Dependency Injection) або "Інверсія управління" (Inversion of Control).

Введення залежностей та управління компонентами

Jakarta EE – стандарт включає в себе IoC (Inversion of Control – інверсія управління) та DI (Dependency Injection – введення залежностей), але імплементація їх може бути досить складною та вимагати більше конфігурації. Spring - однією з ключових особливостей Spring є спрощене введення залежностей та управління компонентами. Завдяки анотаціям та конфігураційним файлам, Spring дозволяє легко управляти компонентами та їхніми залежностями. Як відомо, такий підхід дозволяє реалізувати в застосуванні поширені паттерни проєктування[3]: "Стратегія", "Стан", "Шаблонний метод", "Абстрактна фабрика", "Фабричний метод", "Фасад", "Фронт-контроллер", "Шукач сервісів" та ін.

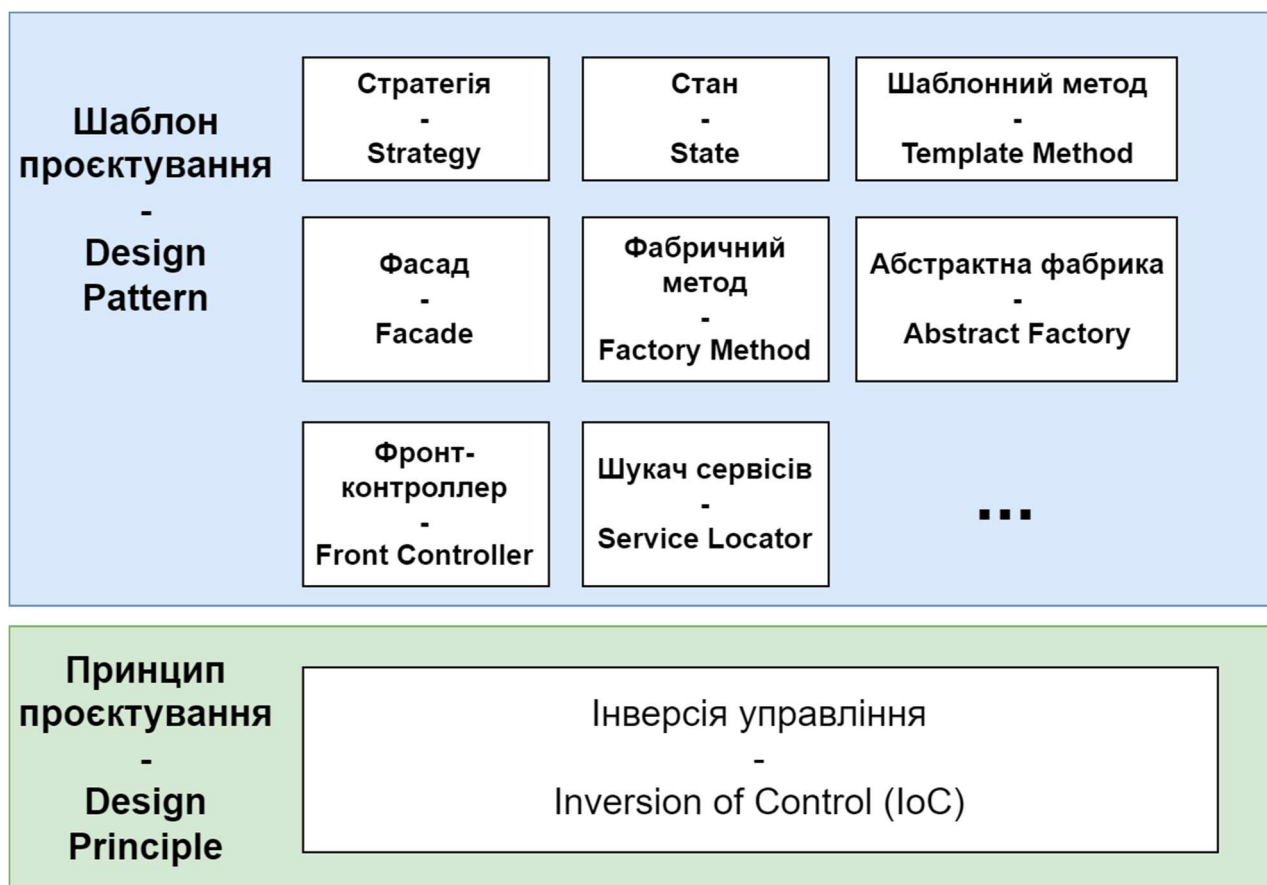


Рисунок 1.6 – Застосування шаблонів та принципів проектування

Таким чином, слід зазначити, що підтримка введення залежностей та інверсії управління є важливими факторами при виборі платформи. У Spring Framework за це відповідає модуль ядра DI (Dependency Injection) а у Jakarta – CDI (Context Dependency Injection). Навіть із назви зрозуміло, що ядра в обох платформах абсолютно ідентичні за характеристиками. Так воно і є. Фактично в платформі Jakarta EE спочатку і використовувався DI, який був запозичений зі Spring. Але після того, як його включили прямо в Core, використовувати його окремо стало неможливо. Тому розробники Jakarta створили свій DI з анотаціями. Багато експертів вважають, що реалізація DI в Jakarta не гірша за аналогічну у Spring, а можливо, навіть трохи краще. Але відмінності настільки незначні, що їх можна знехтувати.

Spring Beans та EJB (Enterprise Java Beans)

Spring Beans та EJB – це дві різні моделі компонентів, які використовуються для створення бізнес-логіки в Java-застосуваннях.

Spring Beans – це прості Java-об'єкти, які управляються контейнером Spring. Вони можуть бути легко інтегровані та використані в інших компонентах застосування, а також взаємодіяти з іншими бінами через механізм введення залежностей.

EJB, з іншого боку, є потужними компонентами Jakarta EE, які надають розширену підтримку для розподіленого виконання застосувань. Вони включають в себе різноманітні можливості, такі як розподілений збирач сміття (Garbage Collector), автентифікацію, управління транзакціями та інші аспекти, що сприяють побудові розподілених систем.

Отже, хоча Spring Beans пропонують простий та гнучкий спосіб управління об'єктами та їх залежностями в межах застосування, EJB володіють більш розширеними функціональними можливостями, спрямованими на підтримку розподіленої архітектури застосувань. Обираючи між ними, важливо враховувати специфіку проекту та його вимоги до розподіленості, безпеки та масштабованості.

Spring MVC та Java Server Faces (JSF)

Для створення інтерфейсів користувача, хоча вони й мають багато спільного, кожна з платформ має власні особливості та підходи.

Spring MVC, як правило, використовується з різноманітними шаблонними фреймворками (Template Engine), такими як Thymeleaf, Mustache або фронтенд-фреймворки типу React чи Angular написані мовою JavaScript. Це дозволяє розробникам створювати шаблони, які можна легко інтегрувати в клієнтську частину застосування та взаємодіяти з ними.

JSF – це більш інтегроване рішення для побудови користувацького інтерфейсу, що надає засоби для розробки GUI-додатків та підтримку AJAX прямо з коробки. Крім того, для реалізації SPA-додатків використовуються додаткові бібліотеки, такі як PrimeFaces або IceFaces, які дозволяють створювати красиві та зручні інтерфейси.

Обираючи між Spring MVC та JSF, важливо враховувати конкретні вимоги проекту, особливості розробки та команду розробників. Spring MVC забезпечує гнучкість у виборі технологій для шаблонів, тоді як JSF надає більш інтегрований підхід для побудови GUI та має вбудовану підтримку AJAX.

Spring Data і JPA (Java Persistence API)

Робота з даними у сучасному світі Java давно не обмежується стандартним JDBC. JPA є стандартом Jakarta EE для роботи з реляційними базами даних через ORM (Object-Relational Mapping). В той час як Spring Data – це надбудова над JPA, яка надає простий та ефективний спосіб взаємодії з даними за допомогою ряду абстракцій і інструментів, що спрощують розробку та підтримку застосувань.

Spring Data дозволяє розробникам працювати з базами даних на вищому рівні абстракції, пропонуючи різноманітні інструменти для виконання операцій збереження, оновлення та видалення даних без прив'язки до конкретної реалізації JPA або SQL.

Отже, Spring Data спрощує роботу з JPA, надаючи більш високорівневий і зручний спосіб взаємодії з базами даних, що робить його привабливим для розробників. Це дозволяє писати більш зрозумілий та менш залежний від конкретної реалізації код, зменшуючи кількість шаблонного коду та полегшуючи розробку програмного забезпечення. Треба зазначити, що на платформі Jakarta наразі відсутній будь-який аналог Spring Data. Jakarta Data – нова специфікація, запропонована для включення в Jakarta EE 11, випуск якої

заплановано на першу половину 2024 року. Ця специфікація спрощує доступ до даних та зменшує кількість необхідного шаблонного коду, реалізуючи патерн Repository.

Розробники можуть визначити інтерфейс, що представляє сховище (репозиторій), та сутність, яка відображає таблицю в базі даних. Реалізація Jakarta Data буде надавати реальну реалізацію репозиторію, спрощуючи процес взаємодії з базою даних та забезпечуючи зручний доступ до неї без необхідності писати велику кількість стандартного коду. Це може сприяти прискоренню та спрощенню розробки застосунків, що працюють з базами даних.

Spring Boot та Enterprise Server

Існуючі платформи пропонують два різних підходи до створення та розгортання веб-застосунків, що розробляються мовою Java.

Spring Boot – фреймворк, який спрямований на прискорення процесу розробки шляхом надання уніфікованого та автоматизованого способу налаштування та розгортання застосунків. Він надає різноманітні можливості та конфігурації, що спрощують створення інтернет-застосунків та мікросервісів.

Enterprise Server – рішення, яке зазвичай включає в себе набір різноманітних технологій та фреймворків Jakarta EE, направлених на побудову корпоративних застосунків. Це включає в себе сервери застосунків, що мають інтегровані різні фреймворки, такі як EJB, JPA, CDI та інші, у складних та зручних конфігураціях.

Spring Boot спрощує розгортання за допомогою своїх уніфікованих інструментів та автоматизації, дозволяючи розробникам швидше створювати та запускати застосунки. У той же час, Enterprise Server надає повний набір різних фреймворків та технологій, спрямованих на побудову корпоративних

систем, але може вимагати більшої складності у конфігурації та управлінні, особливо у великих проектах. Обираючи між ними, важливо враховувати конкретні потреби проекту та рівень зручності розробки та управління.

Розширюваність та гнучкість

В екосистемі Jakarta EE існують різні сервери, що імплементують специфікації Jakarta EE, проте обмеження стандарту може призвести до меншої гнучкості у виборі та налаштуванні програмних середовищ.

У свою чергу Spring надає більшу гнучкість у виборі компонентів та конфігурації. Існує можливість вибору різних модулів (Spring Boot, Spring MVC, Spring Data тощо), яка дає змогу створювати різноманітні типи застосувань.

Підтримка та спільнота

Jakarta EE – як стандарт, Jakarta EE потребує використання конкретних специфікацій, але підтримка може бути менш активною порівняно з Spring. Оновлення, виправлення помилок та недоліків, а також удосконалення можуть займати більше часу.

Spring має широку та активну спільноту розробників, що сприяє швидкому виявленню та виправленню помилок, оновлень та поширення знань через велику кількість документації та ресурсів.

Висновки

Наведений аналіз щодо використання різних стеків у розробці програмного забезпечення корпоративних систем показав що кожен стек має свої сильні сторони та найкраще підходить для певних видів проектів.

Jakarta EE відмінно підходить для розробки монолітних застосунків, які легко масштабуються. З іншого боку, Spring ідеально підходить для менших за розміром застосунків зі складним інтерфейсом користувача або для мікросервісної архітектури.

Також важливо враховувати актуальність AJAX у сучасних застосуваннях. Використання Spring MVC та Thymeleaf може потребувати значних зусиль для розробки функцій AJAX. Особливо у великих проектах це може бути витратно за часом. У такому разі альтернативою можуть бути React/Angular або традиційний JSF з PrimeFaces. Останній варіант, хоч і менш ефектний, але також є відмінним варіантом, особливо для бек-офісних додатків. PrimeFaces надає широкий спектр готових бізнес-компонентів, що може значно полегшити процес розробки.

Проаналізувавши вимоги до системи, зважаючи на відносно невеликий її об'єм, та обмеження у ресурсах, а також з урахуванням потреби до обробки різноманітних даних, що є лише в Spring Data, було прийнято рішення вести розробку концепції системи та її реалізацію з орієнтуванням на зв'язку Spring Boot + JavaScript фреймворк Vue для представлення даних.

1.5 Розробка концепції інформаційної системи

За результатом сформованих вимог розроблено концепцію інформаційної системи та її модель, відповідно до функціональних вимог.

