

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістра

(назва освітнього ступеня)

на тему: Алгоритмічне та програмне забезпечення комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти

Виконав(ла): студент(ка) 6 курсу, групи СІМ-61

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

(підпис)

Дзюбак Ю.Ю.

(прізвище та ініціали)

Керівник

(підпис)

Лецишин Ю.З.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Луцик Н.С.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Осухівська Г.М.

(прізвище та ініціали)

Рецензент

(підпис)

(прізвище та ініціали)

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Осухівська Г.М.

(підпис)

(прізвище та ініціали)

« ____ » _____ 2023 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студенту Дзюбаку Юрію Юрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Алгоритмічне та програмне забезпечення комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти

Керівник роботи Лецишин Юрій Зіновійович, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 01 » 12 2023 року № 4/7-1132

2. Термін подання студентом завершеної роботи 26.12.2023 р.

3. Вихідні дані до роботи Інформація про особливості ведення навчального процесу у коледжі (НЕМК), апаратне забезпечення для сервера комп'ютеризованої системи, програмне забезпечення для розробки комп'ютеризованої системи

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Огляд методів автоматизації ведення обліку успішності та відвідування занять здобувачами освіти. 2. Проектування архітектури комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти. 3. Розробка та тестування комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти.

4. Охорона праці та безпека в надзвичайних ситуаціях. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Актуальність і мета дослідження.

2. Об'єкт і предмет, наукова новизна і практична цінність дослідження

3. Логічна модель спроектованої бази даних.

4. EER-діаграма створеної бази даних.

5. Структура модулів програмної реалізації комп'ютеризованої системи.

6. Алгоритм роботи програмного забезпечення комп'ютеризованої системи.

7. Основні інтерфейси програми та сформовані додатні до друку сторінки журналу.

8. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Охорона праці</i>	<i>Осухівська Г.М.</i>		
<i>Безпека в надзвичайних ситуаціях</i>	<i>Стадник І.Я.</i>		

7. Дата видачі завдання 20.11.2023

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Розробка та аналіз вимог технічного завдання</i>	<i>23.09-31.09</i>	
2	<i>Аналіз організаційної структури закладу освіти</i>	<i>25.09-02.10</i>	
3	<i>Проектування схеми бази даних та архітектури програмного забезпечення</i>	<i>02.10-14.10</i>	
4	<i>Розробка програмного забезпечення комп'ютеризованої системи</i>	<i>14.10-15.11</i>	
5	<i>Налаштування апаратної складової комп'ютеризованої системи</i>	<i>16.11-20.11</i>	
6	<i>Тестування і виправлення помилок комп'ютеризованої системи</i>	<i>20.11-01.12</i>	
7	<i>Безпека життєдіяльності, основи охорони праці</i>	<i>02.12-06.12</i>	
8	<i>Оформлення кваліфікаційної роботи</i>	<i>06.12-12.12</i>	
9	<i>Попередній захист кваліфікаційної роботи</i>	<i>12.12-17.12</i>	
10	<i>Захист кваліфікаційної роботи</i>	<i>26.12-28.12</i>	

Студент

_____ (підпис)

Дзюбак Ю.Ю.

_____ (прізвище та ініціали)

Керівник роботи

_____ (підпис)

Лецишин Ю.З.

_____ (прізвище та ініціали)

АНОТАЦІЯ

Алгоритмічне та програмне забезпечення комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти // Кваліфікаційна робота магістра // Дзюбак Юрій Юрійович // ТНТУ, Комп'ютерна інженерія, група СІм-61 // Тернопіль, 2023 // с. – 93, рис. – 45, табл. – 2, аркушів А1 – 8, бібліогр. – 20.

Ключові слова: електронний журнал, комп'ютеризована система, сервер, crud-застосунок, java, javafx, ui/ux дизайн, база даних, mysql, pdf-формат.

У кваліфікаційній роботі розроблено проект комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти до складу якої входять серверна частина та програмне забезпечення управління процесом контролю обліку успішності та відвідування занять.

Серверна частина побудована на основі фізичного сервера HP ProLiant DL160 G6 під керуванням операційної системи Windows Server 2012 R2 Standard Server. Забезпечення зв'язку між робочими станціями і сервером бази даних реалізовано через існуючу локальну мережу навчального закладу за допомогою базових протоколів передачі даних використовуваних інструментів розробки прикладного застосунку.

Керування процесом контролю та обліку навчальних занять забезпечено на рівні програмного забезпечення, розробленого на основі архітектури "клієнт-сервер" засобами мови програмування Java. База даних побудована на основі реляційного підходу та з використанням системи керування базами даних MySQL Server.

ABSTRACT

Algorithmic and software support for a computerized system for tracking academic performance and attendance of education seekers // Master's graduation thesis // Dziubak Yuriy Yuriyovich // TNTU, Computer engineering, group SIm-61 // Ternopil, 2023 // p. – 93, fig. – 45, tab. – 2, posters A1 – 8, ref. – 20.

Keywords: digital classbook, computerized system, server, crud-application, java, javafx, ui/ux design, database, mysql, pdf-format.

In the qualification work, a project of the computerized system of the of accounting for performance and attendance of students, which structure includes a server part and the software for management of process for control of the account performance and attendance is developed.

The server part is based on the physical server HP ProLiant DL160 G6 running Windows Server 2012 R2 Standard Server. Communication between workstations and the database server is implemented through the existing local network of the educational institution using the basic data transmission protocols of the tools used to develop the application.

Management of the process of control and accounting of training sessions is provided at the software level, developed on the basis of the "client-server" architecture by means of the Java programming language. The database is based on a relational approach and using the MySQL Server database management system.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 ОГЛЯД МЕТОДІВ АВТОМАТИЗАЦІЇ ВЕДЕННЯ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ	12
1.1 Особливості використання електронних систем обліку успішності та відвідування занять в навчальному процесі.....	12
1.2 Огляд існуючих рішень електронних систем обліку успішності та відвідування занять на ринку	13
1.3 Доцільність розробки комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для потреб конкретного навчального закладу	16
1.4 Висновки до розділу	19
РОЗДІЛ 2 ПРОЕКТУВАННЯ АРХІТЕКТУРИ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ.....	20
2.1 Аналіз та виділення інформаційних об'єктів предметної області	20
2.2 Проектування архітектурної моделі та бази даних системи.....	24
2.3 Обґрунтування вибору інструментів розробки ПЗ системи.....	27
2.4 Обґрунтування вибору рішень для реалізації серверного забезпечення системи	28
2.5 Висновки до розділу	30
РОЗДІЛ 3 РОЗРОБКА ТА ТЕСТУВАННЯ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ.....	31
3.1 Розробка бази даних системи.....	31
3.2 Методи та інструменти для розробки системи	33
3.3 Реалізація інтерфейсів клієнтського застосунку.....	36
3.4 Розробка методу друку "фізичного" журналу на основі даних системи.....	49
3.5 Компіляція програмного проекту у виконуваний десктоп-застосунок.....	51

3.6 Встановлення та налаштування ОС і СУБД на сервер системи.....	54
3.7 Тестування та внесення корективів в систему	56
3.8 Висновки до розділу	59
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ.....	60
4.1 Охорона праці	60
4.2 Забезпечення безпеки життєдіяльності при роботі з ПК	64
4.3 Моделювання уразливості об'єкта економіки та його елементів до дії вторинних вражаючих факторів ядерного вибуху.....	65
4.4 Висновки до розділу	68
ВИСНОВКИ.....	69
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	70
Додаток А Тези конференцій	73
Додаток Б Лістинги коду основних класів програми.....	76

ВСТУП

Актуальність теми. В освітньому середовищі, як і в усіх видах людської діяльності, зараз є актуальними сучасні комп'ютерно-інформаційні технології. Ефективне використання комп'ютерно-інформаційних технологій мають вирішальний вплив на будь-яку сферу діяльності, зокрема на сферу освіти, що є надважливою для будь-якої країни.

Так звана «цифровізація» (англ. digitalization) – це заміна "традиційних" рішень шляхом використання набагато ефективніших комп'ютерно-інформаційних систем. Зменшення паперового документообігу та збільшення ефективності ведення робіт де він використовується і є однією із цілей впровадження комп'ютерно-інформаційних систем.

Прикладом системи, що все частіше впроваджують у навчальний процес, є електронний класний журнал успішності та відвідуваності, що автоматизує контроль успішності та відвідуваності навчальних занять здобувачами освіти, замінюючи традиційні паперові журнали, що несе за собою наступні переваги:

- захист від спотворень даних (використання заздалегідь внесених даних);
- можливість доступу до даних з будь-якого місця де є мережа;
- можливість проведення ефективної аналітики даних.

Метою кваліфікаційної роботи магістра є покращення процесу ведення обліку успішності на відвідуваності навчальних занять шляхом дослідження особливостей організації даного обліку у вищому навчальному закладі для подальшої трансформації цього процесу в електронний вигляд шляхом розробки алгоритмічного та програмного забезпечення для комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти. Для досягнення цієї мети в рамках наукових досліджень сформульовано такі завдання:

1. Проаналізувати особливості організації обліку процесу оцінювання та відвідуваності у вищому навчальному закладі для складання переліку вимог і задач, котрі має вирішувати розроблювана система;

2. На основі результатів аналізу предметної області спроектувати логічну модель інформаційних об'єктів розроблюваної системи для подальшого проектування на їх основі бази даних системи;

3. Розробити архітектуру та алгоритми роботи системи обліку успішності та відвідуваності занять здобувачами освіти для подальшої розробки програмного забезпечення системи;

4. Обрати та обґрунтувати вибір технологій та інструментів для реалізації системи для того, щоб розробити систему у відповідності до поставлених вимог та задач, котрі вона має вирішувати;

5. Розробити та протестувати комп'ютеризовану систему обліку успішності та відвідуваності занять здобувачами освіти для покращення процесу ведення обліку успішності на відвідуваності навчальних занять у навчальному закладі;

Об'єктом наукових досліджень є процес обліку успішності та відвідуваності занять здобувачами освіти у якому головним інформаційним об'єктом є класний журнал (або журнал групи), що є основним документом обліку успішності та відвідуваності навчальних занять здобувачами освіти. Предметна область даної сфери діяльності характеризується особливостями інформаційних потоків, що мають свої набори об'єктів, які утворюють в рамках навчального процесу основу такої інформаційної системи як класний журнал.

Предметом наукових досліджень є основні правила і вказівки ведення обліку навчальних занять і успішності, що описані у виписці з Положення про ведення журналів навчальних занять ДВНЗ «НЕМК». Саме на основі них будуть розроблені принципи систематизації інформації та, власне, функціонування самої інформаційної системи.

Методи дослідження для розв'язку поставлених задач роботи:

- 1) метод структурного аналізу;
- 2) методи інвентаризації, сегрегації та систематизації;
- 3) метод моделювання даних;
- 4) метод висхідного проектування (БД);
- 5) порівняльний метод;

б) методи ui/ux; black-, white-, greybox; регресійного, модульного та функціонального тестування.

Наукова новизна отриманих результатів дослідження полягає в наступному:

1. Реалізовано власну архітектуру комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти, шляхом дослідження і аналізу організації обліку процесу оцінювання та відвідуваності у вищому навчальному закладі, інформаційного аналізу предметної області та виділення інформаційних об'єктів, що стали основою структури системи, яка в точності відтворює інформаційну модель журналу обліку успішності та відвідування занять і максимально ефективно замінює традиційне його ведення.

2. Розроблено власний алгоритм формування сторінок журналу на основі заповнених у базу даних системи, у відповідності до шаблонів що використовуються у навчальному закладі, для подальшого їх друку і формування фізичних журналів для архівування, ефективно замінюючи їх ручне заповнення.

Практичне значення одержаних результатів

Практичне значення одержаних результатів полягає в набутті вагомої кількості навичок, знань та досвіду, котрі є дуже важливими для майбутньої професійної діяльності.

Систему, що є розвитком розробленого проекту вже введено у повноцінну експлуатацію у ДВНЗ НЕМК, чим повністю замінено ведення традиційних паперових журналів груп.

Публікації

1. Ю. Дзюбак, Ю. Лещинин Обґрунтування доцільності створення комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти під потреби окремого вищого навчального закладу. Матеріали XII Міжнародної науково-практичної конференції молодих учених та студентів Тернопільського національного технічного університету імені Івана Пулюя

«Актуальні задачі сучасних технологій» (6-7 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 410.

1. Ю. Дзюбак, Ю. Лецишин Комп'ютеризована система «Classbook» для обліку успішності та відвідування занять здобувачами освіти. Матеріали XII Міжнародної науково-практичної конференції молодих учених та студентів Тернопільського національного технічного університету імені Івана Пулюя «Актуальні задачі сучасних технологій» (6-7 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 411.

РОЗДІЛ 1

ОГЛЯД МЕТОДІВ АВТОМАТИЗАЦІЇ ВЕДЕННЯ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ

1.1 Особливості використання електронних систем обліку успішності та відвідування занять в навчальному процесі

У закладах освіти традиційно має місце великий обсяг різноматного, переважно традиційного (паперового) документообігу. Кожного дня витрачається значна кількість часу на заповнення вручну того ж класного журналу.

Сьогодні рівень інформаційних технологій, надає все більше можливостей до автоматизації документації, котра тепер може мати такий же офіційний статус, як і у фізичних (паперових) документів.

Тож, за допомогою комп'ютерно-інформаційних технологій можливо розробити комп'ютеризовану систему контролю успішності та відвідуваності здобувачів освіти, цифровізуючи процес контролю успішності та відвідуваності навчальних занять в навчальному закладі. Основним документом, що ведеться з метою контролю успішності та відвідуваності навчальних занять є класний журнал.

Журнал успішності та відвідуваності навчальних занять – основний державний документ контролю успішності та відвідуваності навчальних занять учнями та студентами в закладах освіти усіх типів. Для кожної групи (або класу) ведення окремого журналу успішності та відвідуваності занять кожним діючим викладачем (вчителем) є обов'язковим. Електронний журнал успішності та відвідуваності занять є віртуальним аналогом паперового документу, проте за потреби його дані також можна сформувати у фізичний документ.

Електронний журнал успішності та відвідуваності навчальних занять – це комп'ютерно-інформаційна система, що цифровізує процес ведення традиційного (паперового) журналу успішності та відвідуваності навчальних занять.

Так як електронний журнал успішності та відвідуваності навчальних занять в ідеалі є повноцінною комп'ютерно-інформаційною системою, а комп'ютерно-інформаційна система являє собою сукупність інструментів для створення, збереження та обробки інформації, з метою забезпечення інформаційних потреб користувачів, тож вся інформація щодо навчального процесу заповнюється в систему за один раз, щоб в майбутньому лише вносити необхідні доповнення чи виправлення існуючих даних за необхідності й використовувати ці дані для різномітних потреб.

1.2 Огляд існуючих рішень електронних систем обліку успішності та відвідування занять на ринку

На комерційному ринку наявна велика кількість різнопланових реалізацій систем обліку успішності та відвідування навчальних занять. У західних країнах цифровізація навчального процесу більш розвинена, ніж у навчальних закладах України. Причина – лише початок розвитку і інтеграції інформаційних технологій у навчальний процес, проте гостра необхідність дистанційного навчання є стимулом цифровізації освітньої діяльності та використання комп'ютерно-інтегрованих технологій та середовищ дистанційної освіти.

Оскільки, в Україні ступінь розвитку цифровізації освітньої діяльності починає стрімко розвиватись – зростає кількість сервісів для дистанційної освіти, але вона досі не така значна, як у розвинутих країнах заходу. При цьому, багато установ, які вводять дистанційну освіту і сервіси для цифровізації навчального процесу в Україні, є філіями чи послідовниками закордонних інститутів і навчальних закладів середнього рівня.

Враховуючи виклики та нагальні потреби, з якими стикається система освіти України в умовах боротьби з пандемією COVID-19 та військового часу, а також з метою вдосконалення управління системою освіти, Міністерство освіти і науки України поставила перед ДНУ «Інститутом освітньої аналітики» (наказ Міністерства освіти і культури від 09.06.2020 № 781) розпочати роботу над

розвитком безкоштовних державних електронних журналів та електронних щоденників [1].

Також, в Україні часто впроваджується програмне забезпечення, розроблене приватними компаніями. Наприклад, освітня система Moodle [2] досить популярна як за кордоном, так і в нашій країні – це освітня платформа, що об'єднує всіх учасників освітнього процесу у спеціалізовану систему, яка формує своєрідне освітнє середовище, в якому є свої електронні класні журнали.

Іншими успішними комерційними впровадженнями «Електронних» журналів в Україні є:

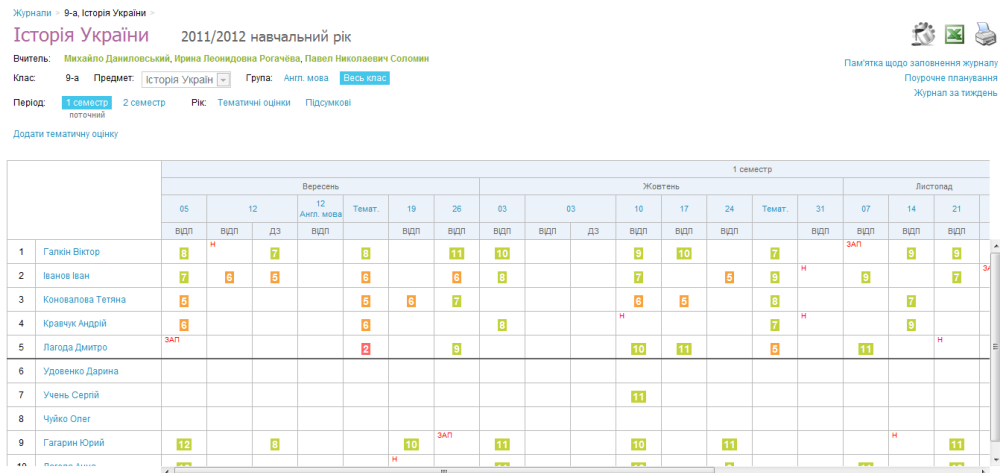
– «Електронні щоденники та журнали» платформи e-schools.info [3] – послуга від ТОВ «Київське енергетичне агентство» за підтримки МОН України надає навчальним закладам України можливість впевнено розвивати та покращувати ведення навчального процесу розпочавши користуватися сучасним сервісом електронних журналів і щоденників (рис. 1.1);

Біологія, 9 "А"			Вересень																											
Уроки за предметом Редагувати розклад Генератор уроків																														
Вчителі: Шуть Ганна Євгенівна, Джудай Анна Василівна																														
1-й семестр Семестрові та річні відмітки			Журнал на весь рік																											
Семестр: 1 вересня 2016 р. - 25 грудня 2016 р. До кінця семестра 63 уроку.			Режими збереження відіток (детальніше): По одній																											
Весь клас																														
Темати уроків:																														
/ Середня / Рейтинг			02 пт	04 вск	05 пн	07 ср	09 пт	11 вск	12 ян	12 пн	14 ср	16 пт	18 вск	19 пн	19 пн	21 ср	23 пт	25 вск	26 пн	26 пн	28 ср	30 пт	02 вск	03 пн	05 ср					
1. Іванов Максим	7	8.43	7					11	7					9	10	7	7	8												
2. Іоанно Юлія	4	9	10	10					7			9					10													
3. Байтрак Микола	5	9.2	8			8			9	8					10			11												
4. Баранкі Ганна	4	8.75	8	9						8		8					10													
5. Богданов Олексій	7	8.57	9	7			9	10	9	8		9	8			8														
6. Богданова Анастасія	7	7.43	8	н			7		8	6	8	7		8		8														
7. Боравльов Олексій	8	8.38	9			8			7	9		10	7	8		9	9													
8. Гонта Катерина	11	8.36	8	8		8	9		9	8		8	8	8	6	10	10													
9. Горіна Каріна	9	8.78	8	9	8	8	8	9	н	8	н	7			н	10	12													
10. Григорський Ігор	10	9.5	9	9		8		12	6	12	8			9	11	11	9													
11. Довбуш Аліна	7	8	8	7	8	9		9	9	7	7																			
12. Ковальчук Ярослав	9	8.33	8	8		10	9	8		7	8	8		8	9															
13. Колтєнко Сергій	7	9.29	9	8		10				8	9		9		9	12														
14. Левківський Сергій	8	8.88	9	8		9			11	7	8	10		9			9													
15. Навал Роман	9	7.11	8	8		8			7	7	н	8	8		5	6	7													

Рис. 1.1. Електронний журнал успішності від платформи E-schools.info

– проект «Щоденник» (ТОВ «Щоденник») [4] – безкоштовна всеукраїнська освітня платформа «Щоденник.ua» – унікальний електронний освітній простір для

усіх учасників навчального процесу. Електронний журнал має перелік різноманітних функцій для успішного контролю успішності та відвідування навчальних занять (рис. 1.2);



		1 семестр															
		Вересень					Жовтень					Листопад					
		05	12	12	Темат.	19	26	03	03	10	17	24	Темат.	31	07	14	21
		відп.	відп.	дз.	відп.	відп.	відп.	відп.	дз.	відп.	відп.	відп.	відп.	відп.	відп.	відп.	відп.
1	Галкін Віктор	8	7		8	11	10			9	10		7	зап.	9	9	
2	Іванов Іван	7	8	8		8	8			7		8	8	Н	9		7
3	Коновалова Тетяна	8			8	8	7			8	8		8				7
4	Кравчук Андрій	8			8			8		Н			7	Н		8	
5	Лагода Дмитро	зап.			2		9			10	11		8		11		Н
6	Удовенко Дарина																
7	Учень Серій									11							
8	Чуйко Олег																
9	Гагарин Юрій	12	8			10	зап.	11		10		11					11

Рис. 1.2. Електронний журнал успішності від платформи «Щоденник.ua»

– портал ukrschools.com.ua – платний проект що використовує модель річної підписки, розроблений для комфорту всіх учасників освітнього процесу. Інтерфейс системи максимально подібний до традиційних паперових журналів та щоденників. Має інтуїтивно зрозумілі та зручні розділи.

