

Київський національний торговельно-економічний університет

Кафедра комп'ютерних наук та інформаційних технологій

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Розробка програмного забезпечення для створення та ведення договорів підприємства»

Студента 4 курсу, 13 групи,
Спеціальності
122«Комп'ютерні науки»

підпис студента

Наумов Дмитро
Максимович

Науковий керівник

підпис керівника

Демідов Павло
Георгійович

Гарант освітньої програми

підпис керівника

Демідов Павло
Георгійович

Київ 2021

Київський національний торговельно-економічний університет

Факультет інформаційних технологій
Кафедра комп'ютерних наук та інформаційних систем
Спеціальність 122 «Комп'ютерні науки»

Затверджую

Зав. кафедри _____ Пурський О.І.
«15» грудня 2020р.

Завдання на випускню кваліфікаційну роботу студента

Наумова Дмитра Максимовича

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи
«Розробка програмного забезпечення для створення та ведення договорів підприємства».

Затверджена наказом ректора від «18» грудня 2020р. № 3780

2. Строк здачі студентом закінченої роботи 31 травня 2021 року

3. Цільова установка та вихідні дані до роботи

Мета роботи: розробити програмне забезпечення для створення та ведення договорів підприємства

Об'єкт дослідження: процеси створення та ведення договорів на підприємстві

Предмет дослідження: моделі, методи та інформаційні технології створення та ведення договорів

4. Перелік графічного матеріалу _____

5. Консультанти по роботі із зазначенням розділів, за якими здійснюється консультування:

Розділ	Консультант(прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Демідов П.Г.	22.12.2020 р.	22.12.2020 р.
2	Демідов П.Г.	22.12.2020 р.	22.12.2020 р.
3	Демідов П.Г.	22.12.2020 р.	22.12.2020 р.

6. Зміст випускної кваліфікаційної роботи (проекту) (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. АНАЛІЗ ТА ПІДХОДИ ДО СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ В ДІЯЛЬНОСТІ ПІДПРИЄМСТВА

- 1.1. Характеристика типового процесу створення та ведення договорів на підприємстві.
- 1.2. Огляд і аналіз існуючого програмного забезпечення управління договорами на підприємстві.
- 1.3. Засоби розробки програм та баз даних в створенні систем управління договорами на підприємстві.

РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ

- 2.1. Постановка задачі.
- 2.2. Проектування візуальної частини програмного забезпечення
- 2.3. Нормативно-довідкові документи системи ведення договорів.

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ

- 3.1. Загальна характеристика програмного забезпечення системи створення та ведення договорів.
- 3.2. Віконний інтерфейс доступу до функцій ведення нормативно-довідкової інформації та договорів підприємства.
- 3.3. Перевірка коду, компіляція та створення програм.

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТОК

7. Календарний план виконання роботи

№ Пор	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		За планом	Фактично
1	2	3	4
1	Вибір теми випускної кваліфікаційної роботи	05.10.2019	05.10.2019
2	Розробка та затвердження завдання на випускну кваліфікаційну роботу	18.12.2020	18.12.2020
3	Вступ	03.02.2021	
4	Розділ 1. Проектування проекту по розробці програмного забезпечення для створення та ведення договорів	26.02.2021	
5	Розділ 2. Проектування програмного забезпечення для створення та ведення договорів	06.04.2021	
6	Розділ 3. Розробка програмного забезпечення для створення та ведення договорів	20.05.2021	
7	Висновки	27.05.2021	
8	Здача випускної кваліфікаційної роботи на кафедрі науковому керівнику	29.05.2021	
9	Попередній захист випускної кваліфікаційної роботи	04.06.2021	
10	Виправлення зауважень, зовнішнє рецензування випускної кваліфікаційної роботи	05.06.2021	
11	Представлення готової зшитої випускної кваліфікаційної роботи на кафедрі	17.06.2021	
12	Публічний захист випускної кваліфікаційної роботи	За розкладом роботи ЕК	

8. Дата видачі завдання «22» грудня 2020 р

9. Керівник випускної кваліфікаційної роботи

Демідов П.Г.

(прізвище, ініціали, підпис)

Анотація

У випускній кваліфікаційній роботі здійснено комплексну розробку програмного забезпечення для створення та ведення договорів на підприємстві з метою підвищення ефективності його діяльності. Розроблені засоби дозволяють створити базу даних та виконувати в ній функції створення, редагування, накопичення та перегляду договорів підприємства. В програмі реалізовані режими представлення накопиченої бази договорів в стандартних форматах текстових процесорів.

Ключові слова: розробка програмного забезпечення, підприємство, програма, база даних.

Annotation

In the final qualification work, a comprehensive development of software for creating and maintaining contracts at the enterprise in order to increase the efficiency of the enterprise. A method for editing table data from a database and opening contracts has been developed. Connected to different databases.

Keywords: software development, enterprise, program, database.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1. АНАЛІЗ ТА ПІДХОДИ ДО СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ В ДІЯЛЬНОСТІ ПІДПРИЄМСТВА	10
1.1. Характеристика типового процесу створення та ведення договорів на підприємстві.	10
1.2. Огляд і аналіз існуючого програмного забезпечення управління договорами на підприємстві.	17
1.3. Засоби розробки програм та баз даних в створенні систем управління договорами на підприємстві.	22
РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ	41
2.1. Постановка задачі.	41
2.2. Проектування візуальної частини програмного забезпечення	41
Візуальна частина програмного забезпечення має бути інтуїтивна та зрозуміла.	41
2.3. Нормативно-довідкові документи системи ведення договорів.	41
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ	47
3.1. Загальна характеристика програмного забезпечення системи створення та ведення договорів.	47
Програмне забезпечення допомагає в створенні, зберіганні та редагуванні договорів або інших документів на підприємстві.	47
3.2. Віконний інтерфейс доступу до функцій ведення нормативно-довідкової інформації та договорів підприємства.	47
3.3. Перевірка коду, компіляція та створення програм.	54
ВИСНОВКИ	57
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	58
ДОДАТОК	62

ВСТУП

Розробка програмного забезпечення (англ. *software engineering, software development*) — це рід діяльності (професія) та процес, спрямований на створення та підтримку працездатності, якості та надійності програмного забезпечення, використовуючи технології, методологію та практики з інформатики, керування проектами, математики, інженерії та інших областей знання.[57]

Мета і завдання дослідження: Метою даного дослідження є розробити програмне забезпечення для створення та ведення договорів підприємства

Для досягнення поставленої мети необхідно було вирішити наступні **завдання:**

- провести комплексне аналітичне дослідження схожих існуючих програм;
- дослідити методи розробки програмного забезпечення;
- розробити візуальну модель програмного забезпечення для ведення договорів;
- розробити метод створення та ведення договорів;
- розробити автоматизовану систему ведення договорів;

Об'єкт дослідження: процеси створення та ведення договорів на підприємстві

Предмет дослідження: моделі, методи та інформаційні технології створення та ведення договорів

Методи дослідження: Теоретичною основою дослідження є загальнонауковий аналітичний метод, а також системний підхід і праці провідних вчених з проблем дослідження ведення договорів на підприємстві. Інформаційну базу дослідження становлять статистичні дані Державного комітету статистики України. Для практичного вирішення поставлених задач використовувалися такі методи:

- загальнонауковий аналітичний метод;
- методи теорії БД для формування інформаційно-логічної моделі

предметної області та БД;

- методи алгоритмічного програмування, для створення автоматизованої системи ведення договорів.

Практичне значення. Створену програму можна буде використовувати на підприємстві для ведення договорів. Автоматизація договорів на підприємстві дуже важлива роль у житті підприємства, бо якщо не будуть вестися договори – не буде підприємства, як такого. Автоматизація при створенні та веденні договорів облегшує роботу працівників.

РОЗДІЛ 1. АНАЛІЗ ТА ПІДХОДИ ДО СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ В ДІЯЛЬНОСТІ ПІДПРИЄМСТВА

1.1. Характеристика типового процесу створення та ведення договорів на підприємстві.

Правові засади розробки, укладення та виконання колективних договорів, які спрямовані на врегулювання трудових відносин та соціально-економічних інтересів працівників і власників визначає Закон України "Про колективні договори і угоди".

Метою укладення колективного договору є врегулювання виробничих, трудових і соціально-економічних відносин, а також узгодження інтересів трудящих, власників та уповноважених ними органів.

Так, колективний договір укладається на підприємствах, в установах, організаціях незалежно від форм власності і господарювання, які використовують найману працю.

Також, колективний договір може укладатися в структурних підрозділах підприємства в межах компетенції цих підрозділів. Під такими підрозділами маються на увазі цехи, відділи, сектори тощо.

Однією із сторін колективного договору є власник або уповноважений ним орган, другою стороною колективного договору є профспілковий чи інший уповноважений на представництво трудовим колективом орган, а у разі відсутності такого органу – представники трудящих, які обрані і уповноважені трудовим колективом.

Врегулювання відносин між власником або уповноваженим ним органом з працівниками шляхом укладення колективного договору дозволить створити надійний соціальний клімат в колективі та мінімізувати можливість конфліктів і страйків при умові виконання колективного договору.

Крім того, колективний договір служитиме стабілізатором прав власника або уповноваженого ним органу і працівника протягом дії колективного договору.

Доцільно зазначити, що умови колективних договорів, які укладені відповідно до чинного законодавства є обов'язковими для підприємств, на які вони поширюються, та сторін, які їх уклали.

Але, якщо умови колективних договорів погіршують порівняно з чинним законодавством становище працівників, то такі умови є недійсними, і забороняється включати їх до колективних договорів.

Необхідно наголосити на тому, що чинне законодавство України забороняє будь-яке втручання, яке може обмежити законні права працівників та їх представників або заборонити їх здійснення, з боку органів представницької і виконавчої влади та господарського управління, політичних партій, власників або уповноважених ними органів при укладенні та виконанні колективних договорів.

Крім того, не допускається ведення переговорів та укладення колективних договорів від імені працівників організаціями або органами, які створені чи фінансуються власниками або уповноваженими ними органами, політичними партіями.

Також, у разі коли інтереси трудового колективу представляє профспілковий орган, інтереси власника або уповноваженого ним органу не можуть представляти особи, які є членами виборного органу цієї профспілки.

Слід звернути увагу та тому, що укладенню колективного договору обов'язково передують колективні переговори.

Право на ведення таких переговорів і укладення колективних договорів від імені найманих працівників надається професійним спілкам, об'єднанням профспілок в особі їх виборних органів або іншим представницьким організаціям трудящих, які наділені трудовим колективом відповідними повноваженнями.

В законодавстві України відсутня норма, якою б покладався обов'язок про початок переговорів на якусь конкретну сторону.

Так, будь-яка зі сторін не раніше ніж за три місяці до закінчення строку дії колективного договору або у строки, визначені договором, письмово

повідомляє іншу сторону про початок переговорів. Інша сторона протягом семи днів повинна розпочати переговори.

Порядок ведення переговорів з питань розробки, укладення або внесення змін до колективного договору визначається сторонами і оформлюється відповідним протоколом.

Доцільно зазначити, що для ведення переговорів і підготовки проекту колективного договору із представників сторін утворюється робоча комісія.

Сторони можуть переривати переговори з метою проведення консультацій, експертиз, отримання необхідних даних для вироблення відповідних рішень і пошуку компромісів.

Так, сторони колективних переговорів зобов'язані надавати учасникам переговорів всю необхідну інформацію щодо змісту колективного договору, угоди. Учасники переговорів не мають права розголошувати дані, що є державною або комерційною таємницею, і підписують відповідні зобов'язання.

Щодо змісту колективного договору, то його визначають сторони колективного договору в межах своєї компетенції.

Так, у колективному договорі встановлюються взаємні зобов'язання сторін щодо регулювання виробничих, трудових, соціально-економічних відносин, зокрема: зміни в організації виробництва і праці; забезпечення продуктивної зайнятості; нормування і оплати праці, встановлення форми, системи, розмірів заробітної плати та інших видів трудових виплат (доплат, надбавок, премій та ін.); встановлення гарантій, компенсацій, пільг; участі трудового колективу у формуванні, розподілі і використанні прибутку підприємства (якщо це передбачено статутом); режиму роботи, тривалості робочого часу і відпочинку; умов і охорони праці; забезпечення житлово-побутового, культурного, медичного обслуговування, організації оздоровлення і відпочинку працівників; гарантій діяльності профспілкової чи інших представницьких організацій трудящих; умов регулювання фондів оплати праці; забезпечення рівних прав та можливостей жінок і чоловіків.

Також колективний договір може передбачати додаткові порівняно з чинним законодавством гарантії, соціально-побутові пільги.

Проект колективного договору підготовлений робочою комісією, готується з урахуванням пропозицій, що надійшли від працівників, трудових колективів. За результатами підготовки проекту приймається рішення, яке оформляється відповідним протоколом.

Під час ведення колективних переговорів можуть виникати між сторонами розбіжності, для врегулювання яких сторони використовують примирні процедури.