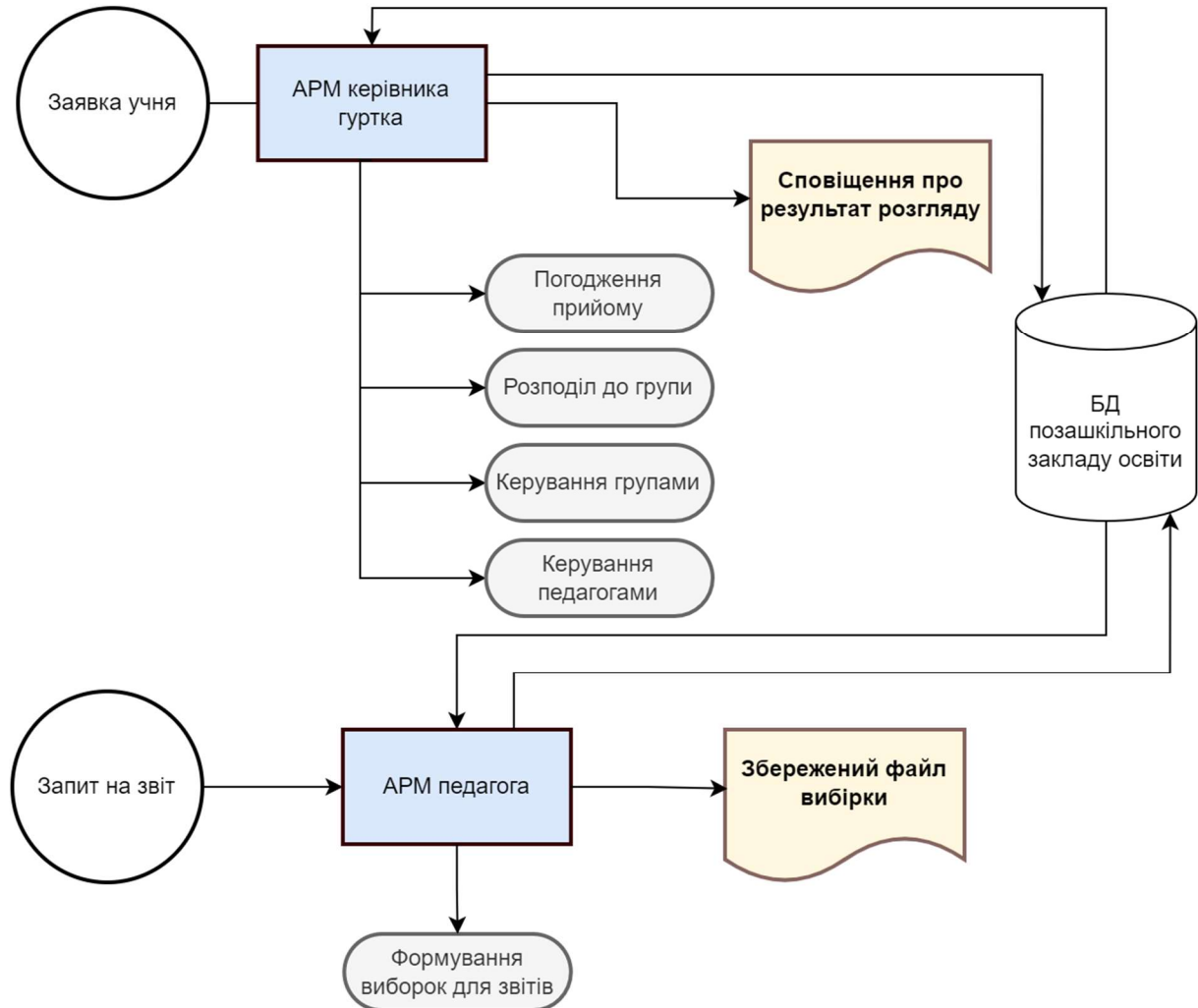


Рисунок 1.7 – Модель інформаційної системи

Виходячи зі схеми моделі, певний користувач, підключившись до інформаційної системи використовуючи додаток, може працювати зі своїм автоматизованим робочим місцем, всі дані для кого будуть отримані і будуть зберігатися в базі даних позашкільного навчального закладу.

За аналізом розробленої моделі виділено основні частини:

- 1) АРМ керівника гуртка;
- 2) АРМ педагога;
- 3) Система управління базою даних та власне, БД;

4) Технічні засоби;

5) Програмні засоби;

Розподілення системи, що розробляється, на окремі АРМ дозволить зробити її структуру більш модульною та здатною для розширення.

Автоматизоване робоче місце (АРМ) складається з комплексу обчислювальної техніки та програмного забезпечення, розташованого прямо на робочому місці працівника і спрямованого на автоматизацію роботи відповідно до його спеціальності.[4]

Створення автоматизованих робочих місць повинне точно відповідати їхнім передбачуваним функціональним цілям. Однак загальні принципи розробки АРМ залишаються незмінними. До них відносяться: системність, гнучкість, стійкість, ефективність.

АРМ керівника гуртка автоматизує наступні задачі:

- 1) Розподіл до групи: при отриманні заявки нового учня, обирає відповідну групу для нього;
- 2) Погодження прийому: здійснюється перевірка наявності місць в групах та коректність вказаних даних;
- 3) Керування групами: здійснюється створення, заміна, видалення груп відповідно до потреб гуртка;
- 4) Керування педагогами: здійснюється прикріплення педагогів до груп, додавання нових, зміна їх даних.

АРМ педагога автоматизує формування вибірок для звітів: здійснюється на основі певного запиту на звіт, дозволяє згенерувати документ для збереження вибірки;

Системою управління базою даних було обрано вільну об'єктно-реляційну систему – PostgreSQL.

Технічні засоби

Для нормальної роботи клієнтської сторони, вона повинна мати наступні технічні характеристики:

- 1) Процесор – Intel Core 2 Duo 2 GHz і продуктивніші;
- 2) Графічна карта – 256 MB;
- 3) Оперативна пам'ять – 2 GB і більше;
- 4) Мережева карта – 100 Mbit;
- 5) Мишка;
- 6) Клавіатура;
- 7) Монітор, роздільна здатність дисплею якого 1024x768 та більше.

Для серверної сторони рекомендованими системними вимогами до комплектуючих комп'ютера є:

- 1) Процесор – Intel Core 2 Duo 2.5 GHz і продуктивніші;
- 2) Графічна карта – 256 MB;
- 3) Оперативна пам'ять – 4 GB і більше;
- 4) Мережева карта – 100 Mbit;
- 5) Місце на жорсткому диску – 20 GB;
- 6) Мишка;
- 7) Клавіатура;
- 8) Монітор, роздільна здатність дисплею якого 1024x768 і більше.

Відповідно, в кожного педагога має бути доступ до робочого комп'ютера підключеного до мережі Інтернет або смартфона з доступом до веб-додатку.

Програмні засоби

Розробка веб-додатку буде виконуватись використовуючи об'єктно-орієнтовану мову програмування Java, використовуючи IDE (інтегроване середовище розробки) IntelliJ IDEA Ultimate Edition. Для розробки веб-застосування, за результатом дослідження було обрано типове рішення для невеликих систем, що інтенсивно використовують дані – фреймворк Spring, для налаштування та зручної конфігурації було обрано один з проєктів Spring – Spring Boot. Для розробки інтерфейсу користувача застосування обрано Vue.JS - JavaScript-фреймворк і його бібліотеку UI-компонентів Buefy. У якості системи контролю версій має використовуватися Git та веб-сервіс GitHub, який інтегрується з середовищем розробки IntelliJ IDEA.

Завдяки тому, що система має бути у вигляді веб-застосування, вона є кросплатформною і для роботи з нею потрібна будь-яка операційна система та будь-який веб-браузер. Операційною системою для комп'ютерів обрано Linux Mint, а для смартфонів – Android, в обох випадках рекомендується використовувати браузер Google Chrome.

Для створення діаграм було обрано онлайн графічний редактор Diagrams.net. Для документації коду потрібно використовувати Javadoc - автоматичний генератор документації в HTML-форматі з коментарів вихідного коду мовою Java, а також Swagger - фреймворк для документування REST-сервісу.

Сервісні засоби, що необхідні для функціонування системи:

- 1) Стабільне інтернет підключення;
- 2) Електронна пошта;

Для інтернет підключення всередині позашкільного навчального закладу використовуватиметься технологія Ethernet.

У процесі роботи позашкільних навчальних закладів з інформаційною системою виникає потреба у отриманні повідомлень про нові заявки до гуртка, а також для взаємодії між співробітниками навчального закладу. Для покриття цього завдання необхідно використати електронну пошту. Для цього було обрано Gmail у якості поштового сервісу – безкоштовна поштова служба від компанії Google.

В навчальному закладі на стаціонарних комп'ютерах зв'язок з сервером буде здійснюватися за допомогою мережі Інтернет. Для доступу до серверу зі смартфона, використовується 4G – четверте покоління мобільного радіозв'язку.

Отже, реалізація інформаційної системи дозволить досягнути цілей закладу позашкільної освіти за рахунок автоматизації типових процесів та раціонального використання часу співробітників. Завдяки пришвидшенню виконання типової роботи і структуруванню даних, педагоги матимуть більше часу на підготовку до занять, створення навчального плану, робочих програм та інших навчальних матеріалів, що покращить рівень навчального процесу.

РОЗДІЛ 2. РОЗРОБКА ТЕХНІЧНОГО ПРОЕКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ УЧНІВ ПОЗАШКІЛЬНИХ НАВЧАЛЬНИХ ЗАКЛАДІВ

2.1 Загальносистемні рішення

Відповідно до зазначеної концепції, визначено, що педагоги, керівник гуртка, адміністрація позашкільного навчального закладу і учні (їх батьки) стануть користувачами інформаційної системи. Діаграма прецедентів (Use-case diagram) відображає інформаційні процеси користувачів та їх взаємодію.

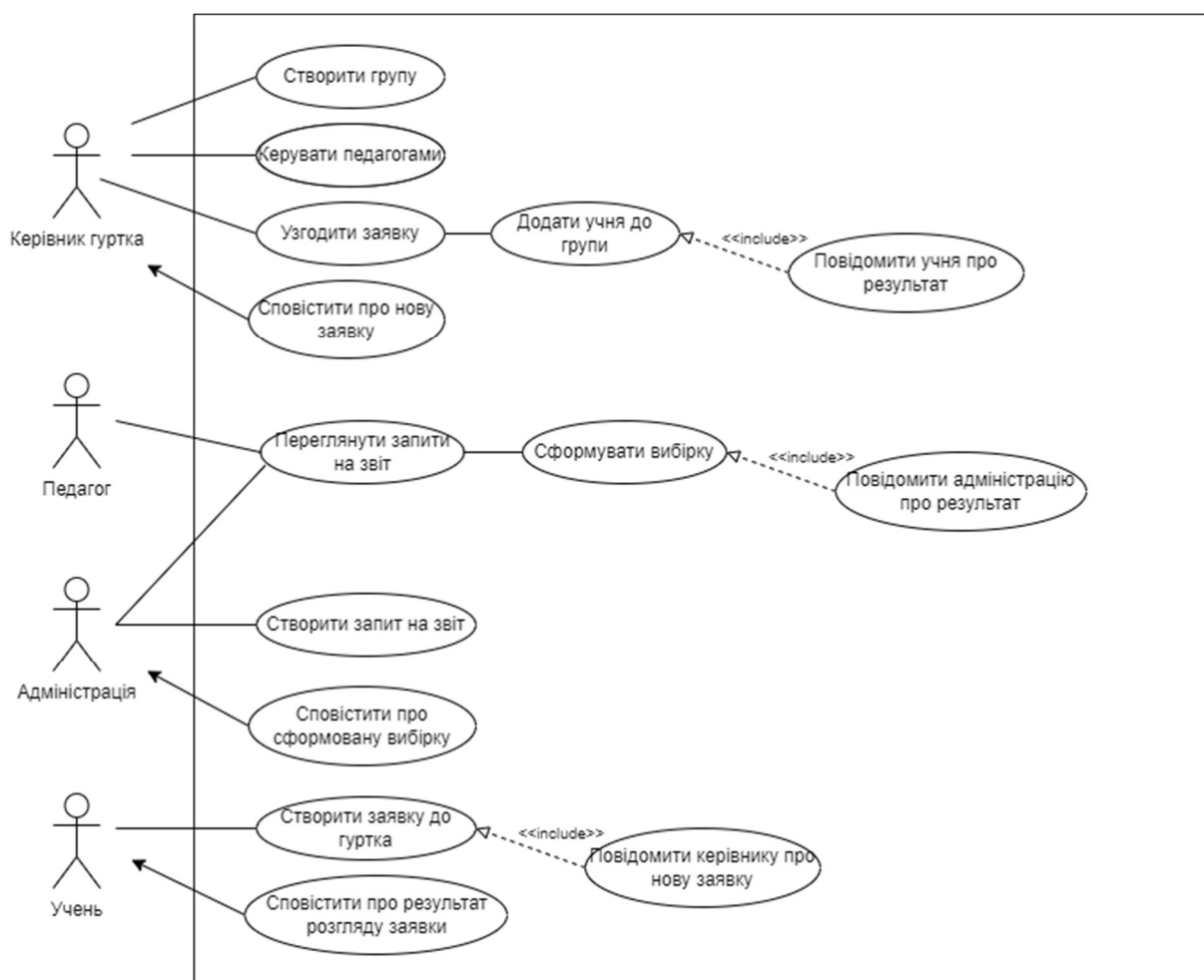


Рисунок 2.1 – Діаграма прецедентів

Використання інформаційної системи розглянуто у таблицях 2.1-2.3, де представлені різні сценарії використання:

Таблиця 2.1. Створення групи

Створення групи	
Призначення	Використовується для створення нової групи
Користувач	Керівник гуртка
Передумова	Необхідне створення нової групи для учнів
На що реагує	Натискання на кнопку «Створити групу».
Сценарій	Керівник гуртка натискає кнопку «Створити групу», заповнює інформацію про групу, натискає кнопку «Зберегти».
Що відбудеться	Інформація про нову групу буде збережена в базі даних.

Таблиця 2.2. Створення запиту на звіт

Створення запиту на звіт	
Призначення	Використовується для створення запиту на звіт
Користувач	Адміністрація
Передумова	Було отримано запит на звіт від навчального закладу, необхідно зібрати актуальні дані по гурткам.
На що реагує	Натискання на кнопку «Створити запит на звіт».
Сценарій	Адміністрація натискає кнопку «Створити запит на звіт», заповнює інформацію про звіт, призначає вчителя, натискає кнопку «Відправити».
Що відбудеться	Призначений вчитель отримує повідомлення про новий запит на звіт. Інформація про запит буде збережена в базі даних.

Таблиця 2.3. Узгодження заявки

Узгодження заявки	
Призначення	Використовується для узгодження отриманої заявки до гуртка
Користувач	Керівник гуртка
Передумова	Учнем було створено заявку до гуртка, отримано повідомлення про нову заявку. Необхідна перевірка заявки та розподілення учня до групи.
На що реагує	Натискання на кнопку «Перевірити заявки».
Сценарій	Керівник гуртка натискає кнопку «Перевірити заявки», перевіряє дані введені в заявці, змінює статус заявки, за потребою призначає до групи і натискає на кнопку «Зберегти».
Що відбудеться	Результат розгляду заявки відправлено учню. Інформація про результат перевірки збережена в базі даних.

На діаграмі прецедентів відображено, що деякі завдання можуть бути виконані користувачами індивідуально, тоді як інші вимагають спільної участі кількох користувачів. Наприклад, для схвалення заявки керівник гуртка повинен попередньо створити заявку, заповнену потенційним учнем, який вводить основні дані та інформацію про себе. Після цього система повідомляє керівника гуртка, який перевіряє заявку на відповідність критеріям зарахування, наявності місць у групах гуртка та приймає рішення. Результат розгляду повідомляється учневі, і, у разі позитивного рішення, учень додається до відповідної групи. Такий підхід до обробки заявок дозволяє негайно формувати групи в гуртках і сповіщати учнів про результати розгляду, що сприяє підвищенню ефективності роботи керівника гуртка та спрощує інформування учнів про їх зарахування.