Зазвичай, не використовуються комерційні платні освітні середовища у вищих навчальних закладах України, бо такі заклади часто практикують розробку та впровадження власних або замовних розробок, інформацію про опис чи деталі яких не допускають у загальний доступ. Перелік кількох ВНЗ, де є впроваджені системи власної розробки:

- Київський національний економічний університет [5];
- Буковинський державний медичний університет [6];
- Національної академії Державної прикордонної служби України імені Богдана Хмельницького [7];
- Національний педагогічний університет імені М.П. Драгоманова [8].

1.3 Доцільність розробки комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для потреб конкретного навчального закладу

Доцільність розробки комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для потреб конкретного навчального закладу має сторону переваг розробки такої системи та, звісно, сторону недоліків та труднощів такого підходу, які необхідно буде вирішити, щоб власна розробка могла з'явитися та ефективно функціонувати.

Переваги розробки комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для потреб конкретного навчального закладу:

- на відміну від універсальних систем, власна розробка буде створена за конкретними вимогами сторони-замовника, враховуючи всі вимоги та особливості у своїй архітектурі та функціоналі;
- унікальність власної розробки;
- реалізація системи власними студентами навчального закладу чи хоча б їх часткове їх залучення до розробки такого продукту однозначно підвищить престижність даного навчального закладу та мотивацію і конкурентоздатність студентів на ринку праці;
- мінімум матеріальних затрат на реалізацію системи власними зусиллями;
- можливість подальшого вдосконалення та функціонального доповнення створеної системи власними силами;
- можливість перетворення створеної системи на універсальну або модифікувати її під конкретні вимоги інших навчальних закладів для можливості подальшої комерціалізації системи.

Недоліки та труднощі розробки комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для потреб конкретного навчального закладу:

- складність розробки подібної системи використовуючи власні можливості;
- значні затрати часу при розробці подібної системи використовуючи власні можливості;
- якісний унікальний продукт розроблений професіоналами з подальшим супроводом буде відповідно коштувати, тож при виборі реалізації подібної системи шляхом замовлення її у професіоналів головним недоліком буде висока ціна;
- ймовірність невдалої реалізації та неефективності розробленої системи використовуючи власні можливості через відсутність достатнього професійного досвіду у виконавців. Проте, такий варіант можливий і при замовленні у професіоналів через ряд інших причин, таких як недосконало складене ТЗ та ін.;
- великі затрати часу і зусиль на налагодження та тестування системи.

Але, все ж, беручи до уваги особливості ведення навчального процесу в різних навчальних закладах (враховуючи і певну самостійність у прийнятті рішень щодо ведення навчального процесу), доцільним є розробка власної системи обліку успішності та відвідування занять здобувачами освіти, що надасть можливість наступної адаптації і модифікації до змін чи нових потреб.

Власне, розробка проекту комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для потреб окремого навчального закладу буде втілюватись згідно конкретних вимог, обмежень та завдань, що має вирішувати система.

Призначення комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти – переведення у цифровий формат обліку успішності та відвідуваності навчальних занять здобувачами освіти. Призначення комп'ютеризованої системи відображається набором наступного функціоналу:

- введення/редагування даних успішності здобувачів освіти;
- введення/редагування даних відвідуваності здобувачами освіти навчальних занять;

- введення/редагування списків здобувачами освіти, груп, дисциплін, викладчів та інших довідкових даних системи;

- швидкий доступ до потрібної інформації щодо організації навчального процесу у розрізі комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти;

- можливість формування необхідної аналітичної інформації у розрізі комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти;

- можливість друку та формування фізичного журналу на основі даних введених у систему обліку успішності та відвідування занять здобувачами освіти.

Комп'ютеризовану систему обліку успішності та відвідування занять здобувачами освіти передбачається використовувати у ДВНЗ «Нововолинський електромеханічний коледж» задля забезпечення максимальної ефективності організації навчального процесу у навчальному закладі.

Основними цілями реалізації комп'ютеризованої системи є:

- скорочення операцій, що виконуються «вручну» і відповідних дублюючих операцій для підвищення рівня ефективності ведення та організації навчального процесу;

- забезпечення зручності доступу, відображення та використання інформації для її редагування, введення нової, отримання аналітичних даних і т.д.;

- можливість забезпечення адаптації системи до змінних умов та врахування особливостей організації і ведення навчального процесу;

- швидкий та ефективний пошук необхідної інформації у системі;

- розрізний доступ користувачів системи згідно наданих ролей до чітко визначеного функціоналу системи та лише відповідної інформації;

- оперативний контроль та моніторинг даних системи адміністрацією навчального закладу для нагляду за організацією та перебігом навчального процесу наживо.

Метою даної роботи є розробка комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для ДВНЗ

«Нововолинський електромеханічний коледж», що передбачає розробку бази даних системи та десктоп-застосунку для переведення процесу ведення журналу успішності та відвідуваності у цифровий вид.

Для досягнення поставленої мети потрібно вирішити ряд пов'язаних послідовних задач, що є своєрідними етапами розробки проекту:

1) проаналізувати особливості організації обліку процесу оцінювання та відвідуваності у вищому навчальному закладі;

2) на основі результатів аналізу предметної області спроектувати логічну модель інформаційних об'єктів розроблюваної системи;

3) розробити архітектуру та алгоритми роботи системи обліку успішності та відвідуваності занять здобувачами освіти;

4) обрати та обґрунтувати вибір технологій та інструментів для реалізації системи;

5) розробити та протестувати комп'ютеризовану систему обліку успішності та відвідуваності занять здобувачами освіти.

Дані етапи будуть описані у наступних розділах роботи.

1.4 Висновки до розділу

В даному розділі було описано особливості використання електронних систем обліку успішності та відвідування занять в навчальному процесі.

Розглянуто існуючі рішення електронних систем обліку успішності та відвідування занять на ринку.

Обґрунтовано доцільність розробки комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти для потреб конкретного навчального закладу, сформовано список переваг і недоліків розробки такої системи, а, також, складено перелік цілей та задач для реалізації комп'ютеризованої системи.

РОЗДІЛ 2

ПРОЕКТУВАННЯ АРХІТЕКТУРИ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ

2.1 Аналіз та виділення інформаційних об'єктів предметної області

Класний журнал (або журнал групи у ВНЗ) це головний інформаційний об'єкт дослідження, бо він є основним документом ведення обліку успішності та відвідуваності здобувачів освіти. Тож, центром предметної області є класний журнал, що характеризується набором інформаційних об'єктів, котрі утворюють його сутність.

Журнал обліку успішності та відвідуваності здобувачів освіти –документ, в якому відбувається фіксування результатів досягнень здобувачів освіти, а, також, відвідування навчальних занять.

Наявність даного документу під час ведення навчального процесу - обов'язкова.

Журнали зберігаються в навчальному відділенні (денному/заочному - залежно від виду навчальної програми) навчального закладу.

Основним документом, що регламентує ведення журналу є відповідні положення навчального закладу з яких можна сформулювати опис предметної області (рис. 2.1).

Положення про ведення журналів навчальних занять ДВНЗ «НЕМК»:

- 1) Ведення журналу здійснюється викладачами коледжу та кураторами груп. Викладачі несуть відповідальність за правильність і достовірність записів;
- 2) Записи в журналах проводяться виключно державною мовою, з предметів іноземних мов можна частково записувати зміст заняття та домашнє завдання мовою, що вивчає предмет, з інформатично-комп'ютерних предметів – допускається варіант заповнення англійською мовою;
- 3) Записи вносяться синім чорнилом акуратно і уважно. Будь-які виправлення на сторінках журналу проводити не можна, в разі помилки поряд вноситься виправлений варіант, що засвідчується потім підписом і скріплюється печаткою відповідним представником адміністрації;
- 4) Куратор групи заповнює розділ «Облік відвідування»;
- 5) Відсутність студента на занятті позначається літерою «н» і здійснюється в кінці заняття, у разі спізнення ставиться літера «с», якщо студент спізнився і отримав оцінку, то оцінка ставиться через риску дробу, наприклад «с/5». Якщо студент відпрацював пропущене заняття, у відповідну клітинку до літери «н» через дріб ставиться оцінка, наприклад, «н/5»;
- 6) Головними видами оцінювання навчальних досягнень здобувачів освіти є поточне та підсумкове (тематичне, модульне, семестрове, річне), державна підсумкова атестація, залік, екзамен. Тематичне оцінювання здійснюється у відповідності до робочих навчальних програм не рідше як двічі на семестр;
- 7) Незадовільна Семестрова оцінка, Залік, Екзамен може підлягати коригуванню. Скоригована оцінка виставляється без дати у колонку з надписом Скоригована поруч із колонкою I семестр, II семестр, Залік, Екзамен. Задовільна оцінка може бути скоригована тільки в особливих випадках і тільки з дозволу директора коледжу.

Рис. 2.1. Положення про ведення журналів навчальних занять ДВНЗ «НЕМК»

Журнал обліку навчальних занять, переважно складається з кількох основних розділів, першим з яких є "Зміст", що відображає навантаження на певний навчальний період (рис. 2.2).

Викладач _____					
(прізвище та ініціали)					
№ з/п	Дата	Кількість годин	Тема заняття	Домашнє завдання	Підпис викладача

Рис. 2.4. «Ліва» сторінка розділу дисципліни

Необхідні для відтворення журналу у електронному виді інформаційні об'єкти предметної області та перелік функцій котрі від цих об'єктів будуть прямо залежними формуються саме на основі описаних вище складових "традиційного" журналу обліку навчальних занять (табл. 2.1).

Таблиця 2.1

Опис інформаційних складових моделі предметної області

Список інформаційних об'єктів	Викладачі Групи та підгрупи Студенти Дисципліни Види занять Навчальні роки Семестри Системи оцінювання Оцінки Відмітки відвідуваності Навантаження окремих груп Сторінки журналу (сформовані з навантаження занять та реалізовані облік успішності й відвідуваності студентів)
Перелік функцій, де будуть використані інформаційні об'єкти	Етап А: введення необхідних для ведення журналу довідкових даних адміністратором Етап Б: ведення викладачем журналу Б1: створення заняття викладачем згідно навантаження Б2: заповнення даних про заняття Б3: заповнення даних про відвідуваність та успішність заняття Етап В: використання введених даних (наочне відображення, друк і т. д.)

Тому згідно цього необхідно розробити системні інтерфейси із розподіленим доступом для наступних категорій користувачів:

- представників адміністрації навчального закладу;
- викладачів;
- кураторів груп;
- студентів (здобувачів освіти).

Також буде реалізована можливість друку заповнених «сторінок журналу» для подальшого формування з них паперових журналів для архіву.

Основою комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти є саме інтерфейс для викладачів, бо саме із нього формується основа класного журналу, тому найбільше уваги буде приділено опису його реалізації.

2.2 Проектування архітектурної моделі та бази даних системи

Згідно отриманих результатів дослідження та аналізу предметної області було спроектовано логічну модель комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти (рис. 2.5).

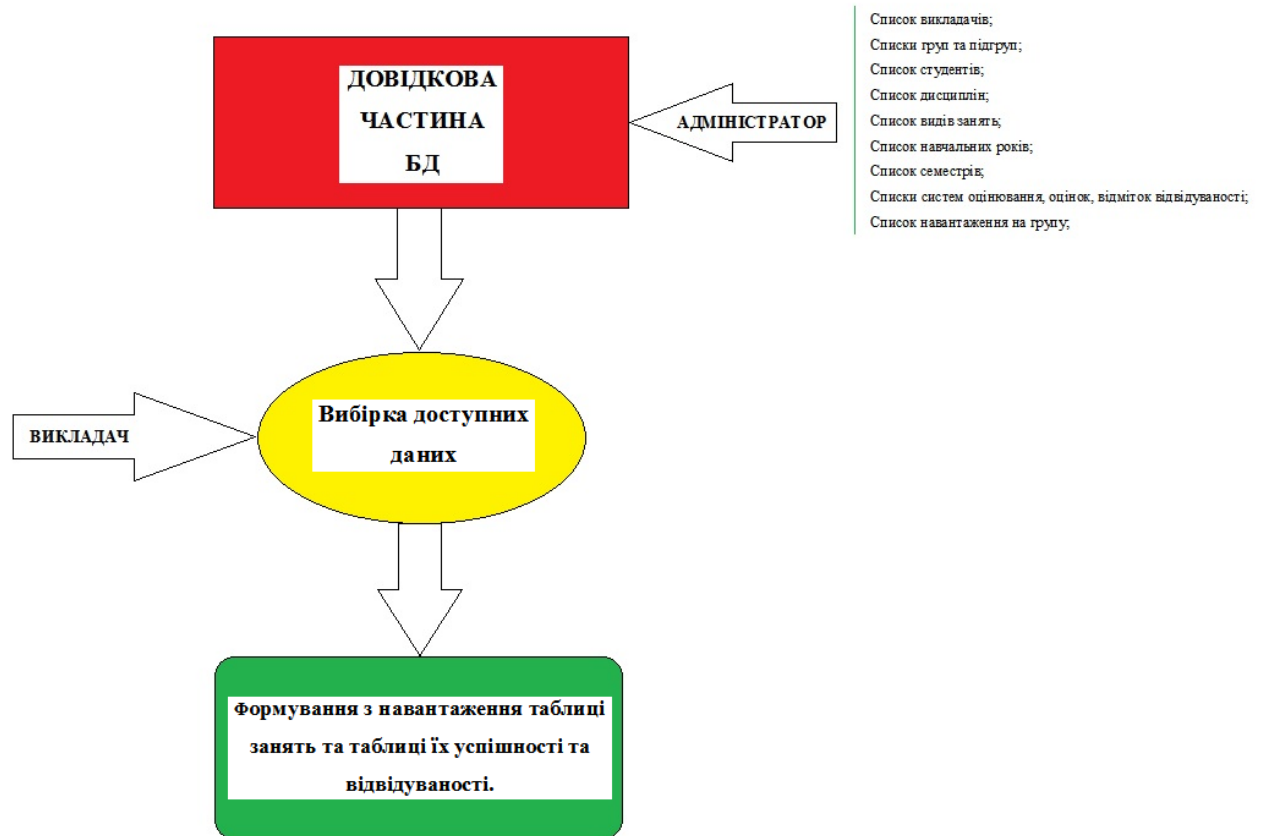


Рис. 2.5. Спроектвана логічна модель комп'ютеризованої системи

На основі логічної моделі предметної області разом було здійснено проектування реляційної бази даних системи.

Шляхом експериментів та дискусій, на основі результатів аналізу предметної області та дослідження процесу ведення навчального процесу було спроектовано модель бази даних засобами Microsoft Access (рис. 2.6).

Протягом проектування було враховано особливості та обмеження предметної області, що вплинули на архітектуру БД:

1) Студент може бути студентом виключно однієї групи. Якщо та сама людина йде на паралель, то вона стає іншим студентом, що теж вноситься в систему. Зумовлено таке обмеження необхідністю чіткої ідентифікації студента за номером залікової книжки;

7) Коли кількість фактично вчитаних годин співпаде із кількістю відведених годин для дисципліни згідно навантаження, то з'являтиметься відповідне повідомлення про вчитку годин на сторінки дисципліни;

8) Викладання одної дисципліни у різних семестрах фактично прийматиметься системою як дві різні дисципліни, для цього існуватиме довідкова таблиця «Довідник семестрів». Аналогічно окремими дисциплінами є дисципліни із властивостями різних типів занять;

9) За замовчуванням кількість годин будь-якого заняття - 2, але при потребі можна внести інше значення;

10) Системи оцінювання занять містять у наборі значень оцінок міститимуть додаткові універсальні оцінки «зарах.» і «незарах.»;

2.3 Обґрунтування вибору інструментів розробки ПЗ системи

Для ролі сервера баз даних було обрано MySQL Server 8.0, бо MySQL – одна з найпопулярніших СУБД, що використовується для створення веб-сайтів та застосунків та має підтримку великого переліку мов програмування.

MySQL Workbench – це назва власне СУБД MySQL забезпечує моделювання даних, розробку баз/скриптів на мові SQL та набір інструментів для адміністрування/налаштування сервера і користувачів та резервного копіювання.

Особливості та переваги MySQL Workbench:

- візуальне проектування, моделювання, генерування та керування базами даних;

- консоль для адміністрування середовищ MySQL. Можливість використання візуальних інструментів адміністрування та налаштування серверів, користувачів, резервного копіювання та перевірки стану баз даних;

- набір інструментів для підвищення продуктивності MySQL-додатків;

- наявність рішень для міграції Microsoft SQL Server, Microsoft Access, Sybase ASE, PostgreSQL та інших таблиць, об'єктів і даних RDBMS в MySQL із підтримкою апгрейду застарілих версій MySQL до найсвіжіших версій.

Для ролі середовища розробки ПЗ системи було обрано IDE – IntelliJ IDEA, що зараз, вважається найкращим середовищем розробки для мови Java.

IntelliJ IDEA – закрите, консервативне та надійне IDE, котре підходить для вирішення будь-яких завдань.

Основні особливості і переваги IDE IntelliJ IDEA:

- автоаналіз програмного коду – IDE аналізує код в усіх файлах на всіх мовах, що використовуються в розробці проекту. Результатом цього аналізу є надання допомоги при написанні коду, перевірка помилок в коді, рефакторинги і т.д.;

- ергономічне, зручне і ефективне користування середовищем – ніщо не відволікає і неважає розробнику у творчому процесі;

- «з коробки» доступний набір різноманітних інструментів для розробників для спрощення робочого процесу;

- розробка на різних мовах, підтримка різних фреймворків та технологій – крім Java, IDE «з коробки» надає підтримку провідних бібліотек, фреймворків і популярних мов розробки.

2.4 Обґрунтування вибору рішень для реалізації серверного забезпечення системи

Необхідності в придбанні нового, окремого, сервера для розроблюваної комп'ютеризованої системи нема, бо вона не потребуватиме значних обчислювальних ресурсів для свого функціонування.

Тож, у ролі сервера було обрано наявний у навчальному закладі HP ProLiant DL160 G6 (рис. 2.7 - 2.8) – що є досить швидким та продуктивним (табл. 2.2) і, водночас, недорогим сервером, стійкового типу, що добре оснащений для виконання різноманітних завдань.

DL160 G6 – є хорошим рішенням для роботи з ПЗ для вирішення задач виробничо-комерційного рівня.



Рис. 2.7. Сервер HP ProLiant DL160 G6



Рис. 2.8. Сервер HP ProLiant DL160 G6 у серверній стійці навчального закладу

Таблиця 2.2

Короткі технічні характеристики обраного сервера

Технічні характеристики HP ProLiant DL160 G6	
Процесор(и)	2x Intel Xeon L5640, 2.266 GHz, Hexa Core
Материнська плата	HP ProLiant DL160 G6 (LGA 1366)
RAM	32 GB DDR3
HDD	1 TB
БЖ	600 Вт
ОС	Windows Server 2012 R2 Standard Server

У навчальному процесі даний сервер використовується як термінальний. Наявна роль використовує невелику частину його обчислювальних потужностей, тому це не перешкоджає використати його паралельно для забезпечення серверної частини для комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти.

Для розділу ролей сервера було прийнято рішення зробити «сервер на сервері» з використанням системи віртуалізації VMware Workstation Pro. На віртуальну машину буде встановлено окрему серверну ОС де буде розгорнуто і налаштовано базу даних для системи обліку успішності та відвідуваності занять здобувачами освіти. Також, віртуалізація серверної частини комп'ютеризованої системи надасть властивість мобільності – у випадку виходу із ладу чи апгрейду сервера не потрібно перевстановлювати ОС і знову розгортати і налаштовувати БД.

VMware Workstation – це система віртуалізації ОС, що підтримує різні платформ та надає можливість запуску на ПК кількох різних ОС одночасно. Кожна віртуальна машина може мати власну будь-якою іншу ОС. Також VMware Workstation є корисною при тестуванні клієнт-серверних застосунків.

Для ролі серверної ОС було обрано Windows Server (2012), що є висококласною серверною операційною системою із широкими можливостями, на основі котрої побудовано велика кількість центрів зберігання та обробки даних.

2.5 Висновки до розділу

В даному розділі було проаналізовано та виділено головні інформаційні об'єкти предметної області.

На основі проаналізованої предметної області та згідно з виділеними інформаційними об'єктами спроектовано архітектурну модель та базу даних комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти.

Обґрунтовано вибір інструментів розробки ПЗ системи, а, також, вибір рішень для реалізації серверного забезпечення системи.