Так, якщо в ході переговорів сторони не дійшли згоди з незалежних від них причин, то складається протокол розбіжностей, до якого вносяться пропозиції сторін про заходи, необхідні для усунення цих причин, а також про строки відновлення переговорів.

На протязі трьох днів після складання протоколу розбіжностей сторони проводять консультації, формують із свого складу примирну комісію, а у разі недосягнення згоди звертаються до посередника, обраного сторонами.

Примирна комісія або посередник у термін до семи днів розглядає протокол розбіжностей і виносить рекомендації по суті спору.

Проте, у разі недосягнення згоди між сторонами відносно внесення рекомендації допускається організація та проведення страйків у порядку, що не суперечить законодавству України.

Також, законодавством передбачено, що для підтримки своїх вимог під час проведення переговорів щодо розробки, укладення чи зміни колективного договору, інші уповноважені працівниками органи можуть проводити у встановленому порядку збори, мітинги, пікетування, демонстрації. Крім того, необхідно зазначити, що на період переговорів законодавством передбачено такі гарантії та компенсації.

Так, особи, які беруть участь в переговорах як представники сторін, а також спеціалісти, запрошені для участі в роботі комісій, на період переговорів та підготовки проекту звільняються від основної роботи із збереженням

середнього заробітку та включенням цього часу до трудового стажу. Всі витрати, пов'язані з участю у переговорах і підготовці проекту, компенсуються в порядку, передбаченому законодавством про працю, колективним договором, угодою.

Підготовлений проект колективного договору обговорюється у трудовому колективі і вноситься на розгляд загальних зборів (конференції) трудового колективу.

Проте, у разі, якщо збори (конференція) трудового колективу відхилять проект колективного договору або окремі його положення, то сторони відновлюють переговори для пошуку необхідного рішення. Термін таких переговорів не повинен перевищувати 10 днів, лише після цього проект в цілому вноситься на розгляд зборів (конференції) трудового колективу.

У разі схвалення проекту колективного договору загальними зборами (конференцією) трудового колективу він підписується уповноваженими представниками сторін не пізніше як через 5 днів з моменту його схвалення.

Колективний договір набирає чинності з дня його підписання представниками сторін або з дня, зазначеного у колективному договорі. Згідно з положеннями чинного законодавства, підписаний сторонами колективний договір підлягає повідомній реєстрації районними державними адміністраціями, районними у містах Києві та Севастополі державними адміністраціями, виконавчими комітетами сільських, селищних та міських Рад (Положення про порядок повідомної реєстрації галузевих і регіональних угод, колективних договорів, затверджене постановою Кабінету Міністрів України від 5 квітня 1994 р. № 225).

Як зазначено в цьому Положенні повідомна реєстрація колективних договорів проводиться з метою забезпечення можливості для врахування їх умов під час розгляду трудових спорів (індивідуальних і колективних), що можуть виникнути за результатами застосування норм цих договорів уповноваженими на те органами, і засвідчує автентичність примірників і копії, поданих на реєстрацію.

Реєстрація проводиться в двотижневий строк з дня одержання колективного договору. Не пізніше наступного дня після реєстрації два примірники колективного договору повертаються сторонам, що його підписали і подали на реєстрацію. Копія договору зберігається реєструючим органом.

Органи, що реєструють колективні договори, не можуть вимагати від сторін цього договору внесення до нього змін або відмовити в його реєстрації. У повідомній реєстрації може бути відмовлено лише у тому разі, якщо подані на реєстрацію примірники і копія колективного договору не будуть автентичними.

Слід звернути увагу та тому, що зміни і доповнення до колективного договору протягом строку їх дії можуть вноситися тільки за взаємною згодою сторін в порядку, визначеному колективним договором.

Чинним законодавством України встановлено контроль за виконанням колективного договору. Такий контроль проводиться безпосередньо сторонами, що їх уклали, або уповноваженими ними представниками. У разі здійснення контролю сторони зобов'язані надавати необхідну для цього наявну інформацію.

Доцільно зазначити, що положення колективного договору поширюються на всіх працівників підприємств незалежно від того, чи є вони членами профспілки, і є обов'язковими як для власника або уповноваженого ним органу, так і для працівників підприємства.

Крім того, слід наголосити на тому, що після закінчення строку дії колективний договір продовжує діяти до того часу, поки сторони не укладуть новий або не переглянуть чинний, якщо інше не передбачено самим договором.

Також, колективний договір зберігає чинність у разі зміни складу, структури, найменування уповноваженого власником органу, від імені якого укладено цей договір.

У разі реорганізації підприємства він зберігає чинність протягом строку, на який його було укладено, або може бути переглянутий за згодою сторін.

У разі зміни власника підприємства чинність колективного договору зберігається протягом строку його дії, але не більше одного року. У цей період сторони повинні розпочати переговори про укладення нового чи зміну або доповнення чинного колективного договору.

Варто відзначити, що у разі ліквідації підприємства колективний договір діє протягом усього строку проведення ліквідації.

Слід звернути увагу на те, що на новоствореному підприємстві колективний договір укладається за ініціативою однієї із сторін у тримісячний строк після реєстрації підприємства, якщо законодавством передбачено реєстрацію, або після рішення про заснування підприємства, якщо не передбачено його реєстрацію.

Власник або уповноважений ним орган зобов'язаний ознайомити з колективним договором усіх працюючих. Доцільно зазначити, що сторони, які підписали колективний договір, щорічно в строки, передбачені колективним договором, звітують про його виконання.

На осіб, які представляють власника або уповноважений ним орган чи профспілки або інші уповноважені трудовим колективом органи і з вини яких порушено чи не виконано зобов'язання щодо колективного договору, накладається штраф, і вони також несуть дисциплінарну відповідальність.

Разом з цим, на вимогу профспілок, іншого уповноваженого трудовим колективом органу власник, або уповноважений ним орган зобов'язаний вжити заходів, передбачених законодавством, до керівника, з вини якого порушуються чи не виконуються зобов'язання по колективному договору.

Також, чинним законодавством України передбачено відповідальність у вигляді штрафу та дисциплінарну відповідальність за ухилення від участі в переговорах по укладенню, зміні або доповненню колективного договору, за порушення і невиконання колективного договору, за ненадання інформації, необхідної для колективних переговорів і здійснення контролю.

Порядок і строки накладення штрафів регламентуються Кодексом України про адміністративні правопорушення.

Справи з цих питань розглядаються судом за поданням однієї із сторін колективного договору, відповідних комісій або з ініціативи прокурора.

1.2. Огляд і аналіз існуючого програмного забезпечення управління договорами на підприємстві.

На сьогоднішній день існує значна кількість програм, в яких реалізовані функції ведення договорів. Опишемо далі найбільш відомі.

Програма IS-Pro - це українська програма від компанії-розробника “Інтелект-Сервіс”, створена для автоматизації управління підприємствами та бюджетними організаціями різного профілю.

Закладені в ній принципи допомагають оптимізувати всі основні процеси роботи підприємства від управління логістикою і складськими ресурсами до ведення бухгалтерії та податкового обліку.

Програма ISpro – багатофункціональний програмний комплекс, що забезпечує оптимізацію всіх процесів від бухгалтерського до управлінського обліку

Гнучка система-конструктор дозволяє обрати найбільш оптимальну конфігурацію для Вашого підприємства. Ви самі обираєте, які процеси роботи автоматизувати

Розробник постійно відслідковує зміни українського законодавства та світових стандартів стратегій управління.

команда забезпечить впровадження, адаптацію та подальшу технічну підтримку в роботі з програмою

За роки роботи з програмою створили широку базу інструкцій та роз’яснень на основі питань та потреб, що найчастіше виникають у наших клієнтів проводять як базове навчання працівників по роботі з програмою, так і спеціалізовані вебінари за запитами наших клієнтів (Рис. 1.1).

Модулі системи

Модулі системи відповідають за окремі процеси управління підприємством, та являються самостійними підсистемами. У кожному окремому випадку ми створюємо унікальну конфігурацію на основі тих модулів програми, що найбільш оптимально забезпечать потреби клієнта.

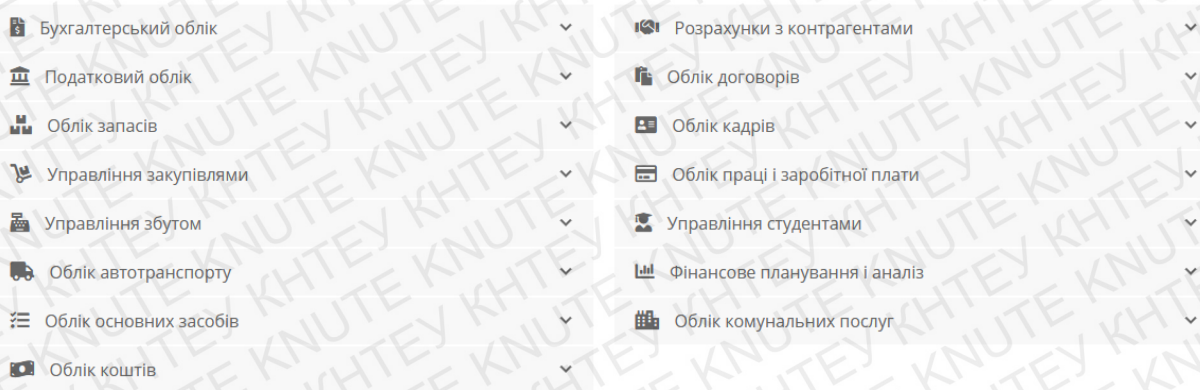


Рис. 1.1. Модулі системи

Програма РБюджет це програма для бюджетних установ була створена спеціально для роботи бухгалтерів в невеликих бюджетних бухгалтерських установах і має досить широкий набір функцій. Опишемо їх далі.

-Планування. Ведення планових показників (формується весь комплекс кошторисної документації, включно до додатків до рішення сесії) балансування бюджету. Можливість внесення даних у розрізі підвідомчих установ (наприклад школи і дитсадки для відділів освіти).

-Фінансування. Введення юридичних, фінансових зобов'язань, платіжних доручень, розподілів, розпоряджень (з експортом в dbf файли) контроль залишків кошторису, юридичних і фінансових зобов'язань. Формується картка аналітичного обліку отриманих асигнувань, картка аналітичного обліку касових видатків. Можливість обліку касових видатків в розрізі підвідомчих установ.

-Звітність. Основні форми звітності з можливістю автоматичного занесення даних та перевіркою правильності введення. Можливість формування зведеного звіту на основі звітів підвідомчих установ та введення звітів по підвідомчих установах.

-Бух облік. Внесення проведень (унікальна розробка «плаваючої» аналітики), формування меморіальних ордерів та книги журнал-головна з будь яким рівнем деталізації по рівнях аналітики нижчого рівня. Автоматичне

проведення фінансування та касових видатків та деяких видів проведень.
Можливість формування картки обліку фактичних видатків.

-Облік активів. Формування інвентаризаційних описів. Автоматичне введення надходжень згідно накладних, функції внутрішньої передачі між матеріально відповідальними особами, списання. Формування оборотної відомості, документів внутрішньої передачі та списання. Зручний поточний контроль наявних активів в розрізі рахунків і МВО. Перенесення даних у вигляді проведень у модуль бухгалтерського обліку.

-Облік договорів. Ведення обліку договорів заключених з сторонніми організаціями, додаткових угод, актів / накладних та експортувати дані для розміщення на сайті Є-Дата.

-Планування заробітної плати. Внесення даних для планування заробітної плати співробітників установи на рік з можливістю аналізувати до кошторисних призначень та формувати документи: штатний розпис (в різних формах), тарифікаційний список.

-Облік робочого часу. Внесення даних про використаний робочий час співробітників установи помісячно з можливістю формувати документ: Табелю обліку робочого часу.

-Заробітна плата. Нарахування заробітної плати співробітників на основі даних планування заробітної плати та обліку робочого часу з можливістю формувати документ: відомість розрахунку заробітної плати.

Наведемо далі характеристику таких поширених інформаційних систем, як BAS та 1С:Підприємство 8.

Багато підприємств на етапі планування та проведення робіт з автоматизації обліку не повною мірою розуміють усі переваги, які вони отримують від такої діяльності. Тому часто рішення про впровадження затримується, а отримані результати автоматизації скромні, дають незначний ефект лише за рахунок зменшення ручної праці, але не створюють значних конкурентних переваг та навіть викликають розчарування.