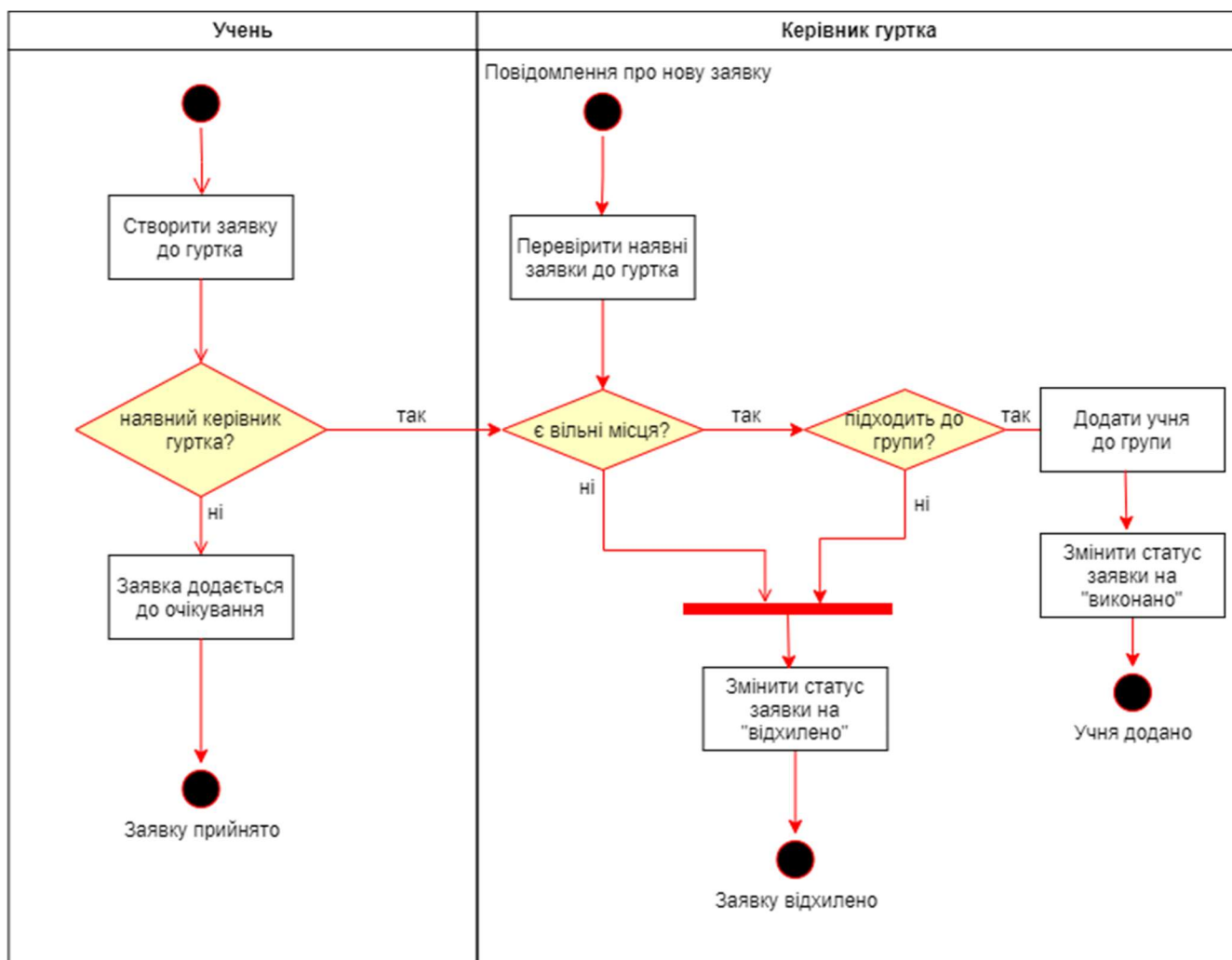


Рисунок 2.2 – Діаграма діяльності процесу обробки заявки до гуртка

На діаграмі діяльності представлено спосіб спільної роботи з заявками на вступ до гуртка. Спочатку учень створює заявку, вказавши в ній свої дані. У випадку відсутності доступного керівника гуртка для обробки заявок, заявка додається до черги очікування та зберігається. Якщо керівник гуртка доступний, він отримує повідомлення про нову заявку, переглядає існуючі заявки, перевіряє наявність вільних місць у групах та аналізує введені учнем дані. Базуючись на цих відомостях, він визначає, чи відповідає учень критеріям прийому до будь-якої групи. У випадку успішної перевірки, учень включається до конкретної групи, а статус його заявки змінюється на "Виконано", інформація зберігається. Якщо немає вільних місць або учень не відповідає

критеріям групи, керівник змінює статус заявки на "Відхилено". Учень отримує повідомлення про результат розгляду своєї заявки.

Для ефективної колективної роботи інформаційна система повинна відповідати наступним вимогам:

- 1) Мати простий і інтуїтивно зрозумілий користувацький інтерфейс;
- 2) Забезпечувати високу надійність інформаційної системи;
- 3) Бути готовою до одночасної роботи багатьох користувачів;
- 4) Виявляти універсальність в своїй функціональності;
- 5) Мати легкий процес початку використання для навчального закладу;
- 6) Використовувати загальноприйняті стандарти розробки.

Простий та інтуїтивно зрозумілий користувацький інтерфейс. Легкий і зрозумілий інтерфейс для користувачів є ключовою умовою для ефективної роботи інформаційної системи, оскільки такий інтерфейс дозволяє кожному користувачеві використовувати систему без необхідності додаткової допомоги чи інструкцій. Для забезпечення цієї вимоги інтерфейс повинен бути графічним, тобто користувач взаємодіє з системою через графічні зображення та візуальні вказівки на екрані. Простий графічний інтерфейс дозволяє користувачеві легко розуміти необхідні дії та працювати з інформаційною системою.

Надійність інформаційної системи. Надійність — властивість технічних об'єктів зберігати протягом встановленого часу значення всіх параметрів, які характеризують здатність виконувати потрібні функції в заданих режимах та умовах застосування, технічного обслуговування, зберігання та транспортування. Під технічними об'єктами розуміють пристрої, прилади, механізми, машини, комплекси обладнання, будівельні конструкції і споруди, технологічні операції і процеси, системи зв'язку, інформаційні системи, автоматизовані системи управління технологічними процесами тощо. [5]

Для забезпечення надійності даних інформаційної системи на сервері слід створити підсистему, яка відповідає за реєстрацію та надійну автентифікацію користувачів. В разі необхідності також слід встановити антивірусне програмне забезпечення.

Для забезпечення розрахованості на роботу з багатьма користувачами системи, доступ до певних дій має бути доступний лише користувачам, які мають відповідні права доступу. Кожен користувач, в залежності від свого типу, має отримувати конкретні права для виконання дій та запитів у базі даних.

Універсальність інформаційної системи виявляється у тому, що її можна використовувати іншими закладами зі схожою структурою для виконання подібних завдань. Наприклад, певні елементи системи, які відповідають за генерацію вибірок, можуть бути використані і в навчальних закладах, і в приватних компаніях, що організовують курси та тренінги.

Простий початок користування інформаційною системою в навчальному закладі є важливим критерієм. Введення системи повинно відбуватися легко і без проблем.

Використання загальноприйнятих стандартів розробки дозволяє іншим особам, які будуть займатися підтримкою проекту, ефективно взаємодіяти з ним. Це спрощує подальші завдання з розширення, модернізації системи та виправлення можливих неполадок.

2.2 Рішення з інформаційного забезпечення

Концептуальне проектування — побудова семантичної моделі предметної області, тобто інформаційної моделі найбільш високого рівня абстракції. Така модель створюється без орієнтації на якусь конкретну СУБД і модель даних. [6]. Основуючись на вивченні предметної області, вимог до інформаційної системи та потреб замовника, була створена концептуальна модель даних і відтворена взаємодія об'єктів. Об'єкти в цій концептуальній моделі відповідають реальним об'єктам в даній області.

Усі об'єкти в предметній області мають певні атрибути, що відображають їхні характеристики. Обираються необхідні атрибути об'єкта, які визначають дані, що пов'язані з об'єктом. Ті атрибути, за якими можна однозначно визначити об'єкт, називаються ідентифікаторами об'єкта.

В предметній області виділяються різні сутності, такі як вчитель, учень, група, гурток, статус, заявка до групи, запит на звіт.

У результаті концептуального проектування було створено модель бази даних інформаційної системи для позашкільного навчального закладу у формі ER-діаграми, яку можна бачити на рисунку 2.3.

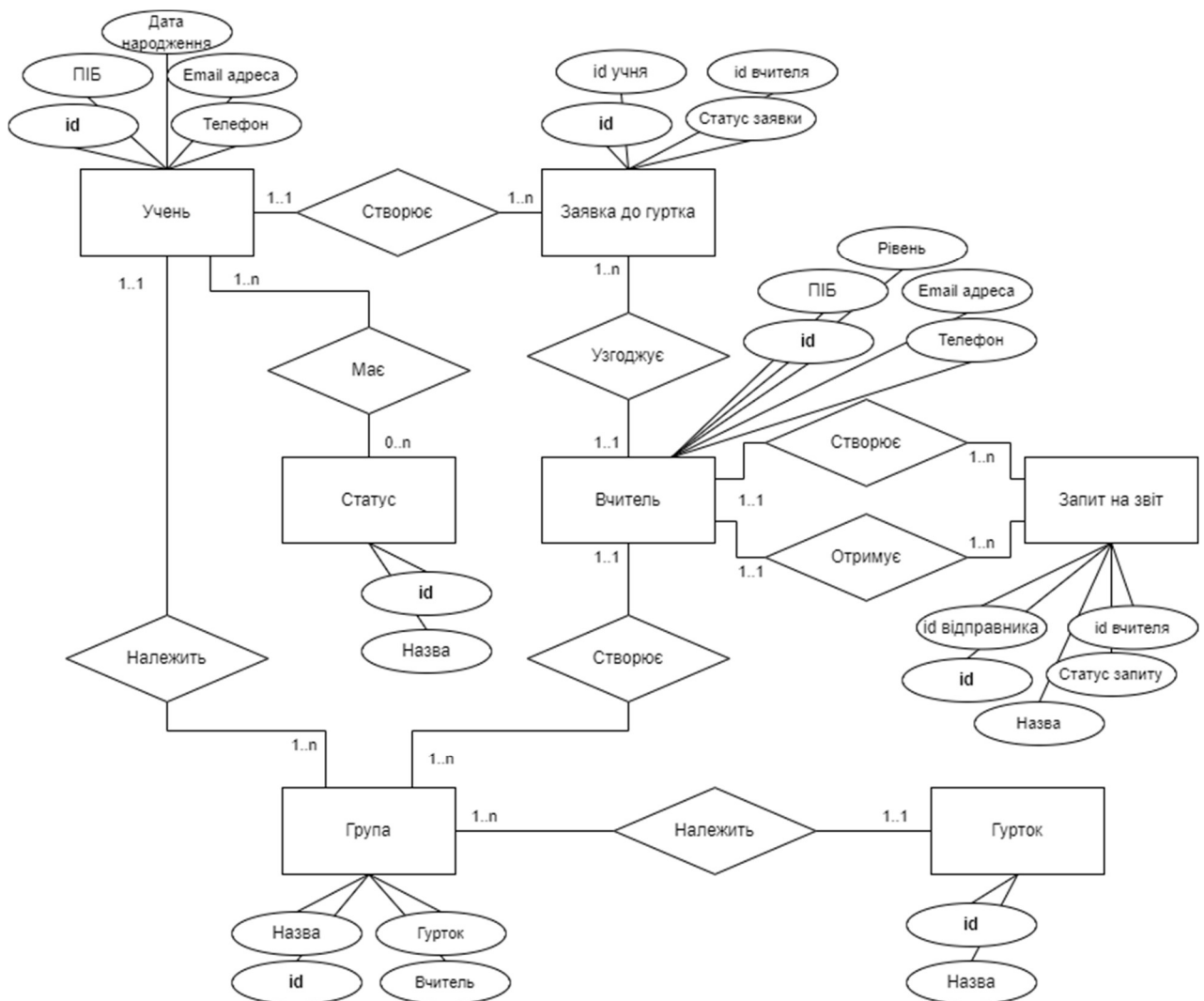


Рисунок 2.3 – Концептуальна ER-діаграма

Під час розробки встановлюється основа можливої бази даних, визначається необхідна структура цієї бази, спираючись на вибрану предметну область. Логічна структура бази даних визначається у вигляді реляційних таблиць, включаючи їх

склад та логічні зв'язки. При формуванні структури кожної таблиці визначаються поля, що відображають атрибути конкретного об'єкта, і додається опис, тип даних, обмеження та інші властивості. Крім того, визначається унікальний ключ таблиці, який може включати одне або кілька полів та ідентифікувати окремий запис в таблиці.

Фізична структура бази даних - це організація створеної бази даних для комп'ютера, визначаючи тип, властивості та обмеження даних, які будуть зберігатися в пам'яті комп'ютера. Фізичне моделювання бази даних включає визначення файлів та їх наповнення вихідними даними відповідно до предметної області та обмежень даних. Проектування бази даних зазвичай включає створення таблиць у системі управління базами даних (СУБД). Також при фізичному проектуванні бази даних необхідно визначити імена таблиць та імена полів таблиць.

Фізична структура бази даних інформаційної системи обліку учнів позашкільних навчальних закладів представлена на рисунку 2.4.