РОЗДІЛ 3

РОЗРОБКА ТА ТЕСТУВАННЯ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ

3.1 Розробка бази даних системи

В процесі розробки реляційної БД потрібно обов'язково не забувати про основні складові даної моделі:

- структуру даних;
- операції, що виконуються над даними;
- обмеження, від яких залежить цілісність даних.

Таблиці – це основна структура даних в реляційній моделі БД, що в реляційній теорії мають назву «відношення». Саме від терміну «відношення» (англ. «relation») і було сформовано назву моделі – реляційна.

Інформаційно-логічну модель досліджуваної предметної області відображає логічна модель БД.

Реляційними таблицями відображаються відповідні інформаційні об'єкти моделі даних. Реквізити кожного відповідного інформаційного об'єкта визначають структуру таблиць, в яких кожне поле відповідає реквізітам об'єктів.

Унікальний ключ реляційної таблиці формують головні реквізити об'єкту. Для кожного поля таблиці задаються: тип даних, розмір даних і ін. властивості. Складовими інформаційних об'єктів відповідають записи таблиць, що формуються при ініціалізації таблиці.

На основі розробленої логічної моделі БД у середовищі MySQL Workbench було створено базу даних системи (рис. 3.1).

Згідно логічної моделі БД було відтворено таблиці та зв'язки між ними. Схема (EER-діаграма) реалізованої БД, що містить всі зв'язки, поля, розміри полів та типи даних, має наступний вигляд (рис. 3.2).

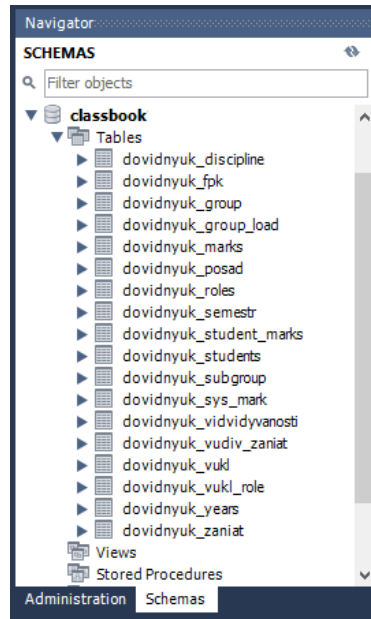


Рис. 3.1. Створена БД із списком таблиць

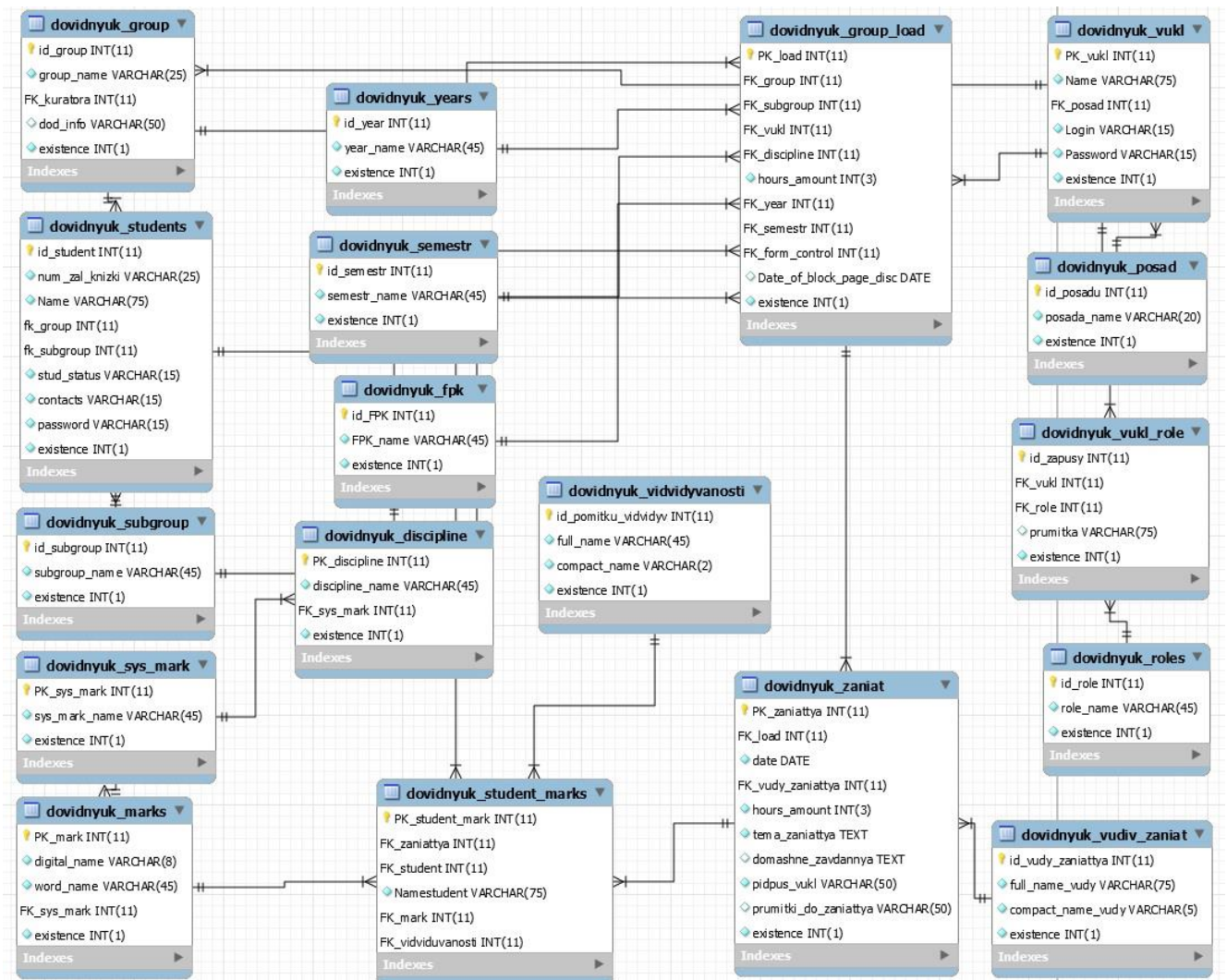


Рис. 3.2. EER-діаграма створеної БД у MySQL Workbench

3.2 Методи та інструменти для розробки системи

«Клієнт-серверна» архітектура виникла в процесі розвитку уявлень про процеси обробки даних з використанням розподілених обчислювальних процесів та стала загальним поняттям про взаємозв'язок обчислювальних систем в мережах, котрі, зазвичай, поділяються на два види:

- джерело запитів (клієнтська сторона);
- джерело відповідей (серверна сторона).

За допомогою заздалегідь визначеного протоколу у мережі відбувається взаємодія між учасниками такої архітектури.

Крім хостування БД, сервер бази даних повинен виконувати основний об'єм обробки даних – це є основною суттю концепції такої архітектури. Процес пошуку даних на сервері здійснюється по запиту клієнта. Отримані дані передаються мережею від сервера до клієнта. Мова запитів SQL (Structured Query Language) відповідає за специфіку «клієнт-серверної» взаємодії.

На сервері виконується запит до бази даних, котрий надійшов від клієнта. По мережі результат запиту повертається до клієнта (рис. 3.3).

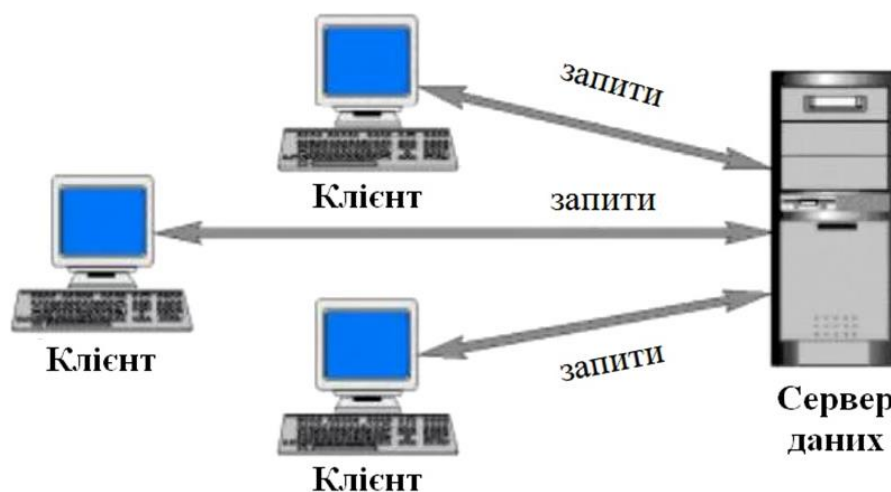


Рис. 3.3. «Клієнт-серверна» архітектура

Процес запиту до БД, в основному, складається з наступних етапів:

- 1) Запит даних клієнтом;

- 2) Транслявання запиту в SQL;
- 3) Передавання SQL-запит мережею на сервер;
- 4) Здійснення процесу пошуку сервером бази даних;
- 5) Повернення клієнту запитаних даних;
- 6) Пред'явлення даних користувачеві.

Переваги використання «клієнт-серверної» архітектури:

- відносно невелика вартість системи;
- достатні обчислювальні можливості;
- легке налаштування під конкретні завдання;
- менші вимоги до обсягу оперативної пам'яті у клієнтів;
- менші вимоги до обсягу дискового простору у клієнтів;
- сервери можуть зберігати великий обсяг даних;
- спрощується процес резервного копіювання даних;
- простіший процес керування всією системою і контролем її безпеки.

Для розробки клієнтського застосунку головним інструментом була мова програмування Java.

Java – розроблена компанією Sun Microsystems у 1995 році мова програмування та обчислювальна платформа, що розвинулась до рівня одної із основних мов програмування, що забезпечує потреби значної частини цифрового світу, забезпечуючи надійну основу, для побудови багатьох сервісів і програм. Покладатися на Java також продовжують і найсучасніші, найінноваційніші продукти та цифрові послуги.

Для розробки на мові Java необхідно мати Java Runtime Environment (JRE) та Java Development Kit (JDK).

Це основні є інструменти розробки – на мові Java, бо JRE містить і виконує віртуальну машину Java (JVM), зберігає у собі основні класи та Java бібліотеки. JRE – це база, що забезпечує виконання та функціонування програмного забезпечення написаного на Java.

Платформа JavaFX стала основою для створення UI програми.

JavaFX — це платформа клієнтських додатків нового покоління з відкритим кодом для настільних, мобільних і вбудованих систем, побудованих на Java. Це спільне зусилля багатьох осіб і компаній з метою створення сучасного, ефективного та повнофункціонального інструментарію для розробки багатофункціональних клієнтських програм. JavaFX є наступником, вже застарілої, базової графічної бібліотеки Swing. Кокурентами JavaFX є Microsoft Silverlight, Adobe Flash, та ін.

Для реалізації функціоналу застосунку було використано описані нижче бібліотеки та драйвери.

MySQL Connector Java (driver або external library) – забезпечує підключення для клієнтських програм, розроблених мовою програмування Java, за допомогою MySQL Connector/J, драйвера, який реалізує API підключення до бази даних Java (JDBC), а також MySQL X DevAPI.

Connection – забезпечує з'єднання із обраною базою даних. База даних об'єкта Connection може надавати інформацію, що описує його таблиці, підтримувану граматику SQL, збережені процедури, можливості цього з'єднання тощо.

DriverManager – клас у Java, що керує набором драйверів Java Database Connectivity (JDBC), які доступні для використання програмою.

ResultSet – об'єкт Java, який містить результати виконання SQL-запиту. Іншими словами, він містить рядки, які задовольняють умови запиту.

Statement – об'єкт, що є одною "командою", яка виконується інтерпретатором Java. За замовчуванням інтерпретатор Java виконує один оператор за іншим у порядку їх написання.

SQLException – клас і його підтипи надають інформацію про помилки та попередження, які виникають під час доступу до джерела даних.

ObservableList – спеціальна реалізація інтерфейсу List у JavaFX SDK. Це спостережуваний список, який надає можливість приєднати "слухачів" для зміни вмісту списку. ObservableList відіграє важливу роль у розробці JavaFX, оскільки

він використовується для основних компонентів, таких як TableView, ComboBox тощо.

FXCollections – службовий class, що описує роботу із колекціями в JavaFX, що визначаються пакетом javafx.collections, який складається з інтерфейсів і класів різноматних видів списків.

ArrayList – реалізація інтерфейсу List зі змінним розміром масиву. Реалізує всі необов’язкові операції зі списком і дозволяє всі елементи, включаючи null.

JFoenix – open-source бібліотека для Java, що реалізує за допомогою компонентів Java дизайн матеріалів Google.

iTextPDF – одна із найвдаліших реалізацій бібліотеки Java, для створення, читання та роботи з PDF-файлами.

Створюваний застосунок, фактично, є CRUD-застосунком.

CRUD (англ. «create», «read», «update», «delete») – аббревіатура, котра означає чотири базові функції, що використовуються у роботі з базами даних: створення, читання, модифікація, видалення.

Приклади, де детально пояснено реалізацію CRUD-застосунку на основі JavaFX та MySQL, що в подальшому були використані як зразок реалізації довідкових інтерфейсів застосунку, були знайдені на платформі Youtube на каналі «kenLovesToCode» [9].

3.3 Реалізація інтерфейсів клієнтського застосунку

Спершу було створено проект в IDE для якого було задано наступну структуру («скелет») (рис. 3.4).

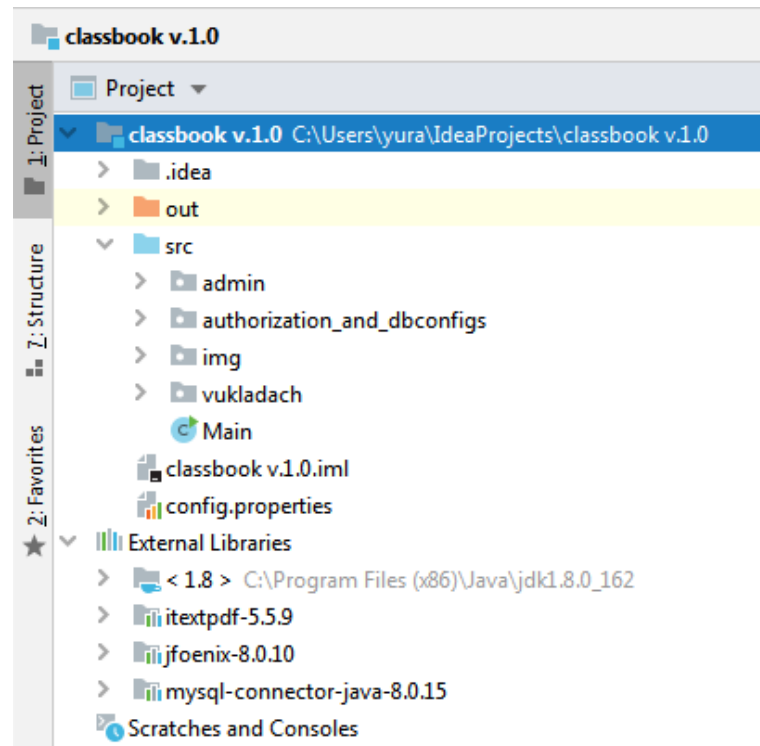


Рис. 3.4. Базова структура проекту затосунку

Наступним було створення класу `DBConnection` (рис. 3.5), в якому було описано і ініційовано функції відповідальні за зв'язок із БД.

```

public class DBConnection {
    private final String dburl;
    private Connection connect;
    private String host;
    private String port;
    private String db;
    public DBConnection() {
        FileInputStream fis;
        Properties property = new Properties();
        String path = System.getProperty("user.dir");
        try {fis = new FileInputStream(path+"/config.properties");
            property.load(fis);
            host = property.getProperty("db.host");
            port = property.getProperty("db.port");
            db = property.getProperty("db.name");
        } catch (IOException e) {
            System.err.println("Error: Config file is not allowed!");
        }
        dburl =
"jdbc:mysql://" + host + ":" + port + "/" + db + "?allowPublicKeyRetrieval=true&useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDateTimeCode=false&serverTimezone=UTC&useSSL=false";

        public Connection getConnection() {
            try {
                connect = DriverManager.getConnection(dburl, username, password);
            } catch (Exception e) { e.printStackTrace(); }
            return connect;
        }

        public void close(Connection connect, PreparedStatement pstmt,
ResultSet rs) {
            try {if(connect != null)connect.close();
                if(pstmt != null)pstmt.close();
                if(rs != null)rs.close();
            } catch (Exception e) { e.printStackTrace(); }}

        public void close(Connection connect, PreparedStatement pstmt)
{ try { close(connect, pstmt, null);
        } catch (Exception e) { e.printStackTrace(); } }

        public void close(PreparedStatement pstmt) {
            try { close(null, pstmt, null);
            } catch (Exception e) { e.printStackTrace(); }}
    }
}

```

Рис. 3.5. Лістинг коду класу DBConnection

Важливим пунктом даної реалізації є, те що дані для доступу до бази даних не зберігаються у кодї, а у окремому property-файлі, призначення якого – збереження будь-яких статичних даних, що можуть використовуватись під час виконання програми, до прикладу, адреса, дані авторизації до роботи із БД і ін.

Наступним етапом після реалізації класу доступу до БД була розробка дизайну інтерфейсів та, власне, самих інтерфейсів. Першим було реалізовано інтерфейс адмінпанелі застосунку (рис. 3.6).

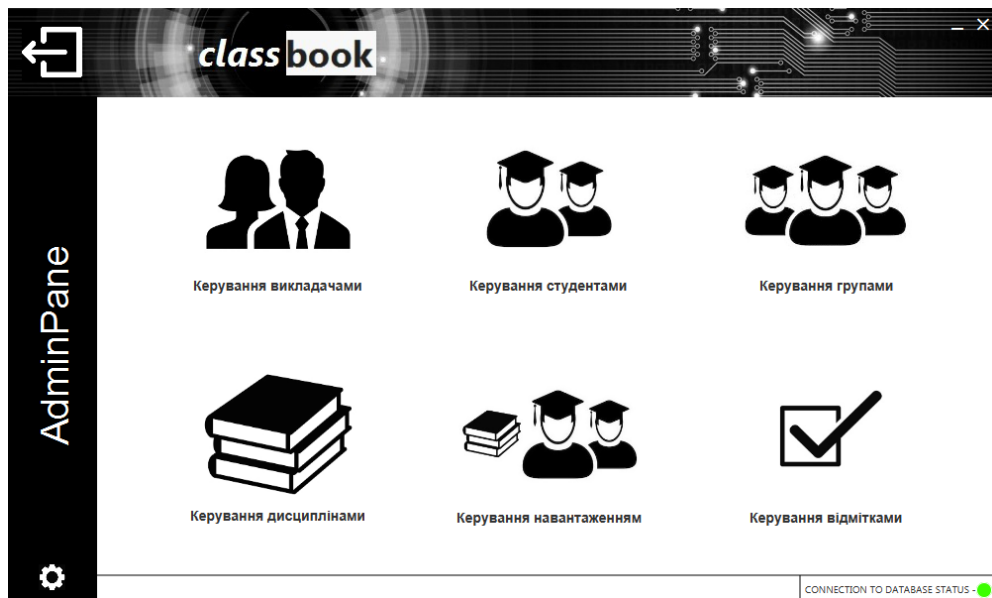


Рис. 3.6. Форма адмінпанель застосунку

Для створення інтерфесів адмінпанелі застосунку було використано два основних типи форм: повнорозмірна і підформа. Як приклад наведено форми «Керування навантаженням» (рис. 3.7) та «Керування навчальними роками» (рис. 3.8).

Керування навантаженням

Група: Підгрупа: Викладач: Дисципліна: Система оцінювання: Кількість годин: Навчальний рік: Семестр: Форма підсумкового контролю: Дата блокування стор. дисципліни:

Група	Підгрупа	Викладач	Дисципліна	Сист. оцінов.	К. год.	Навч. рік	Семестр	ФПК	Дедлайн
1-КТ-17	1	Юрій Дзюбак	Технічне обслуговування EOM і ...	5	126	2020-2021	7	екзамен	2021-01-15
2-КТ-16	1	Наталія Ромашевська	WEB дизайн	5	2	2020-2021	7	залік	2020-10-31
2-КТ-16	1	Юрій Дзюбак	Технічне обслуговування EOM і ...	5	34	2020-2021	7	залік	2020-10-31
1-КТ-17	1	Наталія Ромашевська	WEB дизайн	5	2	2020-2021	7	залік	2020-10-31

CONNECTION TO DATABASE STATUS: ●

Рис. 3.7. Повнорозмірна форма «Керування навантаженням»

Список навчальних ро...

Навчальні роки

Новий навчальний рік:

Навчальні роки
2020-2021
2021-2022

Рис. 3.8. Підформа «Керування навчальними роками»

За аналогічними моделями панелей реалізовано всі форми для адмінпанелі. Кожна форма є відображення відповідної таблиці чи кількох пов'язаних таблиць у базі даних.

Кожна форма застосунку потребує наявності трьох індивідуальних Java-класів:

- FXML-клас із описом зовнішніх елементів, тобто UI форми із прописаними ID компонентів;
- Controller-клас форми, що відповідає за виконання методів (події), котрі виконуються при роботі з формою;
- DataAccess-клас, де описані змінні, які приймають зчитані дані з БД та через котрі здійснюється передача даних у БД.

Також, необхідні методи дописуються у спільний клас `DataAccessObject` для всіх форм, який здійснює безпосередній контакт із БД (в даному випадку для операцій `create`, `read` та `update`).