Найпоширеніші помилки, які заважають найбільш ефективно автоматизувати бізнес:

- використання мало функціонального комерційного або безкоштовного програмного забезпечення;
- острівна автоматизація - впровадження різних ділянок фінансового, управлінського, бухгалтерського обліку без можливості інтеграції даних в єдину систему управління підприємством;
- впровадження програм силами власних співробітників або локальних фахівців, які не мають високої кваліфікації та відповідного досвіду.
- Якщо ви очікуєте від автоматизації віддачі не 10—20 %, а бажаєте збільшити її в декілька разів, правильним підходом стане застосування програм обліку Business Automation Software (BAS), «1С:Підприємство 8.3» як структурної основи організації. Це максимально сприятиме використанню таких нематеріальних активів, як: правильний добір та організація роботи персоналу, управління фінансовими потоками, удосконалення структури відносин із клієнтами, партнерами. Програми обліку BAS, 1С:Підприємство також дадуть можливість визначати стратегічні цілі, ефективно проводити операційну діяльність, мати чітке розуміння ринкових змін у режимі реального часу та можливість адаптуватися до них і реагувати швидше, ніж це зроблять конкуренти.
- Компанія НЕТСОФТ спеціалізується на постачанні типових і галузевих рішень ліцензійного програмного забезпечення лінійки Business Automation Software (BAS), 1С:Підприємство. Надає послуги автоматизації, впровадження, підтримки, інформаційно-технологічного супроводу (ІТС), широкого спектра стандартних і доопрацьованих прикладних рішень (конфігурацій) програм Business Automation Software (BAS), 1С:Підприємство для України.
- У своєму складі має сертифікованих співробітників — професіонал, спеціаліст, спеціаліст-консультант за такими напрямками, як бухгалтерський облік, торгові складські операції, виробничий облік, конфігурування

програмування програм Business Automation Software (BAS), 1С:Підприємство 8.3, керівник проєктів.

- Основною метою нашої діяльності є: побудова систем бухгалтерського та управлінського обліку, збільшення продуктивності роботи бізнесу замовника, удосконалення бізнес-процесів, підвищення ефективності використання ресурсів, поліпшення систем мотивації співробітників.

- На сторінках сайту розміщені огляди популярних конфігурацій систем Business Automation Software (BAS), 1С:Підприємство з метою первинного знайомства та вибору відповідного програмного продукту для впровадження. Також розділ сайту «Статті» містить більш детальні кейси, які можуть бути цікаві для директорів і керівників компаній, головних бухгалтерів та фінансових директорів, співробітників ІТ-відділів.

- На сайті ви можете купити 1С:Підприємство 8.3 та купити Business Automation Software (BAS) або замовити демонстрацію прикладного рішення, яке вас зацікавило (Рис. 1.2).

ПОПУЛЯРНІ ПРОГРАМИ ТА ПОСЛУГИ BUSINESS AUTOMATION SOFTWARE (BAS)

Опис послуг, ціни на обслуговування і підтримку програм Business Automation Software (BAS)	від 660	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Роздрібна торгівля</u>	6 690	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Бухгалтерія</u>	2 240	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Бухгалтерія КОРП</u>	18 000	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Малий бізнес</u>	8 400	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Управління торгівлею</u>	8 400	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Комплексне управління підприємством</u>	34 000	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Будівництво. Бухгалтерія</u>	8 400	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS АГРО. Бухгалтерія</u>	11 700	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Управління авто транспортом. Стандарт</u>	12 300	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS ERP</u>	180 000	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Управління холдингом</u>	210 000	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ
<u>BAS Документообіг КОРП</u>	62 700	КУПИТИ	ЗАМОВИТИ ДЕМОНСТРАЦІЮ

Рис. 1.2. Програми та послуги BAS.

1.3. Засоби розробки програм та баз даних в створенні систем управління договорами на підприємстві.

До інструментального програмного забезпечення відносяться засоби розробки програмного забезпечення. Це системи програмування, що включають програмні засоби, необхідні для автоматичної побудови машинного коду. Вони є інструментами для програмістів - професіоналів і дозволяють розробляти програми на різних мовах програмування.

До складу засобів розробки програмного забезпечення входять наступні програми:

- асемблери - комп'ютерні програми, що здійснюють перетворення програми у формі вихідного тексту на мові асемблера в машинні команди у вигляді об'єктного коду;
- транслятори - програми, виконують трансляцію програми;

- компілятори - програми, що переводять текст програми на мові високого рівня в еквівалентну програму на машинній мові;
- інтерпретатори - програми, що аналізують команди або оператори програми і тут же виконують їх;
- компоновщики (редактори зв'язків) - програми, які виробляють компоновку - приймають на вхід один або кілька об'єктних модулів і збирають по ним здійснений модуль;
- препроцесори вихідних текстів - це комп'ютерні програми, що приймають дані на вході, і видають дані, призначені для входу іншої програми, наприклад такий, як компілятор;
- окладники (debugger) - програми, що є модулем середовища розробки або окремим додатком, призначеним для пошуку помилок у програмі;
- спеціалізовані редактори вихідних текстів - програми, необхідні для створення і редагування вихідного коду програм. Спеціалізований редактор вихідних текстів може бути окремим додатком або вбудованим в інтегроване середовище розробки та ін.

Мови, що представляють алгоритми у вигляді послідовності читаних (Не двійково-кодованих) команд, називаються алгоритмічними мовами. Алгоритмічні мови підрозділяються на машинно-орієнтовані, процедурно-орієнтовані і проблемно-орієнтовані.

Машинно-орієнтовані мови відносяться до мов програмування низького рівня - програмування на них найбільш трудомістким, але дозволяє створювати оптимальні програми, які максимально враховують функціонально-структурні особливості конкретного комп'ютера. Програми на цих мовах, при інших рівних умовах, будуть більш короткими і швидкими. Крім того, знання основ програмування на машинно-орієнтованій мові дозволяє спеціалісту докладним чином розібратися з архітектурою комп'ютера. Більшість команд машинно-орієнтованих мов при трансляції (перекладі) на машинний (двійковий) мова генерують одну машинну команду.

Процедурно-орієнтовані і проблемно-орієнтовані мови відносяться до мов високого рівня, що використовують макрокоманди. Макрокоманда при трансляції генерує багато машинних команд (для процедурно орієнтованої мови це співвідношення в середньому "1 до десяткам машинних команд", а для проблемно-орієнтованого - "1 до сотень машинних команд". Процедурно орієнтованими мовами програмування є самими використовуваними (Basic, Visual Basic, Pascal , Borland Delphi, C та ін.). У цьому випадку програміст повинен описувати всю процедуру вирішення завдання, тоді як проблемно-орієнтовані мови (їх називають також непроцедурного) дозволяють лише формально ідентифікувати проблему і вказати склад, структури представлення і формати вхідний і вихідний інформації для завдання.

При виконанні інструкцій програм комп'ютера необхідно перетворити зручні для людського сприйняття оператори, написані на якомусь мові програмування, у форму, задній хід для комп'ютера. Інструментальне програмне забезпечення має спеціальні програми, що транслюють (translate) текст програм, написаних на різних мовах програмування, в машинні коди, які потім виконуються комп'ютером. Цей вид програмного забезпечення називається компілятором або інтерпретатором. Текст програми, написаної на мові програмування високого рівня, до того як бути перетвореним в машинні коди, називається вихідним кодом (source code). Компілятор (compiler) перетворює вихідний код в машинні коди, звані об'єктним кодом (object code) - програмою на вихідному мові транслятора. Перед виконанням відбувається процес редагування зв'язків (linkage editing), що полягає в тому, що модулі вихідний програми об'єднуються з іншими модулями об'єктного коду, що містять, наприклад, дані. Результуючий завантажувальний модуль - це команди, що безпосередньо виконуються комп'ютером. Деякі мови програмування містять не компілятор, а інтерпретатор (interpreter), який перетворює кожне окреме вираз вихідного коду в машинні коди і відразу виконує їх. Інтерпретатор зручний на етапі налагодження програми, оскільки забезпечує швидко зворотний зв'язок при виявленні помилки у вихідному коді.

До інструментального ПО відносять також деякі системи управління базами даних (СУБД). СУБД - це спеціалізований комплекс програм, призначений для організації та ведення баз даних. Так як системи управління базами даних не є обов'язковим компонентом обчислювальної системи, їх не відносять до системного програмного забезпечення. А так як окремі СУБД здійснюють лише службову функцію при роботі інших видів програм (веб-сервери, сервери додатків), їх не завжди можна віднести до прикладного програмного забезпечення. З цих причин їх часто відносять до інструментального програмного забезпечення.

Основні функції таких СУБД:

- управління даними у зовнішній пам'яті (на дисках);
- керування даними в оперативній пам'яті з використанням дискового кеша;
- фіксація змін у спеціальних журналах, резервне копіювання і відновлення бази даних після збоїв;
- підтримка мов БД (мова визначення даних, мова маніпулювання даними).

База даних (БД) - упорядкований набір логічно взаємопов'язаних даних, що використовується спільно, та призначений для задоволення інформаційних потреб користувачів. У технічному розумінні включно й система управління БД.

Система управління базами даних (СУБД) - це комплекс програмних і мовних засобів, необхідних для створення баз даних, підтримання їх в актуальному стані та організації пошуку в них необхідної інформації.

Централізований характер управління даними в базі даних передбачає необхідність існування деякої особи (групи осіб), на яку покладаються функції адміністрування даними, що зберігаються в базі.

Головним завданням БД є гарантоване збереження значних обсягів інформації та надання доступу до неї користувачеві або ж прикладній програмі. Таким чином БД складається з двох частин: збереженої інформації та системи

управління нею. З метою забезпечення ефективності доступу записи даних організовують як множину фактів (елемент даних).

Існує величезна кількість різновидів баз даних, що відрізняються за критеріями (наприклад, в Енциклопедії технологій баз даних [21] визначаються понад 50 видів БД).

Відзначимо тільки основні класифікації.

Класифікація БД за моделлю даних:

- ієрархічні,
- мережеві,
- реляційні,
- об'єктні,
- об'єктно-орієнтовані,
- об'єктно-реляційні.

Класифікація БД за технологією фізичного зберігання:

- БД у вторинній пам'яті (традиційні);
- БД в оперативній пам'яті (in-memory databases);
- БД у третинній пам'яті (tertiary databases).

Класифікація БД за вмістом:

- географічні.
- історичні.
- наукові.
- мультимедійні.

Класифікація БД за ступенем розподіленості:

- централізовані (зосереджені);
- розподілені.

Окреме місце в теорії та практиці займають просторові (англ. spatial), тимчасові, або темпоральні (temporal) і просторово-часові (spatial-temporal) БД.

Ієрархічні бази даних можуть бути представлені як дерево, що складається з об'єктів різних рівнів. Верхній рівень займає один об'єкт, другий - об'єкти другого рівня і т.д.

Між об'єктами існують зв'язки, кожен об'єкт може включати в себе декілька об'єктів більш низького рівня. Такі об'єкти перебувають у відношенні предка (об'єкт більш близький до кореня) до нащадка (об'єкт більш низького рівня), при цьому можлива ситуація, коли об'єкт-предок не має нащадків або має їх декілька, тоді як у об'єкта-нащадка обов'язково тільки один предок. Об'єкти, що мають загального предка, називаються близнюками.

Мережеві бази даних подібні до ієрархічних, за винятком того, що в них є покажчики в обох напрямках, які з'єднують споріднену інформацію.

До основних понять мережевої моделі бази даних відносяться: рівень, елемент (вузол), зв'язок.

Вузол - це сукупність атрибутів даних, що описують деякий об'єкт. На схемі ієрархічного дерева вузли представляються вершинами графа. У мережній структурі кожен елемент може бути пов'язаний з будь-яким іншим елементом.

Незважаючи на те, що ця модель вирішує деякі проблеми, пов'язані з ієрархічною моделлю, виконання простих запитів залишається досить складним процесом.

Також, оскільки логіка процедури вибірки даних залежить від фізичної організації цих даних, то ця модель не є повністю незалежною від програми. Іншими словами, якщо необхідно змінити структуру даних, то потрібно змінити і додаток.

Реляційна модель орієнтована на організацію даних у вигляді двовимірних таблиць. Кожна реляційна таблиця являє собою двовимірний масив і має наступні властивості:

- кожен елемент таблиці - один елемент даних;
- всі осередки в стовпчику таблиці однорідні, тобто всі елементи в стовпчику мають однаковий тип (числовий, символний тощо);
- кожен стовпчик має унікальне ім'я;
- однакові рядки в таблиці відсутні;
- порядок проходження рядків і стовпчиків може бути довільним.

Об'єктна СУБД ідеально підходить для інтерпретації складних даних, на відміну від реляційних СУБД, де додавання нового типу даних досягається ціною втрати продуктивності або за рахунок різкого збільшення термінів і вартості розробки додатків. Об'єктна база, на відміну від реляційної, не вимагає модифікації ядра при додаванні нового типу даних. Новий клас і його екземпляри просто надходять у зовнішні структури бази даних. Система управління ними залишається без змін.

Об'єктно-орієнтована база даних (ООБД) - база даних, в якій дані оформлені у вигляді моделей об'єктів, що включають прикладні програми, які управляються зовнішніми подіями. Результатом поєднання можливостей (особливостей) баз даних і можливостей об'єктно-орієнтованих мов програмування є об'єктно-орієнтовані системи управління базами даних (ООСУБД). ООСУБД дозволяють працювати з об'єктами баз даних також, як з об'єктами у програмуванні в об'єктно-орієнтованих мовах програмування. ООСУБД розширює мови програмування, прозора вводячи довготривалі дані, управління паралелізмом, відновлення даних, асоційовані запити й інші можливості.