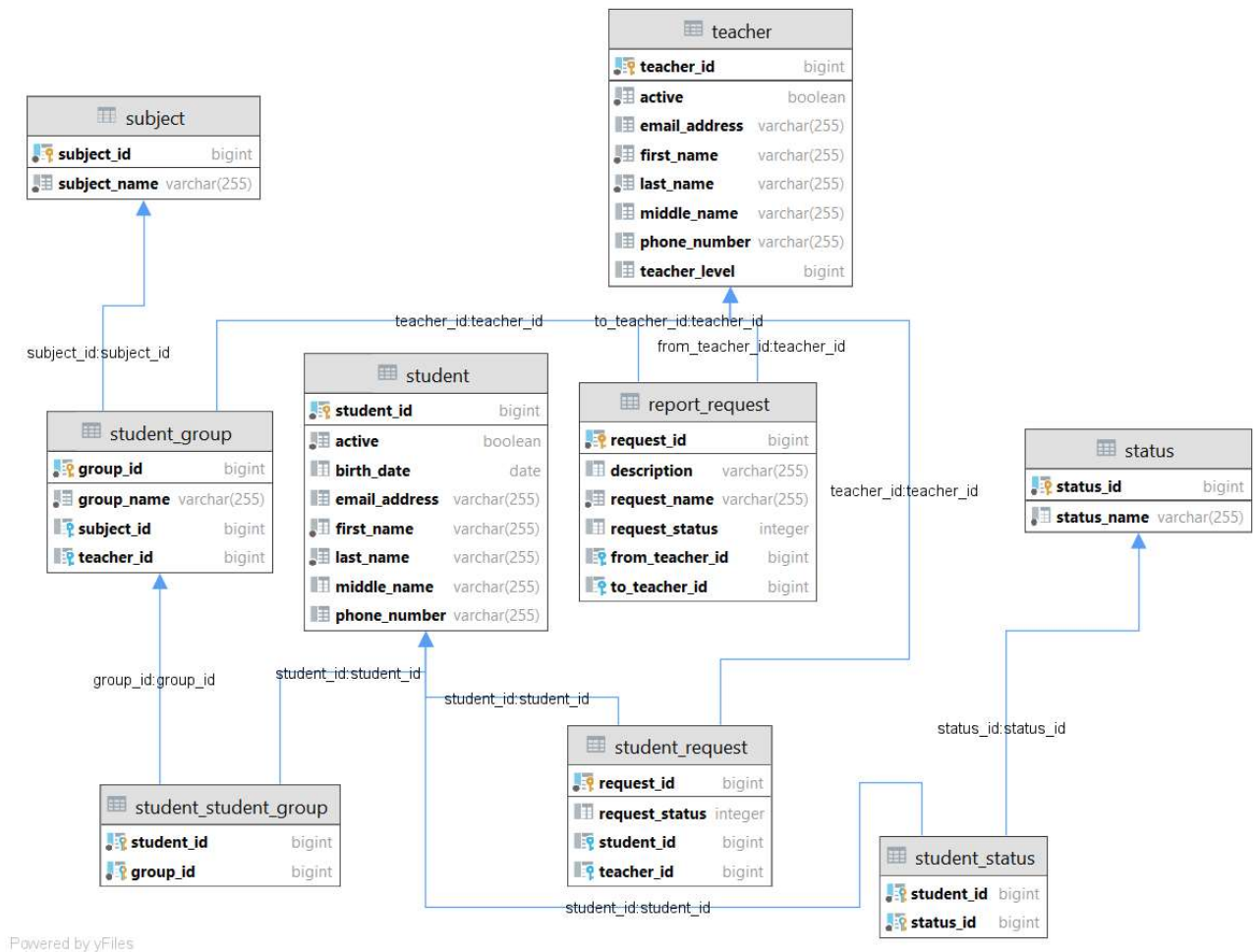


Рисунок 2.4 – Фізична модель бази даних

Була створена фізична модель, яка включає в себе структуру бази даних, об'єкти системи, взаємозв'язки між ними, а також характеристики та властивості об'єктів. Для цього були визначені поля та первинні ключі.

Сутності системи представлені за допомогою рядків таблиць бази даних і відображаються як об'єкти класів у програмах серверної та клієнтської частин. Взаємозв'язки та зв'язки між цими класами інформаційної системи для позашкільного навчального закладу відображено на діаграмі класів, яка представлена на рисунку 2.5. Треба зазначити, що оскільки для зберігання даних використовується реляційна система управління базами даних PostgreSQL, а для реалізації бізнес-логіки – об'єктно-орієнтована мова програмування Java, то для коректної взаємодії цих технологій потрібно використовувати відповідні механізми перетворень. Найпоширенішими є механізми JDBC (Java Data Base Connectivity) та

JPA (Java Persistence API). Як було визначено раніше, для підвищення продуктивності розробки, використано підхід JPA на основі відповідного фреймворку Spring Data JPA.

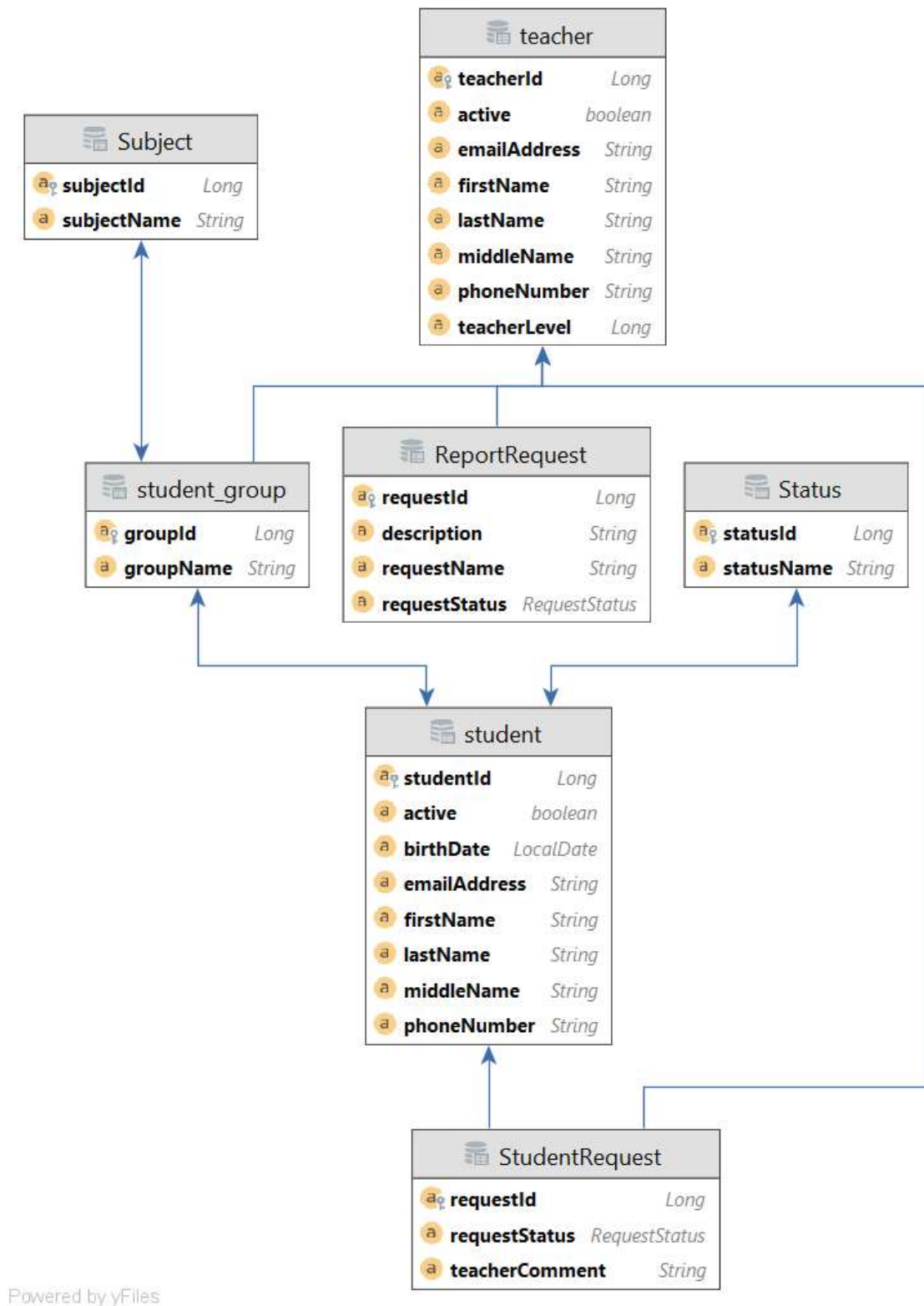


Рисунок 2.5 – Діаграма класів

2.3 Рішення з програмного забезпечення

Для створення бази даних в інформаційній системі використовується система управління базами даних PostgreSQL. PostgreSQL — це потужна система об'єктно-реляційних баз даних з відкритим вихідним кодом, яка використовує та розширює мову SQL у поєднанні з багатьма функціями, які безпечно зберігають і масштабують найскладніші робочі навантаження даних. [7]

Для будь-яких проектів на стадії ініціалізації, SQL буде розумним рішенням. Причина в тому, що нещодавно розпочаті проекти зазвичай не мають структурованих вимог і можуть змінювати їх з часом. У NoSQL ці зміни важко реалізувати, тоді як бази даних SQL є більш гнучкими.

PostgreSQL також пропонує багато додаткових переваг. Одна з них - її надійність. Не менш важливою є цілісність даних у базі даних. З PostgreSQL не має потреби турбуватися про це. Останнє, але не менш важливе — це продуктивність. Це найбільш визначальний фактор. [8]

PostgreSQL повністю відповідає стандартам ACID. Ця система управління базами даних також є сумісною з різними платформами, які використовують різні основні мови програмування та проміжне програмне забезпечення.

Код для створення таблиць можна знайти в додатку Г.

Веб-додаток інформаційної системи розроблено мовою програмування Java. Java є широко використовуваною об'єктно-орієнтованою мовою програмування та програмною платформою, яка підтримується на різних пристроях, таких як ноутбуки, мобільні пристрої, ігрові консолі, медичні пристрої та інші. Однією з ключових переваг розробки програм на Java є її кросплатформеність.

Для написання програми на Java необхідно встановити Java Development Kit (JDK). Після написання програми мовою Java, компілятор перетворює її в

байт-код Java — набір інструкцій для віртуальної машини Java (JVM), яка є частиною середовища виконання Java (JRE). Байт-код Java є переносним і може виконуватися в будь-якій системі, яка підтримує JVM, що забезпечує можливість запускати Java-код на різних платформах.

Для розробки програмних модулів системи було використано інтегроване середовище розробки (IDE) IntelliJ IDEA Ultimate Edition. IntelliJ IDEA містить значну кількість інтегрованих інструментів, що дозволяють розробникам працювати більш ефективно та продуктивно. Один з таких інструментів — це високорозвинений редактор, який надає користувачам зручний інструментарій під час написання коду.

Як головні особливості IntelliJ IDEA, що розроблена компанією JetBrains слід відмітити широкий спектр інструментів, які полегшують розробку програмного забезпечення, а також підвищують ефективність та прискорюють процес розробки, великий набір плагінів (доповнень), що орієнтовані на різні аспекти розробки. Наприклад, модуль взаємодії з БД дозволяє ефективно проєктувати структуру БД, налаштовувати зв'язки між таблицями та легко візуалізувати представлення БД у вигляді діаграми; модуль взаємодії з серверами контролю версій дозволяє вести облік змін та організовувати взаємодію розробників у процесі колективної розробки; плагін JPA Buddy призначений для автоматизації робіт зі створення об'єктного представлення БД, розробки запитів та репозиторіїв, об'єктів передачі даних.

Також для розробки, згідно обґрунтування у розділі 1, була використана група фреймворків Spring. Spring Framework (Spring) — програмне забезпечення з відкритим вихідним кодом, яке забезпечує підтримку інфраструктури для розробки різноманітних застосувань мовами Java, Kotlin та Groovy. Spring допомагає розробникам створювати високопродуктивні програми за допомогою простих об'єктів Java (POJO). Spring вважається надійним, недорогим і гнучким фреймворком. Spring покращує ефективність кодування та скорочує загальний час розробки застосувань, оскільки він легкий, ефективний у використанні системних ресурсів і має велику підтримку

[9].

Для швидкого налаштування та узгодження версій бібліотек, що використовуються у застосуванні, був задіяний один з найсучасніших проєктів Spring – Spring Boot. Spring Boot – це відкритий веб-фреймворк для Java, заснований на принципах мікросервісної архітектури. Фреймворк Spring Boot надає повністю готове до використання середовище, яке можна повністю налаштувати за допомогою заздалегідь створеного в його базі коду. За допомогою архітектури мікросервісів розробник отримує повністю самодостатнє застосування, включаючи вбудовані сервери застосувань.

Для взаємодії з об'єктами бази даних використано Spring Data JPA. Spring Data JPA, який є частиною екосистеми Spring Data, спрощує реалізацію репозиторіїв на основі JPA. Цей модуль надає розширену підтримку доступу до даних на основі JPA, що полегшує створення додатків на основі Spring, які використовують технології доступу до даних.

Для перевірки автентифікації користувачів використовується Spring Security. Spring Security – це фреймворк, спрямований на створення механізмів автентифікації та авторизації для додатків, розроблених з використанням фреймворку Spring.

Для створення документації API використовувався інструмент Swagger. Swagger дозволяє описати структуру API, щоб машини могли їх читати. Здатність API описувати власну структуру є коренем усієї приголомшливості Swagger. Читаючи структуру API, автоматично створюється інтерактивна документація API. Також можливе автоматичне створення клієнтської бібліотеки для API багатьма мовами та досліджування інших можливостей, наприклад автоматичне тестування. [10]

Для створення інтерфейсу користувача використано Vue.js. Vue.js – це фреймворк JavaScript, який використовує шаблон MVVM для створення інтерфейсів користувача на основі моделей даних за допомогою реактивного зв'язування. Для зручного створення графічного інтерфейсу використано Vuefy – бібліотеку компонентів інтерфейсу користувача, яка базується на

Vue.js і Bulma, двох відомих фреймворках. В основі бібліотеки Vuefy лежать два принципи: простота у використанні та легкість з точки зору використаних ресурсів.

У роботі було створено 12 модулів інформаційної системи. Їхнє призначення відображено в таблиці 2.4.

№	Назва модуля	Призначення модуля
1	Додавання вчителя	Форма вводу даних про вчителя закладу.
2	Редагування даних вчителя	Форма для перегляду даних про вчителя з можливістю редагувати певні поля.
3	Створення групи	Форма вводу даних для створення нової групи.
4	Редагування групи	Форма перегляду групи з можливістю редагування. Містить перелік учнів, що належать до цієї групи.
5	Створення заявки до гуртка	Форма вводу даних про учня, з подальшим відправленням заявки до керівника гуртка..
6	Керування заявками до гуртка	Відображає весь список заявок до гуртка, з можливістю редагування даних про учня та розподілення учня до групи.
7	Створення запиту на створення звіту	Форма вводу даних про запит на звіт до вчителя.
8	Формування вибірки для звіту	Відображає список учнів з необхідної перспективи за певними критеріями.
9	Додавання учня	Форма вводу для даних про учня.
10	Редагування даних про учня	Форма для перегляду даних про учня з можливістю редагувати деякі поля
11	Створення нової категорії учня	Форма для створення нової категорії, до якої можуть відноситись учні.
12	Редагування категорії	Форма для перегляду даних про категорію з можливістю редагування.

Таблиця 2.4. Призначення програмних модулів інформаційної системи

Наприклад, у додатку Г розташований вихідний код модуля, який взаємодіє з групами, включаючи обробку інформації про учнів та вчителів.

Методи розробки програмного забезпечення

При розробці інформаційної системи було вирішено застосовувати інкрементальний метод розробки через те, що потреба в системі для навчального закладу виникає до початку наступного навчального року. Це надає можливість систематично визначити вимоги до інформаційної системи та розробляти стратегію розробки, маючи достатньо часу для цього. Початково розробляється стратегія розробки з графіком впровадження нового функціоналу системи. Інкрементальна модель дозволяє почати з невеликої частини системи та поступово розширювати її на кількох етапах.