Власна фантазія та фоторедактори були головними інструментами для реалізації загального стилю програми та створення фонових зображень. UI панелей реалізовувався за допомогою редактору для об'єктів JavaFX під назвою `Scene Builder`, який дає змогу з легкістю створювати компоненти керування інтерфейсу JavaFX, графічні діаграми, геометричні фігури та контейнери для елементів, для того щоб якісно та швидко реалізовувати інтерфейси користувача.

Наступним було розроблено дизайн та функціонал головної панелі для авторизації у систему (рис. 3.9).

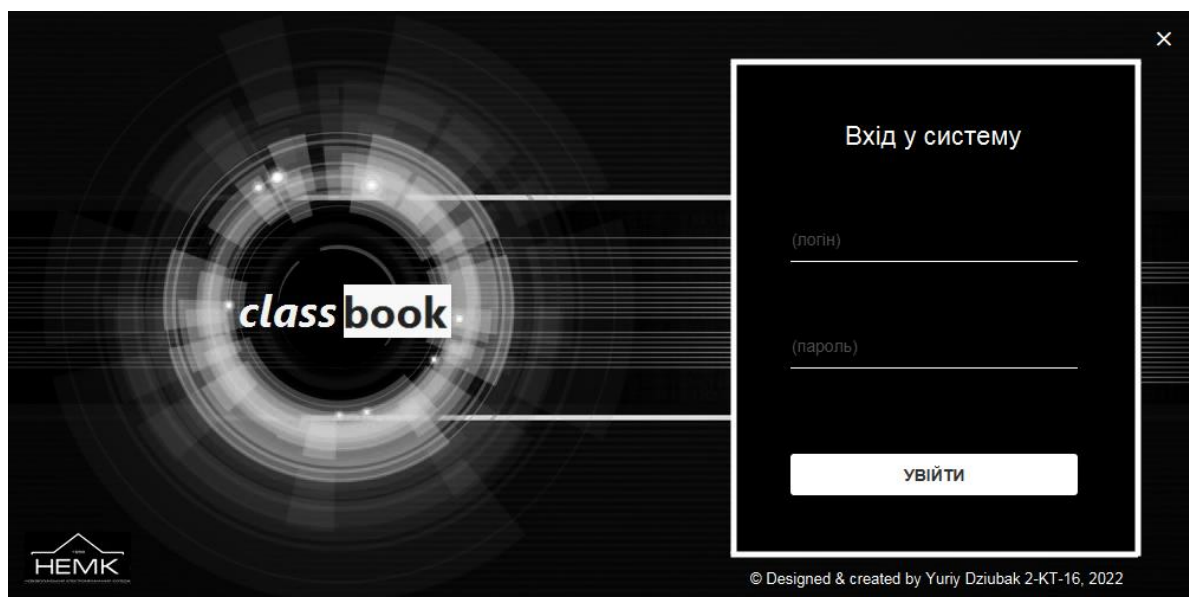


Рис. 3.9. Головна панель авторизації у систему

Для даної панелі були реалізовані функції, що здійснюють перевірку ідентифікаторів користувача, котрі будуть вводиться у відповідні поля. Для випадків коли будуть вноситись некоректні дані було додано візуальні повідомлення про помилки, до прикладу, якщо одне чи два поля заповнено неправильно, то символи заповненого(-них) поля(-ів) буде підсвічено червоним, а у випадку коли лише одне з полів заповнено вірно, то його буде підсвічено зеленим (рис. 3.10).

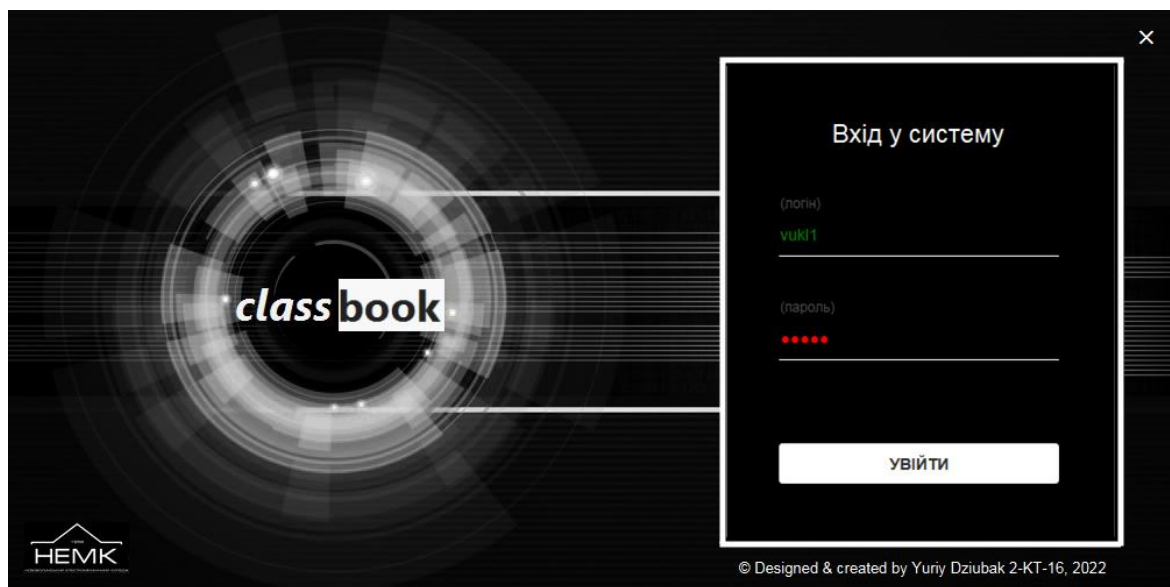


Рис. 3.10. Головна панель авторизації у систему з введеним некоректним паролем користувача

Під час реалізації інтерфесу було враховано необхідність перевірки застосунком статусу з'єднання із БД на сервері (рис. 3.11).

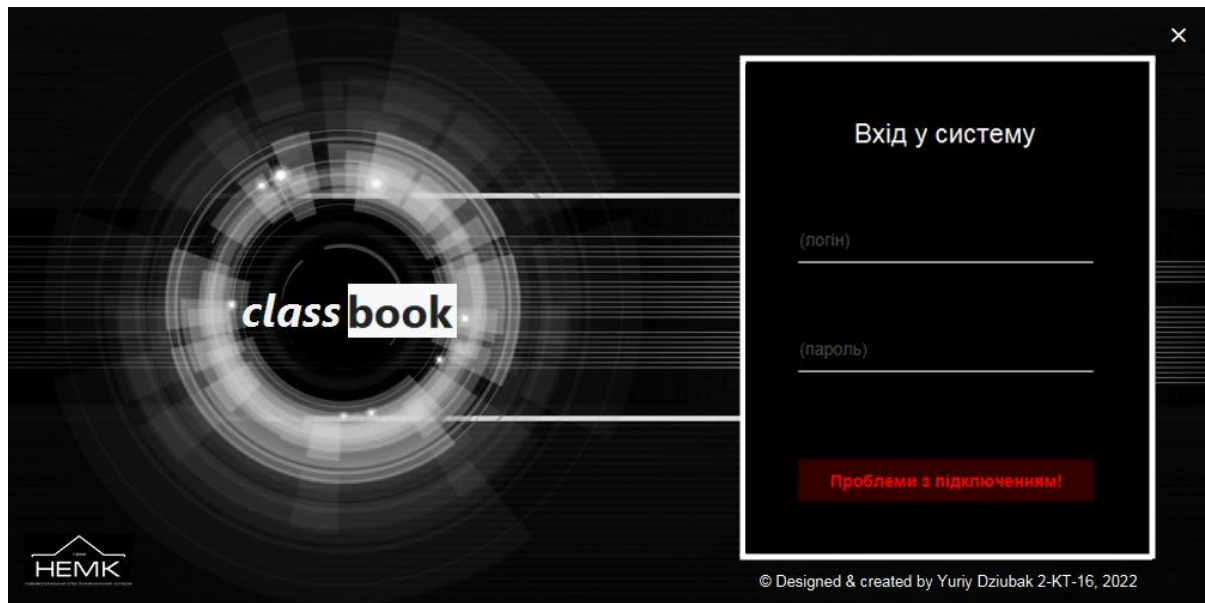


Рис. 3.11. Головна панель авторизації у систему із відображенням повідомлення про відсутність з'єднання з БД

Наступним було реалізовано стартову форму – персональний акаунт користувача системи. На даній панелі відображаються доступні ролі та ПІБ авторизованого користувача для поточного сеансу (рис. 3.12). У систему було додано наступні ролі:

- «Викладач» – головна роль, яка дублює дії викладача при роботі із традиційним паперовим журналом;
- «Представник адміністрації навчального закладу» – додаткова роль, що дає можливість переглядати дані журналу усіх груп;
- «Куратор» – додаткова роль, що дає можливість переглядати дані журналу призначеної до кураторства групи.

Додаткові ролі мають особливість у тому, що вони не мають можливості внесення чи зміни даних (крім приміток) у відомостях, котрі доступні їм до перегляду.

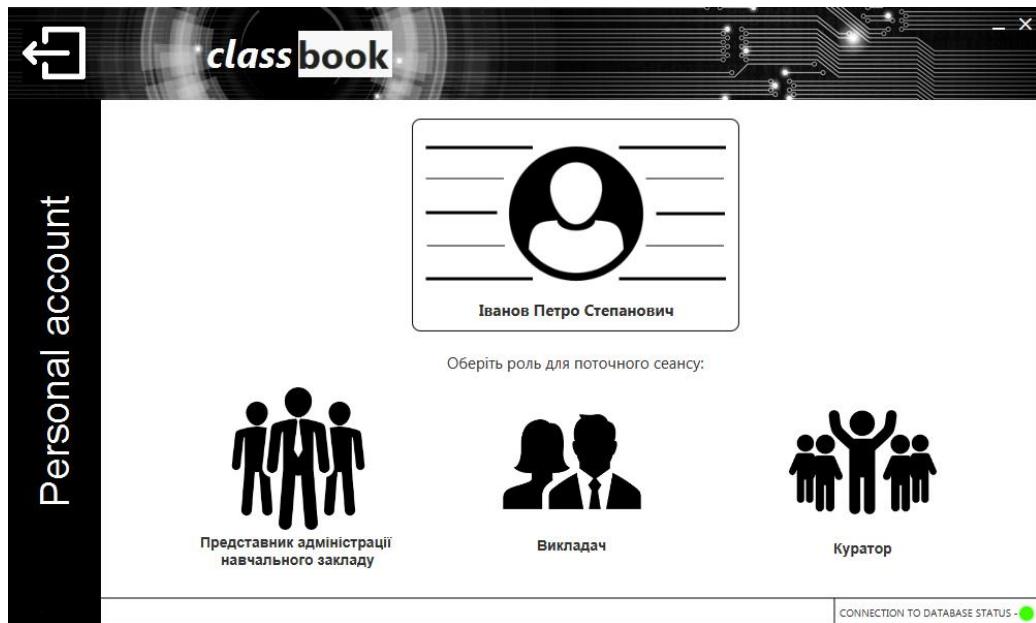


Рис. 3.12. Форма "Персональний акаунт" користувача

Далі було реалізовано форму для вибору викладачем/представником адміністрації навчального закладу/куратором необхідних йому доступних згідно ролі даних, що містяться у довіднику навантаження для подальшого створення заняття (рис. 3.13).

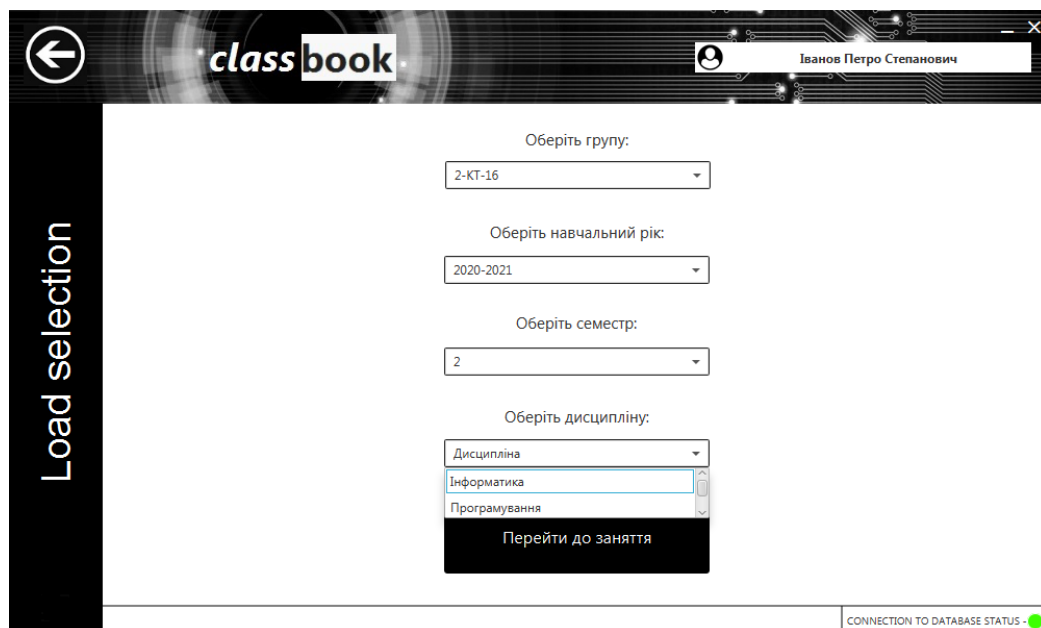


Рис. 3.13. Форма вибору доступних даних для створення/перегляду/редагування заняття

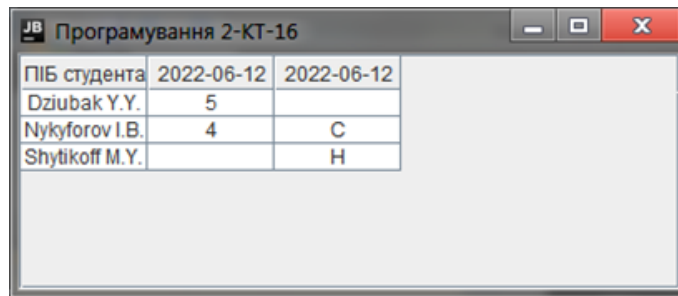
Наступною було реалізовано панель «Керування заняттям», що, фактично, представляє собою «праву» сторінку традиційного паперового журналу (рис. 3.14-3.16).

The screenshot shows the 'classbook' interface. The header includes a back arrow, the 'classbook' logo, and the user name 'Іванов Петро Степанович'. The main area is titled 'Керування заняттям' (Lesson Management) for the discipline 'Програмування' (Programming). On the left, there is a form with fields for 'Дата проведення заняття' (Lesson date), 'Вид заняття' (Lesson type), 'Кількість годин' (Hours), 'Тема заняття' (Topic), 'Домашнє завдання' (Homework), 'Підпис' (Signature), and 'Примітки' (Remarks). Below the form are buttons for 'Очистити поля' (Clear fields) and 'Зберегти зміни' (Save changes). The main table has columns: '№', 'Дата проведення', 'Вид заняття', 'Кількість годин', 'Тема заняття', 'Домашнє завдання', 'Підпис', and 'Примітки'. The table is currently empty, displaying 'No content in table'. At the bottom, there is a status bar with 'Кількість год. до вичитки: 104', a blue button 'Перейти до оцінювання/відвідуваності заняття' (Go to evaluation/attendance), and 'ФПК: екзамен'. A 'CONNECTION TO DATABASE STATUS' indicator is visible in the bottom right corner.

Рис. 3.14. Панель «Керування заняттям»

This screenshot shows the same 'classbook' interface as Figure 3.14, but with the form fields filled with data. The 'Дата проведення заняття' field contains '12.06.2020', 'Вид заняття' is set to 'Л', 'Кількість годин' is '2', 'Тема заняття' is 'Тема 1.', 'Домашнє завдання' is 'Домашнє завдання 1.', 'Підпис' is 'Іванов', and 'Примітки' is empty. The 'Очистити поля' button is now disabled. The table remains empty with 'No content in table'. The status bar at the bottom shows 'Кількість год. до вичитки: 104', the same blue button, and 'ФПК: екзамен'. The database connection status is still visible in the bottom right corner.

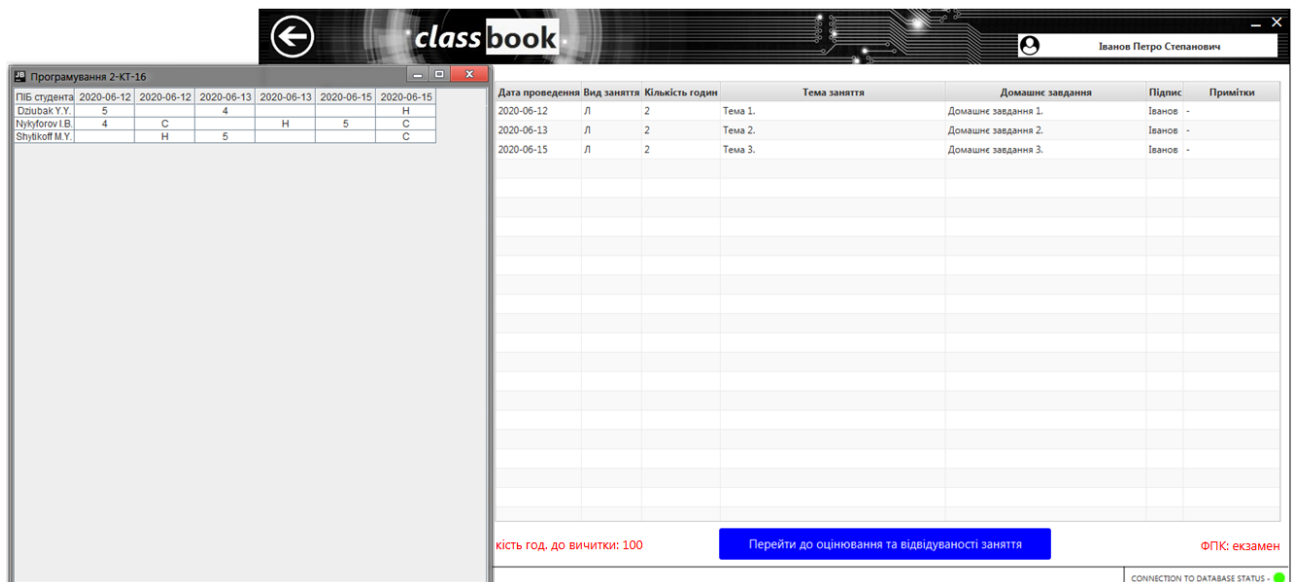
Рис. 3.15. Заповнення даних панелі «Керування заняттям»



ПІБ студента	2022-06-12	2022-06-12
Dziubak Y.Y.	5	
Nykyforov I.B.	4	С
Shytkoff M.Y.		Н

Рис. 3.19. Панель із сформованими даними обліку оцінювання/відвідуваності студентів згідно обраної дисципліни

Для зручнішого використання було додано можливість користування обома видами форм одночасно (рис. 3.20):



ПІБ студента	2020-06-12	2020-06-12	2020-06-13	2020-06-13	2020-06-15	2020-06-15
Dziubak Y.Y.	5		4			Н
Nykyforov I.B.	4	С		Н	5	С
Shytkoff M.Y.		Н	5			С

Дата проведення	Вид заняття	Кількість годин	Тема заняття	Домашнє завдання	Підпис	Примітки
2020-06-12	Л	2	Тема 1.	Домашнє завдання 1.	Іванов -	
2020-06-13	Л	2	Тема 2.	Домашнє завдання 2.	Іванов -	
2020-06-15	Л	2	Тема 3.	Домашнє завдання 3.	Іванов -	

Рис. 3.20. Паралельне використання панелей оцінювання/відвідуваності заняття і «Керування заняттям»

У першій релізній версії застосунку структура проекту мала наступний вигляд (рис. 3.21):

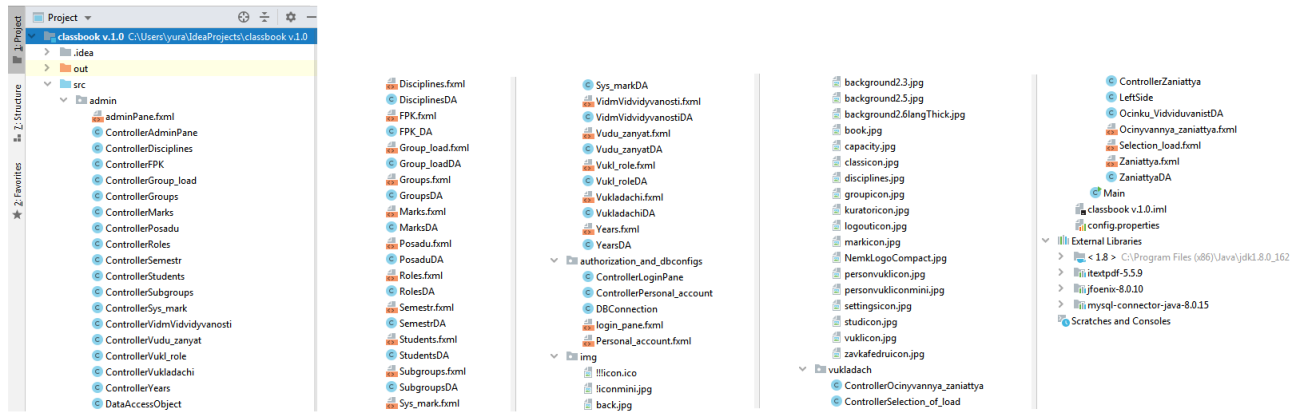


Рис. 3.21. Структура проекту застосунку

Далі відбувалась розробка засобів для можливості друку зформованих даних системи, для можливості створення на їх основі «традиційного» (фізичного) класного журналу, для подальшого його зшивання та розміщення в архіві навчального закладу.