Об'єктно-орієнтовані бази даних звичайно рекомендовані для тих випадків, коли потрібна високопродуктивна обробка даних, які мають складну структуру.

Система, яка забезпечує об'єктну інфраструктуру і набір реляційних розширювачів, називається "об'єктно-реляційною".

Об'єктно-реляційні системи поєднують переваги сучасних об'єктно-орієнтованих мов програмування з такими властивостями реляційних систем як множинні представлення даних і високорівневі непроцедурні мови запитів.

За технологією обробки даних бази даних поділяються на централізовані й розподілені.

Централізована база даних зберігається у пам'яті однієї обчислювальної системи. Якщо ця обчислювальна система є компонентом мережі ЕОМ,

можливий розподілений доступ до такої бази. Такий спосіб використання баз даних часто застосовують у локальних мережах ПК.

Розподілена база даних складається з декількох, можливо пересічних або навіть дублюючих одна одну частин, які зберігаються в різних ЕОМ обчислювальної мережі. Робота з такою базою здійснюється за допомогою системи управління розподіленою базою даних (СУРБД).

За способом доступу до даних бази даних поділяються на бази даних з локальним доступом і бази даних з віддаленим (мережевим) доступом.

Системи централізованих баз даних з мережевим доступом припускають різні архітектури подібних систем:

- файл-сервер;
- клієнт-сервер.

Файл-сервер. Архітектура систем БД з мережевим доступом передбачає виділення однієї з машин мережі в якості центральної (сервер). На такій машині зберігається спільно використовувана централізована БД. Усі інші машини мережі виконують функції робочих станцій, за допомогою яких підтримується доступ користувальницької системи до централізованої бази даних. Файли бази даних відповідно до призначених для користувача запитів передаються на робочі станції, де в основному і проводиться обробка. При великій інтенсивності доступу до одних і тих же даних продуктивність інформаційної системи падає. Користувачі можуть створювати також на робочих станціях локальні БД, які використовуються ними монопольно.

Клієнт-сервер. У цій концепції мається на увазі, що крім зберігання централізованої бази даних центральна машина (сервер бази даних) повинна забезпечувати виконання основного обсягу обробки даних. Запит на дані, який видається клієнтом (робочою станцією), породжує пошук і вилучення даних на сервері. Витягнуті дані (але не файли) транспортуються по мережі від сервера до клієнта. Специфікою архітектури клієнт-сервер є використання мови запитів SQL.

Області застосування баз даних. Історично системи управління базами даних орієнтувалися на вирішення завдань, пов'язаних у першу чергу з транзакційною обробкою структурованої інформації. Безумовно, найкращим, перевіреним часом рішенням тут була і залишається реляційна модель СУБД. Однак в останні роки область застосування баз даних незмінно розширювалася. З одного боку, потрібно керувати більш широким набором форматів даних, переходячи до вирішення спільних проблем управління корпоративною інформацією. З іншого - саме СУБД беруть на себе основні функції інтеграції даних і додатків корпоративних систем. (За даними Gartner Group, інформаційні відділи підприємств витрачають до 40% свого бюджету на вирішення завдань інтеграції діючих компонентів баз даних.) Саме цим пояснюється активний інтерес до обговорення архітектурних принципів і можливостей реалізації баз даних різних моделей - постреляційних, об'єктно-реляційних, XML.

Якщо постаратися класифікувати існуючі області застосування баз даних, а так само оцінити перспективи їхнього розвитку в даний час, то можна отримати приблизний список найбільш поширених класів:

- документографічні й документальні застосовуються у всіх базах органів влади та управління;
- бази даних з промислової, будівельної та сільськогосподарської продукції;
- бази даних з економічної та кон'юнктурної інформації (статистична, кредитно-фінансова, зовнішньоторговельна);
- фактографічні бази соціальних даних, які включають відомості про населення і про соціальні середовища;
- бази даних транспортних систем;
- довідкові дані для населення та установ (енциклопедії та довідники, розклади літаків і поїздів, адреси та телефони громадян і організацій);
- ресурсні бази даних, що включають фактографічну інформацію про природні ресурси (земля, вода, надра, біоресурси, гідрометеорологія, вторинні ресурси і відходи, екологічна обстановка);

- фактографічні бази і банки наукових даних, щоб забезпечити фундаментальні наукові дослідження;
- фактографічні бази даних у галузі культури і мистецтва;
- лінгвістичні бази даних, тобто машинні словники різного типу і призначення.

Останнім часом утворилися нові важливі області застосування баз даних, і кожна з них представляє принципово нове середовище, до якого необхідно адаптувати технології СУБД. Ці області отримали на ринку назви інтелектуально аналізу даних (data mining), сховищ даних (data warehousing), репозитаріїв даних (data repository).

Інтелектуальний аналіз даних. Ідея інтелектуально аналізу даних (data mining), тобто добування інформації з величезних масивів даних, накопичених зовсім для інших цілей, викликає сьогодні підвищений ентузіазм. Наприклад, авіакомпанії домагаються оптимального заповнення рейсів за рахунок аналізу накопичених раніше даних про резервування квитків. Можна навести одну історію про те, як була виявлена несподівана кореляція між покупками пива і покупками серветок у післяобідній період. Власник магазину наблизив один до одного відділи, які торгують пивом і серветками, а між ними помістив ще прилавки з картопляними чіпсами. У результаті збільшилися продажі усіх трьох видів товару.

Із запитамі, характерними для систем інтелектуально аналізу даних, пов'язана низка незвичайних проблем.

1. Вони включають, як правило, агрегацію величезних обсягів даних.
2. Вони мають нерегламентований характер; їх формулюють особи, відповідальні за прийняття рішень, коли їм необхідно виявити будь-які неочевидні взаємозв'язки.
3. У додатках, пов'язаних, наприклад, з торгівлею цінними паперами, дуже важливий малий час відповіді. Суть проблеми полягає тут у тому, щоб скоротити загальний час, необхідний для написання, налагодження та виконання запиту.

4. Досить часто користувач не в змозі точно сформулювати запит - йому просто потрібно виявити "що-небудь цікаве".

Таким чином, з видобутком даних пов'язані такі дослідницькі напрями:

- методи оптимізації складних запитів, які включають, наприклад, агрегацію та групування;
- методи підтримки "багатовимірних" запитів, що відносяться до даних, організованих у вигляді "куба", в осередках якого знаходяться потрібні дані;
- методи оптимізації використання третинної пам'яті;
- мови запитів дуже високого рівня, а також інтерфейси для підтримки користувачів, які не є експертами, і яким потрібні відповіді на нерегламентовані запити.

Сховища даних.

У сховищі даних накопичуються дані з однієї або більше баз даних. Існує безліч потенційних застосувань, а також підходів до організації сховищ даних. Наприклад, великий магазин може підтримувати сховище даних на основі транзакційних даних про касові операції для цілей видобутку даних. У сховищі даних може зберігатися інформація з багатьох баз даних для використання в надзвичайних ситуаціях. Наприклад, в єдиному сховищі даних підтримуються відомості про цивільну інфраструктуру (дороги, мости, трубопроводи тощо), оскільки, наприклад, після землетрусу навряд чи вдасться отримати цю інформацію з міст, що знаходяться поблизу епіцентру. Ще один приклад - використання сховища даних як "матеріалізованого уявлення" інтегрованої інформації. Альтернативою медіаторних систем, які дають цілісне уявлення даних, витягнутих з безлічі джерел, можуть служити сховища даних, що забезпечують фізичне зберігання інтегрованих даних. На відміну від сховищ, медіатори надають інформацію, розсилаючи запити декільком джерелам, подібно до того, як це відбувається при реалізації уявлень.

Деякі дослідницькі проблеми, що стосуються сховищ даних, збігаються з тими, які характерні для інтеграції даних у цілому, але є і деякі специфічні проблеми.

1. Інструменти для створення насосів даних (data pump), тобто модулів, що функціонують за середовищем джерел даних і поставляють у сховище ті зміни, які істотні з точки зору сховища; при цьому дані мають транслюватися у відповідності з глобальною моделлю і схемою сховища.

2. Методи "чистки даних" (data scrubbing), які забезпечують узгодження даних, видалення елементів, що відповідають різним уявленням одного й того ж об'єкта (наприклад "Sally Tones" і "SA Tones"), а також видалення неправдоподібних значень.

3. Засоби для створення і підтримки метасловника, який інформує користувачів про способи отримання даних.

. Репозитарії - програми, що відносяться до категорії репозитаріїв, характеризуються тим, що вони призначені для зберігання і управління як даними, так і метаданими, тобто інформацією про структуру даних. Приклади репозитаріїв - бази даних для підтримки комп'ютерного проектування, включаючи CASE (системи проектування програмного забезпечення), а також системи управління документами. Відмінна риса цих систем - часті зміни метаданих, характерні для будь-якого середовища проектування.

У репозитарії необхідно підтримувати безліч уявлень однієї й тієї ж або схожої інформації. Наприклад, програмний модуль має уявлення у вигляді вихідного коду, об'єктного коду, проміжного коду, готової програми, таблиць використань/визначень, документації. Зв'язки між усіма цими уявленнями повинні відстежуватися репозитарієм так, щоб зміни в одному з них автоматично поширювалися на інші уявлення того ж об'єкта.

Репозитарії повинні підтримувати поняття версій (моментальних знімків елементів даних, що змінюються в часі) і конфігурацій (версійних колекцій версій). Наприклад, різні релізи програмної системи будуть зазвичай формуватися як конфігурації з певних версій файлів вихідного коду.

Сховище повинне підтримувати еволюцію структури інформації та її метаданих таким чином, щоб при додаванні нових властивостей даних або нових зв'язків не була потрібна повна перекомпіляція.

Ось кілька прикладів додатків нового покоління, які визначають потреби у нових засобах розробки баз даних і можливості їхнього застосування.

База даних Системи спостереження Землі (EOSDIS).

Система спостереження Землі (EOS - Earth Observing System) являє собою безліч супутників, які запускає NASA починаючи з 1998 р. Їх призначення - збір інформації, необхідної для дослідників, зайнятих вивченням довгострокових тенденцій стану атмосфери, океанів, земної поверхні. Супутники поставляють інформацію в обсязі 1/3 Пбайт (Petabyte - 10¹⁵ байт) в рік. Передбачається, що ці дані будуть інтегруватися з уже існуючою інформацією, а також з даними з інших джерел (іноземні супутники, наземні станції спостереження) і накопичуватися в базі даних EOSDIS (EOS Data and Information System) небачених раніше масштабів.

EOSDIS призначена для інформаційного обслуговування, як фахівців, так і неспеціалістів. Передбачається, наприклад, що доступ до неї матимуть навіть школярі, які зможуть знайомитися з моделями формування погодних умов, з впливом вулканічних явищ тощо. Ось найбільш складні завдання, що виникають у зв'язку з цим проектом.

Електронна комерція на даний час існує ряд проектів, загальна мета яких - надати потенційним споживачам оперативний доступ до каталогів товарів з наступним електронним оформленням покупок. Передбачається, що можливою проміжною ланкою подібних систем буде електронний брокер. Брокери акумулюють дані з множинних джерел шляхом збору інформації, наприклад, з декількох каталогів предметів одягу. Кінцевому покупцю такий брокер запропонує оперативне оформлення покупок.

Як і проект EOSDIS, система електронної комерції передбачає мережеву взаємодію величезного числа учасників торгових угод. Різниця полягає у тому, що в EOSDIS є один головний постачальник інформації і безліч

її споживачів, а торгова система має на увазі наявність безлічі постачальників і безлічі споживачів. Крім того, учасники в даному випадку можуть відчувати певну взаємну недовіру і, можливо, мають свої приватні закриті інформаційні системи. Найбільш складні проблеми, пов'язані з проектами цього роду - система електронної комерції повинна мати високонадійні засоби розподіленої аутентифікації та переказу грошових сум.

Інформаційна система охорони здоров'я. Лікаряю в процесі роботи необхідний доступ до безлічі джерел інформації. Наприклад, історії хвороби одного пацієнта можуть перебувати в різних лікарнях, клініках, страхових установах. Для отримання повної картини їх усі варто зібрати. Точно так само існує безліч систем і баз даних, що надають інформацію про ліки, лікувальні процедури, діагностичні засоби.

Записи лікаря, результати обстежень, інформація про рахунки за лікування, договори медичного страхування для кожного пацієнта повинні фіксуватися в електронній формі та залишатися доступними для подальшого використання. Впровадження сучасних інформаційних технологій у галузі охорони здоров'я надасть кардинальний вплив на такі характеристики медичного обслуговування, як вартість, якість, повсюдна доступність. Ось низка проблем, які виникають у зв'язку з реалізацією подібної системи - інтеграція різнорідних джерел вже накопиченої інформації, засоби контролю доступу, що забезпечують необхідний рівень конфіденційності, інтерфейси доступу до інформації, зручні для різних категорій працівників охорони здоров'я.