Інформаційна система використовує об'єктно-орієнтоване програмування (ООП) як методологію програмування. Це забезпечує зберігання та обробку структурованих даних у формі об'єктів, що полегшує розробку. Методологія ООП дозволяє структурувати розробку за модулями, сприяє полегшенню підтримки та розширенню проекту в майбутньому.

Слід відмітити, що багаторічне використання об'єктно-орієнтованого підходу проектування та розробки програмного забезпечення, дозволило виявити його сильні і слабкі сторони.

До слабких сторін ООП часто відносять такі проблеми, як складність, надмірну абстракцію, недостатній контроль над пам'яттю, проблеми спадкування, проблеми з розширенням і підтримкою.

У складних системах ООП може стати важко зрозуміти взаємозв'язки між об'єктами та класами. Це може спричинити заплутаність та ускладнення розуміння коду.

Іноді велика кількість абстракцій у ООП може призвести до того, що код стає занадто загальним, важким для використання та розуміння.

У деяких мовах програмування з ООП, зокрема з автоматичним управлінням пам'яттю, як Java, можуть виникати проблеми з ефективністю та недостатнім контролем пам'яті.

У деяких випадках, коли використовується спадкування класів, може виникати складність у відстеженні, як зміни в базовому класі впливають на його похідні класи.

Великі проекти з великою кількістю класів можуть бути важкими для розширення та підтримки з часом. Додавання нового функціоналу або змін до існуючих може призвести до непередбачуваних наслідків.

Проте, використовуючи сучасні шаблони проєктування (Design Patterns) можна знівелювати ці недоліки та максимально використати переваги ООП, до яких відносять модульність та повторне використання коду, успадкування та поліморфізм, інкапсуляцію та захищеність даних, ефективні підходи до управління великими проєктами та підтримку паралельного програмування.

Операційна система

В комп'ютерах позашкільного навчального закладу використовується операційна система Linux, з конкретним дистрибутивом — Mint.

Лінукс (англ. Linux, повна назва — GNU/Linux) — загальна назва UNIX-подібних операційних систем на основі однойменного ядра. Це один із найвидатніших прикладів розробки вільного (free) та відкритого (з відкритим кодом, open source) програмного забезпечення (software). На відміну від власницьких операційних систем (на кшталт Microsoft Windows та MacOS X), їхні вихідні коди доступні всім для використання, зміни та поширення абсолютно вільно. [11]

Дистрибутив – це набір файлів, призначених для установки, який включає всі необхідні компоненти, інсталятор та інші додаткові елементи.

Linux Mint вирізняється такими перевагами:

- легко налаштовується;
- придатний для користувачів-початківців;
- включає усе необхідне з Linux;
- оснащений найкращим менеджером програмного забезпечення;
- має попередньо встановлені програми.

Типові програмні засоби

Для використання інформаційної системи використовується веб-додаток, який доступний через браузер, такий як Google Chrome – швидкий і сучасний браузер, розроблений компанією Google. Браузер відзначається стабільністю, високою безпекою, швидкістю, багатством розширень та простим інтерфейсом користувача.

Для створення діаграм використовується графічний редактор diagrams.net — це технологічний стек з відкритим вихідним кодом для розробки додатків створення діаграм, який є одним з найбільш використовуваних програмних засобів для кінцевих користувачів у браузері. Для генерації фізичної моделі бази даних використовується вбудований функціонал в IntelliJ IDEA (IDE).

Документація API виконується за допомогою технології Swagger. Документація коду здійснена використовуючи Javadoc — генератор документації в HTML-форматі з коментарів вихідного коду мовою Java від Sun Microsystems. Javadoc — стандарт для документування класів Java. Більшість середовищ розробки програмного забезпечення автоматично генерують HTML-документацію, використовуючи Javadoc. [12]

Сервісне програмне забезпечення

У позашкільному навчальному закладі для забезпечення інтернет-з'єднання використовується Wi-Fi. Wi-Fi є торговою маркою Wi-Fi Alliance та загальноприйнятою назвою для стандарту передавання цифрових даних по радіоканалах, відомого як IEEE 802.11. Обладнання, яке відповідає стандарту IEEE 802.11, може бути протестоване Wi-Fi Alliance та отримати відповідний сертифікат і право використовувати логотип Wi-Fi. На сьогоднішній день широко використовується протокол IEEE 802.11n.

Встановлення Wireless LAN доцільне для побудови мереж, де розгортання кабельної системи є неможливим або економічно недоцільним.

Поточні реалізації Wi-Fi дозволяють отримати швидкість передавання даних понад 100 Мбіт/с, при цьому користувачі можуть переміщуватися між точками доступу на території покриття мережі Wi-Fi, використовуючи мобільні пристрої (КПК, смартфони, PSP і ноутбуки), оснащені клієнтськими приймально-передавальними пристроями Wi-Fi та отримувати доступ в Інтернет. [13]

Для обміну сповіщеннями та повідомленнями про нові запити між працівниками навчального закладу використовується електронна пошта. У більшості випадків установи обирають Gmail як поштовий сервіс – зручний та простий сервіс від Google, який є сумісним з іншими сервісами Google і надає безкоштовне сховище в хмарі.

Стратегія резервного копіювання передбачає наявність інформаційного сховища, або "вторинного місця для зберігання даних", яке об'єднує резервні копії "джерел" даних. Репозиторій може бути простим, як будь-який зі списку носіїв резервних копій, або може включати комп'ютеризований індекс, каталог або реляційну базу даних.

Дані резервної копії необхідно зберігати, що вимагає схеми чергування резервних копій, яка є системою резервного копіювання даних на комп'ютерних носіях, яка обмежує кількість резервних копій різних дат, що зберігаються окремо, шляхом відповідного повторного використання носія для зберігання даних та шляхом перезапису резервних копій, які більше не потрібні. Схема визначає, як і коли кожен фрагмент певного сховища використовується для операції резервного копіювання і як довго він зберігається, коли на ньому зберігаються дані резервної копії. [14]

- Існує кілька видів резервного копіювання, включаючи:
 - Повне резервне копіювання;
 - Диференційне резервне копіювання;
 - Інкрементне резервне копіювання;
 - Клонування;

- Резервне копіювання як образ;
- Резервне копіювання в режимі реального часу;
- Холодне резервування;
- Гаряче резервування.

Сховищами для резервних копій можуть бути:

- Стрічкова бібліотека або стример – резервні копії записуються на магнітну стрічку стримера;
- Дискове сховище – резервні копії записуються на диски, які можуть бути об'єднані в RAID або на дискову систему зберігання;
- Virtual Tape Library – резервні копії записуються на диски, проте саме сховище виглядає для клієнта як стрічкова бібліотека;
- "Хмарний" бекап – резервні дані записуються за допомогою "хмарної" технології через онлайн-служби спеціалізованих провайдерів.

Для резервного копіювання бази даних інформаційної системи обліку учнів позашкільних навчальних закладів використовується утиліта `pg_dump`, вбудована в PostgreSQL. Ця утиліта спрощує процес створення резервних копій баз даних і використовується як користувачем із відповідними правами на читання бази даних, яку потрібно зробити резервною.

Отже, розроблена інформаційна система відповідає вимогам, пройшла тестування, перевірку та має захист від втрати даних.

2.4 Рішення з технічного забезпечення

У сучасних інформаційних системах різних компаній та установ часто використовуються два основних типи носіїв даних: жорсткі магнітні диски (HDD) та твердотільні накопичувачі (SSD). HDD представляє собою традиційний жорсткий диск, тоді як SSD – це сучасний твердотільний накопичувач.

Порівнюючи ці два типи носіїв, основними перевагами SSD є висока швидкість роботи та висока надійність. HDD є більш застарілою технологією та основною їх перевагою є більш доступна ціна.

Щодо обсягу зберігання, SSD доступні в розмірах від 120 ГБ до 30,72 ТБ, тоді як жорсткі диски можуть мати обсяг від 250 ГБ до 20 ТБ. З урахуванням вартості на один гігабайт жорсткі диски перевершують SSD, але зі зниженням цін на SSD ця різниця стає менш помітною. Тим не менш, твердотільні накопичувачі забезпечують більшу продуктивність на серверах, що дозволяє досягти бажаних результатів за допомогою меншої кількості пристроїв. Це призводить до того, що SSD мають більш низьку загальну вартість володіння (TCO).

Надійність, яка визначається збереженням даних у непошкодженому стані, є ще однією перевагою SSD. Оскільки у SSD відсутні рухомі частини, вони загалом надійніше за HDD. Відсутність руху дозволяє SSD уникати проблем, пов'язаних з вібрацією та температурою, що робить їх більш стійкими.

SSD зазвичай споживають менше енергії і мають триваліший термін служби акумулятора, оскільки доступ до даних відбувається набагато швидше, а пристрій частіше простоює. Завдяки дискам, що обертаються, жорсткі диски вимагають більше енергії під час запуску, ніж SSD. [15]

NAND – це енергонезалежна флеш-пам'ять, яка може зберігати дані, навіть якщо вона не підключена до джерела живлення. Можливість зберігати дані, коли живлення вимкнено, робить NAND чудовим варіантом для внутрішніх, зовнішніх і портативних пристроїв. USB-накопичувачі, твердотільні накопичувачі та SD-карти використовують флеш-технологію, забезпечуючи пам'ять для таких пристроїв, як мобільний телефон або цифрова камера.

На ринку є кілька типів NAND. Простіше кажучи, що відокремлює кожен тип, так це кількість бітів, які можна зберегти в комірці. Біти

представляють електричний заряд, який може містити лише одне з двох значень, 0 або 1, увімкнено/вимкнено.

Основними відмінностями між типами NAND є вартість, ємність і витривалість. Витривалість визначається кількістю циклів програмування-стирання (P/E), які може пройти елемент флеш-пам'яті, перш ніж він почне зношуватися. Цикл P/E – це процес стирання та запису осередку, і чим більше циклів P/E може витримати технологія NAND, тим краще витривалість пристрою.

Поширеними типами флеш-сховища NAND є SLC, MLC, TLC і 3D NAND.

SLC NAND

Плюси: Найвища витривалість. Мінуси: Дорогі та невеликі ємності

Однорівневий осередок (SLC) NAND зберігає лише 1 біт інформації на клітинку. У клітинці зберігається або 0, або 1, і в результаті дані можуть бути записані та отримані швидше. SLC забезпечує найкращу продуктивність і найвищу витривалість із 100 000 циклів P/E, тому він прослужить довше, ніж інші типи NAND. Однак його низька щільність даних робить SLC найдорожчим типом NAND.

MLC NAND

Плюси: дешевше, ніж SLC. Мінуси: повільніше і менш витривалий, ніж SLC

NAND багаторівневої комірки (MLC) зберігає кілька біт на клітинку, хоча під терміном MLC зазвичай мають на увазі, що кількість дорівнює 2 бітам на клітинку. MLC має вищу щільність даних, ніж SLC, тому його можна виробляти на більші ємності. MLC має гарне поєднання ціни, продуктивності та витривалості. Однак MLC більш чутливий до помилок даних із 10 000 циклів P/E і, отже, має меншу витривалість порівняно з SLC. MLC зазвичай зустрічається в споживчих продуктах, де витривалість менш важлива.

TLC NAND

Плюси: Найдешевша і велика ємність - Мінуси: Низька витривалість

Трирівневий осередок (TLC) NAND зберігає 3 біти на клітинку. Додаючи більше бітів на клітинку, це зменшує вартість і збільшує ємність. Однак це негативно позначається на продуктивності та витривалості, оскільки має лише 3000 циклів Р/Е. Багато споживчих товарів використовують саме TLC, оскільки це найдешевший варіант.

3D NAND

За останні десять років 3D NAND була однією з найбільших інновацій на ринку флеш-пам'яті. Виробники флеш-пам'яті розробили 3D NAND, щоб виправити проблеми, з якими вони стикалися, зменшивши масштаб 2D NAND, щоб досягти більшої щільності за меншою ціною. У 2D NAND клітинки, які зберігають дані, розміщені горизонтально, поруч. Це означає, що кількість простору, на якому можна розмістити клітини, обмежена, а спроба зменшити клітини зменшує їх надійність. [16]

Отже, для сервера інформаційної системи вибрали SSD TLC через їхню високу швидкість та для зниження вартості впровадження. Для резервного копіювання бази даних використовують HDD диски, оскільки вони більш економічні, а обсяги даних можуть бути значними. На клієнтських комп'ютерах достатньо обладнання з жорсткими дисками, оскільки їм не потрібно зберігати великі обсяги даних, які зберігаються на сервері.

Вибір технічного обладнання здійснено на основі системних вимог:

- 1) Комп'ютер для клієнта: Intel Core i3-10100 3.6 GHz, 4 GB DDR4, 120 GB HDD;
- 2) Комп'ютер для сервера: Intel Xeon Silver 4219 3.2 GHz, 16 GB DDR4, SSD 120 GB, HDD 500 GB;
- 3) Джерело безперебійного живлення, для серверу: LogicPower LPY-W-PSW-2000VA+;
- 4) Монітор: Lenovo ThinkVision E24-28;
- 5) Комплект клавіатура і мишка: Logitech Desktop MK120

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ УЧНІВ ПОЗАШКІЛЬНИХ НАВЧАЛЬНИХ ЗАКЛАДІВ

Розроблено технічне завдання на розробку, що описане в додатку А, а також розроблено програму і методику випробувань – додаток Б.

В інформаційній системі реалізовано декілька типових модулів для керування різними частинам, наприклад, сторінка керівництва з можливістю переглядати дані про усі структури навчального закладу та вносити зміни до даних про студентів, групи, вчителів та інші. Приклади екранних форм зображено у додатку В.

Інструкція користувача

В інструкції описуються основні дії користувача. Область використання інформаційної системи обліку учнів – позашкільні навчальні заклади. Система автоматизує ведення даних про учнів, групи, створення заявок до гуртків, запити на звіти.

Завдяки інтуїтивно зрозумілому інтерфейсу програмного забезпечення, користувач з будь-яким рівнем підготовки зможе легко почати користуватися інформаційною системою. Модулі та функції на сторінках проекту розташовані структуровано, відповідно до розділів. Всі дані зображено у вигляді таблиць, натискання на будь-який рядок відобразить більш детальну інформацію, можливість редагування, збереження та видалення даних. Усі елементи, що пов'язані зі збереженням інформації або її додаванням мають зелений колір, а пов'язані з видаленням – червоний. Також деякі таблиці мають поля для фільтрування. Приклади елементів графічного інтерфейсу, через які виконується управління інформаційною системою представлено у таблиці 3.1.