3.4 Розробка методу друку "фізичного" журналу на основі даних системи

Першим етапом розробки методу друку сторінок журналу на основі даних системи була розробка алгоритму, що формує документи у форматі придатному для друку заповнених даних системи, враховуючи відповідність цього формату стандартам вигляду журналів у навчальному закладі, для чого потрібно було розробити відповідний алгоритм формування шаблону цих документів.

Для розробки описаних алгоритмів було обрано iTextPDF, що є бібліотекою для Java, за допомогою якої можна розробляти Java-програми, що створюють, конвертують та обробляють документи PDF-формату.

Враховуючи те що, друкувати необхідно «ліві» та «праві» сторінки журналу (рис. 2.2 - 2.3), потрібно було реалізувати відповідні шаблони для зформованих документів. У відповідних класах програми, що відповідають за заповнення сформованих даних «лівої» та «правої» сторінок журналу, було розміщено функції, що відповідають за формування відповідних шаблонів, заповнення цих шаблонів та подальше їх зберігання.

Причина такої реалізації в тому, що дані класи формують відповідні таблиці для відображення заповнених даних системи і вони є цифровими відображеннями «правої» та «лівої» сторінок журналу, що значить, що алгоритм для правильного формування цих сторінок не треба буде дублювати для створення друкопридатних, сформованих згідно шаблонів, документів – раціонально для цього використати інструменти бібліотек для зчитування даних, уже зформованих, відповідних цифрових таблиць (рис. 3.20).

У Додатку А в методі AddToPDF() класу ControllerZaniattya наведено код алгоритму, що формує "праву" сторінку журналу.

Даний алгоритм є більш простим аніж алгоритм, що формує «ліву» сторінку, тому що, «права» сторінка має фіксовану кількість стовпців, а кількість стовпців «лівої» прямо залежна від визначеної кількості годин дисципліни, а кількість цих годин – змінна. Тобто, можна описати це так – «ліва» сторінка, заповнюється у горизонтальній площині документу, а «права» сторінка у вертикальній.

Тому, така особливість заповнення даних серйозно ускладнює алгоритм заповнення шаблону «лівої» сторінки. Також, важливо враховувати те, що кількість навчальних занять на обох "сторінках" журналу має бути відповідною – тобто у випадку, якщо під час формування документу на одній із сторінок журналу закінчиться простір, то тоді інша сторінка теж повинна автоматично перейти на новий лист формованого документу.

Враховуючи усі наведені вище особливості і обмеження було реалізовано абсолютно новий і унікальний алгоритм заповнення сформованого (беручи як до уваги шаблон «правої») шаблону «лівої» сторінки, код якого наведено в Додатку Б у методі AddToPDF() класу LeftSide.

На панелях «Керування оцінюванням заняття» і «Керування заняттям» було додано відповідні кнопки (рис. 3.22), котрі викликають вище описані методи, у результаті натискання на які відбувається генерація PDF-документів «лівої» та «правої» сторінок журналу (рис. 3.23).



Рис. 3.22. Додані на панелі кнопки для генерації PDF-документів

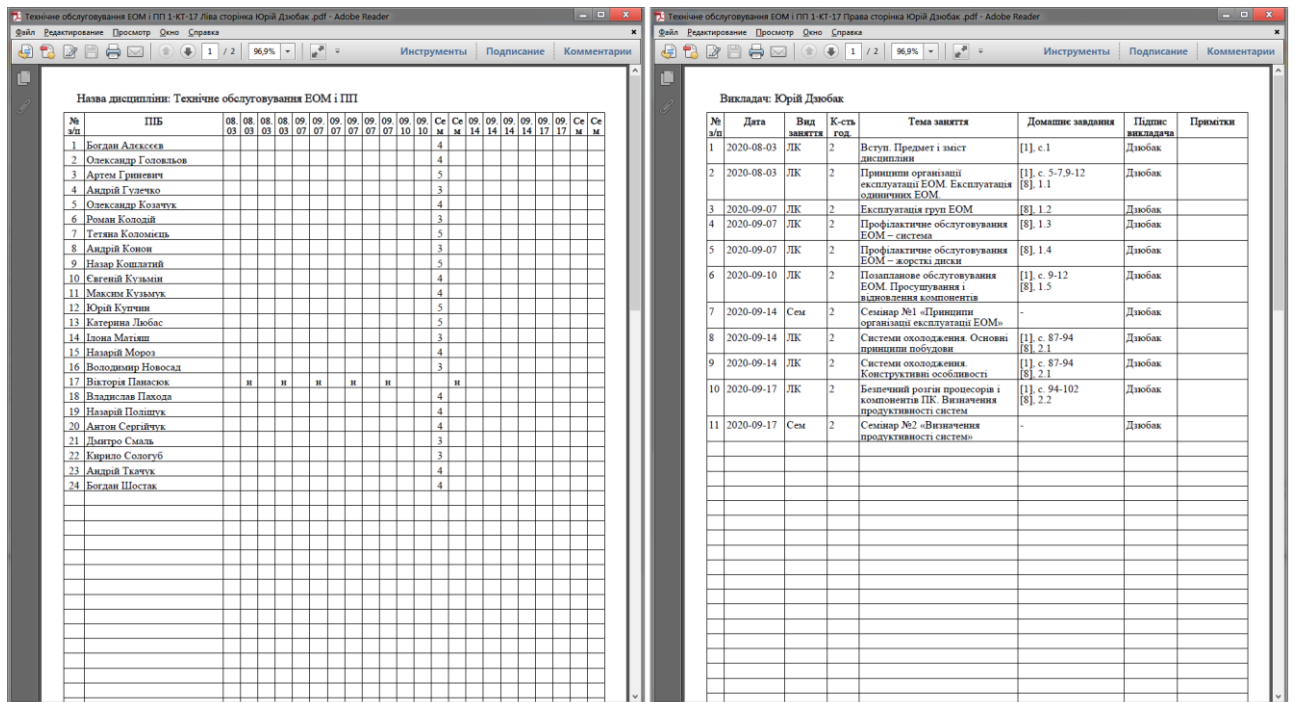


Рис. 3.23. Згенеровані на основі відповідних шаблонів і заповнених даних системи «ліва» та «права» сторінки журналу у PDF форматі

3.5 Компіляція програмного проекту у виконуваний десктоп-застосунок

Компіляція програмного проекту у виконуваний десктоп-застосунок було реалізовано за допомогою інтегрованих інструментів середовища розробки IntelliJ IDEA.

Інструменти для цього знаходяться за наступним шляхом: Project Structure -> Artifacts -> Add -> JavaFx Application -> From module 'classbook v.1.0' (рис. 3.24).

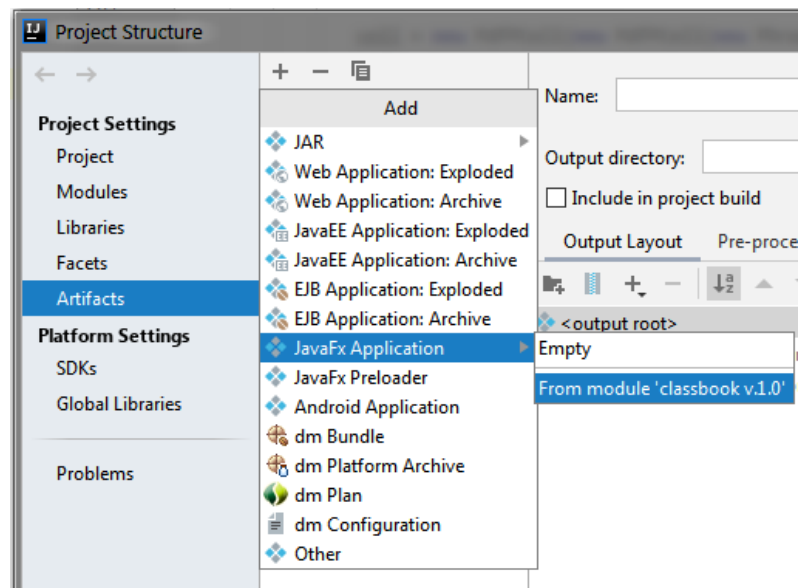


Рис. 3.24. Перший етап для збирання проекту

Потім потрібно було сформуванати список вихідних файлів проекту, для збирання їх у виконуваний файл (рис. 3.25).

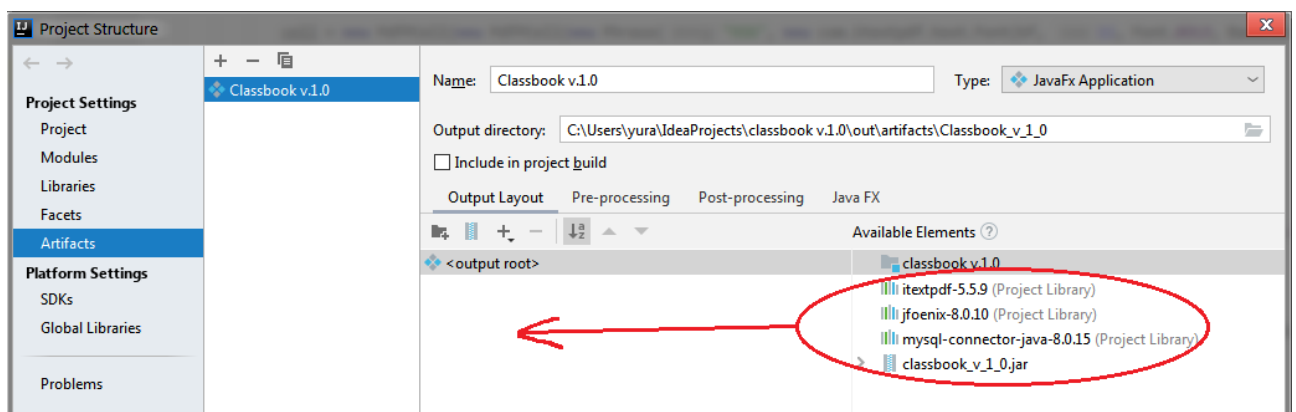


Рис. 3.25. Формування список вихідних файлів проекту для збирання їх у виконуваний файл

Наступним етапом задавались необхідні параметри і вказувався шлях до файлу іконки застосунку у відповідному меню параметрів (JavaFx) (рис. 3.26).

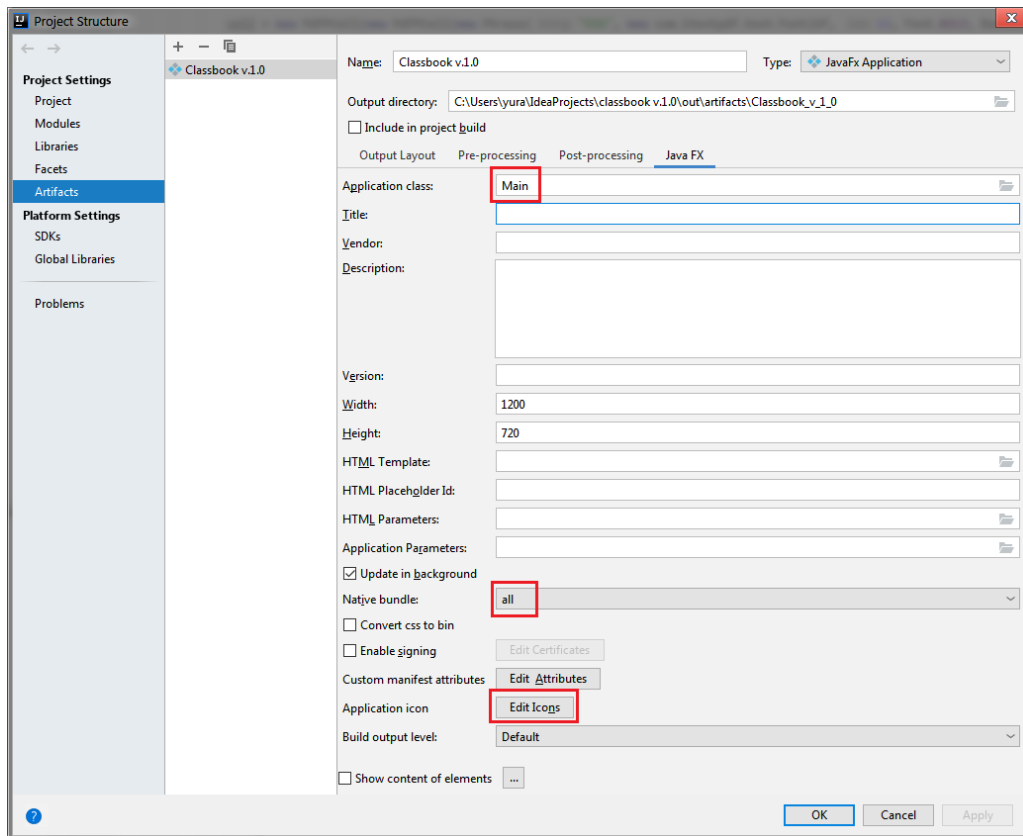


Рис. 3.26. Вибір необхідних параметрів у відповідному меню JavaFx

Тоді у вкладці Build → Build Artifacts виконувалась подія Build (рис. 3.27).

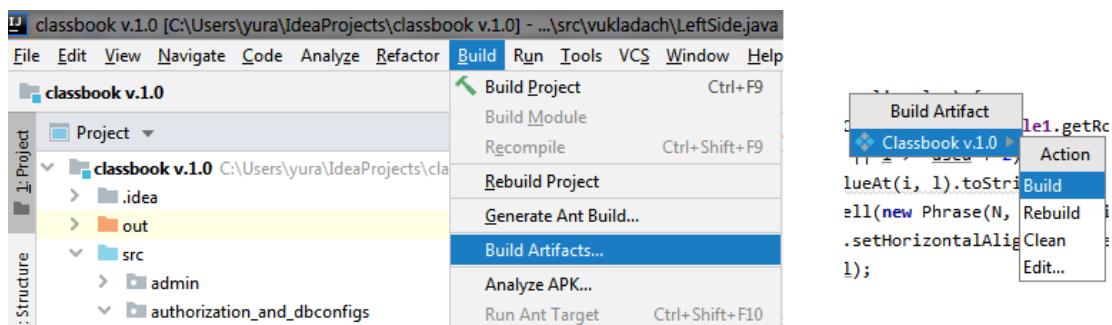


Рис. 3.27. Вибір функції Build Artifact

У випадку, коли все вказано вірно і виконання Build Artifact пройде без проблем, то у папці проекту в репозиторії out → artifacts з'явиться зібраний застосунок із виконуваним exe-файлом (рис. 3.28).

Для використання зібраного застосунку необхідно із кореня папки проекту у репозиторій app потрібно перенести config-файл із даними доступу до БД (рис. 3.29).

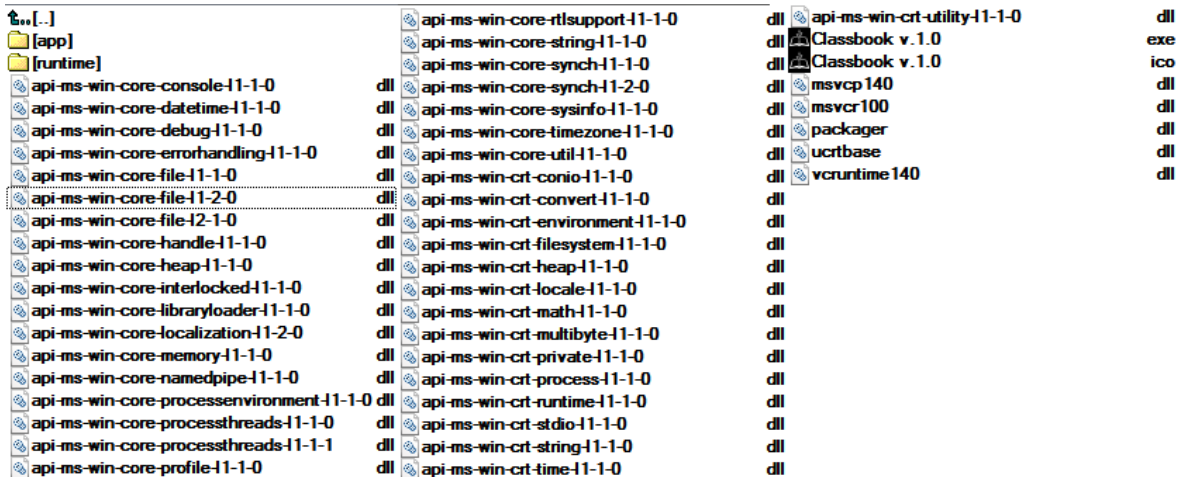


Рис. 3.28. Файлова структура зібраного застосунку із ехе-файлом

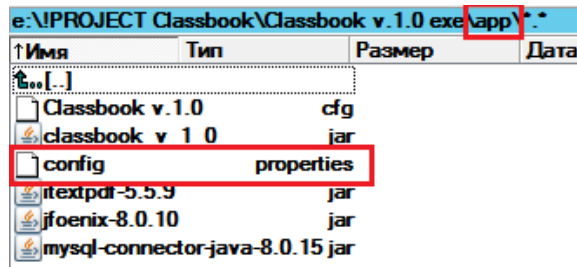


Рис. 3.29. Доданий в папку app config-файл із даними доступу до БД

Після компіляції застосунків буде придатним до використання на будь-якому ПК із ОС Windows 7+.

3.6 Встановлення та налаштування ОС і СУБД на сервер системи

Першим етапом, було встановлено і налаштовано систему віртуалізації VMware Workstation на вже існуючій конфігурації сервера. Потім в середовищі системи віртуалізації було створено і налаштовано нову віртуальну машину (рис. 3.30), котра буде використана для встановлення ОС для сервера комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти.

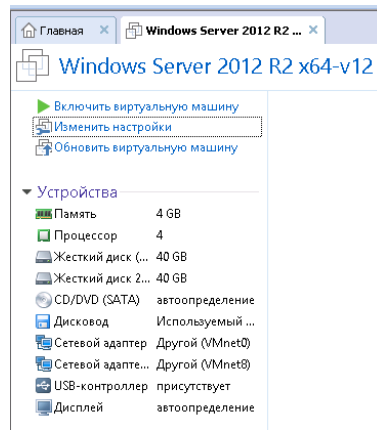


Рис. 3.30. Створена і налаштована віртуальна машина для встановлення ОС для сервера комп'ютеризованої системи

Наступним кроком було встановлення на створену віртуальну машину обраної ОС – Windows Server 2012 R2 із подальшим базовим налаштуванням його мережі і сервісів.

Потім було встановлено MySQL і його компоненти: MySQL сервер, СУБД WorkBench, MySQL Notifier та MySQL Connectors. Після чого було імпортовано dump-файл із сконфігурованою БД для комп'ютеризованої системи (рис. 3.31).

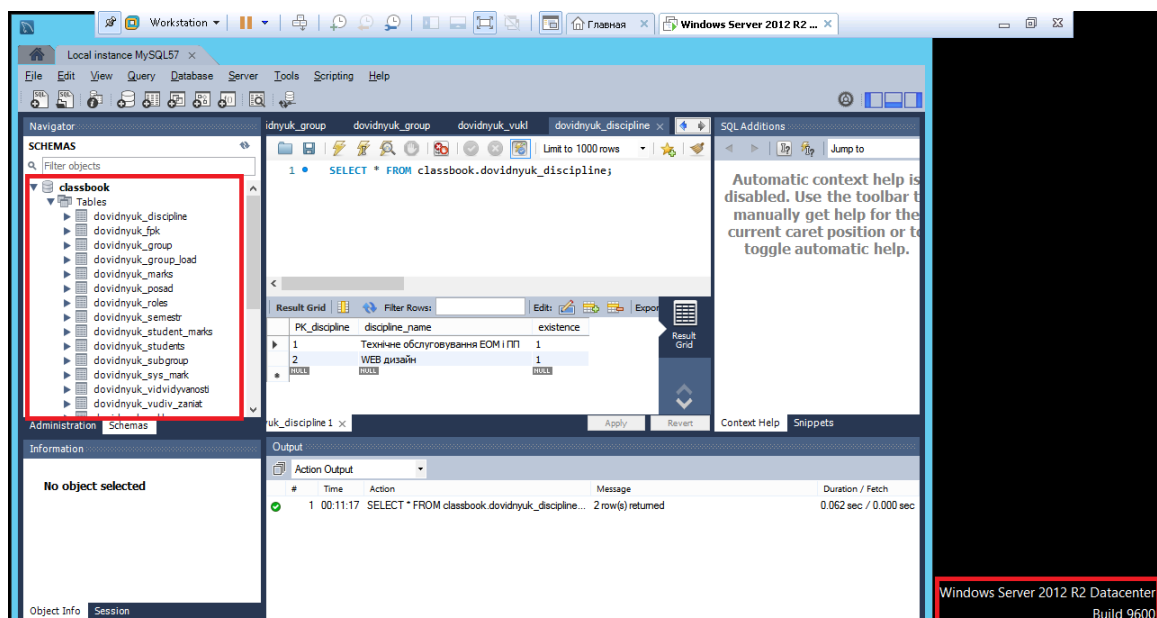


Рис. 3.31. Встановлений і налаштований MySQL із імпортованою БД комп'ютеризованої системи у встановленій ОС Windows Server 2012 R2

Завершальним кроком було створення користувача БД у середовищі MySQL із подальшим наданням йому прав на користування базою даних використовуючи наступні команди (рис. 3.32).

```
CREATE USER 'classbookuser'@'%' IDENTIFIED BY '*****';
GRANT ALL PRIVILEGES ON classbook.* to 'classbookuser';
```

Рис. 3.32. Лістинг SQL-скриптів наданням для надання прав користувачу на користування базою даних

Вкінці було здійснено перевірку зв'язку клієнтського застосунку із налаштованим сервером системи (рис. 3.33).