Електронні публікації у видавничому бізнесі, як і у сфері охорони здоров'я, очікується в найближчому майбутньому ряд глибоких змін. Стає можливим, наприклад, зберігання книг та статей в електронному вигляді та оперативна доставка їх споживачам високошвидкісними мережевими каналами. Далі, саме поняття публікації істотно розширюється - документ може містити графічні, аудіо-чи відео-включення, анотацію, інші супровідні елементи. Загальний обсяг інформації, яка доступна вже сьогодні, перевищує розміри

бази даних EOSDIS, а в найближчому майбутньому очікується його зростання приблизно на порядок.

Коллективне проектування. Великі й складні проекти, наприклад, в області літакобудування, реалізуються сьогодні об'єднаними зусиллями декількох незалежних компаній. Час життя інформації, що відноситься до подібних проектів, може вимірюватися десятиліттями, оскільки вона необхідна для підтримки, модифікації та розвитку. Конструкторські рішення, перш ніж стати фізичною реальністю, можуть проходити стадії комп'ютерного моделювання - для дослідження робочих властивостей, зручності складання виробів, правильності функціонування. Еволюція конструкторських схем починається задовго до випуску першого виробу і продовжується ще довгий час після цього, що призводить до розростання інформаційної конфігурації, яка повинна відображати поточний стан розробки, експериментальні версії, історичний розвиток. Для різних сфер конструювання характерне використання різнорідних конструкторських інструментальних систем, заснованих на різних моделях і системах позначень. Причому процес конструювання може тривати довше, ніж існують застосовувані інструменти, а значить, компоненти однієї й тієї ж конструкції можуть розроблятися із застосуванням різних версій інструментальної системи.

Як і в деяких із згадуваних раніше сфер, тут також постає задача інтеграції різнорідних джерел історично накопиченої інформації. Коллективне проектування вимагає нових форм управління спільним доступом до баз даних і механізмів поділу інформації. Для регулювання різнорідних процесів, що виконуються спільно, таких як моделювання та конструювання, необхідні засоби управління потоками робіт, засновані на чітко визначених взаємодіях допомогою довготривалих транзакцій.

Коротка характеристика деяких СУБД. MySQL - вільна система управління базами даних. MySQL є власністю компанії Oracle Corporation, що отримала її разом з поглиненою Sun Microsystems, яка здійснює розробку і підтримку додатку. Розповсюджується під GNU General Public License і під

власною комерційною ліцензією, на вибір. Крім цього розробники створюють функціональність на замовлення ліцензійних користувачів, саме завдяки такому замовленню майже в найраніших версіях з'явився механізм реплікації.

Цю систему управління базами даних з відкритим кодом було створено як альтернатива комерційним системам. MySQL із самого початку була дуже схожою на mSQL, проте з часом вона все розширювалася і зараз MySQL - одна з найпоширеніших систем управління базами даних. Вона використовується, у першу чергу, для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

MySQL є рішенням для малих і середніх додатків. Зазвичай MySQL використовується як сервер, до якого звертаються локальні або віддалені клієнти, проте до дистрибутиву входить бібліотека внутрішнього сервера, що дозволяє включати MySQL до автономних програм. Вихідні коди сервера компілюються на багатьох платформах. Найповніше можливості сервера виявляються в UNIX-системах, де є підтримка багатонитевості, що підвищує продуктивність системи в цілому.

Гнучкість СУБД MySQL забезпечується підтримкою великої кількості типів таблиць: користувачі можуть вибрати як таблиці типу MyISAM, що підтримують повнотекстовий пошук, так і таблиці InnoDB, що підтримують транзакції на рівні окремих записів. Більш того, СУБД MySQL поставляється із спеціальним типом таблиць EXAMPLE, що демонструє принципи створення нових типів таблиць. Завдяки відкритій архітектурі й GPL-ліцензуванню, в СУБД MySQL постійно з'являються нові типи таблиць. MySQL характеризується великою швидкістю, стійкістю і простотою використання.

Для некомерційного використання MySQL є безкоштовною. Можливості сервера MySQL:

- простота у встановленні та використанні;
- підтримується необмежена кількість користувачів, що одночасно працюють із БД;
- кількість рядків у таблицях може досягати 50 млн.;

- висока швидкість виконання команд;
- наявність простої та ефективної системи безпеки.

PostgreSQL - об'єктно-реляційна система управління базами даних. Є альтернативою як комерційним СУБД (Oracle Database, Microsoft SQL Server, IBM DB2 та інші), так і СУБД з відкритим кодом (MySQL, Firebird, SQLite).

Порівняно до інших проектів з відкритим кодом, такими як Apache, FreeBSD або MySQL, PostgreSQL не контролюється якоюсь однією компанією, її розробка можлива завдяки співпраці багатьох людей та компаній, які хочуть використовувати цю СУБД та впроваджувати у неї найновіші досягнення.

СУБД Oracle - це найпотужніший програмний комплекс, що дозволяє створювати додатки будь-якої складності. Ядром цього комплексу є база даних, що зберігає інформацію, кількість якої за рахунок наданих засобів масштабування практично безмежна. З високою ефективністю працювати з цією інформацією одночасно може практично будь-яка кількість користувачів (за наявності достатніх апаратних ресурсів), не проявляючи тенденції до зниження продуктивності системи при різкому збільшенні їхньої кількості.

Механізми масштабування в СУБД Oracle останньої версії дозволяють безмежно збільшувати потужність і швидкість роботи сервера Oracle і своїх додатків, просто додаючи нові й нові вузли кластеру. Це не вимагає зупинки працюючих додатків, не вимагає переписування старих додатків, розроблених для звичайної одномашинної архітектури. Крім того, вихід з ладу окремих вузлів кластера також не призводить до зупинки програми.

Вбудовування до СУБД Oracle JavaVM, повномасштабної підтримки серверних технологій (Java Server Pages, Java-сервлети, модулі Enterprise JavaBeans, інтерфейси прикладного програмування CORBA), призвели до того, що Oracle на сьогоднішній день де-факто є стандартом СУБД для Internet.

Ще однією складовою успіху СУБД Oracle є те, що вона поставляється практично для всіх існуючих на сьогодні операційних систем. Працюючи під Sun Solaris, Linux, Windows або на іншій операційній системі з продуктами Oracle не буде виникати ніяких проблем у роботі. СУБД Oracle однаково добре

працює на будь-якій платформі. Таким чином, компаніям, які розпочинають роботу з продуктами Oracle не доводиться міняти мережеве оточення. Існує лише невелика кількість відмінностей при роботі з СУБД, обумовлених особливостями тієї або іншої операційної системи. У цілому ж це завжди та ж сама безпечна, надійна і зручна СУБД Oracle.

Microsoft SQL Server - система управління реляційними базами даних, розроблена корпорацією Microsoft. Основна використовувана мова запитів - Transact-SQL, створена спільно Microsoft та Sybase. Transact-SQL є реалізацією стандарту ANSI / ISO щодо структурованої мови запитів (SQL) із розширеннями. Використовується для роботи з базами даних розміром від персональних до великих баз даних масштабу підприємства, конкурує з іншими СУБД у цьому сегменті ринку.

При взаємодії з мережею Microsoft SQL Server і Sybase ASE використовують протокол рівня додатків під назвою Tabular Data Stream (TDS, протокол передачі табличних даних). Протокол TDS також був реалізований у проєкті FreeTDS з метою забезпечити різні додатки можливістю взаємодії з базами даних Microsoft SQL Server і Sybase.

Для забезпечення доступу до даних Microsoft SQL Server підтримує Open Database Connectivity (ODBC) - інтерфейс взаємодії додатків з СУБД. SQL Server надає можливість підключення користувачів через веб-сервіси, що використовують протокол SOAP. Це дозволяє клієнтським програмам, не призначеним для Windows, кросплатформно з'єднуватися з SQL Server.

Microsoft Office Access або просто Microsoft Access - реляційна СУБД корпорації Microsoft. Має широкий спектр функцій, включаючи зв'язані запити, зв'язок із зовнішніми таблицями і базами даних. Завдяки вбудованій мові VBA, в самому Access можна писати застосунки, що працюють з базами даних.

Основні компоненти MS Access:

- будівник таблиць;
- будівник екранних форм;

- будівник SQL-запитів (мова SQL в MS Access не відповідає стандарту ANSI);
- будівник звітів, що виводяться на друк.

Вони можуть викликати скрипти мовою VBA, тому MS Access дозволяє розробляти програми і бази даних практично "з нуля" або написати оболонку для зовнішньої БД.

MS Access є файл-серверною СУБД і тому застосовується лише до маленьких додатків. Відсутній ряд механізмів, необхідних у багатокористувацьких БД, таких, наприклад, як тригери.

Істотно розширює можливості MS Access з написання додатків механізм зв'язку з різними зовнішніми СУБД: "зв'язані таблиці" (зв'язок з таблицею СУБД) і "запити до сервера" (запит на діалекті SQL, який "розуміє" СУБД). Також MS Access дозволяє будувати повноцінні клієнт-серверні додатки на СУБД MS SQL Server. При цьому є можливість поєднати з притаманною MS Access простотою інструменти для управління БД і засоби розробки.[48]

РОЗДІЛ 2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ

2.1. Постановка задачі.

Задачею випускної кваліфікаційної роботи є розробка програмного забезпечення для створення та ведення договорів на підприємстві з метою підвищення ефективності його діяльності. Необхідно розробити інструментальні засоби, які дозволять створити базу даних та виконувати в ній функції створення, редагування, накопичення та перегляду договорів підприємства. В програмі необхідно реалізувати режими представлення накопиченої бази договорів в стандартних форматах текстових процесорів.

Розробка програми має бути виконана в інтегрованому середовищі Visual Studio 2019 на мові програмування Visual Basic. База даних зроблена засобами СУБД Microsoft Access 2016.

В базу даних занести договори, які можуть укладатись на підприємстві, а також повинна бути можливість їх додавати та видаляти. Перегляд договорів має бути у форматі PDF. Також має бути можливість виводу договору на друкування на принтер.

Має бути можливість пошуку договору по назві або по ідентифікаційному коду, який буде надаватися договору.

2.2. Проектування візуальної частини програмного забезпечення

Візуальна частина програмного забезпечення має бути інтуїтивна та зрозуміла.

Для цього має використовуватися прості віконні інтерфейси, системи взаємодії з користувачем, такі як: текстове поле, кнопка, переглядач PDF файлів, переглядач таблиць із бази даних.

2.3. Нормативно-довідкові документи системи ведення договорів.

Нормативно-методичне забезпечення - це сукупність документів організаційного, організаційно-методичного, організаційно-розпорядницького, технічного, техніко-економічного та економічного характеру, а також

нормативно-довідкові матеріали та інші документи, затверджені у встановленому порядку компетентними органами або керівництвом готелю.

Нормативно-методичне забезпечення створює умови для ефективного процесу управління персоналом. Воно полягає в організації розробки і застосування методичних документів, а також ведення нормативного господарства в системі УП.[68]

Нормативно-методичні матеріали групуються за їхнім змістом. Розрізняють нормативно-довідкові документи; документи організаційного, організаційно-розпорядницького та організаційно-методичного характеру; документи технічного, техніко-економічного та економічного характеру. Перша група включає норми і нормативи, необхідні при вирішенні задач готелю та планування праці у сфері надання послуг та управління.

Документи другої групи регламентують задачі, функції, права, обов'язки підрозділів і окремих працівників системи управління персоналом; містять методи і правила виконання робіт з управління персоналом. Наприклад, Цивільний кодекс, КЗпП, регулюючі документи (положення, інструкції, правила) державних органів, накази, положення, інструкції, що видаються керівником організації або відповідним підрозділом з питань праці, чисельності та оплати праці тощо.

Документи технічного, техніко-економічного та економічного характеру містять правила, стандарти, норми, вимоги, що регламентують планування приміщень і робочих місць, надання послуг, тарифні ставки, коефіцієнти доплати, типові норми часу, заходи з техніки безпеки та інше.

Найважливішим внутрішнім організаційно-регламентуючим документом є Положення про підрозділ - документ, що регламентує діяльність кожного структурного підрозділу кадрової служби: його завдання, права, функції, відповідальність.

На основі типових документів з урахуванням особливостей організації працівники служби управління персоналом розробляють документи для

внутрішнього користування. Так, важливими організаційно-розпорядницькими документами є Правила внутрішнього розпорядку, що включають такі розділи:

- загальні положення;
- порядок прийому і звільнення робітників та службовців;
- основні обов'язки робітників та службовців;
- основні обов'язки адміністрації;
- робочий час і його використання;
- винагорода за успіхи в роботі;
- відповідальність за порушення трудової дисципліни.

У розпорядженні кадрової служби знаходяться всі нормативні акти, на підставі яких складаються документи внутрішнього користування:

- колективний договір;
- правила внутрішнього трудового розпорядку;
- положення про підрозділи (відділи, служби, групи) та ін.