Таблиця 3.1. Елементи графічного інтерфейсу користувача

№	Зовнішній вигляд	Опис
1		Створення нового запису

2	Прізвище 	Форма пошуку записів за певними критеріями
3	1 Ігор Іванов	Рядок, натиснувши на який відкриється детальніша інформація
4	Обрати дату...	Поле вибору дати
5	Видалити	Кнопка видалення запису
6	Зберегти	Зберегти зміни

Інструкція по формуванню і веденню бази даних

Для підготовки даних спочатку проводиться збір паперових документів, що містять початкові, вже існуючі дані для введення. База даних для інформаційної системи створюється, налаштовується та керується за допомогою графічного клієнту для управління базою даних PostgreSQL - pgAdmin4. Співробітники навчального закладу та керівництво додають дані в систему за допомогою програмного забезпечення, а саме веб-застосунку. Після вводу, дані зберігаються до спільної бази даних, яка використовується для всіх автоматизованих робочих місць та зберігає унікальні та актуальні дані. Редагування та видалення даних можуть виконувати тільки користувачі з відповідним рівнем доступу. Системний адміністратор інформаційної системи та бази даних стежить за роботою бази даних, наявністю вільного місця, стежить за створенням бекапів. Резервне копіювання повинне відбуватися постійно: щоденно, щотижнево, щомісячно та щорічно.

Копіювання, що створюється щоденно, повинно зберігатися тиждень. Щотижневе копіювання зберігатиметься місяць, щомісячні бекапи повинні зберігатися протягом двох років. Щорічні повинні зберігатися постійно.

Підготовка до роботи інформаційної системи

Для початку роботи системи, системний адміністратор повинен ознайомитися з інструкцією по формуванню і веденню бази даних. Після заповнення бази даних та налаштування резервного копіювання, співробітники навчального закладу можуть почати працювати з інформаційною системою використовуючи веб-додаток. Також, для цього потрібно створити облікові записи користувачів, та налаштувати їх ролі, відповідно до яких вони матимуть дозвіл на виконання відповідних дій. Налаштування комп'ютерів та реєстрацію користувачів здійснює системний адміністратор.

Інструкція щодо використання системи

Починаючи роботу, працівник повинен увійти до свого облікового запису, використовуючи свій логін та пароль. Якщо на комп'ютері встановлено пароль безпеки, відходячи від комп'ютера рекомендується блокувати його. Якщо пароль не встановлено, рекомендується виходити з акаунту користувача. Це потрібно для забезпечення безпеки даних, що містить в собі інформаційна система. Після входу, користувач може здійснювати необхідні доступні для нього дії.

В інформаційній системі запроваджено логування (запис дій користувачів), що дозволить швидше знаходити та виправляти недоліки.

Сервер і комп'ютери користувачів повинні мати безперебійне живлення, всі комп'ютери мають бути під'єднані до стабільного Інтернету. Доступ до системи повинні мати тільки співробітники навчального закладу, в системі повинні зберігатися актуальні та коректні дані.

Інструкція на випадок аварійних ситуацій

У випадку тривалої, або критично некоректної роботи програмного забезпечення або технічних засобів, необхідно повідомити про проблему

системному адміністратору, який в залежності від помилки, визначить проблему. Якщо проблема у технічній частині – слід виконати ремонт, віднести на сервісне обслуговування або купити нову. У випадку, якщо проблема відбулась на стороні серверу, у програмному забезпеченні, необхідно звернутися до розробника, що відповідає за підтримку відповідного програмного забезпечення. Розробник знайде спосіб рішення цієї помилки та надає виправлений код або інше рішення. У випадку небажаного втручання до інформаційної системи, порушення цілісності бази даних, втрату даних, системний адміністратор повинен відновити дані в базі даних за допомогою одного з останніх бекапів, що зберігаються. Під час проведення відновлення даних, користувачам забороняється користуватися інформаційною системою, про що повинен повідомити системний адміністратор.

Отже, завдяки реалізації даного проекту, підвищується ефективність роботи співробітників позашкільного навчального закладу, а також зменшується час, що витрачається на типову роботу, за рахунок раціонального використання часу та трудових ресурсів закладу.

РОЗДІЛ 4. РОЗРАХУНОК ЕКОНОМІЧНОГО ЕФЕКТУ ВІД ВПРОВАДЖЕННЯ ІС

4.1 Вступ до проблеми

Інформаційні системи широко використовуються багатьма організаціями України різної форми власності. Ці системи розповсюджуються все більше і більше, змінюючись відповідно до вимог замовників та модернізуючись з плином часу.

Інформаційні системи освітніх установ не є цілком новим винаходом, але як концепція розробки інформаційних систем та надання послуг у галузі ІТ вони тільки починають свій розвиток. Немає сумніву, що з удосконаленням цієї технології все більше організацій, які займаються освітньою діяльністю, застосовуватимуть інформаційні системи для побудови своєї комп'ютерної інфраструктури, виконання розрахункових завдань та іншого.

Для створення інформаційної системи необхідно витратити кошти на її проектування та закупівлю відповідного обладнання, а також на утримання підсистеми в робочому стані. Економічна вигода від використання системи залежить від витрат на її обслуговування. Співвідношення економічної вигоди до витрат на розробку системи показує економічну доцільність капітальних інвестицій. Економічні показники розраховуються за цінами, тарифами та ставками зарплати, які діють на момент обчислення.

4.2 Розрахунок вартості створення інформаційної системи управління контентом web-порталу

Загальні витрати (на проектування, реалізацію та впровадження системи)

$$З = C_{п-р} + C_{тз} + C_{пз} + C_{впр},$$

де $C_{п-р}$ – витрати на проектування і реалізацію;

$C_{\text{тз}}$ – витрати на купівлю, або модернізацію комп'ютерної техніки, різноманітних периферійних пристроїв, засобів комунікації, допоміжного обладнання та оргтехніки (з урахуванням витрат на перевезення, доставлення, монтаж, налагодження та впровадження), виробничо-господарських інструментів;

$C_{\text{пз}}$ – витрати на придбання та установку програмного забезпечення (ПЗ), необхідного для розробки;

$C_{\text{впр}}$ – витрати на первинний запуск системи (навчання персоналу тощо).

Витрати на проектування та реалізацію розраховуються за формулою

$$C_{\text{п-р}} = C_3 + C_{\text{від}} + C_{\text{м}} + C_{\text{н}},$$

де C_3 – заробітна плата розробників (програмістів бекенду, фронтенду, дизайнерів UI, тестувальників тощо);

$C_{\text{від}}$ – відрахування на соціальні заходи (які становлять 47,5% від фонду оплати праці, відповідно до законодавства);

$C_{\text{м}}$ – експлуатація та амортизація, $C_{\text{м}} = Z_{\text{експл}} + Z_{\text{амор}}$;

$C_{\text{н}}$ – 50-150% від витрат на оплату праці становлять накладні витрати у розмірі (охоплюють оплату праці керуючого персоналу з нарахуваннями; оплату відряджень пов'язаних з виконанням робіт; оплату консультаційно-інформаційних послуг; оплату ремонту і технічного обслуговування основних фондів, крім персональних комп'ютерів (ПК та ноутбуків)).

1) Розробка підсистеми вимагає 2 місяці (експертна оцінка, яка базується на досвіді розробки систем такого класу). Заробітна платня розробників за місяць дорівнює 12000 грн (відповідно до сучасних розцінок та попиту на спеціалістів такого профілю). Кількість розробників – одна людина.

$$C_3 = 1 * 18000 * 2 = 36000 \text{ грн.}$$

$$C_{\text{від}} = 36000 * 0,475 = 17100 \text{ грн.}$$

2) Витрати на амортизацію ЕОМ, на яких виконується розробка:

$$Z_{\text{амор}} = (C_{\text{еом}} / T_{\text{сл}}) * (T_{\text{разр}} / 12),$$

де $C_{\text{еом}} = 30000$ грн. – балансова вартість ЕОМ;

$T_{\text{сл}} = 5$ років – термін служби ЕОМ;

$T_{\text{разр}} = 0,5$ року – час, необхідний для створення системи, виражений в місяцях, і тому ділиться на 12, щоб одержати час, що виміряний в роках.

$Z_{\text{амор}} = (20000/5) * 0,5 = 2000$ (грн.);

3) витрати на експлуатацію персонального комп'ютера, на якому виконується розробка, складаються з оплати споживаної ним електричної енергії:

$Z_{\text{експл}} = P_{\text{ЕОМ}} * T_{\text{разр}} * N_{\text{рд}} * N_{\text{рч}} * E_{\text{ст}}$,

де $P_{\text{ЕОМ}} = 0,6$ кВт/год – потужність персонального комп'ютера;

$T_{\text{разр}} = 2$ місяці – тривалість розробки;

$N_{\text{рд}} = 21$ день – кількість робочих днів у місяці;

$N_{\text{рч}} = 8$ годин – кількість годин у робочому дні;

$E_{\text{ст}} = 1,8$ грн – ціна за 1 кВт електроенергії

$Z_{\text{експл}} = 0,6 * 2 * 21 * 8 * 1,8 = 362,88$ (грн.)

$C_{\text{м}} = 2000 + 362,88 = 2362,88$ (грн.)

$C_{\text{н}} = 36000 * 0,5 = 18000$ (грн.)

$C_{\text{п-р}} = 36000 + 17100 + 2362,88 + 18000 = 73362,88$ (грн.)

Витрати на ліцензійне програмне забезпечення не враховуються у даному розрахунку, оскільки в роботі системи буде використовуватись вільне, безкоштовне ПЗ з відкритими вихідними кодами. Витрати на мережеве та серверне обладнання – апаратне забезпечення складають 18594 грн (придбання серверу та комутаційного обладнання і матеріалів). Для запровадження системи потрібно враховувати витрати на освіту персоналу, витрати на встановлення та налаштування мережі, вартість обладнання, вартість програмного забезпечення.

Витрати на впровадження системи приведені в таблиці 4.1.

Таблиця 4.1 – Витрати на впровадження системи.

Назва витрати	Сума, грн
Вартість обладнання	18594,00

Вартість програмного забезпечення	-
Вартість підготовки кадрів	500,00
Монтаж і налаштування мереж	300,00
Разом на впровадження системи	19394,00

Таким чином, загальні витрати на проектування, кодування, реалізацію та впровадження системи складають:

$$З = C_{п-р} + C_{тз} + C_{пз} + C_{вп}p = 73362,88 + 19394 = 92756,88 \text{ (грн.)}$$

Витрати на розробку та експлуатацію системи в перший рік експлуатації

$$З_{се} = З + З_{ер},$$

де $З_{ер}$ – експлуатаційні витрати на апаратне (технічне) забезпечення за рік.

$$З_{ер} = A_{об} + З_{м} + З_{е},$$

де $A_{об}$ – амортизаційні відрахування на обладнання (устаткування);

$З_{м}$ – витрати на основні і допоміжні матеріали;

$З_{е}$ – витрати на електроенергію.

Амортизаційні відрахування на обладнання (устаткування) становлять 60% балансової вартості в рік (відповідно до діючого законодавства)

$$A_{об} = 19394 * 0,6 = 11636,4 \text{ (грн.)}$$

В масштабі підприємства річні витрати на основні та допоміжні матеріали (флеш-накопичувачі, тонер для принтера, папір тощо) визначаються в розмірі 3% вартості основного устаткування

$$З_{м} = 19394 * 0,03 = 581,82 \text{ (грн.)}$$

Річний обсяг роботи персонального комп'ютера у нормо-годинах

$$\Phi_{м} = 253,3 * T_{с},$$

де $T_{с}$ – середнє добове завантаження обладнання (близько 5 годин); 253,3 – середня кількість робочих днів у році.

$$\Phi_{м} = 253,3 * 5 = 1266,5 \text{ (годин.)}$$

Витрати на електроенергію під час 1266,5 годин роботи устаткування в рік при потужності ЕОМ 0,6 кВт/год та ціні 1,8 грн за 1 кВт електроенергії складуть

$$З_е = 1266,5 * 1,8 * 0,6 = 1367,82 \text{ (грн.)}$$

Експлуатаційні витрати на технічне забезпечення за рік

$$З_{ер} = 11636,4 + 581,82 + 1367,82 = 13585,64 \text{ (грн.)}$$

Отже, у перший рік витрати на створення та експлуатацію системи складуть

$$З_{се} = 92756,88 + 13585,64 \approx 106342,52 \text{ (грн.)}$$

4.3 Розрахунок економічної ефективності

Економічна ефективність використання інформаційної системи полягає у наступному:

зменшення часу на обслуговування системи за рахунок її відмовостійкості та автоматичного масштабування;

можливість надання користувачам інформації в оренду.

Це зменшення відбудеться за рахунок зменшення складності користування системою, збільшення надійності, стабільності та рівня безпеки. Визначимо, що це зменшення складе 50% від сучасного.

Однак, не піддається прямій грошовій оцінці зменшення кількості помилкових та необережних рішень, підвищення оперативності керування, поліпшення організації роботи та своєчасне отримання необхідної інформації про пріоритетність варіантів рішень тощо.

Використання даної системи дозволяє вивільнити 0,1 робочого часу. Так як з системою працює біля 50 користувачів, то система умовно вивільнить 5 робітників.

Визначимо пряму економічну ефективність, ґрунтуючись на тому, що впровадження системи вивільнить 5 робітників. Річна платня за рахунок вивільнення робочого часу складе

$$\Delta C_n = 5000 * 5 * 12 - 12674,16 = 287325,84 \text{ (грн.)}$$

Річний економічний ефект розраховується за наступною формулою:

$$E_{год} = \Delta C_n - E_n * k,$$

де E_n – нормативний коефіцієнт економічної ефективності капітальних вкладень у галузь і для обчислювальної техніки приймається 0,5;

k – додаткові капітальні вкладення з урахуванням витрат на проектування, створення і функціонування системи.

$$E_{\text{год}} = 287325,84 - 0,5 * 106342,52 = 234154,58 \text{ (грн.)}$$

Строк окупності системи

$$T = k / \Delta C_n$$

Тобто

$$T = 106342,52 / 234154,58 = 0,45 \text{ (року)}$$

Таким чином, витрати на створення інформаційної системи позашкільного навчального закладу окупляться протягом 0,45 року.

4.4 Підсумки

Проведені розрахунки показали, що витрати на створення системи позашкільного навчального закладу (проектування, реалізація і впровадження) складуть 92756,88 грн; експлуатаційні витрати щорічно складатимуть 234154,58 грн. Загалом в перший рік витрати на модернізацію й експлуатацію складуть 106342,52 грн.

Вивільнюється 5 умовних фахівців.

Вивільнена річна зарплата працівників складає 287325,84 грн.

Витрати на створення системи окупляться протягом 0,45 року (5,5 місяців).