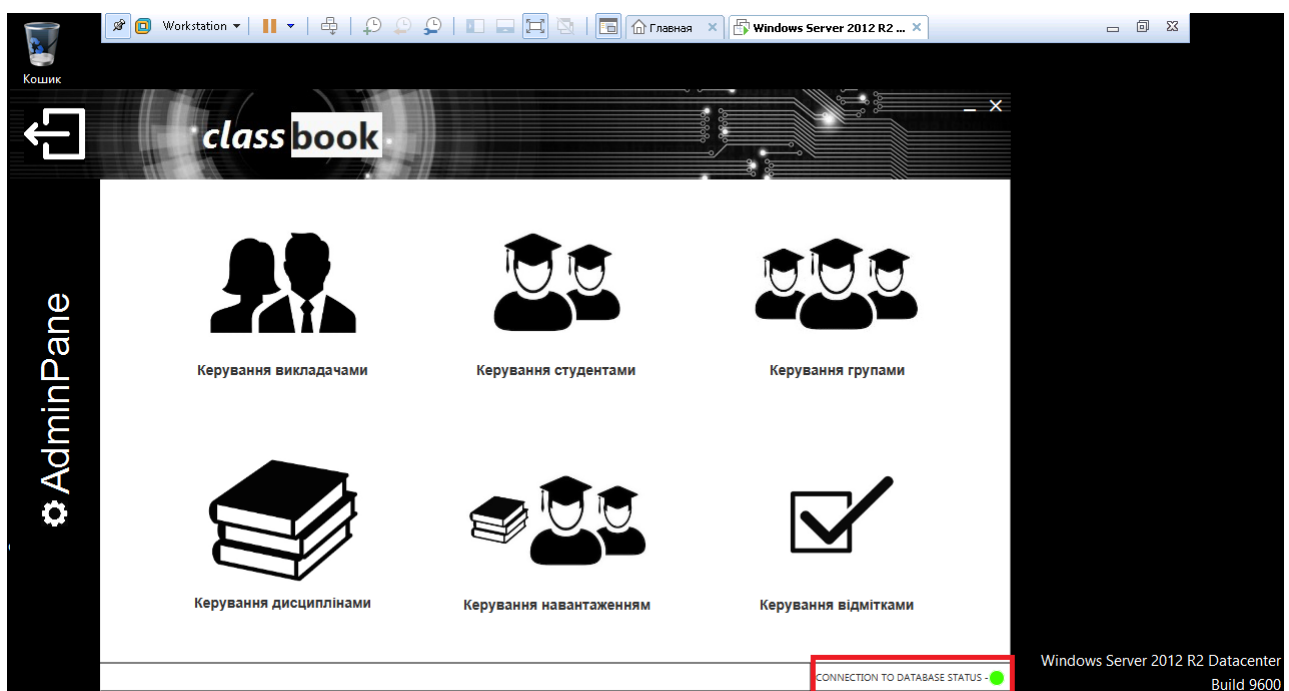


Рис. 3.33. Індикація успішно встановленого зв'язку клієнтського застосунку із серверною частиною системи

3.7 Тестування та внесення корективів в систему

Тестування системи відбувалося і безпосередньо під час розробки функціональних частин, так і після реалізації всього функціоналу застосунку, в

ході нього було виявлено і «пофікшено» кілька «багів», та скориговано/модифіковано певні моменти:

1. Було вирішено проблему зсуву іd записів при видаленні записів із довідкової таблиці, котрий призводив до некоректного вибору Foreign Key для запитів системи і елементів інтерфейсу типу Combobox. Вирішенням даної проблеми стало додавання в усі таблиці поля «активність» властивості boolean з наступними варіантами: «1» – запис використовується, «0» – запис видалений. Відповідно, тепер замість фізичного видалення запису, він буде приховуватись (подібні рішення використовується у корпоративних БД для збереження цілісності даних) (згадка про це була у підрозділі 2.2).

Тепер, до основного змісту запису, що видаляється додається атрибут *(DELETED) (рис. 3.34).

	id_year	year_name	existence
▶	1	/	1
	2	2020-2020	1
	3	2019-2020 (DELETED)	0
*	NULL	NULL	NULL

Рис. 3.34. Поява атрибуту *(DELETED) для запису, який було «видалено»

2. У початковій версії БД існувало три поля «Прізвище», «Ім'я» та «По батькові» для користувачів системи. Було прийнято рішення об'єднати дані поля в одне – «ПІБ», бо окремо ці поля ніде не використовуються, а дана зміна дала можливість зручніше використовувати ці дані в довідкових цілях.

3. Було додано наступні функціональні кнопки до усіх довідкових підформ (рис. 3.35) для виконання відповідних операцій.

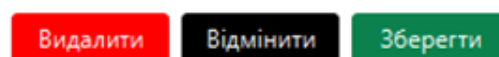


Рис. 3.35. Додані функціональні кнопки до довідкових підформ

4. Виявлено та «пофікшено» два «баги» пов'язані із редагуванням записів:

1) у момент, коли запис вибирався для редагування, після чого редагувався і зберігався, то залишався зв'язок даних із полями введення, і, у випадку коли одразу після виконаних дій відбувалась була спроба введення нових даних у очищені поля введення, то ці дані змінювали ті, що було перед цим відредаговані. Вирішення проблеми стало додаванням функціонального блоку коду, що має забезпечувати унеможливлення умови можливості помилкового редагування даних після завершення операції їх зберігання, яка іноді випадково спрацьовувала;

2) під час вибору редагування якогось існуючого запису некоректно відображались id записів у Combobox, що провокувало помилку ключа. Вирішено дану проблему за допомогою додаткового ініціювання елементів Combobox, що було описано у методі редагування.

5. Було прийнято рішення прибрати форму відображення «лівої» сторінки журналу з використанням таблиці JTable, а замість неї використовувати опцію формування PDF-документа, бо так значно зручніше переглядати зведену відомість оцінювання/відвідуваності, бо відображення даної інформації у такому варіанті є більш звичною для викладачів. Але генерація таблиці JTable було залишено через необхідність її використання як шаблону для формування PDF-документу «лівої» сторінки.

Даний список є далеко не повним відображенням того, що було модифіковано і виправлено в ході розробки, бо розвиток проекту був дуже стрімким і він продовжується й надалі.

Версія комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти, що описана у даній роботі є пілотним тестовим проектом, котрий є базою для подальшого розвитку і модернізації даної системи.

Введення комп'ютеризованої системи в повноцінну експлуатацію планувалось після розробки веб-версії із безпековими політиками для користування системою через Інтернет та інтерфейсу для студентів. В даний час систему, що є розвитком даного пілотного проекту уже введено у повноцінну

експлуатацію у ДВНЗ НЕМК, чим було повністю замінено ведення традиційних паперових журналів груп.

3.8 Висновки до розділу

В даному розділі було описано процес розробки комп'ютеризованої системи обліку успішності та відвідуваності занять здобувачами освіти, а саме:

- здійснено опис методів та інструментів для розробки системи;
- описано реалізацію інтерфейсів клієнтського застосунку;
- описано процес реалізації методу друку "фізичного" журналу на основі даних системи;
- описано процес компіляції програмного проекту у виконуваний десктоп-застосунок;
- описано встановлення та налаштування ОС і СУБД на сервер системи;
- описано результати тестування та внесені корективи в систему.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

Під час розробки алгоритмічного та програмного забезпечення комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти усі процеси повинні виконуватись із врахуванням вимог техніки безпеки на робочому місці, пожежної безпеки, відповідно з діючими нормативно-правовими актами та встановленими нормами щодо роботи з комп'ютерною технікою. Дотримання норм і правил охорони праці є важливим аспектом у контексті дотримання норм організації робочого місця, забезпечення комфортних та зручних умов праці осіб, які беруть участь у процесі, а це вимагає дотримання вимог з охорони праці при роботі на персональному комп'ютері.

Основними регламентуючими нормативними документами охорони праці користувачів ПК є Положення про розробку інструкцій з охорони праці, затвердженого наказом Держпраці від 29.01.1998 № 9, Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держпраці від 26.01.2005 № 15, Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями, затверджених наказом Мінсоцполітики від 14.02.2018 № 207, Державні санітарні правил і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПіН 3.3.2.007-98, затверджені постановою Головного державного санітарного лікаря України від 10.12.1998 № 7, Загальні вимоги стосовно забезпечення роботодавцями охорони праці працівників, затверджені наказом ДСНС від 25.01.2012 № 67 (НПАОП 0.00-7.11-12).

Згідно даних нормативних документів працівника, який використовує персональний комп'ютер (далі — користувач), інструктують перед початком

роботи (первинний інструктаж), а потім через кожні 6 місяців (повторний інструктаж). Результати інструктажу заносять до Журналу реєстрації інструктажів з питань охорони праці на робочому місці (у журналі має бути підпис особи, яка інструктує, та користувача).

Користувач зобов'язаний дбати про особисту безпеку і здоров'я, а також про безпеку і здоров'я навколишніх при виконанні будь-яких робіт, а також під час перебування на території підприємства.

До роботи на персональному комп'ютері допускають осіб, які пройшли інструктажі з питань охорони праці та пожежної безпеки.

Користувач зобов'язаний:

- виконувати правила внутрішнього трудового розпорядку;
- не допускати за своє робоче місце сторонніх осіб;
- не виконувати вказівок, які суперечать правилам охорони праці та пожежної безпеки;
- знати правила надання домедичної допомоги;
- знати розташування та вміти користуватись первинними засобами пожежогасіння;
- вміти працювати з ПК.

Основні небезпечні та шкідливі виробничі фактори, що можуть впливати на користувача:

- підвищений рівень статичної електрики;
- нерівномірність розподілу яскравості в полі зору;
- підвищена яскравість світлового зображення;
- ураження електричним струмом;
- напруження зору та уваги;
- тривалі статичні навантаження.

У приміщеннях із ПК має бути природне і штучне освітлення.

При розміщенні робочих місць необхідно унеможливити пряме засвічування екрана природним освітленням.

При природному освітленні слід передбачити наявність сонцезахисних засобів (плівка, жалюзі, штори тощо).

Світлові відблиски із клавіатури, екрана та інших частин ПК у напрямку очей користувача неприпустимі.

Основним обладнанням робочого місця є ПК або ноутбук, монітор, клавіатура, маніпулятор, робочий стіл, стілець (крісло).

При розміщенні елементів робочого місця слід враховувати:

- робочу позу користувача;
- простір для розміщення користувача;
- можливість огляду елементів робочого місця;
- можливість огляду простору поза межами робочого місця;
- можливість робити записи, розміщувати на робочому столі документацію та матеріали, які використовує користувач.

Розміщення елементів робочого місця не має заважати рухам та переміщенню для експлуатування ПК.

Монітор встановлюють так, щоб відстань від поверхні екрана до очей користувача була 600-700 мм залежно від розміру екрана.

Клавіатуру розміщують на робочому або окремому столі на відстані 100-300 мм від краю з боку користувача. Положення клавіатури та кут її нахилу залежить від побажання користувача (як правило, в межах 5-15°). Не допускати хитання клавіатури.

Конструкція робочого столу має бути такою, щоб оптимально розмістити на робочій поверхні обладнання, що використовують, з урахуванням кількості, розмірів, конструктивних особливостей і характеру його роботи.

Крісло має забезпечувати підтримання раціональної робочої пози під час виконання основних виробничих операцій та можливість зміни пози. Тип робочого крісла обирають залежно від характеру та тривалості роботи.

Раціональна поза користувача:

- ступні розташовані на підлозі або на підставці для ніг;
- стегна зорієнтовані у горизонтальній площині;

- верхні ділянки рук вертикальні;
- кут ліктьового суглоба у межах 70—90°;
- зап'ястя зігнуті під кутом не більше ніж 20°;
- нахил голови у межах 15—20°, а часті її повороти виключені.

Для забезпечення оптимальної робочої пози користувача необхідно:

- засоби праці, з якими користувач має тривалий або найбільш частий зоровий контакт, розмістити у центрі зони зорового спостереження та моніторного поля;
- забезпечити відстань близько 500 мм між найважливішими засобами праці, з якими користувач працює найчастіше.

ПК встановлювати на рівній твердій поверхні (столі). Не дозволено встановлювати ПК та оргтехніку на хитких підставках чи на похилій поверхні.

ПК не встановлювати впритул до стіни, перегородки тощо. Не допускати загородження вентиляційних отворів ПК сторонніми предметами.

Розетка біля ПК має бути в доступному місці, щоб в аварійних випадках можна було своєчасно його відімкнути від електромережі. Не рекомендовано використовувати подовжувачі.

Під час переміщення ПК, периферійних пристроїв витягти вилку живлення з розетки.

Не допускати ушкодження чи модифікування шнура живлення. Заборонено ставити важкі речі на шнур живлення, тягнути чи надмірно перегинати його, скручувати та зав'язувати шнур живлення у вузол.

ПК під'єднувати до електромережі лише за допомогою справних штепсельних з'єднань та електророзеток заводського виробництва.

Штепсельні з'єднання та електророзетки мають бути зі спеціальними контактами для під'єднання нульового захисного провідника. Їхня конструкція має забезпечувати з'єднання нульового захисного провідника раніше, ніж з'єднання фазового та нульового робочого провідників. Порядок роз'єднань при вимкненні має бути зворотнім.

Заборонено під'єднувати електрообладнання до звичайної двошнурової електромережі.

За невиконання норм і вимог щодо охорони праці на підприємстві працівники можуть нести відповідальність згідно з чинним законодавством.

4.2 Забезпечення безпеки життєдіяльності при роботі з ПК

Заходи щодо усунення небезпеки ураження електричним струмом зводяться до правильного розміщення устаткування та електричних кабелів. Інші заходи щодо забезпечення електробезпеки, збігаються з загальними заходами пожежо- та електробезпеки.

В якості профілактичних заходів для забезпечення пожежної безпеки слід використовувати приховану електромережу, надійні розетки з пожежобезпечних матеріалів, силові мережі живлення устаткування оснащувати кабелями, розрахованими на підключення в 3-5 разів більшого навантаження, включати й виключати живлення обладнання за допомогою штатних вимикачів. Треба регулярно робити очистку внутрішніх частин комп'ютерів, іншого устаткування від пилу, розташовувати комп'ютери на окремих столах.

Екран дисплея повинен бути розташованим перпендикулярно до напрямку погляду. Якщо він розташований під кутом, то стає причиною сутулості. Рекомендовано сидіти на відстані витягнутої руки від монітора. Ще одним моментом, який стосується зору, є необхідність створення неоднорідного поля зору. Для цього можна розвісити на поверхнях (стінах) плакати та картини, виконані у спокійних тонах. Наприклад, пейзажі.

Важливою є форма спинки крісла, яка повинна повторювати форму спини. Висота крісла повинна бути такою, щоб користувач не почував тиску на куприк або стегна. Крісло бажано обладнати бильцями. Його потрібно встановити так, щоб не треба було тягтися до клавіатури. Періодично

користувачу необхідно рухатися, вчасно змінювати положення тіла і робити перерви у роботі.

При напруженій роботі за комп'ютером щогодини необхідно робити перерву на 15 хвилин через кожен годину і треба займатися іншою справою. Декілька разів на годину бажано виконувати серію легких вправ для розслаблення.

Режим праці та відпочинку при роботі з персональною електронно-обчислювальною машиною (ПЕОМ) залежить від категорії трудової діяльності. Всі роботи з ПЕОМ ділять на три категорії. Перша - епізодичне зчитування і робота з інформацією не більше 2-х годин за 8-годинну робочу зміну. Друга - зчитування інформації або творча робота не більше 4-х годин за восьми годинну зміну. Третя - зчитування інформації або творча робота тривалістю більше 4-х годин за зміну.

Отже, щоб запобігти негативним впливам необхідно знати й небезпечні сторони самого комп'ютера і правила безпечної роботи, знати засоби запобігання небезпек. Вони пов'язані перед усім із загально відомими небезпечними факторами - ураженнями електричним струмом та пожежонебезпечністю.

4.3 Моделювання уразливості об'єкта економіки та його елементів до дії вторинних вражаючих факторів ядерного вибуху

Визначення стійкості роботи об'єкта економіки та його елементів при дії вторинних вражаючих факторів ядерного вибуху може бути виконано за допомогою моделювання уразливості об'єкта при впливі вражаючих факторів такого типу надзвичайної ситуації (далі - НС) на основі використання результатів розрахункових даних.

Основними вражаючими факторами НС такого типу є: повітряна ударна хвиля, світлове випромінювання, проникаюча радіація, радіоактивне зараження та електромагнітний імпульс. Всі ці вражаючі фактори можуть в різній мірі

впливати на функціонування об'єкту після НС. Тому оцінювати стійкість об'єкта потрібно по відношенню до кожного з вражаючих факторів.

Вражаючими вторинними факторами від ядерних вибухів небезпечному віддаленні від об'єктів економіки є: пожежі, спровоковані вибухи, затоплення, забруднення атмосфери та місцевості і т. ін. Втрати від вторинних вражаючих факторів у ряді випадків можуть значно перебільшувати втрати, які можуть бути отримані в результаті дії основних факторів НС.

Джерела вторинних вражаючих факторів на об'єкті й в небезпечному віддаленні від нього повинні виявлятися заздалегідь з метою завчасного прийняття заходів, що направлені на виключення чи зменшення вражаючої дії.

Чи не основним вторинним вражаючим фактором ядерного вибуху є радіоактивне зараження, тому, воно є одним із основних до врахування при моделюванні НС такого характеру. Як критерій стійкості об'єкта до впливу радіоактивного зараження приймається граничне значення рівня радіації на об'єкті на початку радіоактивного зараження, при якому можлива виробнича діяльність у звичайному режимі (повними змінами, повний робочий день) і при цьому персонал не отримає дозу опромінення більше чи допустимої.

Прогнозування можливого становища на об'єкті проводиться в такій послідовності:

- виявляють всі можливі джерела вражаючих факторів, як внутрішні, так і зовнішні;
- визначають найкоротшу відстань від об'єкта до кожного джерела вторинного ураження (на місцевості або на мапі чи плані);
- визначають характер вражаючої дії вторинного фактора (пожежа, затоплення, загазованість т. ін.);
- встановлюють чи вираховують час від моменту появи до моменту початку дії на об'єкт вторинного вражаючого фактора;
- визначають тривалість дії вражаючого фактора й можливі розміри втрат.

Одержані результати аналізують і роблять конкретні висновки для розробки організаційних, інженерно-технічних та технологічних заходів щодо виключення або обмеження дії на роботу об'єкта вторинних вражаючих факторів складючи шляхи підвищення стійкості роботи підприємств в надзвичайних ситуаціях, такі як:

- забезпечення надійного захисту робітників і службовців від вражаючих факторів НС;
- захист основних виробничих фондів від вражаючих факторів НС, в тому числі і від вторинних;
- забезпечення сталого постачання всім необхідним для випуску запланованої продукції;
- підготовка до відновлення порушеного виробництва;
- підвищення надійності і оперативності управління виробництвом.

Підвищення стійкості роботи об'єкта народного господарства в НС досягається завчасним проведенням комплексу інженерно-технічних, технологічних і організаційних заходів, спрямованих на максимальне зниження впливу вражаючих факторів НС і створення умов для швидкої ліквідації наслідків.

Основні критерії, за якими здійснюється якісна оцінка потенційних наслідків для кожного небезпечного стану об'єкту економіки, наступні:

- клас 1 — безпечний: не спричиняє незворотних наслідків, пошкодження обладнання та нещасних випадків з людьми;
- клас 2 — граничний: призводить до порушень в роботі, може бути компенсованим або взятим під контроль без ушкодження обладнання або нещасних випадків з персоналом;
- клас 3 — критичний: призводить до великих порушень у роботі, ушкодження обладнання або створення небезпечної ситуації, яка потребує негайних заходів для порятунку персоналу;
- клас 4 — катастрофічний: призводить до втрати обладнання та (чи) загибелі без масового травмування персоналу.

Отже, моделювання можливого становища на об'єкті економіки під впливом вторинних вражаючих факторів дозволяє ефективніше розробити заходи щодо захисту персоналу та самого об'єкта економіки.

4.4 Висновки до розділу

В даному розділі було:

- проведено огляд санітарно-гігієнічних правил і норм роботи з ПК;
- проведено огляд правил пожежної і електро- безпеки при роботі з ПК;
- проведено огляд ергономічних стандартів та регламентованих перерв для відпочинку при організації праці, пов'язаної з використанням ПК;
- проведено аналіз основних та вторинних вражаючих факторів НС (ядерного вибуху);
- проведено аналіз та оцінку стійкості роботи об'єкта економіки за допомогою моделювання уразливості об'єкта при впливі вторинних вражаючих факторів НС (ядерного вибуху);
- проведено аналіз шляхів підвищення стійкості роботи об'єктів економіки в режимі НС (наслідків ядерного вибуху).