Найважливішим організаційним документом є Колективний договір, що розробляється за особистої участі підрозділу служби з управління персоналом. Колективний договір - це угода, що укладається трудовим колективом і адміністрацією щодо врегулювання їхніх взаємин у процесі діяльності на календарний рік.

До документів організаційно-методичного характеру належать ті, що регламентують виконання функцій з управління персоналом. Сюди входять:

- Положення щодо формування кадрового резерву готелю;
- Положення щодо організації адаптації працівників;
- Рекомендації щодо організації підбору і добору персоналу;
- Положення щодо врегулювання взаємин у колективі;
- Положення щодо оплати і стимулювання праці;
- Інструкція з дотримання правил техніки безпеки тощо.

Основним документом у кадровій службі є посадова інструкція-документ, що регламентує діяльність у рамках кожної управлінської посади, який містить вимоги до працівника, що займає цю посаду. Вона може бути

складена на основі типових вимог до посади, що містяться в кваліфікаційному довіднику посад керівників, фахівців, службовців, але з урахуванням особливостей певного готелю і соціально-економічних умов його діяльності.

Як й інші традиційні інженерні дисципліни, розробка програмного забезпечення має справу з проблемами якості, вартості та надійності. Деякі програми містять мільйони рядків вихідного коду, які, як очікується, повинні правильно виконуватися в умовах, що змінюються. Складність ПЗ порівнянна зі складністю найбільш складних з сучасних машин, таких як літаки.

Учасники процесу розробки програмного забезпечення:

- користувач;
- замовник
- дизайнер
- керівник проекту
- аналітик
- тестувальник
- постачальник.

Найбільш поширеними проблемами, що виникають в процесі розробки ПЗ, вважають:

- недолік прозорості. У будь-який момент часу складно сказати, в якому стані знаходиться проект і який відсоток його завершення. Дана проблема виникає при недостатньому плануванні структури (чи архітектури) майбутнього програмного продукту, що найчастіше є наслідком відсутності достатнього фінансування проекту: програма потрібна, скільки часу займе розробка, якими є етапи, чи можна якісь етапи виключити або заощадити — наслідком цього процесу є те, що етап проектування скорочується;
- Недолік контролю. Без точної оцінки процесу розробки зриваються графіки виконання робіт і перевищуються встановлені бюджети. Складно оцінити обсяг виконаної роботи і роботи, що залишилася.

Дана проблема виникає на етапі, коли проект, завершений більш ніж наполовину, продовжує розроблятися після додаткового фінансування без оцінки ступеня завершеності проекту.

- Недолік трасування.
- Недолік моніторингу. Неможливість спостерігати хід розвитку проекту не дозволяє контролювати хід розробки в реальному часі. За допомогою інструментальних засобів менеджери проектів приймають рішення на основі даних, що надходять в реальному часі.

Дана проблема виникає в умовах, коли вартість навчання менеджменту володінню інструментальними засобами порівнянна з вартістю розробки самої програми.

- Неконтрольовані зміни. У споживачів постійно виникають нові ідеї щодо розроблюваного програмного забезпечення. Вплив змін може бути суттєвим для успіху проекту, тому важливо оцінювати пропоновані зміни та реалізовувати тільки схвалені, контролюючи цей процес за допомогою програмних засобів.

Дана проблема виникає внаслідок небажання кінцевого споживача використовувати ті чи інші програмні середовища. Наприклад, коли при створенні клієнт-серверної системи споживач висуває вимоги не тільки до операційної системи на комп'ютерах-клієнтах, а й на комп'ютері-сервері.

- Недостатня надійність. Найскладніший процес — пошук і виправлення помилок у програмах на ЕОМ. Оскільки число помилок у програмах заздалегідь невідомо, то заздалегідь невідома і тривалість налагодження програм і відсутність гарантій відсутності помилок в програмах. Слід зазначити, що залучення доказового підходу до проектування ПЗ дозволяє виявити помилки в програмі до її виконання. У цьому напрямку багато працювали Кнут, Дейкстра і Вірт. Професор Вірт при розробці Паскаля і Оберона за рахунок строгості їх синтаксису домігся математичної доказовості виконання і правильності програм, написаної на цих мовах.

Дана проблема виникає при неправильному виборі засобів розробки. Наприклад, при спробі створити програму, що вимагає коштів високого рівня, за допомогою засобів низького рівня. Наприклад, при спробі створити засоби автоматизації з СУБД на асемблері. У результаті вихідний код програми виходить занадто складним і погано піддається структуруванню.

- Неправильний вибір методології розробки програмного забезпечення. Процес вибору необхідної методології може проблемно відбитися на всіх показниках програмного забезпечення — це його гнучкість, вартість і функціональність. Так звані гнучкі методології розробки допомагають вирішити основні проблеми, однак, варто відзначити, що і каскадна модель (waterfall) так само має свої переваги. У деяких випадках найбільш доцільним буде застосування гібридних методологій у зв'язці Agile + каскадна модель + MSF + RUP і т. д.

- Відсутність гарантій якості і надійності програм через відсутність гарантій відсутності помилок в програмах аж до формальної здачі програм замовникам.

Дана проблема не є проблемою, що відноситься виключно до розробки ПЗ. Гарантія якості — це проблема вибору постачальника товару (продукту).

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ ТА ВЕДЕННЯ ДОГОВОРІВ

3.1. Загальна характеристика програмного забезпечення системи створення та ведення договорів.

Програмне забезпечення допомагає в створенні, зберіганні та редагуванні договорів або інших документів на підприємстві.

У випускній кваліфікаційній роботі розроблені інструментальні засоби, які дозволяють створювати базу даних та виконувати в ній функції створення, редагування, накопичення та перегляду договорів підприємства. В програмі реалізовані режими представлення накопиченої бази договорів в стандартних форматах текстових процесорів.

Розробка програми виконана в інтегрованому середовищі Visual Studio 2019 на мові програмування Visual Basic. База даних зроблена засобами СУБД Microsoft Access 2016.

В базу даних заносяться договори, які укладаються на підприємстві, а також є можливість їх добавляти та видаляти. Перегляд договорів виконується у форматі pdf. Також є можливість виводу договору на друкування на принтері. Має можливість пошуку договору по назві або по ідентифікаційному коду, який надається.

3.2. Віконний інтерфейс доступу до функцій ведення нормативно-довідкової інформації та договорів підприємства.

На рисунку 3.1 зображено вікно програми, на якій знаходиться таблиця із бази даних, кнопка «Редагувати» (для редагування договору),

та панель у верхній частині вікна з трьома кнопками :«Редагування договору», «Договори» (таблиця з договорами), «RefreshDB» (кнопка для перезавантаження бази даних).

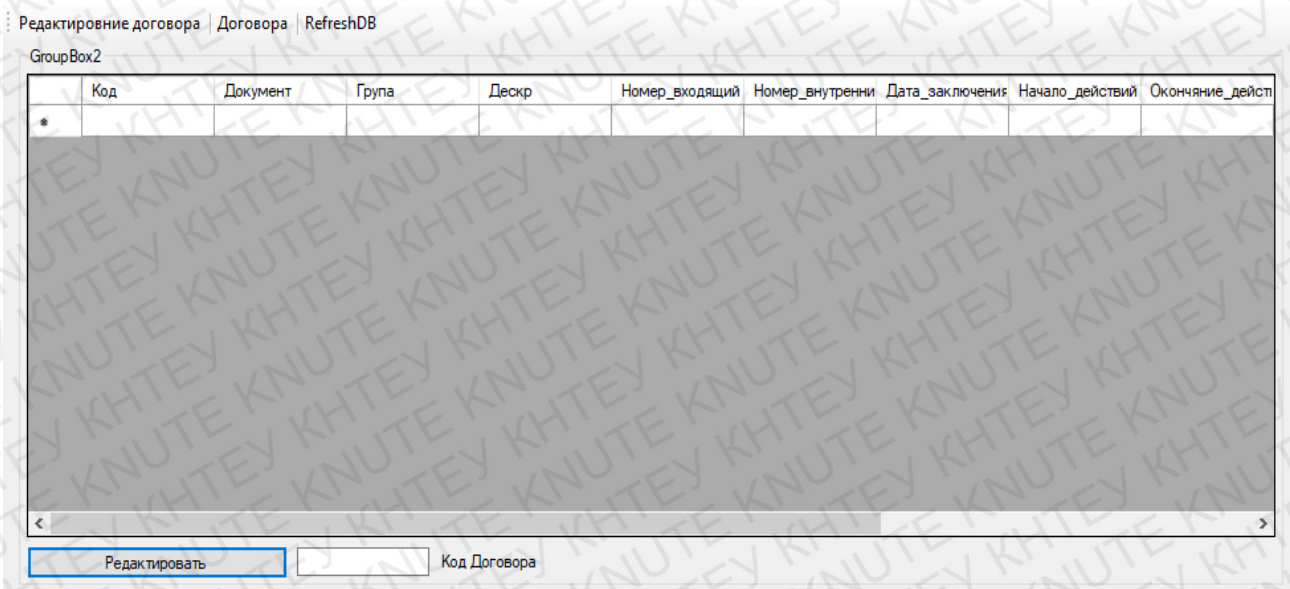


Рис. 3.1. Вікно програми з не завантаженою таблицею.

На рисунку 3.2 зображена вже відкрита таблиця з бази даних за допомогою натискання кнопки «RefreshDB», щоб договір відредагувати, потрібно написати в текстове поле поряд з кнопкою «Редагувати» код договору(як вказано на Рис. 3.3) який вказується в першому стовпчику, та натиснути на кнопку «Редагувати».

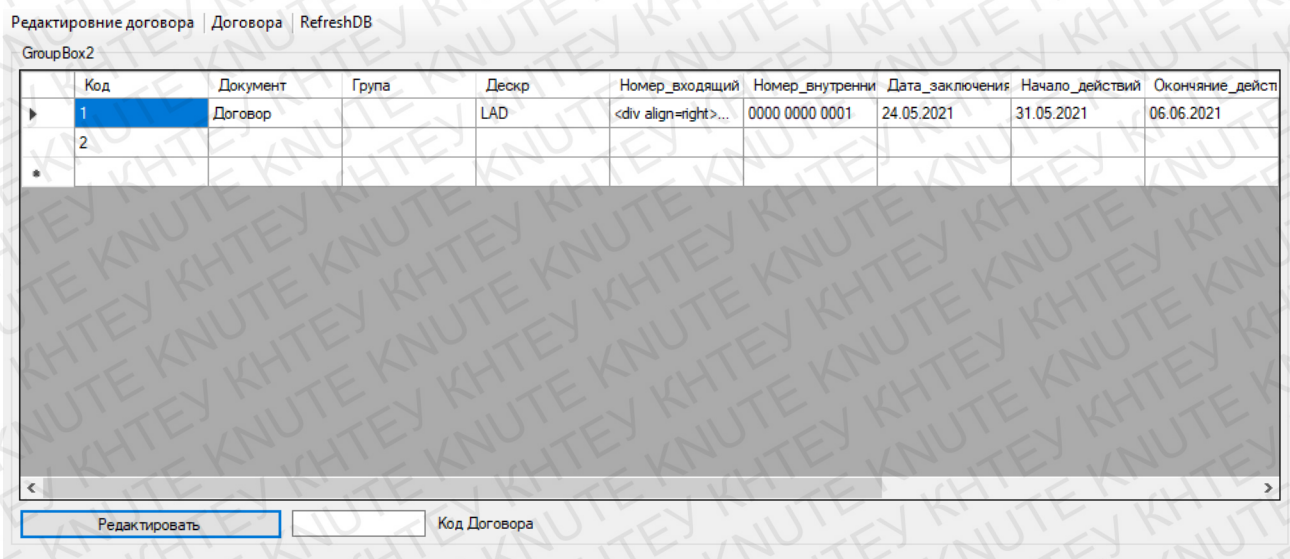


Рис. 3.2. Вікно програми з завантаженою таблицею.

Редактировать договор | Договора | RefreshDB

GroupBox2

	Код	Документ	Группа	Дескр	Номер_входящий	Номер_внутренни	Дата_заклучения	Начало_действий	Окончание_дейст
▶	1	Договор		LAD	<div align=right>...	0000 0000 0001	24.05.2021	31.05.2021	06.06.2021
	2								
*									

Редактировать Код Договора

Рис. 3.3. Зображення, як вибрати договір для редагування.

На рисунку 3.4 зображено вікно редагування договору. В цьому вікні можна відредагувати всі ключові складаючі договору.

Группа Дескр. Входящий номер Дата Статус

Вид

Тип

На период с по на сумму Валюта

налоги

Предмет договора

Ответственный Должность

Курирующее подразделение

Агентский договор

Контрагент 1

Контрагент 2

Спецификация

Рис. 3.4. Вікно редагування договору.