За отриманими даними про річний економічний ефект та термін повернення вкладень можна зробити висновок про економічну доцільність і виправданість розробки інформаційної системи. Варто відзначити, що економічну ефективність цієї системи можна підвищити, пропонуючи її як продукт іншим користувачам. Тоді можна буде зменшити час, за який витрати на розробку системи окупляться, а також отримати прибуток від продажу продукту.

РОЗДІЛ 5. ОХОРОНА ПРАЦІ ТА ТЕХНІКА БЕЗПЕКИ

Вступ

Охорона праці визначається як система заходів, спрямованих на забезпечення безпеки та здоров'я працівників під час виконання роботи. Ця система включає в себе різноманітні правові, соціально-економічні, організаційні, технічні, гігієнічні та медично-профілактичні заходи.

Основною метою охорони праці є мінімізація ризиків травм та професійних захворювань через впровадження відповідних процесів у сфері діяльності підприємства, що впливають на безпеку праці та здоров'я працівників.

Працівник, виконуючи свою роботу, стикається з різними факторами, які можуть впливати на його здоров'я та працездатність. Виробничі фактори поділяються на небезпечні та шкідливі в залежності від наслідків їхньої дії на працівника.

Небезпечний фактор призводить до травми або різкого погіршення здоров'я працівника, тоді як шкідливий фактор може спричинити захворювання або зниження рівня працездатності.

Ці фактори можуть бути фізичними, хімічними, біологічними або психофізіологічними за своєю природою.

Техніка безпеки включає в себе систему технічних засобів, правил та умов, які гарантують безпечні умови праці та запобігають небезпеці під час виконання роботи. Дотримання вимог техніки безпеки є важливим заходом для забезпечення безпеки працівників.

Керівництво підприємств повинно вживати відповідних заходів з техніки безпеки, враховуючи встановлені правила та норми безпеки. Порушення цих вимог може призвести до адміністративної або кримінальної відповідальності згідно з законодавством про охорону праці в Україні, визначеним Конституцією та відповідним Законом.

Статтею 43 Конституції України зазначено: «Кожен має право на належні, безпечні та здорові умови праці, на заробітну плату, не нижчу від визначеної законом».

Законом «Про охорону праці» визначаються основні положення щодо реалізації конституційного права працівників на охорону їх життя і здоров'я у процесі трудової діяльності, на належні, безпечні і здорові умови праці, регулює за участю відповідних органів державної влади відносини між роботодавцем і працівником з питань безпеки, гігієни праці та виробничого середовища і встановлює єдиний порядок організації охорони праці в Україні. [17].

Будь-яке підприємство повинно прагнути створювати комфортні умови для роботи своїх працівників, особливо щодо чистоти повітря у виробничих приміщеннях. Основні шкідливі виробничі фактори включають хімічні речовини та промисловий пил, які можуть бути розділені на кілька категорій, таких як промислові отрути, сільськогосподарські отрутохімікати, побутова хімія, препарати лікувального призначення та отруйні речовини.

Промисловий пил, який є проблемою для деяких виробництв, може негативно впливати на здоров'я шляхом сприяння розвитку захворювань легень. Таким чином, забезпечення захисту від цих шкідливих факторів виробничого середовища є надзвичайно важливим завданням.

Незважаючи на всі заходи, спрямовані на мінімізацію впливу шкідливих факторів, досягнення ідеальних умов праці завжди залишається неможливим у зв'язку з особливостями технологічних процесів та виробництва. Тому для керівництва підприємств важливо пріоритетно зосереджуватися на:

- Усуненні небезпечних факторів або зниженні ризику їхнього впливу;
- Використанні максимально безпечних методів роботи;
- Боротьбі з небезпечними факторами та їх джерелами;
- Забезпеченні ефективного використання засобів індивідуального захисту, якщо це необхідно.

За статистикою Управління Держпраці, за час незалежності України травмувалося понад 925 тисяч людей, з них загинуло більше 33 тисяч осіб. Отже, охорона праці є постійно актуальним питанням, особливо враховуючи постійне впровадження нових технологій на підприємствах.

Важливо, щоб викладачі та інші працівники освітніх закладів були ознайомлені з правилами та стандартами охорони праці, враховуючи проведення обов'язкових навчальних інструктажів та курсів для підвищення рівня обізнаності в цій галузі.

Існуючі види інструктажів з охорони праці приведено на таблиці 5.1.

Таблиця 5.1. Види інструктажів з охорони праці

Вид інструктажу	Умови проведення інструктажу
Вступний	Проводиться для осіб, які вперше виходять на разову або штатну роботу в навчальному закладі, а також для осіб, які приймають участь в навчальному процесі або виконують інші доручення підприємства.
Первинний	Проводиться для осіб, прийнятих на роботу до навчального закладу, переведених на іншу посаду в тому самому закладі, функціональні обов'язки яких змінено, а також тих, які перебувають у закладі за програмою обміну досвіду або у відрядженні.
Повторний	Проводиться для окремого співробітника або для групи співробітників, службові

	обов'язки яких потребують постійного оновлення знань.
Позаплановий	Здійснюється у випадку набрання чинності новими регуляторними актами з охорони праці, при здійсненні модернізації робочого процесу, початком роботи з новим обладнанням, при поновленні на роботу після тривалої перерви.
Цільовий	Виконується з метою роз'яснення порядку боротьби зі стихійним лихом або для усунення наслідків аварій, а також для осіб, на яких складається наказ, розпорядження.

5.1 Аналіз небезпечних і шкідливих факторів в позашкільному навчальному закладі

Ефективний аналіз та дослідження виробничого травматизму та захворювань, що характерні для певних професій, можливі лише при детальному розгляді конкретних причин їх виникнення. Ці причини традиційно поділяють на кілька груп: організаційні, технічні, санітарно-гігієнічні, економічні та психофізіологічні.

Виявлення та вирішення проблем, пов'язаних із забезпеченням безпечних умов праці, є однією з ключових задач при розробці нових технологій та систем виробництва. Вивчення та визначення можливих причин нещасних випадків, професійних захворювань та інших негативних ситуацій на підприємстві дозволяють розробляти відповідні заходи та вимоги для їх усунення.

У вчителів позашкільного навчального закладу та адміністрації, які використовують інформаційну систему, пов'язану з комп'ютером, існує значна кількість шкідливих факторів, які можуть впливати на їхню продуктивність. Серед цих факторів:

1. Підвищений рівень шуму може призводити до зниження уваги та гостроти чуттів.
2. Недостатня або надмірна освітленість може спричинити напругу на зір та збільшити ймовірність передчасної стомленості.
3. Правильне організоване освітлення робочого місця має велике значення для підтримки комфортних умов праці та підвищення ефективності.
4. Ризик ураження електричним струмом може виникнути внаслідок неправильної електричної проводки.
5. Високий рівень інфрачервоного випромінювання може призводити до хвороб шкіри.
6. Некоректна організація робочого місця може спричинити незручність та фізичні проблеми для працівників.

Досягнення оптимальних умов праці вимагає уважного вивчення та врахування цих факторів з метою поліпшення комфорту та безпеки працівників.

Під час щоденної роботи за комп'ютером, при недотриманні принципів ергономіки, санітарних вимог і правильного режиму роботи, може виникнути так званий "ергономічний" стан, що визначається як нове захворювання, пов'язане переважно з впливом комп'ютерів на здоров'я користувачів. Серед таких захворювань, зазначених Всесвітньою організацією охорони здоров'я, входять синдром комп'ютерного стресу оператора, травми від повторюваних навантажень і фотоепілептичні приступи.

Державні санітарні правила встановлюють певні вимоги, серед яких:

1. Загальне освітлення повинно бути здійснене за допомогою суцільних або переривчастих ліній світильників.

2. Яскравість великих поверхонь (вікна, світильники тощо), які знаходяться у полі зору, не повинна перевищувати 200 кд/кв. м.

3. Відеомонітори на електронно-променевих трубках можуть бути джерелами електромагнітного випромінювання та статичного електричного поля.

4. Температура повітря в кабінетах, де використовуються комп'ютери, повинна бути 19.5 ± 0.5 град. С, а відносна вологість повітря - $60 \pm 5\%$, і швидкість руху повітря - не більше 0.1 м/с.

5. У кабінетах, де проводиться навчання з використанням комп'ютерів, потрібен 3-кратний обмін повітря за 1 годину, а для охолодження та очищення повітря можуть встановлюватися побутові кондиціонери з позитивним гігієнічним висновком від ГСЕУ МОЗ.

6. Візуальні ергономічні параметри відеомоніторів повинні відповідати вимогам ДСанПіН при проектній відстані спостереження від 400 до 800 мм та при зовнішній освітленості екрана до 250 Лк.

5.2 Розробка заходів щодо зменшення впливу виробничих факторів на працюючого в позашкільному навчальному закладі.

Робоче місце – це простір, обладнаний необхідними засобами, призначений для виконання трудової діяльності працівника або групи працівників, які спільно займаються певною роботою. Його функції та розташування визначаються згідно з трудовими нормативно-правовими актами.

Щоб попередити або усунути причини виробничого травматизму та професійної захворюваності, застосовують технічні та організаційні заходи. До технічних заходів належать розробка та впровадження безпечного устаткування, автоматизація та механізація технологічних процесів, використання автоматичних блокувальних засобів, запобіжних пристосувань та правильне розташування органів керування обладнанням. Організаційні заходи включають в себе правильну організацію роботи, навчання, систему

планового обслуговування обладнання, контроль та нагляд за охороною праці, впровадження безпечних методів та наукової організації праці, дотримання трудового законодавства та проведення заходів із популяризації охорони праці.

Мікрокліматичні умови на робочому місці можуть підтримуватися за допомогою систем опалення та кондиціонування. Важливо, щоб повітря від них не направлялося безпосередньо на працівників чи учнів. Робоче місце та його обладнання повинні відповідати фізичним, антропометричним та психологічним вимогам, а також забезпечувати достатній робочий простір для зручних рухів.

Для мінімізації стомлення працівника під час сидячої роботи за комп'ютером важливо мати всі необхідні засоби у доступній йому зоні.

На робочому місці велика увага приділяється організації стола, що повинен відповідати певним вимогам для забезпечення комфортної та ефективної роботи:

Висота стола: Обирається з урахуванням можливості комфортного сидіння в зручній позі, і при необхідності, опираючись на підлокітники. Столи з регульованою висотою стають популярнішими.

Поверхня стола: Має уникати відблисків та віддзеркалювання світла для збереження поле зору робітника.

Ящики для документів: Конструкція стола повинна передбачати наявність ящиків для зберігання документації.

Рекомендована висота столу: Знаходиться у межах 680-760 мм, а висота поверхні для клавіатури - близько 650 мм.

Окрім того, особлива увага приділяється організації робочого крісла:

Висота сидіння: Повинна бути в межах 420-550 мм вище рівня підлоги.

Поверхня сидіння: М'яка для забезпечення комфорту.

Закруглений передній край: Забезпечує зручне сидіння.

Регульований кут нахилу спинки: Дозволяє користувачеві налаштувати його зручно.

Щодо монітора, його екран повинен мати вірне положення відносно відстані зчитування (0.6-0.7 м) та кута зчитування, який повинен бути на 20 градусів нижче горизонталі до центру екрана.

Згідно з нормами, робоче місце повинно бути розташоване на відстані не менше 1.5 м від стіни з вікнами і 1 м від інших стін. Також не рекомендується розміщувати робоче місце прямо напроти джерела світла.

Щоб зайняти правильну робочу позу при роботі з комп'ютером, слід виконати кілька дій: відрегулювати висоту робочого столу (якщо це можливо), крісла та підставки для ніг (якщо вони є). Найзручнішою вважається поза, при якій ступні розташовані прямо під кутом до гомілок та паралельно підлозі або підставці для ніг, стегна горизонтальні, верхня частина рук вертикальна. Кут згину ліктя повинен бути в межах 70-90°, зап'ястя повинні бути зігнуті під кутом не більше 20°, а голова повинна нахилитися приблизно на 15-20°.

Перед початком роботи необхідно враховувати ряд вимог безпеки, таких як:

1. Яскравість екрана монітору повинна бути не менше 100 кд/м²;
2. Мінімальний розмір точки свічення – не більше 0.4 мм для монохромного монітора та не менше 0.6 мм для кольорового, контрастність зображення – не менше 0.8;
3. При виявленні несправностей розпочинати роботу заборонено.

Під час роботи слід дотримуватися наступних вимог безпеки:

1. Уникати подразнюючих шумів;
2. Сидіти відповідно до робочого місця.

При завершенні роботи важливо дотримуватися наступних вимог:

1. Зберегти усі файли, що перебували в роботі;
2. Вимкнути кондиціювання або опалення, а також освітлення;
3. Прибрати робоче місце відповідно до правил обслуговування обладнання.

Отже, охорона праці в позашкільних навчальних закладах передбачає виконання вимог щодо проектування робочого місця, ергономічних стандартів для робочих місць та дотримання вимог режиму роботи працівника. Створення

комфортних умов праці та правильно організоване робоче середовище сприяють полегшенню праці та зменшенню впливу шкідливих факторів, що покращує продуктивність та ефективність працівників.

РОЗДІЛ 6. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ до проблеми

Захист навколишнього середовища реалізується за допомогою різних, зокрема й юридичних, методів. При цьому правові форми охоплюють переважно всі складові, які становлять природне середовище. Основними сучасними нормативно-правовими документами, які встановлюють основи організації захисту навколишнього природного середовища, є Закони України «Про охорону навколишнього природного середовища» від 25 червня 1991 р., «Про охорону атмосферного повітря» від 16 жовтня 1992 р., «Про природно-заповідний фонд України» від 16 червня 1992 р., «Про тваринний світ» від 3 березня 1993 р., «Про карантин рослин» від 30 червня 1993 р та інші. Крім того, деякі відносини у сфері використання і захисту навколишнього природного середовища регулюються кодексами (земельним, водним, лісовим, про надра), а також Законами України «Про плату за землю» від 3 липня 1992 р., «Про ветеринарну медицину» від 25 червня 1992 р. Вирішальне значення у цьому питанні має схвалений Постановою Верховної Ради «Порядок обмеження, тимчасової заборони (зупинення) чи припинення діяльності підприємств, установ, організацій і об'єктів у разі порушення ними законодавства про охорону навколишнього природного середовища».