ВИСНОВКИ

В результаті виконання даної кваліфікаційної роботи було розроблено алгоритмічне та програмне забезпечення комп'ютеризованої системи обліку успішності та відвідування занять здобувачами освіти.

Основні наукові та практичні результати роботи полягають у наступному:

1. Проаналізовано особливості організації обліку процесу оцінювання та відвідуваності у вищому навчальному закладі для складання переліку вимог і задач, котрі має вирішувати розроблювана система;

2. На основі результатів аналізу предметної області спроектовано логічну модель інформаційних об'єктів розроблюваної системи для подальшого проектування на їх основі бази даних системи;

3. Розроблено архітектуру та алгоритми роботи системи обліку успішності та відвідуваності занять здобувачами освіти для подальшої розробки програмного забезпечення системи;

4. Обрано та обґрунтовано вибір технологій та інструментів для реалізації системи для того, щоб розробити систему у відповідності до поставлених вимог та задач, котрі вона має вирішувати;

5. Розроблено та протестувано комп'ютеризовану систему обліку успішності та відвідуваності занять здобувачами освіти, що створена для покращення процесу ведення обліку успішності на відвідуваності навчальних занять у навчальному закладі;

6. Створено унікальний метод формування документів сторінок журналу на основі заповнених у систему даних які придатні до друку і формування фізичних журналів;

7. Даний проект має хороші перспективи і можливості подальшого розвитку і вдосконалення, частина з яких вже втілена в життя, бо в уже даний час систему, що є розвитком даного пілотного проекту вже введено у повноцінну експлуатацію у ДВНЗ НЕМК, чим було повністю замінено ведення традиційних паперових журналів груп.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Офіційний веб-сайт Міністерства освіти і науки України. ДЕРЖАВНА БЕЗКОШТОВНА СИСТЕМА ЕЛЕКТРОННИХ ЖУРНАЛІВ І ЩОДЕННИКІВ [Електронний ресурс]. – Режим доступу: mon.gov.ua/ua/osvita/zagalna-serednya-osvita/derzhavna-bezkoshtovna-sistema-elektronnih-zhurnaliv-i-shodennikiv (дата звернення 06.12.2023).
2. Офіційний веб-сайт освітнього ресурсу Moodle. Головна [Електронний ресурс]. – Режим доступу: moodle.org/?lang=uk (дата звернення 06.12.2023).
3. Офіційний веб-сайт платформи E-schools. Електронні щоденники та журнали [Електронний ресурс]:. – Режим доступу: e-schools.info/e-service (дата звернення 06.12.2023).
4. Офіційний веб-сайт компанії «Щоденник». Щоденник.ua [Електронний ресурс]. – Режим доступу: [company.shodennik.ua service](http://company.shodennik.ua/service) (дата звернення 06.12.2023).
5. Київський національний економічний університет імені Вадима Гетьмана. Журнал успішності студентів [Електронний ресурс]. – Режим доступу: kneu.edu.ua/ua/Information_for/students/jurnal/ (дата звернення 06.12.2023).
6. Офіційний веб-сайт Буковинського державного медичного університету. Електронний журнал успішності [Електронний ресурс]. – Режим доступу: www.bsmu.edu.ua/studentu/elektronniy-zhurnal-uspishnosti/ (дата звернення 06.12.2023).
7. Офіційний веб-сайт Національного педагогічного університету імені М.П. Драгоманова. Пс-журнал успішності-web [Електронний ресурс]. – Режим доступу: ntu.pri.edu.ua (дата звернення 06.12.2023).
8. Діденко О. В., Купрієнко Д. А. Інформаційні технології і засоби навчання: електронний журнал обліку успішності слухачів (курсантів, студентів) як засіб раціоналізації навчально-виховного процесу [Електронний ресурс]. – Том 47, №3. – 2015. – Режим доступу: www.google.com.ua/url?sa=t&rct=j&q=&esrc=s&source=web&cd=13&cad=rja&uact=8&ved=2ahUKewj6

pPCn9sToAhUml4sKHdz6CLsQFjAMegQIAxAB&url=https%3A%2F%2Fjournal.iitta.gov.ua%2Findex.php%2Filt%2Farticle%2Fview%2F1226%2F924&usg=AOvVaw3dg-f9VyOC4CDFskb-naVc (дата звернення 06.12.2023).

9. Ютуб-канал «kenLovesToCode». Цикл відео JavaFX CRUD [Електронний ресурс]. – Режим доступу: www.youtube.com/channel/UCopWb4iPes0SDAVD1n7CQmw (дата звернення 06.12.2023).

10. Vos, Johan; Gao, Weiqi; Chin, Stephen; Iverson, Dean; Weaver, James L. Pro JavaFX 8: A Definitive Guide to Building Desktop, Mobile, and Embedded Java Clients. Apress. p. 616. ISBN 978-1-4302-6574-0

11. Date, Chris. Database in depth : relational theory for practitioners. O'Reilly. ISBN 0-596-10012-4.

12. Eckel, Bruce. Thinking in Java. Publisher, Pearson Education, 2006 ; ISBN, 8131705579, 9788131705575 ; 1512 pages.

13. Лещишин Ю. З., Романишин Н.Р., Наконечний В. В., Паламарчук А.О. Розробка системи зв'язку як інтегрованого елемента роботизованих систем // Зб. тез доповідей XXI Всеукр. наук.-пр. конф. Житомир, 2016. С. 102.

14. Марків В.А., Осухівська Г.М., Лещишин Ю.З., Луцків А.М. Комп'ютерна система аутентифікації осіб // Матеріали XX наукової конференції ТНТУ ім. І. Пулюя. 2017. С. 90–91.

15. Leschyshyn Y., Scherbak L., Nazarevych O., Gotovych V., Tymkiv P., Shymchuk G. Multicomponent Model of the Heart Rate Variability Change-point // IEEE XVth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH). 2019. P. 110–113.

16. Tymkiv P., Leshchyshyn Y. Algorithm Reliability of Kalman Filter Coefficients Determination for Low-Intensity Electroretinosignal // IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM). 2019. P.1-5.

17. Leschyshyn Y., Semchyshyn O. Periodically correlated heart rate variability detection by Neyman - Pearson criterion // 9th International Conference - The Experience of Designing and Applications of CAD Systems in Microelectronics.

2007. Р. 139–140.

18. Геврик Є.О. Охорона праці. – К.: Ельга, Ніка-Центр, 2003. 280 с.
19. Стадник І.Я., Зварич Н.М. “Оцінка хімічної обстановки при аваріях на хімічно небезпечних об’єктах з викидом (виливом) небезпечних хімічних речовин та застосуванні хімічної зброї ” ТНТУ, 2020. 36С.
20. Методичні рекомендації до виконання кваліфікаційної роботи магістра / Укладачі Лупенко С.А., Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. – Тернопіль: Вид-во ТНТУ ім. І. Пулюя. – 2021. – 34 с.

Додаток А

Тези конференцій

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний технічний університет імені Івана Пулюя (Україна)
Університет імені П'єра і Марії Кюрі (Франція)
Маріборський університет (Словенія)
Технічний університет у Кошице (Словаччина)
Вільнюський технічний університет ім. Гедимінаса (Литва)
Міжнародний університет цивільної авіації (Марокко)
Наукове товариство ім. Т.Шевченка

АКТУАЛЬНІ ЗАДАЧІ СУЧАСНИХ ТЕХНОЛОГІЙ

Збірник
тез доповідей

**XII Міжнародної науково-практичної
конференції молодих учених та студентів**
6-7 грудня 2023 року



УКРАЇНА
ТЕРНОПІЛЬ – 2023

УДК 004

Ю. Ю. Дзюбак; Ю. З. Лещишин, к.т.н., доц.

(Тернопільський національний технічний університет імені Івана Пулюя)

**ОБГРУНТУВАННЯ ДОЦІЛЬНОСТІ СТВОРЕННЯ КОМП'ЮТЕРИЗОВАНОЇ
СИСТЕМИ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ
ЗДОБУВАЧАМИ ОСВІТИ ПІД ПОТРЕБИ ОКРЕМОГО ВИЩОГО
НАВЧАЛЬНОГО ЗАКЛАДУ**

Y. Y. Dziubak; Y. Z. Leschyshyn, Ph.D., Assoc. Prof.

**JUSTIFICATION OF THE FEASIBILITY OF CREATING A COMPUTERIZED
ACCOUNTING SYSTEM FOR SUCCESS AND ATTENDANCE BY STUDENTS AS
REQUIRED BY AN INDIVIDUAL HIGHER EDUCATIONAL INSTITUTION**

На комерційному рівні є велика кількість варіантів реалізації комп'ютеризованих систем (КС) обліку успішності та відвідування занять здобувачами освіти. В західних країнах діджиталізація навчального процесу є розвинутішою, чим в навчальних закладах України. Причиною є те, що наша країна стоїть на початку розвитку інформаційних технологій для сфери освіти. Проте гостра необхідність віддаленої освіти стимулює діджиталізацію освітньої діяльності, введення комп'ютерно-інформаційних освітніх систем.

Оскільки, ступінь розвитку діджиталізації освітньої діяльності тільки починає набирати оберти в Україні, кількість подібних сервісів ще не така значна, як за кордоном. При цьому, багато установ, які вводять дистанційне навчання і сервіси автоматизації навчального процесу, є відділеннями зарубіжних інститутів і середніх навчальних закладів.

Тому доцільність створення системи такого плану під потреби окремого, конкретно виділеного, вищого навчального закладу має дві сторони – сторону переваг створення індивідуальної системи і, водночас, сторону проблем, котрі необхідно буде вирішити задля того, аби така система могла з'явитися та успішно функціонувати.

Переваги створення КС для конкретного вищого навчального закладу:

- система, на відміну від універсальних аналогів, буде побудована конкретно за побажаннями замовників, з урахуванням усіх необхідних особливостей у своїй структурі та функціоналі;
- система буде унікальною;
- створення такої системи студентами, котрі і навчаються у навчальному закладі, для якого буде створюватися дана КС, або хоча б їх залучення до створення такого продукту у певній мірі підвищить престижність даного навчального закладу та їх мотивацію та конкурентоздатність на ринку праці;
- у разі створення системи власними зусиллями на її розробку піде мінімум матеріальних ресурсів;
- створену систему можна буде вдосконалювати та функціонально доповнювати власними силами;
- створену систему можна буде зробити універсальною або модифікувати під конкретні потреби іншого навчального закладу і у такому випадку, ймовірно, її можна буде комерціалізувати.

Недоліки (проблеми) створення КС для конкретного вищого навчального закладу:

- складність створення подібної системи власними силами;
- великі затрати часу при створенні подібної системи власними силами;
- при замовленні подібної унікальної системи у професіоналів головним недоліком буде ціна, бо якісний унікальний продукт з супроводом відповідно коштує;
- вірогідність невдачі та недостатньої досконалості створеної системи власними силами (відсутність досвіду та професійності у виконавців) або навіть при замовленні у професіоналів;
- значні витрати часу та зусиль на налагодження системи та її тестування.

Проте, зважаючи на особливості навчального процесу в кожному вищому навчальному закладі та при наявності відповідних ресурсів все ж доцільним є створення власного програмного продукту із можливістю наступної модифікації та адаптації до змінних потреб.

УДК 004

Ю. Ю. Дзюбак; Ю. З. Лещишин, к.т.н., доц.

(Тернопільський національний технічний університет імені Івана Пулюя)

КОМП'ЮТЕРИЗОВАНА СИСТЕМА «CLASSBOOK» ДЛЯ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ

Y. Y. Dziubak; Y. Z. Leschyshyn, Ph.D., Assoc. Prof.

COMPUTERIZED SYSTEM "CLASSBOOK" FOR RECORDING THE PERFORMANCE AND ATTENDANCE OF STUDENTS

Найкращим варіантом для автоматизації обліку успішності та відвідування занять здобувачами освіти є впровадження електронного журналу [1].

Фактично електронний класний журнал успішності та відвідуваності діджиталізує процес контролю за успішністю та відвідуваністю, дублює записи традиційного, паперового журналу, захищаючи його від спотворень, дає можливість контролювати процес оцінювання, факти відвідування занять, дозволяє у режимі реального часу отримувати доступ до введення, редагування та перегляду даних учасникам освітнього процесу та при потребі перевести їх у паперовий варіант для тривалого архівного зберігання, а також забезпечує можливість проведення заходів аналітичного спрямування з отриманими даними.

Розроблено та впроваджено в тестовому режимі у ДВНЗ «Нововолинський електромеханічний коледж» комп'ютеризовану систему успішності та відвідування занять здобувачами освіти «ClassBook».

Дана система забезпечує автоматизацію обліку успішності та відвідуваності студентів, призначена для комплексного інформаційно-аналітичного забезпечення процесів навчального закладу в частині виконання наступних функцій: введення і редагування даних про успішність студентів; введення і редагування даних про відвідуваність студентів навчальних занять; введення і редагування списків студентів, груп, викладачів, предметів та інших інформаційних об'єктів системи; оперативний доступ до необхідної інформації щодо організації навчального процесу у розрізі класного журналу; можливість формування необхідної аналітичної інформації у розрізі класного журналу.

Основними цілями створеної комп'ютеризованої системи є:

- підвищення рівня ефективності виконання процесів організації навчального процесу шляхом скорочення непродуктивних і дублюючих операцій та операцій, що виконуються «вручну»;
- забезпечення зручності введення, редагування, отримання, використання та відображення інформації;
- врахування особливостей організації навчального процесу та можливість забезпечення її адаптації за умов його зміни;
- пришвидшення пошуку інформації у системі обліку успішності та відвідуваності студентів шляхом її переведення на електронні основи;
- організація розрізненого доступу користувачів системи до визначеного функціоналу та відповідної інформації;
- забезпечення оперативного контролю адміністрацією організації та перебігу навчального процесу шляхом моніторингу інформаційної системи навчального закладу.

Дана інформаційна система розроблялася на основі діючого у коледжі паперового варіанту журналу (рис. 1), та максимально забезпечує відтворення його структури як в електронному, так і в надрукованому вигляді. Зважаючи на це, її розроблялася інфологічна модель бази даних. Фактично створена програма є десктопним CRUD-застосунком (рис. 2) [2].

Матеріали XIII Міжнародної науково-практичної конференції молодих учених та студентів
«АКТУАЛЬНІ ЗАДАЧІ СУЧАСНИХ ТЕХНОЛОГІЙ» – Тернопіль, 6-7 грудня 2023 року

	ДОСЛІДЖЕННЯ ВАРИАНТІВ ПРОЄКТУВАННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА В ІНФОРМАЦІЙНИХ ІНТЕРАКТИВНИХ АНАЛІТИЧНИХ ПАНЕЛЯХ	
23.	В. В. Никитюк, А. В. Орловська, А. К. Карнаухов, В. К. Крилов АНАЛІЗ БІОМЕТРИЧНОЇ СИСТЕМИ СИЛУЕТА КОРИСТУВАЧІВ	387
24.	М. О. Слободян КІБЕРФІЗИЧНА СИСТЕМА ДЛЯ ЕКСПРЕС-АНАЛІЗУ ПСИХОФІЗІОЛОГІЧНОГО СТАНУ НА ОСНОВІ ПУЛЬСОКСИМЕТРІЇ	389
25.	О. Р. Оробчук, І. М. Кивацький АНАЛІЗ РИЗИКІВ ТА ВРАЗЛИВОСТЕЙ В СИСТЕМАХ КЕРУВАННЯ РОЗУМНИМ БУДИНКОМ	391
26.	О. В. Палка ОГЛЯД КРІ РОЗУМНОГО МІСТА	392
27.	Т. А. Липак ЗАСТОСУВАННЯ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ В ЦИФРОВОМУ ЗБЕРЕЖЕННІ КУЛЬТУРНОЇ СПАДЩИНИ	393
28.	Т. О. Крамар, О. М. Дуда МЕТОДИ РЕКОНСТРУКЦІЇ РЕАЛЬНИХ ОБ'ЄКТІВ У ЦИФРОВОМУ СЕРЕДОВИЩІ	395
29.	М. О. Стрембіцький, О. І. Стрембіцька, І. І. Олійник, В. В. Батюк, В. М. Слободян АНАЛІЗ МЕТОДІВ РЕАЛІЗАЦІЇ ЗВ'ЯЗКУ МІЖ ВУЗЛАМИ СТОМАТОЛОГІЧНОЇ УСТАНОВКИ	397
30.	В. Семенюк, В. Сенківський, В. Чичук, Б. Хоміцький, О. Кучма ОГЛЯД ІНСТРУМЕНТІВ БЕЗПЕРЕРВНОЇ ІНТЕГРАЦІЇ В СУЧАСНИХ ПРОЄКТАХ З РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	399
31.	А. Вивюрка, Л. Мариненко, О. Нога, Б. Хоміцький, Т. Ланевич ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ПРОЦЕСІВ СИ/СД В ГНУЧКИХ ТЕХНОЛОГІЯХ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	402
32.	Д. С. Матюк, М. В. Деркач ОЦІНКА СПЕКТРАЛЬНОЇ ЩІЛЬНОСТІ ПОТУЖНОСТІ ВЕГ СИГНАЛУ	404
33.	М. В. Онай, А. І. Северін КОМПЛЕКСНИЙ ПОРІВНЯЛЬНИЙ АНАЛІЗ МЕТОДІВ ЗБЕРЕЖЕННЯ ПРИВАТНОСТІ В МАШИННОМУ НАВЧАННІ	406
34.	А. С. Хом'як ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ОБРОБКИ В РЕАЛЬНОМУ ЧАСІ У СЛУЖБАХ ЧАТ-БОТІВ ЧЕРЕЗ ІНТЕГРАЦІЮ ЧЕРГИ ЗАПИТІВ ДЛЯ РОЗПОДІЛЕННЯ НАВАНТАЖЕННЯ	408
35.	Ю. Ю. Дзюбак, Ю. З. Лещинин ОБГРУНТУВАННЯ ДОЦІЛЬНОСТІ СТВОРЕННЯ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ ПІД ПОТРЕБИ ОКРЕМОГО ВИЩОГО НАВЧАЛЬНОГО ЗАКЛАДУ	410
36.	Ю. Ю. Дзюбак, Ю. З. Лещинин КОМП'ЮТЕРИЗОВАНА СИСТЕМА «CLASSBOOK» ДЛЯ ОБЛІКУ УСПІШНОСТІ ТА ВІДВІДУВАННЯ ЗАНЯТЬ ЗДОБУВАЧАМИ ОСВІТИ	411
37.	В. І. Ковальчук ОГЛЯД СУЧАСНИХ ТЕЛЕКОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ В МЕДИЧНІЙ СФЕРІ	413

Додаток Б

Лістинги коду основних класів програми

Main.java

```

import javafx.application.Application;...
public class Main extends Application {
    private FXMLLoader loader;
    private double xOffset;
    private double yOffset;
    @Override
    public void start(Stage primaryStage) {
        try {
            loader = new FXMLLoader();
loader.setLocation(getClass().getResource("authorization_and_dbcon
figs/login_pane.fxml"));
            ControllerLoginPane controller = new
ControllerLoginPane();
            loader.setController(controller);
            loader.load();
            Scene scene = new Scene(loader.getRoot());
            scene.setOnMousePressed(new EventHandler<MouseEvent>()
{...
                } catch(Exception e) {
                    e.printStackTrace();
                }
            }
        public static void main(String[] args) {
            launch(args);
        }
    }
}

```

DBConnection.java

```

package authorization_and_dbconfigs; ...
public class DBConnection {
    private final String dburl;
    private final String username = "*****";
    private final String password = "*****";
    private Connection connect;
    private String host;
    private String port;
    private String db;
    public DBConnection() {
        FileInputStream fis;
        Properties property = new Properties();
        String path = System.getProperty("user.dir");
        try {
            fis = new FileInputStream(path+"/config.properties");
            property.load(fis);
            host = property.getProperty("db.host");

```

```

        port = property.getProperty("db.port");
        db = property.getProperty("db.name");
    } catch (IOException e) {
        System.err.println("Error: Config file is not
allowed!");
    }
    dburl =
    "jdbc:mysql://" + host + ":" + port + "/" + db + "?allowPublicKeyRetrieval=true&useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDate
timeCode=false&serverTimezone=UTC&useSSL=false";
    }
    public Connection getConnection() {
        try {
            connect = DriverManager.getConnection(dburl, username,
password);
        } catch (Exception e) {
            e.printStackTrace();
        }
        return connect;
    }
    public void close(Connection connect, PreparedStatement pstmt,
ResultSet rs) {
        try {
            if(connect != null)
                connect.close();
            if(pstmt != null)
                pstmt.close();
            if(rs != null)
                rs.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    public void close(Connection connect, PreparedStatement pstmt)
{
        try {
            close(connect, pstmt, null);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    public void close(PreparedStatement pstmt) {
        try {
            close(null, pstmt, null);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

ControllerZaniatty.java