На рисунку 3.5 відкрите вікно 2 на якому є елемент для відображення таблиці, елемент відображення pdf файлів 4 текстових поля(2 з них використовується для вказання шляху до файлів, інші для вказання назви договору та коду договори), також 2 кнопки «Добавити договір» та

«Видалити», на верхній панелі вікна розташовані елементи управління, кнопка «Tools» та «Відкриття договорів».



Рис. 3.5. Вікно у якому можливо відкрити договір у PDF.

На рисунку 3.6 зображена ілюстрація як оновити таблицю через кнопку «RefreshDB».

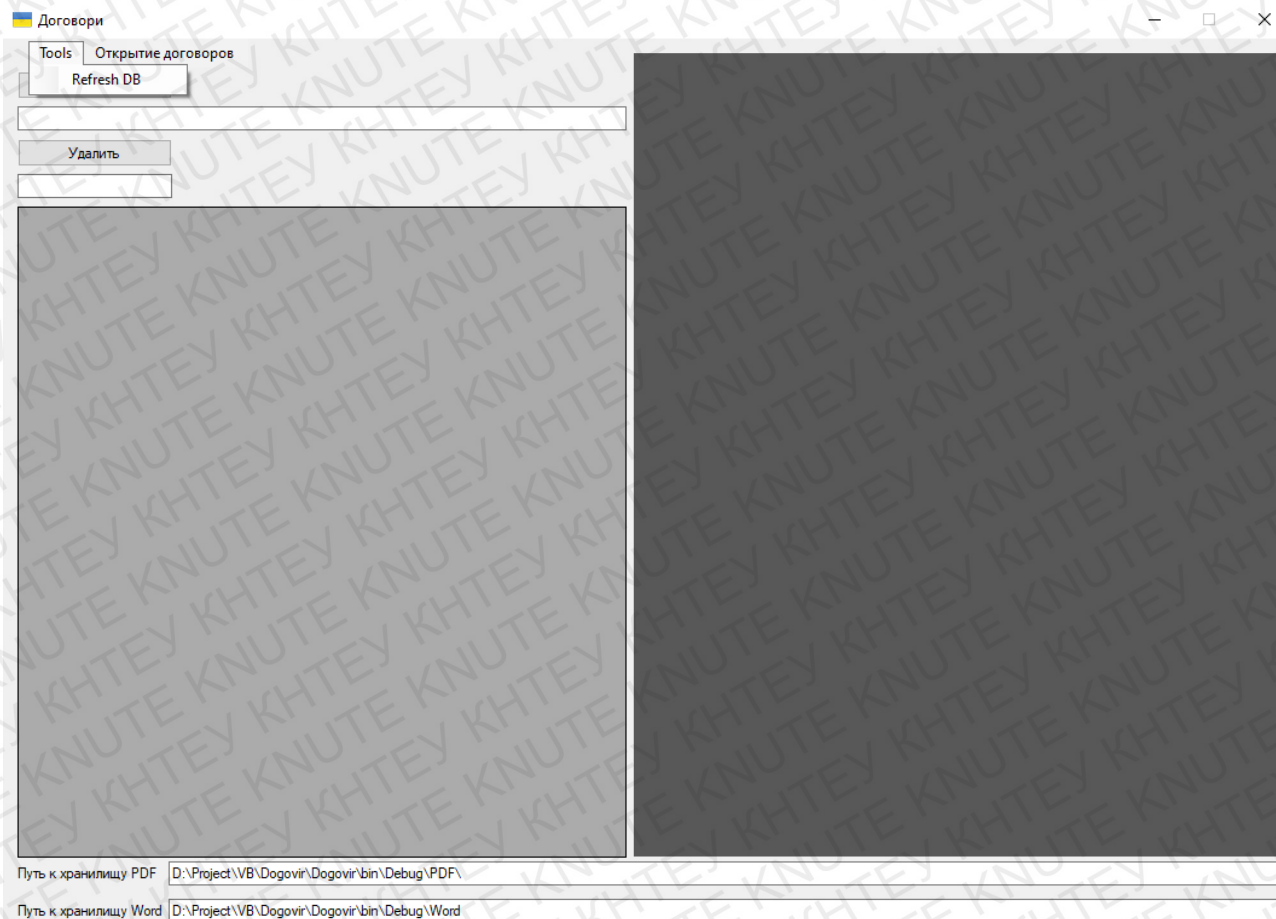


Рис. 3.6. Пример, как перезагрузить таблицу.

На рисунке 3.7 изображено на верхней панели окна рабочего интерфейса две открытые вкладки с кнопками «Відкрити в PDF», «Відкрити в MS Word» по нажатии любой из них з'являється вікно відкриття договорів.



Рис. 3.7. Приклад як відкрити таблицю.

На рисунку 3.8 Вікно відкриття договорів. Відкриває директорію вказану в текстовому полі на головній сторінці вікна 2 у нижній частині віконного інтерфейсу.

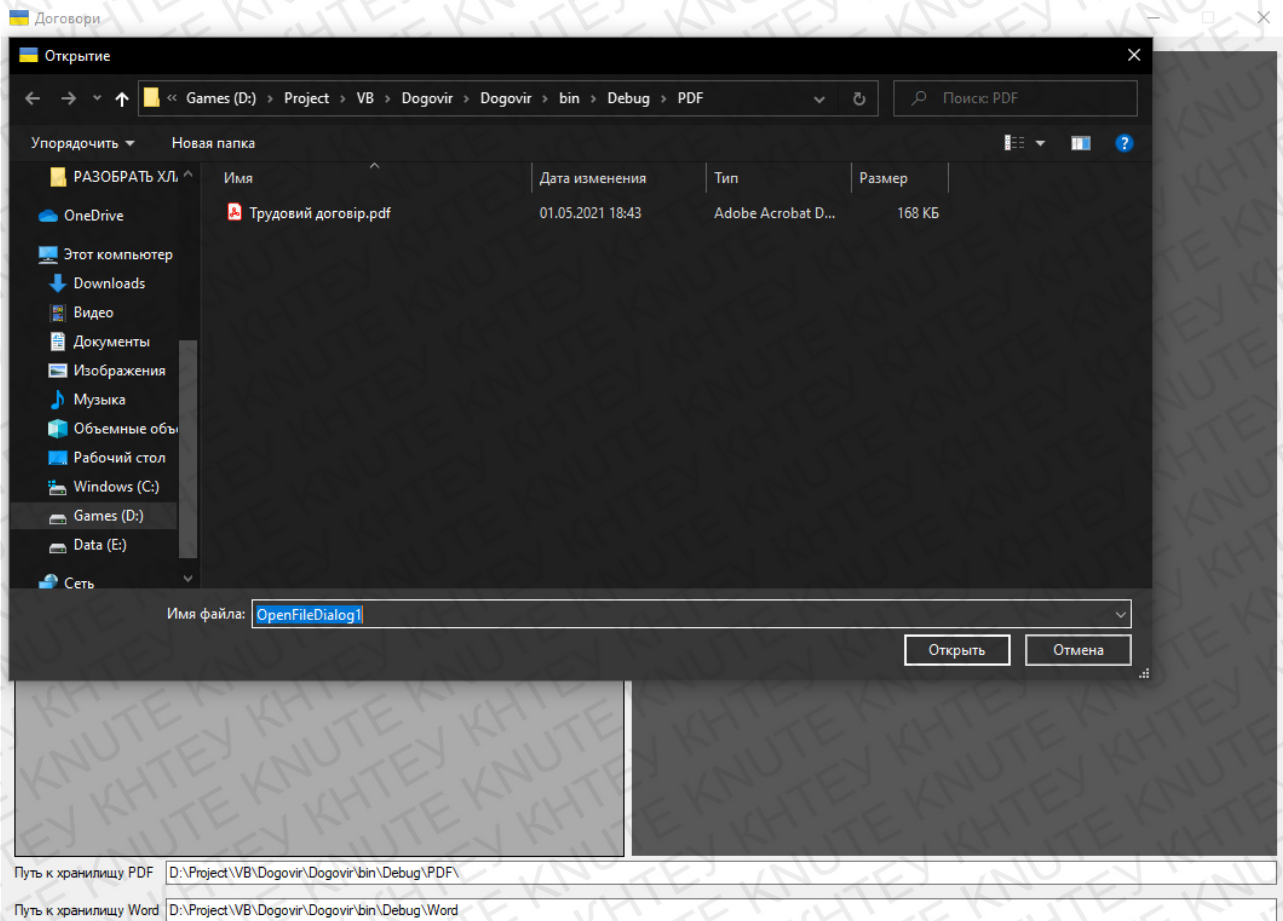


Рис. 3.8. Діалогове вікно, для відкриття договору.

На рисунку 3.9 показано віконний інтерфейс програми у якому відкрито файл pdf у правій частині вікна. З ним можна взаємодіяти, збільшити або зменшити розмір, зберегти, відіслати на печать.

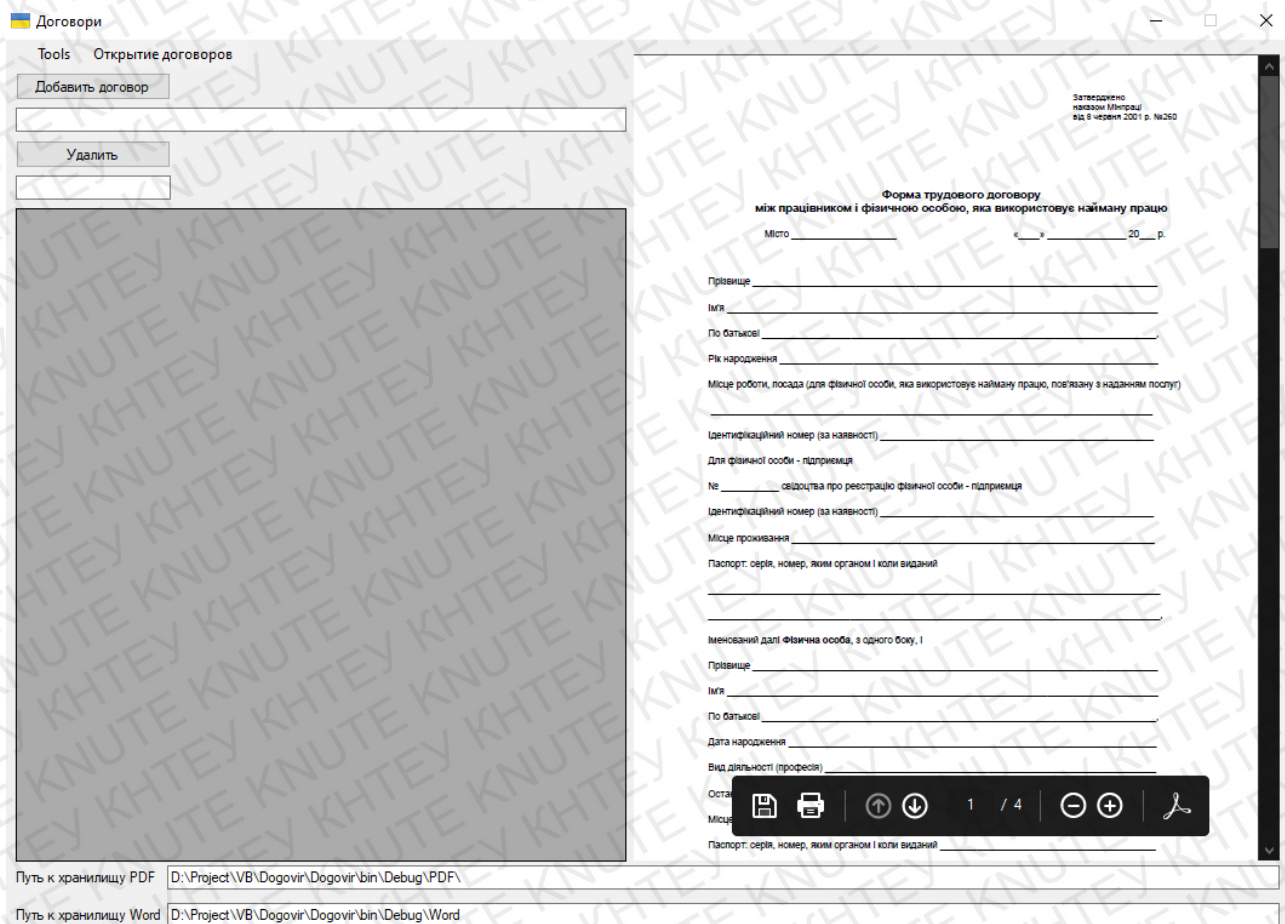


Рис. 3.9. Вікно програми у якому відкритий договір.

3.3. Перевірка коду, компіляція та створення програм.

Створення програмного продукту, після проведення етапу його проектування, потребує виконання наступних кроків: написання на алгоритмічній мові початкового тексту програм, відлагодження цих програм, виконання компіляції програми (створення об'єктного модулю) та створення завантаженого модулю (підключення до об'єктного модулю стандартних модулів). На виконання цих етапів витрачається значна частина часу від всього загального часу на розробку системи в цілому.