Право природокористування має такі види: право використовувати землю, воду, ліси, надра, тваринний світ, природно-заповідний фонд. Право природокористування - це процес розумного використання людиною природних ресурсів для задоволення різних потреб та інтересів. Основними принципами природокористування є його цілеспрямованість, плановість і довготривалість, ліцензування, врахування надзвичайного значення у житті суспільства та ін. При цьому розрізняють такі групи природокористування, як право загального і спеціального використання землі, вод, лісів, надр, тваринного світу та інших природних ресурсів. Суб'єктами права загального користування природними

ресурсами можуть бути, згідно з Законом України «Про охорону навколишнього природного середовища», усі громадяни для задоволення найрізноманітніших потреб і інтересів. Воно здійснюється громадянами безкоштовно і безліцензійно, тобто для цього не потрібен відповідний дозвіл уповноважених органів і осіб. Загальним є, наприклад, використання парків, скверів, водойм, лісів, збір дикорослих ягід, грибів, горіхів і т. ін. Право загального природокористування закріплене у ст. 13 Конституції України: «Кожний громадянин має право користуватися природними об'єктами права власності народу відповідно до закону». Спеціальне використання природних ресурсів є похідним від загального природокористування. На відміну від першого, це використання конкретних природних ресурсів, що здійснюється громадянами, підприємствами, установами і організаціями у випадках, коли відповідна, визначена у законодавстві частина природних ресурсів передається їм для використання. Як правило, така передача має вартість і визначена в часі. Надання природних ресурсів відбувається на основі спеціальних дозволів державних актів на право постійного користування.

Сучасна юридична наука розробляє і розвиває екологічні права і обов'язки, крім прав суб'єктів, як користувачів природи. Так, у Конституції України записано, що «кожен має право на довкілля, яке безпечне для життя і здоров'я, та на компенсацію шкоди, яка завдана порушенням цього права. Кожному гарантується право вільно отримувати і поширювати інформацію про стан довкілля, про якість харчових продуктів і предметів побуту». Подібні положення є й у Законі України «Про охорону навколишнього природного середовища», бо це право одне з основних прав людини. Держава зобов'язана забезпечувати виконання санітарно-гігієнічних заходів, які сприяють поліпшенню та оздоровленню навколишнього природного середовища, що відповідає цьому праву. Усі екологічні права громадян захищаються і відновлюються у судовому порядку. Закон України «Про охорону навколишнього природного середовища» передбачає і певні обов'язки громадян поряд з правами. Так, незалежно від того, є громадяни користувачами природи, чи ні, вони зобов'язані охороняти природу,

раціонально використовувати її запаси, не завдавати шкоди. Крім того, Закон України «Про охорону навколишнього природного середовища» покладає на громадян і підприємства, установи й організації, як суб'єктів спеціального використання природних ресурсів, спеціальні обов'язки. Так, плата за спеціальне користування природою встановлюється на основі нормативів плати і лімітів використання природних ресурсів. Ці нормативи визначаються з урахуванням розповсюдження природних ресурсів, їх якості, можливості використання, місцезнаходження, можливості переробки і зберігання відходів. До того ж суб'єкти спеціального користування природою зобов'язані сплачувати певні кошти за забруднення навколишнього природного середовища, що встановлюються за викиди у атмосферу забруднюючих речовин; скидання забруднюючих речовин на поверхню води, у територіальні і морські води, а також під землю. Контроль у сфері природокористування і охорони навколишнього природного середовища здійснюється шляхом перевірки, нагляду, обстеження, інвентаризації та експертиз. Він може здійснюватися як уповноваженими державними органами, так і громадськими формуваннями. Державний контроль покладається на Ради народних депутатів, державні адміністрації та Міністерство охорони навколишнього природного середовища і його органи на місцях.

Основні пріоритети у сфері охорони довкілля та раціонального використання природних ресурсів включають:

- Забезпечення безпеки ядерних об'єктів і захист населення від радіації, зменшення впливу Чорнобильської катастрофи.
- Покращення стану річкових басейнів та якості питної води в Україні.
- Стабілізація екологічної ситуації у містах та промислових центрах Донецько-Придніпровського регіону.
- Розбудова нових та модернізація існуючих очисних споруд для каналізації.
- Попередження забруднення Чорного та Азовського морів, поліпшення їхнього стану.

- Формування балансованої системи природокористування та екологізація виробничого сектору у промисловості, енергетиці, будівництві, сільському господарстві та транспорті.
- Збереження біорізноманіття та ландшафтів, розвиток заповідної справи.

Для досягнення цих цілей передбачається:

- Зменшення радіаційного забруднення до мінімуму.
- Захист повітря від забруднення, особливо у великих містах та промислових центрах.
- Збереження та захист земельних ресурсів від забруднення та нерационального використання.
- Розвиток територій з природними ландшафтами, підвищення охоронної діяльності у заповідних та рекреаційних зонах.
- Покращення стійкості та екологічних функцій лісів.
- Обробка, утилізація та утилізація промислових та побутових відходів.
- Попередження забруднення водних об'єктів, захист підземних вод від забруднення.
- Збереження та відновлення річок, управління водними ресурсами за басейновим принципом.
- Завершення створення системи моніторингу природного середовища.
- Розробка механізмів природокористування.
- Впровадження ефективних економічних інструментів у природокористування.
- Впровадження системи екологічної освіти, виховання та інформування.

6.2 Забруднення навколишнього середовища підприємством з розробки програмного забезпечення

Фірма в основному забруднює довкілля через використання автотранспорту. Це спричиняє забруднення повітря, а інші аспекти її діяльності

не впливають на якість води або ґрунту. У сучасних умовах автомобільний транспорт є основним джерелом забруднення атмосфери, особливо у великих містах. Міське повітря постійно забруднюється, що призводить до його деформації та змін клімату. Міста характеризуються високим рівнем антропогенних навантажень на довкілля, що призводить до зміни його властивостей, включаючи мікрокліматичні та фізико-хімічні параметри. Температура повітря в містах у середньому вища, ніж за межами міст, і там частіше спостерігаються атмосферні опади, тумани та смоги, що спричинені викидами вихлопних газів та інших забруднень. Атмосфера в містах менш прозора, а тумани та запиленість повітря призводять до зменшення проникнення сонячних променів до землі.

Проте через автомобільні викиди якість повітря в містах погіршилась.

Забруднення атмосфери в містах зростає через високий внесок автотранспорту. Подекуди кожне збільшення на дорозі на одиницю автотранспортного засобу спричиняє викид більше забруднюючих речовин. Недавно обсяг оксидів вуглецю, вуглеводнів, оксидів азоту та сажі в міському повітрі зріс. Однак найбільша загроза, окрім оксидів азоту, становлять сірчані та свинцеві сполуки, які значно зросли у міському повітрі. Міста не готові до такого об'єму автотранспорту: пробіг без зупинок між світлофорами складає лише 400–600 метрів, що призводить до зниження середньої швидкості руху до 12–20 км/год, що в свою чергу збільшує витрати пального в 3–4 рази. Це призводить до збільшення викидів забруднюючих речовин. Автотранспорт сприяє специфічним формам забруднення повітря: під час руху гума зношується, утворюючи пил, який потрапляє в атмосферу. Він не лише забруднює повітря від продуктів згоряння палива, але й сприяє надмірному викиду свинцю у навколишнє середовище. Україна використовує бензин з вмістом свинцю 0,36

г/л, в той час як в країнах, таких як Великобританія, Німеччина та США, ця цифра становить 0,013–0,15 г/л.

6.3 Розробка заходів щодо зменшення забруднення

Природоохоронна діяльність - це будь-яка діяльність, яка має на меті зберегти якість навколишнього середовища на такому рівні, що забезпечує стабільність біосфери. Вона включає як масштабну, проводиму на державному рівні діяльність по охороні природних зразків і різноманітності видів на Землі, організацію наукових досліджень, підготовку фахівців екологів і освіту населення, так і діяльність окремих підприємств по очищенню від шкідливих речовин стічних вод і відхідних газів, зменшенню норм використання природних ресурсів і т.д. Така діяльність здійснюється переважно інженерними методами. Існують два головних напрямки природоохоронної діяльності підприємств. Перший - очищення шкідливих викидів. Цей шлях “у чистій формі” малоефективний, тому що з його допомогою далеко не завжди можна повністю припинити надходження шкідливих речовин у біосферу. Крім того, зменшення рівня забруднення одного компонента навколишнього середовища призводить до збільшення рівня забруднення іншого. Наприклад, встановлення вологих фільтрів при газоочищенні дозволяє зменшити забруднення повітря, але призводить до ще більшого забруднення води. Затримані з відхідних газів і зливальних вод речовини часто отруюють значні земельні площі. Використання навіть найефективніших очисних споруд сильно знижує рівень забруднення навколишнього середовища, але не вирішує цю проблему цілком. У процесі їх роботи утворюються відходи, незважаючи на їх менший обсяг, які мають підвищену концентрацію шкідливих речовин. Більшість таких споруд потребують значних енергетичних витрат, що також може становити небезпеку для навколишнього середовища. Крім того, забруднювачі, на знешкодження яких йдуть величезні засоби, являють собою речовини, на які вже витрачена

праця і які за рідкісним винятком можна було б використовувати в народному господарстві.

Для досягнення високих еколого-економічних показників необхідно поєднати очищення шкідливих викидів з утилізацією зовлених речовин, що дозволить об'єднати ці дві сфери.

Щодо другого напрямку, вирішення причин забруднення потребує створення технологій виробництва з мінімальним утворенням відходів. Наразі не для всіх виробництв знайдено ефективні технічно-економічні рішення щодо значного скорочення відходів та їх утилізації, тому доводиться працювати в обох напрямках.

Щоб покращити заходи з охорони навколишнього середовища, важливо розуміти, що жодні очисні споруди чи технології не зможуть повністю відновити стійкість біосфери, якщо будуть перевищені припустимі межі змін природних систем, що людина не перетворювала. Це підкреслює важливість збереження природних систем для біологічної різноманітності та стабільності екосистем.

Це може бути виявлено шляхом використання більше 1% енергії біосфери і глибокими перетвореннями більше 10% природних територій (правило одного і десяти відсотків). Тому технічні досягнення не вирішують потребу у переосмисленні пріоритетів суспільного розвитку, стабілізації населення, створення заповідних територій та інших, згаданих раніше.

Основними напрямками у захисті атмосфери від викидів автотранспорту є:

- а) розробка та виробництво автомобілів з економічними та екологічно безпечними двигунами, включаючи дизельні;
- б) розвиток ефективних систем нейтралізації відпрацьованих газів;
- в) зменшення токсичності моторних палив;

г) удосконалення організації руху транспорту в містах та покращення дорожнього будівництва для неперервного руху на автошляхах.

ВИСНОВОК

У дипломній роботі розроблено концепцію інформаційної системи позашкільного навчального закладу. При розробці було проведено аналіз предметної області позашкільного навчального закладу, інформаційні процеси та структуру закладу. Сформовано задачі, які дозволяють реалізувати модулі управління учнями, вчителями, групами, заявками до груп та запитами на звіт. На основі сформованих вимог розроблено концепцію інформаційної системи. Було розроблено технічний проект інформаційної системи обліку учнів позашкільних навчальних закладів, а також визначено загальносистемні рішення, рішення з інформаційного, програмного та технічного забезпечення. Реалізовано проект інформаційної системи, створено документацію та коротку інструкцію користувача, а також інструкції з формування і ведення бази даних, початку роботи з інформаційною системою та у аварійних випадках, приведено приклади форми вводу даних та окремих модулів керування системою. Проведено аналіз небезпечних та шкідливих факторів в позашкільному навчальному закладі, а також розроблено заходи і засоби по зменшенню впливу небезпечних та шкідливих факторів та техніка безпеки на робочому місці під час виконання трудових обов'язків. Проведені випробування, за підсумками яких було помічено підвищення ефективності роботи та зменшення часу, що витрачається на типову роботу вчителями позашкільного навчального закладу, за рахунок більш раціонального використання часу та трудових, матеріальних ресурсів, а також автоматизації інформаційних процесів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Заклад позашкільної освіти [Електронний ресурс]: Вікіпедія. Вільна енциклопедія:
https://uk.wikipedia.org/wiki/Заклад_позашкільної_освіти
2. Офіційний сайт Spring Framework [Електронний ресурс]:
<https://spring.io>
3. Design Patterns: Elements of Reusable Object-Oriented Software by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides: Addison Wesley Professional – 1994.
4. Робоча навчальна програма спецкурсу автоматизоване робоче місце - Білоцерківський інститут неперервної професійної освіти, 2021 -
<https://tinyurl.com/arm-programma>
5. Надійність [Електронний ресурс]: Вікіпедія. Вільна енциклопедія:
<https://uk.wikipedia.org/wiki/Надійність>
6. Проектування бази даних [Електронний ресурс]: Вікіпедія. Вільна енциклопедія: https://uk.wikipedia.org/wiki/Проектування_бази_даних
7. PostgreSQL [Електронний ресурс]: <https://www.postgresql.org/about/>
8. Переваги PostgreSQL [Електронний ресурс]: Блог.
<https://fulcrum.rocks/blog/why-use-postgresql-database/>
9. Spring Framework [Електронний ресурс]: TechTarget:
<https://www.techtarget.com/searchapparchitecture/definition/Spring-Framework>
10. Swagger [Електронний ресурс]: Swagger:
<https://swagger.io/docs/specification/2-0/what-is-swagger/>
11. Linux [Електронний ресурс]: Вікіпедія. Вільна енциклопедія:
<https://uk.wikipedia.org/wiki/Linux>
12. Javadoc [Електронний ресурс]: Вікіпедія. Вільна енциклопедія:
<https://en.wikipedia.org/wiki/Javadoc>

13. Wi-Fi [Електронний ресурс]: Вікіпедія. Вільна енциклопедія:
<https://uk.wikipedia.org/wiki/Wi-Fi>
14. Резервне копіювання [Електронний ресурс]: Вікіпедія. Вільна енциклопедія: <https://en.wikipedia.org/wiki/Backup>
15. Порівняння SSD та HDD [Електронний ресурс]: Intel:
<https://www.intel.com/content/www/us/en/products/docs/memory-storage/solid-state-drives/ssd-vs-hdd.html>
16. Порівняння типів пам'яті [Електронний ресурс]: Kingston:
<https://www.kingston.com/en/blog/pc-performance/difference-between-slc-mlc-tlc-3d-nand>
17. Закон про «Охорону праці» [Електронний ресурс]: Сайт Верховної Ради України <https://zakon.rada.gov.ua/laws/show/2694-12#Text>
18. Михелєв О. І., Беркунський О. Є., Казимиренко Ю. О., Гайдаєнко О. В. Систематизація показників та інформаційне забезпечення завантаженості екологічних суден для очищення річкових акваторій // Матеріали XI Всеукраїнської науково-практичної конференції студентів, молодих вчених «Проблеми і перспективи розвитку транспорту». – 2023, Одеса: Одеський національний морський університет (21 квітня). Том 2. – С. 43-46 URL: https://www.onmu.odessa.ua/images/university/news/XI_conf_tr_2023_at2.pdf