```
package vukladach;
```

```

import admin.DataAccessObject;
import com.itextpdf.text.*;
import com.itextpdf.text.pdf.BaseFont;
import com.itextpdf.text.pdf.PdfPCell;
import com.itextpdf.text.pdf.PdfPTable;
import com.itextpdf.text.pdf.PdfWriter;
import com.jfoenix.controls.JFXButton;
import com.jfoenix.controls.JFXDatePicker;
import javafx.beans.property.ReadOnlyObjectWrapper;
import javafx.event.EventHandler;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.geometry.Rectangle2D;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.control.Label;
import javafx.scene.control.TextArea;
import javafx.scene.control.TextField;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.input.MouseButton;
import javafx.scene.input.MouseEvent;
import javafx.scene.layout.AnchorPane;
import javafx.scene.layout.Pane;
import javafx.scene.shape.Circle;
import javafx.stage.Screen;
import javafx.stage.Stage;
import javafx.stage.StageStyle;
import authorization_and_dbconfigs.ControllerPersonal_account;
import authorization_and_dbconfigs.DBConnection;
import javax.swing.*;
import java.awt.*;
import java.awt.Font;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.net.URL;
import java.sql.Connection;
import java.sql.Date;
import java.sql.ResultSet;
import java.sql.Statement;
import java.time.LocalDate;
import java.time.format.DateTimeFormatter;
import java.util.*;

public class ControllerZaniattya extends Component implements
Initializable {

    ...

    private void AddToPDF() throws IOException, DocumentException
{

```

```

String path = "";
JFileChooser j = new JFileChooser();
j.setSelectionMode(JFileChooser.DIRECTORIES_ONLY);
int x = j.showSaveDialog(this);

if (x == JFileChooser.APPROVE_OPTION) {
    path = j.getSelectedFile().getPath();
}

Document doc = new Document();
String docname =
ControllerSelection_of_load.getpDiscname()+"
"+ControllerSelection_of_load.getpGroupname()+" Права сторінка
"+VuklName+".pdf";
PdfWriter.getInstance(doc, new FileOutputStream(path +
docname));
doc.open();
BaseFont bf =
BaseFont.createFont("c:/windows/fonts/times.ttf",
BaseFont.IDENTITY_H, BaseFont.EMBEDDED);

Chunk chunk = new Chunk("Викладач: "+ VuklName, new
com.itextpdf.text.Font(bf, 12, Font.BOLD, BaseColor.BLACK));

Paragraph paragraph = new Paragraph(0);
paragraph.setSpacingAfter(10);
paragraph.add(chunk);
paragraph.setAlignment(Element.ALIGN_LEFT);

PdfPTable tbl1 = new PdfPTable(8);

tbl1.setTotalWidth(550);
tbl1.setLockedWidth(true);

tbl1.setWidths(new int[]{5, 17, 12, 9, 45, 30, 15, 20});
for (int i = 0; i < 49; i++) {
    for (int k = 0; k < 8; k++) {
        tbl1.addCell(new PdfPCell(new Phrase(" ", new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
    }
}

int used = 0;
int n = 0, height = 0;

do {
    doc.add(paragraph);
    PdfPTable tbl = new PdfPTable(8);

    tbl.setTotalWidth(550);
    tbl.setLockedWidth(true);

```



```

tbl.setWidths(new int[]{5, 17, 12, 9, 45, 30, 15,
20});

PdfPCell cell = new PdfPCell(new PdfPCell(new
Phrase("№"+"\\n"+"з/п", new com.itextpdf.text.Font(bf, 11,
Font.BOLD, BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

cell = new PdfPCell(new PdfPCell(new
Phrase("Дата", new com.itextpdf.text.Font(bf, 11, Font.BOLD,
BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

cell = new PdfPCell(new PdfPCell(new
Phrase("Вид"+"\\n"+"заняття", new com.itextpdf.text.Font(bf, 11,
Font.BOLD, BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

cell = new PdfPCell(new PdfPCell(new Phrase("К-
СТЬ"+"\\n"+"год.", new com.itextpdf.text.Font(bf, 11, Font.BOLD,
BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

cell = new PdfPCell(new PdfPCell(new Phrase("Тема
заняття", new com.itextpdf.text.Font(bf, 11, Font.BOLD,
BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

cell = new PdfPCell(new PdfPCell(new
Phrase("Домашнє завдання", new com.itextpdf.text.Font(bf, 11,
Font.BOLD, BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

cell = new PdfPCell(new PdfPCell(new
Phrase("Підпис"+"\\n"+"викладача", new com.itextpdf.text.Font(bf,
11, Font.BOLD, BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

cell = new PdfPCell(new PdfPCell(new
Phrase("Примітки", new com.itextpdf.text.Font(bf, 11, Font.BOLD,
BaseColor.BLACK)));
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
tbl.addCell(cell);

```

```

        for (int i = used; i < 9999; i++)
        {
            if (i < tblview.getItems().size() &&
used+11>=tblview.getItems().size()) {
                String N =
tblview.getColumns().get(1).getCellObservableValue(i).getValue().t
oString();

                String D =
tblview.getColumns().get(2).getCellObservableValue(i).getValue().t
oString();

                String V =
tblview.getColumns().get(3).getCellObservableValue(i).getValue().t
oString();

                String K =
tblview.getColumns().get(4).getCellObservableValue(i).getValue().t
oString();

                String T =
tblview.getColumns().get(5).getCellObservableValue(i).getValue().t
oString();

                String Dz =
tblview.getColumns().get(6).getCellObservableValue(i).getValue().t
oString();

                String P =
tblview.getColumns().get(7).getCellObservableValue(i).getValue().t
oString();

                String Pr =
tblview.getColumns().get(8).getCellObservableValue(i).getValue().t
oString();

                tbl.addCell(new PdfPCell(new Phrase(N, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

                tbl.addCell(new PdfPCell(new Phrase(D, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

                tbl.addCell(new PdfPCell(new Phrase(V, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

                tbl.addCell(new PdfPCell(new Phrase(K, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

                tbl.addCell(new PdfPCell(new Phrase(T, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

                tbl.addCell(new PdfPCell(new Phrase(Dz,
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

                tbl.addCell(new PdfPCell(new Phrase(P, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

                tbl.addCell(new PdfPCell(new Phrase(Pr,
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));

```

```

        n++;
        height = (int) (height +
tbl.getRowHeight(n));

        }
        else if (i < used+11 &&
used+11<tblview.getItems().size()){
            String N =
tblview.getColumns().get(1).getCellObservableValue(i).getValue().t
oString();

            String D =
tblview.getColumns().get(2).getCellObservableValue(i).getValue().t
oString();

            String V =
tblview.getColumns().get(3).getCellObservableValue(i).getValue().t
oString();

            String K =
tblview.getColumns().get(4).getCellObservableValue(i).getValue().t
oString();

            String T =
tblview.getColumns().get(5).getCellObservableValue(i).getValue().t
oString();

            String Dz =
tblview.getColumns().get(6).getCellObservableValue(i).getValue().t
oString();

            String P =
tblview.getColumns().get(7).getCellObservableValue(i).getValue().t
oString();

            String Pr =
tblview.getColumns().get(8).getCellObservableValue(i).getValue().t
oString();

            tbl.addCell(new PdfPCell(new Phrase(N, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(D, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(V, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(K, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(T, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(Dz,
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(P, new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(Pr,

```

```

new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK)));
        }
        else if (tbl1.getTotalHeight() >
tbl.getTotalHeight()){
            tbl.addCell(new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
        }
    }

    doc.add(tbl);
    doc.newPage();

    used = used + 11;
    n = 0;

    if (tblview.getItems().size() - used <=0){
        break;
    }

    } while (true);
doc.close();
Desktop desktop = Desktop.getDesktop();
desktop.open(new File(path+docname));
}
...
}

```

LeftSide.java

```
package vukladach;
```

```

import authorization_and_dbconfigs.DBConnection;
import com.itextpdf.text.*;
import com.itextpdf.text.pdf.BaseFont;
import com.itextpdf.text.pdf.PdfPCell;
import com.itextpdf.text.pdf.PdfPTable;
import com.itextpdf.text.pdf.PdfWriter;

import java.awt.Font;
import java.awt.Desktop;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.net.URL;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;

import javax.swing.*;
import javax.swing.table.DefaultTableCellRenderer;
import javax.swing.table.TableColumn;

public class LeftSide extends JPanel {

    private JFrame frame1;
    private JTable table1;

    private DBConnection database;
    private Connection connection;

    public LeftSide() throws IOException, DocumentException {

        frame1 = new JFrame();
        frame1.setTitle(ControllerSelection_of_load.getpDiscname()
+ " " + ControllerSelection_of_load.getpGroupname());

        try {

            database = new DBConnection();
            connection = database.getConnection();
            Statement statement = connection.createStatement();
            Statement statement2 = connection.createStatement();

            String sql0 = "SELECT FK_student FROM
dovidnyuk_student_marks WHERE FK_zaniattya='" +
ControllerZaniattya.getID_zaniattya() + "' ";
            statement.execute(sql0);
            ResultSet result = statement.getResultSet();
            int id = 0;
            while (result.next()) {

```

```

        id++;
    }

    String sql1 = "SELECT PK_zaniattya FROM
dovidnyuk_zaniat WHERE FK_load='" +
ControllerSelection_of_load.getpID_load() + "' ";
    statement.execute(sql1);
    result = statement.getResultSet();
    int id2 = 2;
    while (result.next()) {
        id2 = id2 + 2;
    }

    String[][] data = new String[id][id2];
    ArrayList<String> dates = new ArrayList<String>();
    dates.add("№");
    dates.add("ПИБ студента");
    String sqlx = "SELECT a.FK_vudy_zaniattya, a.date,
s.compact_name_vudy FROM dovidnyuk_zaniat as a JOIN
dovidnyuk_vudiv_zaniat as s ON
s.id_vudy_zaniattya=a.FK_vudy_zaniattya " +
"WHERE a.FK_load='" +
ControllerSelection_of_load.getpID_load() + "' ORDER BY
PK_zaniattya";

    statement.execute(sqlx);
    result = statement.getResultSet();
    String name, compactname, name_v;
    int lenght;
    int id_v;
    while (result.next()) {
        id_v = result.getInt("FK_vudy_zaniattya");
        name_v = result.getString("compact_name_vudy");

        if (id_v > 2) {
            dates.add(name_v);
            dates.add(name_v);
        }

        else {
            name = result.getString("date");
            lenght = name.length();
            compactname = name.substring(5, lenght);
            compactname = compactname.replace("-",
", ".");

            dates.add(compactname);
            dates.add(compactname);
        }
    }

    Object[] columnNames = dates.toArray();

```

```

table1 = new JTable(data, columnNames);

table1.setAutoResizeMode(JTable.AUTO_RESIZE_OFF);
DefaultTableCellRenderer centerRend = new
DefaultTableCellRenderer();
centerRend.setHorizontalAlignment(JLabel.CENTER);
table1.setDefaultRenderer(Object.class, centerRend);
TableColumn a = table1.getColumnModel().getColumn(0);
a.setPreferredWidth(20);
TableColumn b = table1.getColumnModel().getColumn(1);
b.setPreferredWidth(150);

String sql;

sql = "SELECT a.Namestudent FROM
dovidnyuk_student_marks as a " +
      "JOIN dovidnyuk_students as s ON
a.FK_student=s.id_student JOIN dovidnyuk_marks as d ON
a.FK_mark=d.PK_mark JOIN dovidnyuk_group as g ON
s.fk_group=g.id_group " +
      "JOIN dovidnyuk_vidvidyvanosti as f ON
a.FK_vidviduvanosti=f.id_pomitku_vidvidyv WHERE a.FK_zaniattya='"
+ ControllerZaniattya.getpID_zaniattya() + "' " +
      "AND g.group_name='" +
ControllerSelection_of_load.getpGroupname() + "' ORDER BY
a.FK_student";

statement.execute(sql);
ResultSet rec = statement.getResultSet();
int i = 0, n = 1;

while (rec.next()) {
    data[i][0] = Integer.toString(n);
    data[i][1] = rec.getString("Namestudent");
    i++;
    n++;
}

int j = 2, k = 3;

do {
    String sql3 = "SELECT PK_zaniattya FROM
dovidnyuk_zaniat WHERE FK_load='" +
ControllerSelection_of_load.getpID_load() + "' ORDER BY
PK_zaniattya"; //!!!!!!
    statement.execute(sql3);
    result = statement.getResultSet();
    int PK;
    while (result.next()) {

```

```

        PK = result.getInt("PK_zaniattya");
        String sql2;

        sql2 = "SELECT d.digital_name, f.compact_name
FROM dovidnyuk_student_marks as a " +
            "JOIN dovidnyuk_students as s ON
a.FK_student=s.id_student JOIN dovidnyuk_marks as d ON
a.FK_mark=d.PK_mark JOIN dovidnyuk_group as g ON
s.fk_group=g.id_group " +
            "JOIN dovidnyuk_zaniat as h ON
a.FK_zaniattya=h.PK_zaniattya JOIN dovidnyuk_group_load as k ON '"
+ ControllerSelection_of_load.getpID_load() + "'=k.PK_load JOIN
dovidnyuk_discipline as j ON k.FK_discipline=j.PK_discipline " +
            "JOIN dovidnyuk_vidvidyvanosti as f ON
a.FK_vidvidyvanosti=f.id_pomitku_vidvidyv WHERE h.FK_load='" +
ControllerSelection_of_load.getpID_load() + "' AND
h.PK_zaniattya='" + PK + "' AND g.group_name='" +
ControllerSelection_of_load.getpGroupname() + "' ORDER BY
a.FK_student";

        statement2.execute(sql2);
        ResultSet rec2 = statement2.getResultSet();
        i = 0;
        while (rec2.next()) {
            data[i][j] =
rec2.getString("digital_name");
            data[i][k] =
rec2.getString("compact_name");
            i++;
        }
        k = k + 2;
        j = j + 2;
    }

    break;

} while (true);

rec.close();
result.close();
statement.close();
statement2.close();
connection.close();

} catch (SQLException el) {
    el.printStackTrace();
}

JScrollPane spanel = new JScrollPane(table1);
frame1.add(spanel);
URL iconURL =
getClass().getResource("/img/iconmini.jpg");

```



```

    ImageIcon icon = new ImageIcon(iconURL);
    frame1.setIconImage(icon.getImage());
    frame1.setSize(1200, 470);
    frame1.setVisible(false);
    AddToPDF();
}

private void AddToPDF() throws IOException, DocumentException
{
    String path = "";
    JFileChooser j = new JFileChooser();
    j.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
    int x = j.showSaveDialog(this);

    if (x == JFileChooser.APPROVE_OPTION) {
        path = j.getSelectedFile().getPath();
    }

    Document doc = new Document();
    String docname =
ControllerSelection_of_load.getpDiscname()+"
"+ControllerSelection_of_load.getpGroupname()+" Ліва сторінка
"+ControllerZaniattya.getpVuklName()+".pdf";
    PdfWriter.getInstance(doc, new
FileOutputStream(path+docname));
    doc.open();
    BaseFont bf =
BaseFont.createFont("c:/windows/fonts/times.ttf",
BaseFont.IDENTITY_H, BaseFont.EMBEDDED);

    Chunk chunk = new Chunk("Назва дисципліни: " +
ControllerSelection_of_load.getpDiscname(), new
com.itextpdf.text.Font(bf, 12, Font.BOLD, BaseColor.BLACK));

    Paragraph paragraph = new Paragraph(0);
    paragraph.setSpacingAfter(10);
    paragraph.add(chunk);

    doc.add(paragraph);

    PdfPTable tbl = new PdfPTable(24);

    tbl.setTotalWidth(550);
    tbl.setLockedWidth(true);

    tbl.setWidths(new float[]{3, 20, (float) 2.5, (float) 2.5,
(float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5,
(float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5,
(float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5, (float)
2.5, (float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5, (float)
2.5, (float) 2.5, (float) 2.5});
}

```

```

        PdfPCell cell = new PdfPCell(new PdfPCell(new Phrase("№" +
"\n" + "з/п", new com.itextpdf.text.Font(bf, 10, Font.BOLD,
BaseColor.BLACK))));
        cell.setHorizontalAlignment(Element.ALIGN_CENTER);
        tbl.addCell(cell);

        cell = new PdfPCell(new PdfPCell(new Phrase("ПИБ", new
com.itextpdf.text.Font(bf, 11, Font.BOLD, BaseColor.BLACK))));
        cell.setHorizontalAlignment(Element.ALIGN_CENTER);
        cell.setNoWrap(true);
        tbl.addCell(cell);

String N;

for (int l = 2; l < 24; l++) {
    if (l < table1.getColumnCount()) {
        N = table1.getColumnName(l);
        cell = new PdfPCell(new PdfPCell(new Phrase(N, new
com.itextpdf.text.Font(bf, 10, Font.BOLD, BaseColor.BLACK))));
        cell.setHorizontalAlignment(Element.ALIGN_CENTER);
        tbl.addCell(cell);
    }
    else {
        cell = new PdfPCell(new Phrase(" ", new
com.itextpdf.text.Font(bf, 11, Font.TRUE_TYPE_FONT,
BaseColor.BLACK))));
        tbl.addCell(cell);
    }
}

for (int i = 0; i < 49; i++) {
    for (int l = 0; l < 24; l++) {
        if (l < table1.getColumnCount() && i <
table1.getRowCount()) {
            N = table1.getValueAt(i, l).toString();
            cell = new PdfPCell(new Phrase(N, new
com.itextpdf.text.Font(bf, 11, Font.TRUE_TYPE_FONT,
BaseColor.BLACK))));
            if (l != 1)
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
            tbl.addCell(cell);
        }
        else {
            cell = new PdfPCell(new Phrase(" ", new
com.itextpdf.text.Font(bf, 11, Font.TRUE_TYPE_FONT,
BaseColor.BLACK))));
            tbl.addCell(cell);
        }
    }
}

doc.add(tbl);

```

```

if (table1.getColumnCount() > 24){

    int used = 22, b = 0, g = 0;
    int lim;

    do {

        lim = table1.getColumnCount() - used - 2;
        if (lim>=22){
            lim = 22;
        }
        else lim = lim + 2;

        doc.newPage();
        doc.add(paragraph);

        PdfPTable tbl1 = new PdfPTable(24);

        tbl1.setTotalWidth(550);
        tbl1.setLockedWidth(true);

        tbl1.setWidths(new float[]{3, 20, (float) 2.5,
(float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5,
(float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5,
(float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5, (float) 2.5,
(float) 2.5, (float) 2.5, (float) 2.5});

        cell = new PdfPCell(new PdfPCell(new Phrase("№" +
"\n" + "з/п", new com.itextpdf.text.Font(bf, 10, Font.BOLD,
BaseColor.BLACK))));
        cell.setHorizontalAlignment(Element.ALIGN_CENTER);
        tbl1.addCell(cell);

        cell = new PdfPCell(new PdfPCell(new Phrase("ПИБ",
new com.itextpdf.text.Font(bf, 11, Font.BOLD, BaseColor.BLACK))));
        cell.setHorizontalAlignment(Element.ALIGN_CENTER);
        cell.setNoWrap(true);
        tbl1.addCell(cell);

        for (int l = used + 2; l < used + 2 + 22; l++) {
            if (l < table1.getColumnCount()) {
                N = table1.getColumnName(l);
                cell = new PdfPCell(new PdfPCell(new
Phrase(N, new com.itextpdf.text.Font(bf, 10, Font.BOLD,
BaseColor.BLACK))));
                cell.setHorizontalAlignment(Element.ALIGN_CENTER);
                tbl1.addCell(cell);
            } else {
                cell = new PdfPCell(new Phrase(" ", new
com.itextpdf.text.Font(bf, 11, Font.TRUETYPE_FONT,
BaseColor.BLACK))));
            }
        }
    }
}

```

```

        tbl1.addCell(cell);
    }
}

for (int i = 0; i < 49; i++) {
    for (int l = 0; l < used + 2 + lim; l++) {
        if (l < table1.getColumnCount() && i <
table1.getRowCount()) {
            if (lim > b && l < 2 || l >= used + 2)
{
                N = table1.getValueAt(i,
l).toString();
                cell = new PdfPCell(new Phrase(N,
new com.itextpdf.text.Font(bf, 11, Font.TRUEETYPE_FONT,
BaseColor.BLACK)));
                if (l != 1)
cell.setHorizontalAlignment(Element.ALIGN_CENTER);
                tbl1.addCell(cell);
                b++;
                if (b == 24) b=0;
            } else if (b == lim && 24 - lim > g) {
                cell = new PdfPCell(new Phrase("
", new com.itextpdf.text.Font(bf, 11, Font.TRUEETYPE_FONT,
BaseColor.BLACK)));
                tbl1.addCell(cell);
                g++;
            } else if (24 - lim == g) {
                b = 0;
                g = 0;
                l = -1;
            }
        } else if (table1.getColumnCount() > 1 &&
tbl.getRows().size() > tbl1.getRows().size()) {
            cell = new PdfPCell(new Phrase(" ",
new com.itextpdf.text.Font(bf, 11, Font.TRUEETYPE_FONT,
BaseColor.BLACK)));
            tbl1.addCell(cell);
        }
    }
}

doc.add(tbl1);

used = used + 22;
b = 0; g = 0;

if (table1.getColumnCount() - used - 2 <= 0) {
    break;
}

} while (true);
}

```

```
doc.close();  
frame1.dispose();  
Desktop desktop = Desktop.getDesktop();  
desktop.open(new File(path+docname));  
}  
}
```