Нижче наведені тексти програм, які реалізовані в програмному комплексі створення та ведення договорів підприємства.

Фрагмент програми створення вікна з номером 1.

```
Imports System.Data.OleDb
```

```
Imports System
```

```
Imports System.IO
```

```
Imports System.IO.Compression
```

```
Public Class Form1
```

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Form2.Show()
    conn = New OleDbConnection
    conn.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0; Data Source=Database.mdb;Persist Security Info=False;"
    conn.Open()
End Sub
```

```
Private Sub RefreshDBToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles RefreshDBToolStripMenuItem.Click
    RefreshGrid()
End Sub
```

```
Private Sub RefreshGrid()
    Dim c As New OleDbCommand
    c.Connection = conn
    c.CommandText = "Select * from Договора"

    Dim ds As New DataSet
    Dim da As New OleDbDataAdapter(c)
    da.Fill(ds, "Договора")
    DataGridView1.DataSource = ds
    DataGridView1.DataMember = "Договора"
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    Dim n As String
    n = TextBox1.Text
    Dim c As New OleDbCommand
    c.Connection = conn
    If TextBox1.Text = "" Then
        MsgBox("Ошибка")
    Else
        c.CommandText = "insert into Договора(Name) values('" & n & "')"
        c.ExecuteNonQuery()
        RefreshGrid()
        TextBox1.Clear()
    End If
End Sub
```

```
End Sub
```


Private Sub Button1_Click(sender **As Object**, e **As EventArgs**) **Handles**

Button1.Click

Dim k **As Integer**

Dim c **As New** OleDbCommand

c.Connection = conn

k = TextBox2.Text

c.CommandText = "delete from Договора where N =" & k

c.ExecuteNonQuery()

RefreshGrid()

TextBox2.Clear()

End Sub

Private Sub DataGridView1_CellContentClick(sender **As Object**, e **As**
DataGridViewCellEventArgs) **Handles** DataGridView1.CellContentClick

End Sub

Private Sub Button3_Click(sender **As Object**, e **As EventArgs**)

End Sub

Private Sub ОткрытьPDFToolStripMenuItem_Click(sender **As Object**, e **As**
EventArgs) **Handles** ОткрытьPDFToolStripMenuItem.Click

If OpenFileDialog1.ShowDialog = System.Windows.Forms.DialogResult.OK
Then

AxAcroPDF1.src = OpenFileDialog1.FileName()

End If

End Sub

Private Sub ОткрытьBWordToolStripMenuItem_Click(sender **As Object**, e **As**
EventArgs) **Handles** ОткрытьBWordToolStripMenuItem.Click

Process.Start(TextBox4.Text)

End Sub

End Class

ВИСНОВКИ

Діяльність підприємства в значній мірі залежить від взаємодії з іншими організаціями, які забезпечують ресурсами виробничі процеси підприємстві. Правові засади роботи підприємства з сторонніми організаціями визначаються умовами, які оговорені в договорах.

Договори мають складну структуру та пов'язані, як правило, з значною законодавчою базою. Вони повинні відображати не тільки атрибути взаємодіючих сторін, але і календарний план постачань матеріальних ресурсів, фінансове забезпечення та інші складові.

Трудомісткість створення та ведення договорів потребує впровадження сучасних інформаційних систем для автоматизованих їх підготовки.

У випускній кваліфікаційній роботі розроблені інструментальні засоби, які дозволяють створювати базу даних та виконувати в ній функції створення, редагування, накопичення та перегляду договорів підприємства. В програмі реалізовані режими представлення накопиченої бази договорів в стандартних форматах текстових процесорів. Перегляд договорів виконується у форматі PDF. Також є можливість виводу договору на друкування на принтері. Має можливість пошуку договору по назві або по ідентифікаційному коду, який надається.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Стратегії розвитку регіонів: шляхи забезпечення дієвості / [Біла С. О., Шевченко О. В., Жук В. І. та ін.]; під ред. С. О. Білої – К.: НІСД, 2011. – 88 с.
2. Бережная Е.В. Математические методы моделирования экономических систем / Бережная Е.В., Березной В.И. М.: «Финансы и статистика», 2001. - 368 с.
3. Орлов А.И. Устойчивость в социально-экономических моделях / Орлов А.И. - М.: Наука, 1979. - 218 с.
4. Андрійчук В.Г. Економіка аграрних підприємств [2-ге вид., доп і перероб.] / В.Г. Андрійчук - К.: ІЗМІХ, 2002. – 624 с.
5. Кастельс М. М. Информационная эпоха: экономика, общество и культура [пер. с англ. под науч. ред. О. И. Шкаратана] / Кастельс М.М.: ГУ ВШЭ, 2000. — 608 с.
6. Побурко Я.О. Моніторингові оцінювання складних соціально-економічних явищ розвитку регіону / Я.О. Побурко. - Львів: НАН України. Ін-т регіональних досліджень, 2006. – 306 с.
7. Hatry H.P., Blair L.S., Fisk D.M., et al. How effective are your community services? Procedures for measuring their quality / Harry P. Hatry, Louis H. Blair, Donald M. Fisk, John M. Greiner, John R. Hall, Jr., and Philip S. Schaenman. 2nd ed. – Washington, D.C.: Urban Institute; ICMA, 1992 – 159 p.
8. Morse K. Struyk R.J. Policy Analysis for Effective Development: strengthening transition economies / Kristin Morse, Raymond J. Struyk - New Delhi: Terri Press. Lynne Reinner Publishers Inc., 2006 – 441 p.
9. Hatry H.P., Winnie R.E., Fisk D.M., et al. Practical Program Evaluation for State and Local Governments / Harry P. Hatry, Richard E. Winnie, Donald M. Fisk, Louis H. Blair - Washington, D.C.: Urban Institute; ICMA, 1981 – 123 p.
40. Пурський О.І. Соціально-економічний моніторинг як фактор стабільного і збалансованого розвитку регіону / О.І. Пурський, І.О. Мороз, О.І. Моїсеєнко // Бізнес Інформ. – 2012. – №6. – С. 39-41.

46. Fingleton B., Lopez-Bazo E. Empirical growth models with spatial effects / B. Fingleton, E Lopez-Bazo // Papers Regional Science, Blackwell Publishing. – 2006. - vol. 85(2). – P. 177–198.

47. Федулова Л.І. Інноваційна економіка: Підручник / Л.І.Федулова. - К.: Либідь, 2006. – 480 с.

48. Організація баз даних: практичний курс: Навч. посіб. для студ. / А.Ю. Берко, О.М.Верес; Нац. ун-т «Львів. політехніка».— Л., 2003.— 149 с.— Бібліогр.: 8 назв.

50. Пурський О.І. Визначення інтегральних показників соціально-економічного розвитку регіонів на основі експертних оцінок та методу головних компонент / О.І. Пурський, І.О. Мороз // Проблеми економіки. – 2013. – №2. – С. 230-236.

51. Мартин Дж. Организация баз данных в вычислительных системах / Дж. Мартин. Пер. с англ. – М.: Мир, 1980.–622 с

52. C# 4.0 и платформа .NET 4 для профессионалов / [Кристиан Нейгел, Билл Ивсен, Джей Глинн та ін.]. – М. : Издательский дом "Вильямс", 2011. – 1440 с.

53. Suehring S. MySQL bible / S. Suehring. – NY.: Wiley Publishing Inc., 2002. – 686 p.

54. Аткинсон Л. MySQL. Библиотека профессионала / Л. Аткинсон. Пер. с англ. — М.: Издательский дом "Вильямс", 2002. — 624 с.

55. Кузнецов М.В. MySQL 5 / М.В. Кузнецов, И.В. Симдянов. – БХВ-Петербург, 2010. – 1024 с.

57. Иан Коммервиллем Инженерия программного обеспечения = Software Engineering. — 6-е вид. — М.: «Вильямс», 2002. — С. 642

58. Джек Гринфилд, Кит Шорт, Стів Кук, Стюарт Кент, Джон Крупи Фабрики розробки програм (Software Factories): потокова збірка типових додатків, моделювання, структури та інструменти = Software Factories: Assembling Applications with Patterns, Models, Frameworks, and Tools. — М.: «Діалектика», 2006. — С. 592.

59. Логічне і функціональне програмування : навч. посіб. [для студентів ВНЗ України] / В. М. Заяць, М. М. Заяць ; Нац. ун-т "Львів. політехніка". — Кам'янець-Подільський (Хмельниц. обл.) ; Львів : Гордукова І. Є., 2016. — 398 с. : іл., табл., портр.

60. Иван Братко. Алгоритмы искусственного интеллекта на языке PROLOG (Оригінал Prolog Programming For Artificial Intelligence)

61. John McCarthy. Programs with common sense Symposium on Mechanization of Thought Processes. National Physical Laboratory. Teddington, England. 1958.

62. Fisher Black. A deductive question answering system Harvard University. Thesis. 1964.

63. James Slagle. Experiments with a Deductive Question-Answering Program SACM. December, 1965.

64. Cordell Green. Application of Theorem Proving to Problem Solving IJCAI 1969.

65. Carl Hewitt and Gul Agha. Guarded Horn clause languages: are they deductive and Logical? International Conference on Fifth Generation Computer Systems, Ohmsha 1988. Tokyo. Also in *Artificial Intelligence at MIT*, Vol. 2. MIT Press 1991.

66. Pratt T.W., Zelkovitz M.V. Programming languages, design and implementation (4th ed.). Prentice Hall, 2000 (Пратт Т., Зелкович М., Языки программирования: разработка и реализация.- Спб.: Питер, 2002.-688 с.)

67. Terry Winograd. Procedures as a Representation for Data in a Computer Program for Understanding Natural Language MIT AI TR-235. January 1971.

68. Функційне програмування : Навч. посіб. для студ. вищих навч. закл., що навч. за спец. "Програм. забезп. автоматиз. систем" / В. М. Заяць; Нац. ун-т "Львів. політехніка". - Л. : Бескид Біт, 2003. - 159 с. - Бібліогр.: 15 назв.

69. Судомир В. Використання функційної парадигми при програмуванні мікроконтролерів //Матеріали II Міжнародної студентської науково-технічної

конференції „Природничі та гуманітарні науки. Актуальні питання “. – 2019. –
С. 52-52.

70. Armstrong, Deborah J. (February 2006). The Quarks of Object-Oriented Development. Communications of the ACM 49 (2): 123–128. ISSN 0001-0782.
Процитовано 2006-08-08.

ДОДАТОК

Фрагмент програми створення вікна з номером 2.

```
Imports System.Data.OleDb
Imports System
Imports System.IO
Imports System.IO.Compression
Public Class Form2

    Private Sub Form2_Load(sender As Object, e As EventArgs) Handles MyBase.Load

    End Sub
    Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click
        GroupBox2.Visible = False
        GroupBox1.Visible = True
        GroupBox1.Location = New Point(12, 28)

    End Sub
    Private Sub ToolStripLabel1_Click(sender As Object, e As EventArgs) Handles ToolStripLabel1.Click
        GroupBox2.Visible = False
        GroupBox1.Visible = True
        GroupBox1.Location = New Point(12, 28)

    End Sub
    Private Sub ToolStripLabel3_Click_1(sender As Object, e As EventArgs) Handles ToolStripLabel3.Click
        GroupBox1.Visible = False
        GroupBox2.Visible = True
        GroupBox1.Location = New Point(12, 409)

    End Sub
    Private Sub Button6_Click(sender As Object, e As EventArgs) Handles Button6.Click
        Dim cn As OleDb.OleDbConnection
        Dim tbl As New DataTable
        Dim da As New OleDb.OleDbDataAdapter
        Dim path As String = My.MySettings.Default.DogovirsConnectionString 'путь
к базе данных
        cn = New OleDb.OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=Dogovirs.mdb;Persist Security Info=False;")
        cn.Open()
```



```
da.SelectCommand = New OleDb.OleDbCommand("select * from Договора  
where Сумма = " & 1, cn) ' "Тест" - таблица базы данных;"Код" - столбец  
таблицы; "Text"- строковая переменная для задания условия выборки из базы  
данных.
```

```
da.Fill(tbl)
```

```
TextBox7.Text = tbl.Rows(TextBox18.Text)(12) ' где 0 - это номер строки; 1 -  
номер столбца
```

```
End Sub
```

```
Private Sub ToolStripLabel4_Click(sender As Object, e As EventArgs) Handles  
ToolStripLabel4.Click
```

```
RefreshGrid()
```

```
End Sub
```

```
Private Sub RefreshGrid()
```

```
conn = New OleDbConnection
```

```
conn.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0; Data
```

```
Source=Dogovirs.mdb;Persist Security Info=False;"
```

```
conn.Open()
```

```
Dim c As New OleDbCommand
```

```
c.Connection = conn
```

```
c.CommandText = "Select * from Договора"
```

```
Dim ds As New DataSet
```

```
da.Fill(ds, "Договора")
```

```
DataGridView1.DataSource = ds
```

```
DataGridView1.DataMember = "Договора"
```

```
End Sub
```

```
End Class
```